IPsecure Router KNX

Handbuch

Inhalt

Seite

Inhalt

1 Allgemein

Der Theben IPsecure Router KNX verbindet den KNX-Bus mit einem Ethernet-Netzwerk. Über das Netzwerk können KNX-Telegramme an andere Geräte gesendet oder von diesen empfangen werden. Das Gerät unterstützt das KNX-Secure-Protokoll (KNXnet/IP Security).

1.1 Nutzung des Produkthandbuchs

Das vorliegende Handbuch gibt Ihnen detaillierte technische Informationen über Funktion, Montage und Programmierung des Theben KNX-Geräts. Anhand von Beispielen wird der Einsatz erläutert.

Das Handbuch ist in folgende Kapitel unterteilt:

Kapitel 1	Allgemein
Kapitel 2	Gerätetechnik
Kapitel 3	Inbetriebnahme
Kapitel 4	Planung und Anwendung
Kapitel A	Anhang

1.1.1 Hinweise

In diesem Handbuch werden Hinweise und Sicherheitshinweise folgendermaßen dargestellt:

Hinweis
Bedienungserleichterungen, Bedienungstipps

Beispiele
Anwendungsbeispiele, Einbaubeispiele, Programmierbeispiele

Wichtig
Dieser Sicherheitshinweis wird verwendet, sobald die Gefahr einer Funktionsstörung besteht, ohne Schaden- oder Verletzungsrisiko.

Achtung
Dieser Sicherheitshinweis wird verwendet, sobald die Gefahr einer Funktionsstörung besteht, ohne Schaden- oder Verletzungsrisiko.

 **Gefahr**
Dieser Sicherheitshinweis wird verwendet, sobald bei unsachgemäßer Handhabung Gefahr für Leib und Leben besteht.

 **Gefahr**
Dieser Sicherheitshinweis wird verwendet, sobald bei unsachgemäßer Handhabung akute Lebensgefahr besteht.

1.2 KNXnet/IP Security

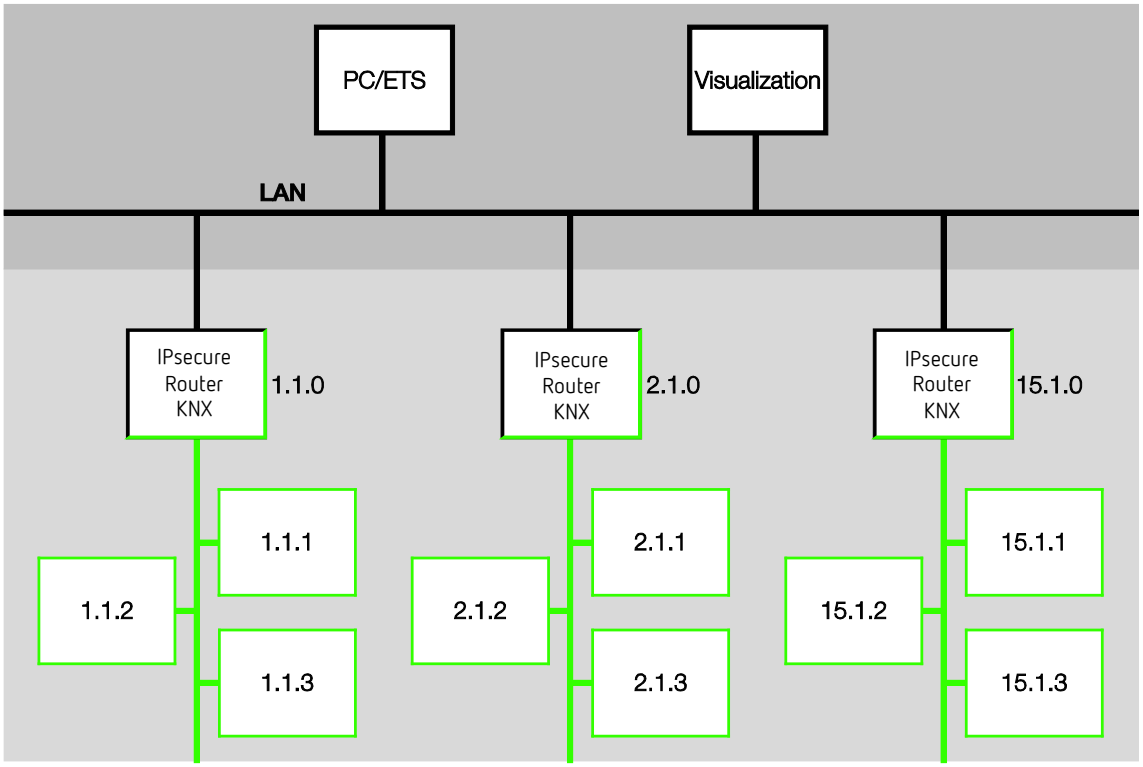
Das Gerät sollte immer im KNX Secure-Modus betrieben werden. So ist sichergestellt, dass die Laufzeitkommunikation auf dem IP-Backbone, die Tunneling Server und die Inbetriebnahme des Gerätes selbst sicher sind.

Siehe auch Kapitel, KNX Secure.

1.3 Produkt- und Funktionsübersicht

Der Theben IPsecure Router KNX verbindet den KNX-Bus mit einem Ethernet-Netzwerk. Über das Netzwerk können KNX-Telegramme an andere Geräte gesendet oder von diesen empfangen werden.

Das Gerät verwendet zur Kommunikation das KNXnet/IP-Protokoll der KNX Association (Routing und Tunneling).



Der Router verfügt über 5 Tunneling Server, siehe Kapitel Die Verwendung der integrierten Tunneling Server. Diese unterstützen sowohl den Busmonitor- als auch den Gruppenmonitorbetrieb. Die Tunneling Server können im KNX Secure-Modus betrieben werden.

Alternativ zur KNX-Standardkommunikation (Multicast) können bis zu 10 IPsecure Router KNX auch über das Unicast-Protokoll miteinander kommunizieren, siehe Kapitel KNX-Telegramme im Netzwerk. Der KNX Secure-Modus ist in dem Fall nicht verfügbar.

Die Spannungsversorgung kann über PoE (Power over Ethernet) nach IEEE 802.3af Class 1 erfolgen oder über eine Hilfsspannung.

Für die IPsecure Router KNX steht das Theben IP-Tool zur Verfügung, mit dem die Router im Netzwerk gefunden werden können (IP-Discovery), die Einstellungen für die Unicast-Kommunikation vorgenommen werden können und im Bedarfsfall die Firmware aktualisiert werden kann, siehe Kapitel Das Theben IP-Tool .

Für das Firmware Update steht eine ETS App (ABB KNX Bus Update) zur Verfügung. Sofern der KNX Secure-Modus bei den Geräten nicht aktiviert ist, kann ein Firmware Update auch mit dem i-bus® Tool erfolgen.

Während des Updatevorgangs muss zusätzlich zum IP-Netzwerk (LAN) auch der KNX-Bus (TP) angeschlossen sein, damit die KNX-Parameter korrekt wiederhergestellt werden können. Andernfalls schlägt der Updatevorgang fehl.

Es muss sichergestellt werden, dass während des Updatevorgangs kein Spannungsausfall (KNX oder IP) auftritt, da ansonsten das Gerät zerstört werden kann.

Das Gerät unterstützt die KNX-Standardfunktion "Überwachung auf Busspannungsausfall". Dies ist eine Netzwerk-Management-Funktion, die z. B. von Visualisierungen verwendet wird (siehe Kap. Überwachung auf Busspannungsausfall.

1.3.1 Überwachung auf Busspannungsausfall

Der IPsecure Router KNX überwacht den KNX-TP-Bus auf Spannungsausfall. Bei einer Zustandsänderung der Busspannung wird ein Broadcastbefehl vom Typ "NetworkParameterWrite" auf das IP-Netzwerk gesendet.

Folgende Werte werden gesendet:

- Busspannungsausfall: "00063301" (hex)
- Busspannungswiederkehr: "00063300" (hex)

Diese Telegramme können z. B. von einer Visualisierung ausgewertet werden.

Typ	Bedeutung	Info
NetworkParameterWrite	Busspannungsausfall TP1	00 06 33 01
NetworkParameterWrite	Busspannungswiederkehr TP1	00 06 33 00

1.3.2 Übersicht Versionen

Gerät	IPsecure Router KNX	
Applikation	IPsecure Router KNX	IPsecure Router KNX
ETS	ETS 4/5	ETS 5
Eigenschaften *IPsecure Router KNX*		
Anzahl Tunneling Server	5	5
Anzahl Unicast-Verbindungen	10	10
Überwachung auf Busspannungsausfall (siehe Kap. Überwachung auf Busspannungsausfall)	■	■
Filterung Gruppentelegramme Hauptgruppe 0...31	■	■
IP-Discovery (IP-Tool)	■	■
Firmware Update (IP-Tool)	■	■*
Parametrierung Unicast (IP-Tool)	■	■*
Firmware Update mit KNX Bus Update App	-	■
Power over Ethernet	■	■
KNX Secure	-	■

■ = Eigenschaft trifft zu

- = Eigenschaft trifft nicht zu

* Nur wenn Gerät nicht im KNX Secure-Modus betrieben wird

2 Gerätetechnik

IPsecure Router KNX

Der IPsecure Router KNX bildet die Schnittstelle zwischen KNX-Installationen und IP-Netzwerken. Er kann als Linien- oder Bereichskoppler eingesetzt werden und dabei das lokale Netzwerk (LAN) für den Austausch von Telegrammen zwischen den Linien/Bereichen nutzen.

Mit der ETS können KNX-Geräte über das LAN programmiert werden (es stehen 5 Tunneling Server zur Verfügung). Das Gerät verwendet das KNXnet/IP-Protokoll der KNX-Association (Routing und Tunneling). Alternativ kann das Gerät über Unicast kommunizieren.
Die Stromversorgung erfolgt über 12 bis 30 V DC oder PoE (Power over Ethernet) nach IEEE 802.3af Class 1.

2.1 Technische Daten

Versorgung	Hilfsspannung U_s	12...30 V DC (+10 % / -15 %) oder PoE (IEEE 802.3af Klasse 1)
	Verlustleistung	Maximal 1,8 W
	Stromaufnahme Hilfsspannung	Maximal 120 mA bei 12 V
	Nennspannung U_n	12 V DC
	Stromaufnahme KNX	< 10 mA
Anschlüsse	KNX	Busanschlussklemme
	Betriebsspannung	Steckklemme
	LAN	RJ45-Buchse für 10/100BaseT, IEEE 802.3 Netzwerke, AutoSensing
Bedien- und Anzeigeelemente	LED rot und Taste	Zur Vergabe der physikalischen Adresse
	LED grün "On"	Anzeige Betriebsbereitschaft
	LED gelb "LAN/Link"	Anzeige Netzwerkverbindung
	LED gelb "Telegram"	Anzeige KNX-Telegrammverkehr
Schutzart	IP 20	Nach DIN EN 60 529
Schutzklasse	II	Nach DIN EN 61 140
Isolationskategorie	Überspannungskategorie	III nach DIN EN 60 664-1
	Verschmutzungsgrad	2 nach DIN EN 60 664-1
KNX-Sicherheitskleinspannung	SELV 30 V DC	
Temperaturbereich	Betrieb	-5 °C...+45 °C
	Lagerung	-25 °C...+55 °C
	Transport	-25 °C...+70 °C
Umgebungsbedingung	maximale Luftfeuchte	95 %, keine Betauung zulässig
	Luftdruck	Atmosphäre bis 2.000 m

IPsecure Router KNX
Gerätetechnik

Design	Reiheneinbaugerät (REG)	Modulares Installationsgerät, Pro*M*
	Abmessungen	90 x 36 x 64 mm (H x B x T)
	Einbaubreite	2 Module à 18 mm
	Einbautiefe	68 mm
Montage	Auf Tragschiene 35 mm	Nach DIN EN 60 715
Einbaulage	Beliebig	
Gewicht	0,1 kg	
Gehäuse, Farbe	Kunststoff, halogenfrei, grau	
Approbation	KNX nach EN 50 090-1, -2	
CE-Zeichen	gemäß EMV- und Niederspannungsrichtlinien	

Gerätetyp	Applikation	maximale Anzahl Kommunikationsobjekte	maximale Anzahl Gruppenadressen	maximale Anzahl Zuordnungen
IPsecure Router KNX	IPsecure Router KNX /...*	0	0	0

* ... = aktuelle Versionsnummer der Applikation. **Bitte beachten Sie hierzu die Softwareinformationen auf unserer Homepage.**

Hinweis

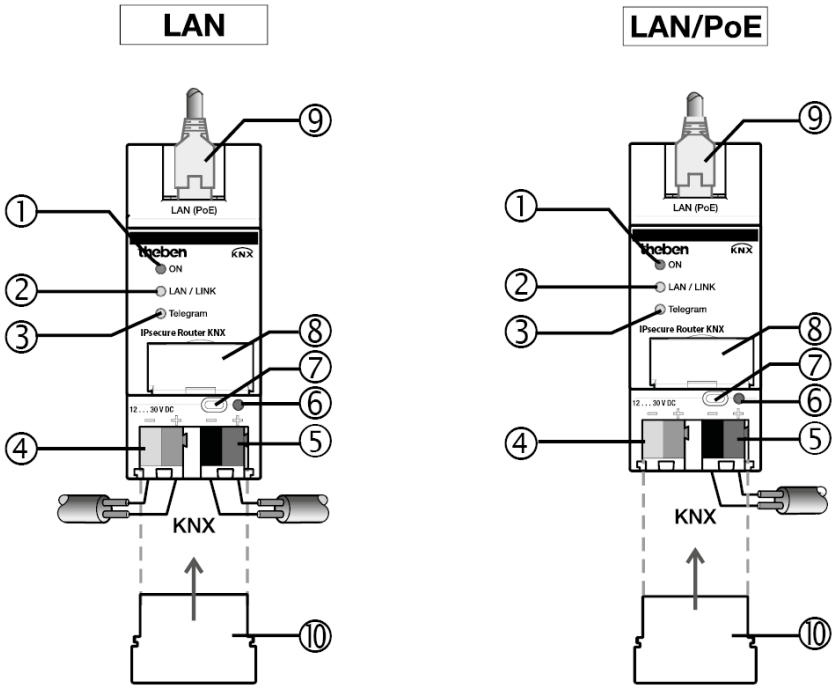
Für die Programmierung sind die ETS und die aktuelle Applikation des Gerätes erforderlich. Soll das Gerät im KNX Secure-Modus betrieben werden, ist zusätzlich der seitlich auf dem Gerät aufgebrachte Inbetriebnahmeschlüssel (FDSK, siehe Kapitel, KNX Secure) erforderlich.

Die aktuelle Applikation finden Sie mit der entsprechenden Softwareinformation zum Download im Internet unter *http://www.theben.de/downloads*. Nach dem Import in die ETS liegt die das Gerät im Fenster *Kataloge* unter *Hersteller/THEBEN AG/Systemgeräte/IPsecure Router KNX*.

Das Gerät unterstützt nicht die Verschließfunktion eines KNX-Geräts in der ETS. Falls Sie den Zugriff auf alle Geräte des Projekts durch einen *BCU-Schlüssel* sperren, hat es auf dieses Gerät keine Auswirkung. Es kann weiterhin ausgelesen und programmiert werden.

Ausnahme: Wenn der KNX Secure-Modus aktiviert ist, kann das Gerät nicht mehr mit einer anderen ETS programmiert werden.

2.2 Anschluss

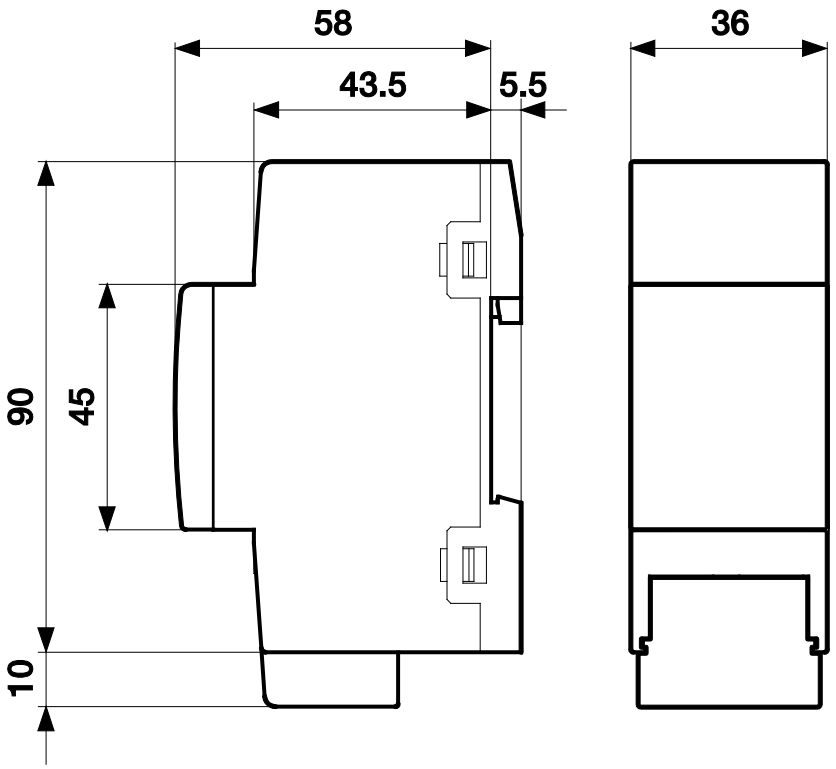


IPsecure Router KNX

1	LED ON	**6**	LED *Programmieren*
2	LED LAN/LINK	**7**	Taste *Programmieren*
3	LED Telegramm	**8**	Schildträger
4	Anschluss Spannungsversorgung	**9**	Anschluss LAN bzw. LAN/PoE
5	Anschluss KNX	**10**	Abdeckkappe

Hinweis
Es ist auch möglich, den Router über den unverdrosselten Spannungsausgang einer KNX-Spannungsversorgung (Typ SV/S) zu versorgen.

2.3 Maßbild



IPsecure Router KNX

2.4 Montage und Installation

Das Gerät ist ein Reiheneinbaugerät zum Einbau in Verteilern zur Schnellbefestigung auf 35-mm-Tragschienen nach DIN EN 60 715.

Das Gerät kann in jeder Einbaulage montiert werden.

Die Verbindung zum Bus erfolgt über die mitgelieferte Busanschlussklemme. Die Klemmenbezeichnung befindet sich auf dem Gehäuse.

Das Gerät ist betriebsbereit, nachdem die Busspannung und die Hilfsspannung angelegt wurden.

Die Zugänglichkeit des Geräts zum Betreiben, Prüfen, Besichtigen, Warten und Reparieren muss gemäß DIN VDE 0100-520 sichergestellt sein.

Inbetriebnahmevoraussetzung

Um das Gerät in Betrieb zu nehmen, werden ein PC mit der ETS 5 oder höher sowie eine Versorgungsspannung von 12 bis 30 V DC benötigt. Alternativ kann die Versorgung über PoE (Power over Ethernet) erfolgen.

Mit dem Anlegen der Busspannung und der Hilfsspannung ist das Gerät betriebsbereit.

Montage und Inbetriebnahme dürfen nur von Elektrofachkräften ausgeführt werden. Bei der Planung und Errichtung von elektrischen Anlagen sowie von sicherheitstechnischen Anlagen für Einbruch- und Branderkennung sind die einschlägigen Normen, Richtlinien, Vorschriften und Bestimmungen des jeweiligen Landes zu beachten.

- Gerät bei Transport, Lagerung und im Betrieb vor Feuchtigkeit, Schmutz und Beschädigung schützen!
- Gerät nur innerhalb der spezifizierten technischen Daten betreiben!
- Gerät nur im geschlossenen Gehäuse (Verteiler) betreiben!
- Vor Montagearbeiten ist das Gerät spannungsfrei zu schalten.

Gefahr

Um gefährliche Berührungsspannung durch Rückspeisung aus unterschiedlichen Außenleitern zu vermeiden, muss bei einer Erweiterung oder Änderung des elektrischen Anschlusses eine allpolige Abschaltung vorgenommen werden.

Auslieferungszustand

Das Gerät wird mit der physikalischen Adresse 15.15.0 ausgeliefert.

Alle physikalischen Adressen der Tunneling-Verbindungen stehen im Auslieferungszustand auf 15.15.100, d.h. nach außen ist nur ein Tunnel sichtbar. Erst nach dem ersten Download werden die in der ETS eingestellen Adressen der Tunneling-Verbindungen übernommen. Die IP-Adresse ist auf automatische Vergabe (DHCP/AutoIP) eingestellt.

Hinweis

Das Gerät wird ab Werk mit der Option *Weiterleiten* ausgeliefert. Das entspricht nicht der Standardeinstellung in der Applikation, erleichtert aber die Inbetriebnahme. Siehe Kapitel Parameterfenster KNX -> LAN.

Nach dem ersten Download wird dann die parametrierte Einstellung übernommen.

Vergabe der physikalischen Adresse

In der ETS erfolgt die Vergabe und Programmierung der physikalischen Adressen und Parameter.

Das Gerät besitzt zur Vergabe der physikalischen Adresse eine Taste *Programmieren*. Nachdem die Taste betätigt wurde, leuchtet die rote LED *Programmieren* auf. Sie erlischt, sobald die ETS die physikalische Adresse vergeben hat oder die Taste *Programmieren* erneut betätigt wurde.

Downloadverhalten

Das Gerät kann auf unterschiedliche Arten programmiert werden: Über einen der integrierten Tunneling Server, über lokalen Download, über KNXnet/IP Routing oder über eine weitere Programmierschnittstelle (USB oder IP).

Hinweis
Wird für die Programmierung eines KNX Secure-Gerätes eine USB-Schnittstelle verwendet, muss diese „Long Frames" unterstützen. Geeignet ist z. B. die USB-Schnittstelle von Theben.

Damit das Gerät programmiert werden kann, muss eine Verbindung zum KNX TP (Twisted Pair) bestehen.

Nach erfolgtem Download startet das Gerät neu und schließt alle offenen Tunneling-Verbindungen. Sofern beim Download die IP-Adresse des Gerätes geändert wurde, müssen die Tunneling-Verbindungen manuell in den Tunneling Clients neu konfiguriert werden. Tunneling Clients stellen die Verbindung zum Server über die IP-Adresse her.

Entladen des Gerätes und Rücksetzen auf Werkseinstellungen

Das Gerät kann auf Werkseinstellungen zurückgesetzt werden. Da es sich um ein Secure-Gerät handelt, ist folgendes zu beachten:

Im KNX Secure-Modus-Betrieb kann das Gerät über die ETS nur dann zurückgesetzt werden, wenn die ETS das Projekt verwendet, mit dem das Gerät parametriert wurde, bzw. wenn im Projekt der Inbetriebnahmeschlüssel vorhanden ist.

Über einen Rechtsklick auf das Gerät in der ETS kann das Gerät entladen werden.

Option: Applikation entladen

- Die IP-Adresse und IP-Konfiguration bleiben erhalten
- Die Unicast-Konfiguration bleibt erhalten, sofern vorhanden
- Die Passwörter und IP Adressen der Tunnelingserver werden gelöscht
- Der Schlüssel für die Multicastkommunikation („Backbone Key") bleibt
- Der von der ETS vergebene Tool Key bleibt erhalten, d.h.für die erneute Programmierung ist der FDSK nicht erforderlich.
- Die physikalische Adresse bleibt erhalten

Option: Physikalische Adresse und Applikation entladen

- Das Gerät wird auf Werkszustand zurück gesetzt
- Für die erneute Inbetriebnahme ist der FDSK notwendig, sofern er nicht von der ursprünglichen Inbetriebnahme noch im ETS Projekt vorhanden ist.

Das Zurücksetzen auf Werkseinstellungen kann auch direkt am Gerät vorgenommen werden. Dies stellt kein Sicherheitsrisiko dar, da das Gerät anschließend nicht mehr Teil der Anlage ist.

- Drücken der Programmiertaste bei nicht verbundenem KNX Bus
- Programmiertaste gedrückt halten und Busklemme aufstecken. Die Programmier-LED blinkt (2 Hz).
- Taste mindestens 5s gedrückt halten und dann loslassen. Die Programmier-LED erlischt, das Gerät startet neu mit Werkseinstellungen.

Verbindet sich nach dem Rücksetzen die ETS mit dem Gerät und der FDSK-Schlüssel des Gerätes ist der ETS noch bekannt, kann der Router erneut programmiert werden. Die ETS meldet in dem Fall, dass das Gerät zurückgesetzt wurde.

Weitere Hinweise zum FDSK (Factory Default Setup Key), siehe Kapitel, KNX Secure.

Reinigen

Das Gerät ist vor dem Reinigen spannungsfrei zu schalten. Verschmutzte Geräte können mit einem trockenen oder leicht mit Seifenlauge angefeuchteten Tuch gereinigt werden. Auf keinen Fall dürfen ätzende Mittel oder Lösungsmittel verwendet werden.

Wartung

Das Gerät ist wartungsfrei. Bei Schäden, z. B. durch Transport und/oder Lagerung, dürfen keine Reparaturen vorgenommen werden.

2.5 Beschreibung der Ein- und Ausgänge

Versorgungsspannungseingang 12 bis 30 V DC

Am Eingang für die Versorgungsspannung darf nur eine Gleichspannung von 12 bis 30 V angeschlossen werden. Wir empfehlen die Verwendung der Netzteile Spannungsvers. 640 mA T KNX aus unserem Sortiment.

Achtung
Die Versorgungsspannung muss 12 bis 30 V DC betragen, oder das Gerät wird über PoE (Power over Ethernet) nach IEEE 802.3af Class 1 versorgt. Bei Anschluss von 230 V kann das Gerät zerstört werden!

KNX-Anschluss

Zum Anschluss an den KNX-Bus wird die mitgelieferte Busanschlussklemme verwendet.

Hinweis
Zur Programmierung ist die ETS 5 in der aktuellsten Version erforderlich.

LAN-Anschluss

Die Netzwerkanbindung erfolgt über eine Ethernet-RJ45-Schnittstelle für LAN-Netzwerke. Die Netzwerkschnittstelle kann mit einer Übertragungsgeschwindigkeit von 10/100 MBit/s betrieben werden. Die Netzwerkaktivität wird durch die LED LAN/LINK auf der Gehäusefrontseite angezeigt.

2.6 Bedienelemente

Es befinden sich keine Bedienelemente am IPsecure Router KNX.

2.7 Anzeigeelemente

Auf der Frontseite des IPsecure Routers KNX befinden sich drei LEDs:

ON | LAN/LINK | Telegram

ON

- Die LED leuchtet wenige Sekunden nach Zuschalten der Hilfsspannung.
- Die LED leuchtet nach dem Zuschalten der Hilfsspannung zunächst dauerhaft. Nach ca. 40 Sekunden beginnt die LED zu blinken, bis der Aufstartvorgang vollständig abgeschlossen ist und die LED wieder dauerhaft leuchtet. Dies kann je nach Größe der Filtertabelle 5 bis 60 Sekunden dauern.

LAN/LINK

- Die LED leuchtet, wenn die Hilfsspannung vorhanden ist und der Router an ein Ethernet-Netzwerk angeschlossen ist.
- Die LED blinkt, wenn das Gerät Aktivität auf dem Netzwerk erkennt, z. B. wenn Daten ausgetauscht werden.

Telegram

- Die LED leuchtet, wenn der Router an ein TP-Netzwerk angeschlossen ist und der Aufstartvorgang (siehe LED "On") vollständig abgeschlossen ist.
- Die LED blinkt, wenn das Gerät Aktivität auf der KNX-Sublinie TP1 (Twisted Pair 1) erkennt, z. B. wenn Daten ausgetauscht werden.

3 Inbetriebnahme

Die Parametrierung des IPsecure Router KNX erfolgt mit der Applikation und der Engineering Tool Software ETS.

Die Applikation ist unter *Theben AG/Systemgeräte/IPsecure Router KNX* zu finden.

Für die Parametrierung wird ein PC oder Laptop mit der ETS und eine Anbindung an den KNX-Bus benötigt.

3.1 Überblick

Die Parametrierung des IPsecure Routers KNX erfolgt mit der Engineering Tool Software ETS 5 in der aktuellsten Version.

Einige Funktionen (Unicast) werden über ein separates Tool (IP-Tool) parametriert.

3.2 Parameter

Dieses Kapitel beschreibt die Parameter des IPsecure Routers KNX anhand der Parameterfenster.

Die Parameterfenster sind dynamisch aufgebaut, so dass je nach Parametrierung und Funktion der Ausgänge weitere Parameter oder ganze Parameterfenster freigegeben werden.

Die Defaultwerte der Parameter sind unterstrichen dargestellt, z. B.:

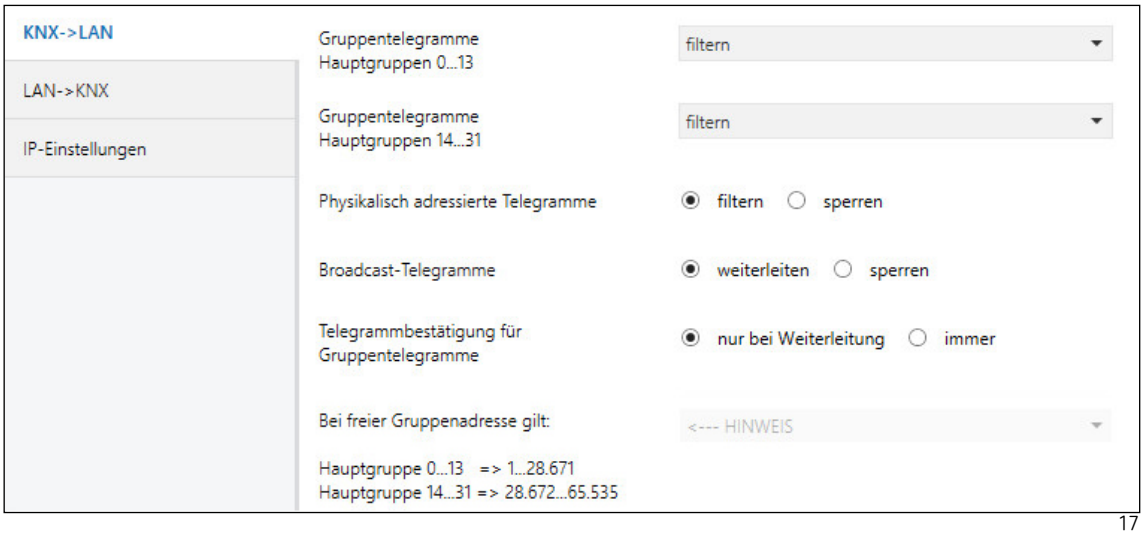
Optionen: ja

<u>nein</u>

3.2.1.1 Parameterfenster *KNX -> LAN*

Im Parameterfenster *KNX -> LAN* kann die Bearbeitung der Telegramme vom KNX-System zum LAN-Netzwerk festgelegt werden.

Hinweis
Das Gerät wird ab Werk mit der Option *Weiterleiten* ausgeliefert. Das entspricht nicht der Standardeinstellung in der Applikation, erleichtert aber die Inbetriebnahme. Nach dem ersten Download wird dann die parametrierte Einstellung übernommen.



Gruppentelegramme Hauptgruppe 0...13

Optionen: filtern
weiterleiten
sperren

Dieser Parameter legt fest, ob Telegramme mit Gruppenadressen der Hauptgruppen 0 bis 13 gefiltert, weitergeleitet oder gesperrt werden sollen.

- *filtern*: Die Telegramme mit Gruppenadressen der Hauptgruppen 0 bis 13 vom KNX zum LAN werden gemäß der Filtertabelle, welche von der ETS automatisch berechnet wird, gefiltert.
- *weiterleiten*: Alle Gruppentelegramme der Hauptgruppen 0 bis 13 werden weitergeleitet, ohne Berücksichtigung der Filtertabelle.

Wichtig
Diese Einstellung ist nur für Inbetriebnahme und Diagnose sinnvoll. Im Normalbetrieb sollte diese nicht verwendet werden. Da durch diese Einstellung die KNX-Linien überlastet werden können, kann es zu einem Telegrammverlust kommen.

- *sperren*: Alle Gruppentelegramme vom KNX zum LAN werden gesperrt, ohne Berücksichtigung der Filtertabelle.

Gruppentelegramme Hauptgruppe 14...31

Optionen: filtern
weiterleiten
sperren

Dieser Parameter legt fest, ob Telegramme mit Gruppenadressen der Hauptgruppen 14 bis 31 gefiltert, weitergeleitet oder gesperrt werden sollen.

- *filtern*: Die Telegramme mit Gruppenadressen der Hauptgruppen 14 bis 31 vom KNX zum LAN werden gemäß der Filtertabelle, welche von der ETS automatisch berechnet wird, gefiltert.
- *weiterleiten*: Alle Gruppentelegramme der Hauptgruppen 14 bis 31 werden weitergeleitet.

Wichtig
Diese Einstellung ist nur für Inbetriebnahme und Diagnose sinnvoll. Im Normalbetrieb sollte diese nicht verwendet werden. Da durch diese Einstellung die KNX-Linien überlastet werden können, kann es zu einem Telegrammverlust kommen.

- *sperren*: Es werden keine Gruppentelegramme der Hauptgruppen 14 bis 31 vom KNX zum LAN übertragen.

Physikalisch adressierte Telegramme

Optionen: filtern
sperren

Dieser Parameter legt fest, ob physikalisch adressierte Telegramme gefiltert oder gesperrt werden.

- *filtern*: Es werden nur die Telegramme vom KNX zum LAN übertragen, welche die Linie des IPsecure Router KNX zum LAN verlassen sollen.
- *sperren*: Physikalisch adressierte Telegramme werden nicht vom IPsecure Router KNX bearbeitet. Bei dieser Einstellung ist es nicht möglich, aus der Linie unterhalb des IPsecure Router KNX heraus in eine andere Linie hinein physikalisch adressierte Telegramme zu schicken, z.B. während der Programmierung.

Broadcast-Telegramme

Optionen: weiterleiten
sperren

Dieser Parameter legt fest, ob Broadcast-Telegramme weitergeleitet oder gesperrt werden.

- *weiterleiten*: Broadcast-Telegramme werden weitergeleitet.
- *sperren*: Broadcast-Telegramme werden nicht vom IPsecure Router KNX bearbeitet. Bei dieser Einstellung ist es nicht möglich, aus der Linie unterhalb des IPsecure Router KNX heraus in eine andere Linie hinein Broadcast-Telegramme zu schicken, z. B. während der Programmierung.

Der Parameter *Broadcast-Telegramme* gilt auch für „System Broadcast-Telegramme". Details siehe Kapitel, System Broadcast.

Telegrammbestätigung für Gruppentelegramme

Optionen: nur bei Weiterleitung
immer

Dieser Parameter legt fest, ob der IPsecure Router KNX Gruppentelegramme mit einem Telegramm bestätigen soll.

- *nur bei Weiterleitung*: Die Gruppentelegramme werden nur bestätigt (ACK senden), wenn sie vom IPsecure Router KNX auch auf das LAN weitergeleitet werden. Damit werden nur Telegramme bestätigt, die auch in der Filtertabelle des IPsecure Router KNX eingetragen sind.
- *immer*: Alle Gruppentelegramme auf dem KNX werden durch den IPsecure Router KNX bestätigt.

Bei freier Gruppenadresse gilt:

Hauptgruppen 0...13 => 1...28.671
Hauptgruppe 14...31 => 28.672...65.535

Hinweis
In der ETS 5 besteht die Möglichkeit, die Gruppenadressen nicht zwei- oder dreistufig zu vergeben, sondern frei. Wird die freie Gruppenadressansicht gewählt, entspricht Hauptgruppe 0...13 dem Untergruppenbereich 1...28.671 und Hauptgruppe 14...31 dem Untergruppenbereich 28.672...65.535. Details hierzu sind in der Hilfe der ETS nachzulesen.

3.2.1.2 Parameterfenster *LAN -> KNX*

Im Parameterfenster LAN -> KNX kann die Bearbeitung der Telegramme vom LAN-Netzwerk zum KNX-System festgelegt werden.

Gruppentelegramme Hauptgruppe 0...13

Optionen: filtern
weiterleiten
sperren

Dieser Parameter legt fest, ob Telegramme mit Gruppenadressen der Hauptgruppen 0 bis 13 gefiltert, weitergeleitet oder gesperrt werden sollen.

- *filtern*: Die Telegramme mit Gruppenadressen der Hauptgruppen 0 bis 13 vom LAN zum KNX werden gemäß der Filtertabelle, welche von der ETS automatisch berechnet wird, gefiltert.
- *weiterleiten*: Alle Gruppentelegramme der Hauptgruppen 0 bis 13 werden weitergeleitet, ohne Berücksichtigung der Filtertabelle.

Wichtig
Diese Einstellung ist nur für Inbetriebnahme und Diagnose sinnvoll. Im Normalbetrieb sollte diese nicht verwendet werden. Da durch diese Einstellung die KNX-Linien überlastet werden können, kann es zu einem Telegrammverlust kommen.

- *sperren*: Alle Gruppentelegramme vom LAN zum KNX werden gesperrt, ohne Berücksichtigung der Filtertabelle.

Gruppentelegramme
Hauptgruppe 14...31

Optionen: filtern
weiterleiten
sperren

Dieser Parameter legt fest, ob Telegramme mit Gruppenadressen der Hauptgruppen 14 bis 31 gefiltert, weitergeleitet oder gesperrt werden sollen.

- *filtern*: Die Telegramme mit Gruppenadressen der Hauptgruppen 14 bis 31 vom LAN zum KNX werden gemäß der Filtertabelle, welche von der ETS automatisch berechnet wird, gefiltert.
- *weiterleiten*: Alle Gruppentelegramme der Hauptgruppen 14 bis 31 werden weitergeleitet.

Wichtig
Diese Einstellung ist nur für Inbetriebnahme und Diagnose sinnvoll. Im Normalbetrieb sollte diese nicht verwendet werden. Da durch diese Einstellung die KNX-Linien überlastet werden können, kann es zu einem Telegrammverlust kommen.

- *sperren*: Es werden keine Gruppentelegramme der Hauptgruppen 14 bis 31 vom LAN zum KNX übertragen.

Physikalisch adressierte Telegramme

Optionen: filtern
sperren

Dieser Parameter legt fest, ob physikalisch adressierte Telegramme gefiltert oder gesperrt werden.

- *filtern*: Es werden nur die Telegramme vom LAN zum KNX übertragen, welche die Linie des IPsecure Router KNX zum LAN verlassen sollen.
- *sperren*: Physikalisch adressierte Telegramme werden nicht vom IPsecure Router KNX bearbeitet. Bei dieser Einstellung ist es nicht möglich, von der Hauptlinie in die KNX TP-Linie hinein physikalisch adressierte Telegramme zu schicken, z. B. während der Programmierung.

Broadcast-Telegramme

Optionen: weiterleiten
sperren

Dieser Parameter legt fest, ob Broadcast-Telegramme weitergeleitet oder gesperrt werden.

- *weiterleiten*: Broadcast-Telegramme werden weitergeleitet.
- *sperren*: Broadcast-Telegramme werden nicht vom IPsecure Router KNX KNX bearbeitet. Bei dieser Einstellung ist es nicht möglich, von der Hauptlinie in die KNX TP-Linie hinein Broadcast-Telegramme zu schicken, z. B. während der Programmierung.

Bei Übertragungsfehlern Telegramme wiederholen

Optionen: ja
nein
benutzerdefiniert

- *ja:* Wird bei der Übertragung eines Telegramms ein Fehler erkannt, wird das Telegramm bis zu drei Mal wiederholt
- *nein:* Die Übertragung wird nicht wiederholt.
- *benutzerdefiniert:* Das Verhalten kann für die unterschiedlichen Telegrammarten individuell eingestellt werden.

  Gruppenadressierte Telegramme wiederholen

  Optionen: ja
  nein

  - *ja:* Wird bei der Übertragung eines gruppenadressierten Telegramms ein Fehler erkannt, wird das Telegramm bis zu drei Mal wiederholt.
  - *nein:* Die Übertragung wird nicht wiederholt.

  Physikalisch adressierte Telegramme wiederholen

  Optionen: ja
  nein

  - *ja:* Wird bei der Übertragung eines physikalisch adressierten Telegramms ein Fehler erkannt, wird das Telegramm bis zu drei Mal wiederholt.
  - *nein:* Die Übertragung wird nicht wiederholt.

  Broadcast-Telegramme wiederholen

  Optionen: ja
  nein

  - *ja:* Wird bei der Übertragung eines Broadcast-Telegramms ein Fehler erkannt, wird das Telegramm bis zu drei Mal wiederholt.
  - *nein:* Die Übertragung wird nicht wiederholt.

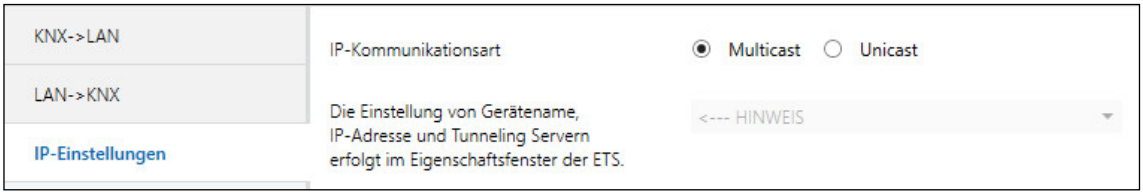
Bei freier Gruppenadresse gilt:

Hauptgruppen 0...13 => 1...28.671
Hauptgruppe 14...31 => 28.672...65.535

Hinweis
In der ETS 5 besteht die Möglichkeit, die Gruppenadressen nicht zwei- oder dreistufig zu vergeben, sondern frei. Wird die freie Gruppenadressansicht gewählt, entspricht Hauptgruppe 0...13 dem Untergruppenbereich 1...28.671 und Hauptgruppe 14...31 dem Untergruppenbereich 28.672...65.535. Details hierzu sind in der Hilfe der ETS nachzulesen.

3.2.1.3 Parameterfenster *IP-Einstellungen*

Im Parameterfenster *IP-Einstellungen* wird eingestellt, wie der IPsecure Router KNX über IP kommuniziert.



IP-Kommunikationsart

Optionen: Multicast
Unicast

Die IP-Kommunikationsart legt fest, welche Art von Telegrammen der IPsecure Router KNX auf das IP-Netzwerk sendet.

- *Multicast*: Dies ist die für KNXnet/IP von der KNX Association festgelegte Kommunikationsart für KNX-IP-Geräte. Diese Einstellung sollte so beibehalten werden und nur geändert werden, wenn durch das vorhandene Netzwerk die Notwendwendigkeit besteht, Telegramme als Unicast zu senden.

  Zur Einstellung der Routing Multicast Adresse siehe Routing Multicast Adresse,

- *Unicast*: Das Routing für dieses Gerät wird abgeschaltet.

  Diese spezielle Kommunikation ist nicht gemäß KNXnet/IP-Spezifikation. Zur Konfiguration wird das Theben IP-Tool benötigt.

  Die Kommunikationsart *Unicast* kann nicht verwendet werden, wenn das Gerät im KNX Secure-Modus betrieben wird. Wird bei aktiviertem KNX Secure-Modus *Unicast* ausgewählt, schaltet die ETS auf *Multicast* um. Die Parametrierung *Unicast* in der Applikation wird dann ignoriert.

  Um die Kommunikationsart *Unicast* zu verwenden, muss der KNX Secure-Modus in der ETS abgeschaltet werden.

Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

Sowohl bei Auswahl *Multicast*, als auch bei Auswahl *Unicast* erscheint folgender Hinweis:

Die Einstellung von Gerätename,
IP-Adresse und Tunneling Servern
erfolgt im Eigenschaftsfenster der ETS.

Bei Auswahl *Unicast* erscheint zusätzlich folgender Hinweis:

Achtung! Diese Einstellung schaltet das Routing für dieses Gerät ab. Die IP-Telegramme werden nun als Unicast an bis zu 9 Zieladressen gesendet.

Die Unicast-Konfiguration erfolgt mit dem Theben IP-Tool.

Siehe Beschreibung Kommunikation Unicast, Kapitel KNX-Telegramme im Netzwerk,

Das IP-Tool kann kostenlos von unserer Homepage (www.theben.de/downloads) geladen werden.

Für das IP-Tool ist keine ETS und auch keine Installation des Falcon erforderlich.

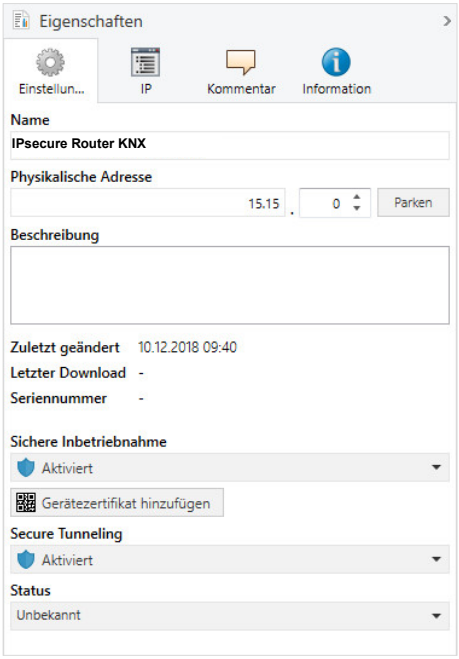
Die Systemanforderungen sind ein Windows-System ab Betriebssystemversion Windows 7 (Service Pack 3) und das .NET Framework ab Version 4.7.2.

Der integrierte Falcon 5.0 unterschützt nur USB und IP-Schnittstellen (kein RS232).

Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

Wichtig
Es ist nicht möglich, die Unicast-Kommunikation bei aktiviertem KNX Secure-Modus zu nutzen. Ist der KNX Secure-Modus aktiviert und der Parameter *IP-Kommunikationsart* auf *Unicast* gesetzt, wird dennoch die Multicast-Kommunikation aktiviert. Bei der Parametrierung der Unicast Adressen über das IP-Tool wird ein entsprechender Hinweis angezeigt.

Die weitere Konfiguration der IP-Parameter (Gerätename, Zuweisung der IP-Adresse per DHCP oder fest) erfolgt im entsprechenden Eigenschaftenfenster der ETS.



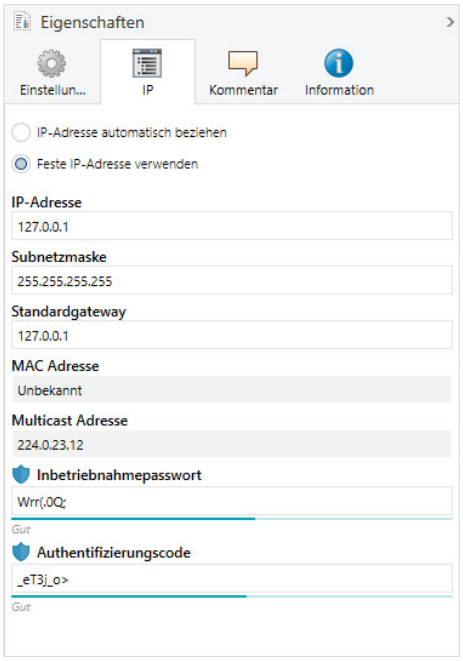
Im Eigenschaftsfenster *Einstellungen* kann der Gerätename eingetragen werden. Im Feld *Name* kann der Gerätename, der ins Gerät geladen wird, geändert werden.

Der Gerätename dient der Identifizierung des Geräts im LAN. Bei einer Suchanfrage, z. B. durch die ETS, meldet jedes KNXnet/IP-Gerät seinen Namen und kann darüber zugeordnet werden. So kann z. B. durch den Namen IPsecure Router KNX, EG, UV7 auch der Einbauort des Geräts mitgeteilt werden.

Hinweis
Bei Auslieferung lautet der Gerätename standardmäßig "IPsecure Router KNX". Nach dem ersten Download wird der Gerätename, der im Eigenschaftsfenster der ETS eingetragen wird, ins Gerät geladen.

Achtung
Es werden nur die ersten 30 Zeichen des Gerätenamens ins Gerät geladen, der Rest wird abgeschnitten.

Im Eigenschaftsfenster *IP* kann die IP-Adresse definiert werden.



Für die Einstellung der IP-Adresse stehen folgende Optionen zur Verfügung:

Optionen: IP-Adresse automatisch beziehen
Folgende IP-Adresse verwenden

- *IP-Adresse automatisch beziehen*: In der Standardeinstellung erwartet der IPsecure Router KNX die Zuweisung einer IP-Adresse durch einen DHCP-Server (dynamic host configuration protocol). Dieser Server vergibt auf Anfrage eine freie IP-Adresse an das Gerät. Ist kein DHCP-Server im Netzwerk verfügbar oder antwortet dieser nicht, so startet das Gerät eine Auto-IP-Prozedur. Es vergibt sich selbst eine Adresse aus dem reservierten Bereich für Auto-IP-Adressen (169.254.xxx.yyy).

  Zu DHCP: siehe Kap. Vergabe der IP-Adresse,

- *Feste IP-Adresse verwenden*: Ist kein DHCP-Server im Netzwerk installiert oder soll die IP-Adresse immer gleich sein, so kann sie auch fest vergeben werden. Bei der Vergabe von festen IP-Adressen ist darauf zu achten, dass jedes Gerät eine unterschiedliche IP-Adresse erhält.

Hinweis
Die Routing Multicast Adresse wird hier nur angezeigt. Zur Einstellung der Routing Multicast Adresse siehe Routing Multicast Adresse,.

Hinweis
Die MAC-Adresse wird nach einem Download aus dem Gerät ausgelesen. Zusätzlich ist die MAC-Adresse auf dem Gerät aufgebracht und kann alternativ über das IP-Tool ermitelt werden.

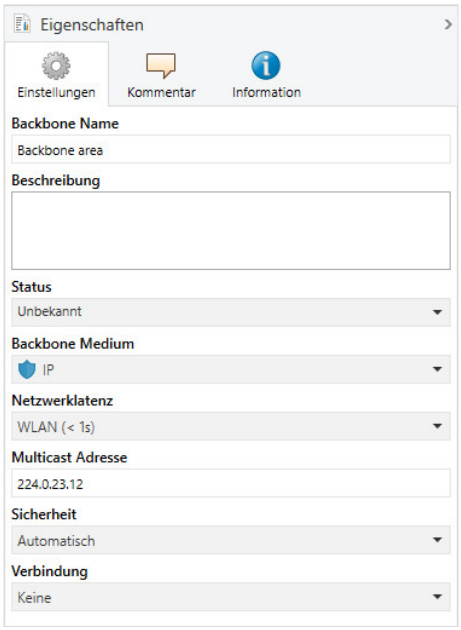
Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

Routing Multicast Adresse
(Standard = 224.0.23.12)

Optionen: 224.0.23.12

Die Routing Multicast Adresse legt die Zieladresse der IP-Telegramme des IPsecure Router KNX fest. Die voreingestellte Adresse 224.0.23.12 ist die für KNXnet/IP von der KNX-Association zusammen mit der IANA festgelegte Adresse für KNX-IP-Geräte. Diese Adresse sollte so beibehalten werden und nur geändert werden, wenn durch das vorhandene Netzwerk die Notwendwendigkeit besteht, eine andere Adresse aus dem Bereich 224.0.0.0 bis 239.255.255.255 (reservierter Bereich für Multicast-Adressen) zu verwenden.

Die Einstellung der Routing Multicast Adresse erfolgt in der ETS in der Ansicht *Topologie* (Auswahl der Topologie, dann kann im Eigenschaftenfenster auf dem Reiter *Einstellungen* die Routing Multicast Adresse eingestellt werden):



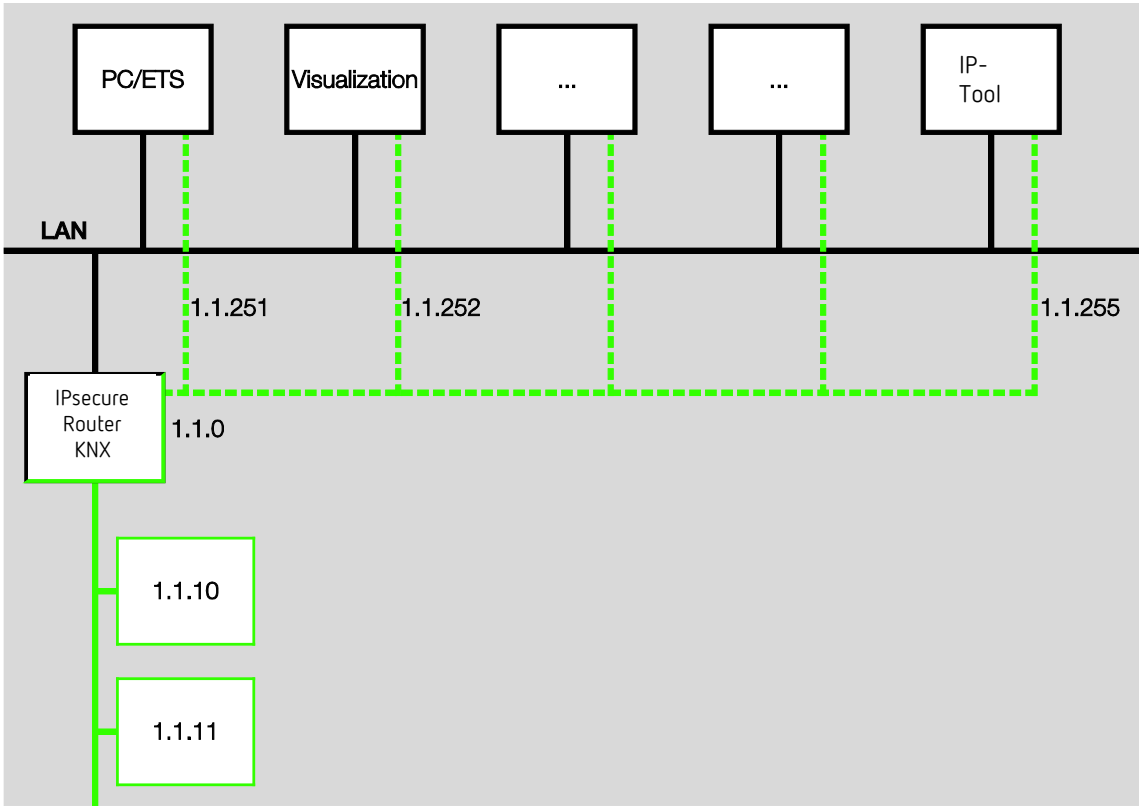
Wichtig
Alle IPsecure Router KNX oder anderen KNXnet/IP-Geräte, die Telegramme am IP-Netzwerk austauschen sollen, müssen die gleiche Routing Multicast-Adresse verwenden.

3.3 Kommunikationsobjekte

Der IPsecure Router KNX hat keine KNX-Kommunikationsobjekte.

3.4 Die Verwendung der integrierten Tunneling Server

Der IPsecure Router KNX bietet 5 zusätzliche physikalische Adressen, die für eine Tunneling-Verbindung verwendet werden können. Diese sogenannten Tunneling Server können mit der ETS als Programmierschnittstelle oder mit einem anderen Client, z.B. einer Visualisierung, verwendet werden.



Bei Tunneling verbindet sich ein Client mit einer Buslinie. Das Tunneling-Verfahren verwendet UDP, beinhaltet aber eine Sicherungsschicht, so dass im Fehlerfall Telegramme wiederholt werden. Ab ETS 5 wird Tunneling V2 unterstützt. Hier wird anstatt UDP TCP verwendet und die Sicherungsschicht des TCP für die Übertragung verwendet.

Hinweis
Die physikalische Adresse für die Tunneling-Verbindung muss in die Topologie passen. Daher müssen die Adressen aus dem Adressbereich der untergeordneten Linie gewählt werden. Bei Auslieferung haben alle Tunneling Server die Adresse 15.15.100. In der ETS 5 werden die ersten fünf freien Adressen in der Linie automatisch vergeben, nachdem der Router in eine Linie eingefügt wurde.

Die Tunneling Server können auch mit KNX Secure verschlüsselt werden. Ist der KNX Secure-Modus aktiviert, benötigt ein Client, das in der ETS vergebene Passwort.
Details siehe Kapitel, KNX Secure.

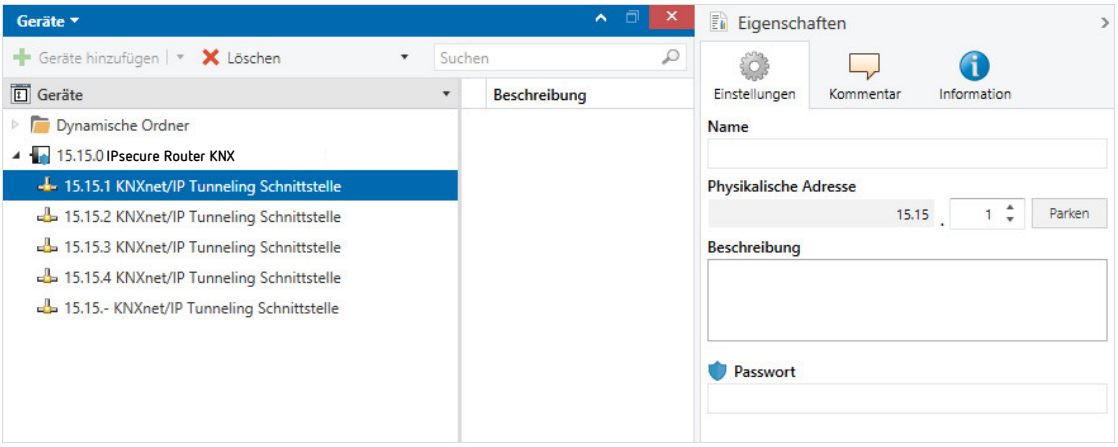
3.4.1 Einstellungen in der ETS 5

In der ETS steht für die Einstellung der zusätzlichen physikalischen Adressen ein zusätzliches Eigenschaftenfenster zur Verfügung.

Die ETS reserviert automatisch nach Einfügen des Routers in die Linie die ersten fünf freien Adressen dieser Linie für die Tunneling Server des Routers. Dies ist eine Eigenschaft der ETS und kann nicht geändert werden.

Nach dem ersten Download stehen die Adressen im Gerät zur Verfügung.

Ist dies nicht erwünscht, kann die Einstellung manuell im Eigenschaftenfenster geändert werden:



Zum Ändern der Adresse die aktuelle Geräteadresse bzw. zusätzliche Adresse markieren und mit den Pfeiltasten nach oben oder unten die gewünschte Ziffer auswählen. Durch Markieren einer anderen Adresse wird die geänderte Adresse gespeichert.

Die geänderten Adressen werden erst nach einem Download vom Gerät übernommen.

Parken

Ist für einen Tunnel die Option Parken aktiviert, so erhält dieser Tunnel die Adresse 15.15.255.

Sofern bei allen Tunneling Servern die Option Parken gewählt wird, erhalten alle Tunneling Server die Adresse 15.15.255. Damit ist nur ein Tunneling Server verfügbar.

3.5 KNX Secure

Der IPsecure Router ist ein KNX-Gerät nach dem KNX Secure-Standard. D. h. das Gerät kann sicher in Betrieb genommen werden, die Kommunikation auf dem IP-Backbone ist sicher (alle KNX-IP-Geräte müssen dazu das KNXnet/IP Security-Protokoll unterstützen) und die Tunnelingverbindungen sind verschlüsselt.

Bei der Inbetriebnahme des Geräts sind daher folgende Dinge zu berücksichtigen.

- Sobald ein KNX Secure-Gerät in ein Projekt importiert wird, muss zwingend ein Projektpasswort vergeben werden. Das Projekt ist damit gegen unbefugten Zugriff geschützt.
  Das Passwort muss sicher aufbewahrt werden – ohne dieses Passwort ist ein Zugriff auf das Projekt nicht möglich (auch nicht durch die KNX-Association)!
- Bei der Inbetriebnahme eines KNX Secure-Gerätes (erster Download) ist ein Inbetriebnahmeschlüssel erforderlich. Dieser Schlüssel (FDSK = Factory Default Setup Key) ist auf einem Aufkleber seitlich auf dem Gerät aufgebracht und muss vor dem ersten Download in die ETS importiert werden.
  - Beim ersten Download des Gerätes öffnet sich in der ETS ein Fenster, die zur Eingabe des Schlüssels auffordert. Das Zertifikat kann alternativ auch mit einem QR-Scanner eingelesen werden (empfohlen).
  - Alternativ können auch die Zertifikate aller Secure-Geräte vorab in die ETS eingegeben werden. Dies erfolgt auf der Projektübersichtsseite auf dem Reiter „Sicherheit".
  - Der FDSK-Aufkleber ist doppelt auf dem Gerät aufgebracht. Ein Teil kann für die Projektdokumentation verwendet werden, der andere kann auf dem Gerät verbleiben.
    Ohne den FDSK kann das Gerät nach einem Reset nicht mehr im KNX Secure-Modus betrieben werden!

Der FDSK wird nur für die Erstinbetriebnahme benötigt. Danach vergibt die ETS neue Schlüssel.
Der FDSK wird erst wieder benötigt, wenn das Gerät auf Werkseinstellungen zurückgesetzt wurde (z. B. wenn das Gerät in einer anderen Anlage mit einem anderen ETS Projekt verwendet werden soll).

Die ETS vergibt an alle KNX IP Secure-Geräte im Projekt den „Backbone" Schlüssel und erzeugt auch für jeden Tunneling Server separate Passwörter. Die Passwörter können bei Bedarf geändert werden.
Die Schlüssel werden von der ETS erzeugt und verwaltet. Bei Bedarf können Schlüssel und Passwörter exportiert werden (z. B. falls ein Client auf einen der Tunnel zugreifen möchte).

Sofern erforderlich, kann der Router auf Werkseinstellungen zurückgesetzt werden, Siehe Kapitel 2.4. Entladen des Gerätes und Rücksetzen auf Werkseinstellungen.

4 Planung und Anwendung

4.1 Der IPsecure Router KNX im Netzwerk

Der IPsecure Router KNX ist für den Einsatz in 10/100-BaseT-Netzwerken nach IEEE 802.3 ausgelegt. Das Gerät besitzt eine AutoSensing-Funktion und stellt die Übertragungsgeschwindigkeit (10 oder 100 MBit) automatisch ein.

4.1.1 Vergabe der IP-Adresse

DHCP/AutoIP

Die IP-Adresse des Geräts kann von einem DHCP-Server bezogen werden. Dazu ist die Einstellung einer automatischen Vergabe der IP-Adresse in der ETS nötig, siehe IP-Einstellungen. Wird bei dieser Einstellung kein DHCP-Server gefunden, startet das Gerät eine AutoIP-Prozedur und vergibt sich selbständig eine IP-Adresse aus dem Bereich 169.254.xxx.yyy.

Die IP-Adresse, die das Gerät beim Starten erhält (per DHCP oder AutoIP), wird beibehalten bis

- zum nächsten Neustart (Aus-/Einschalten oder Neuprogrammierung)
- ein DHCP-Server verfügbar ist
- zum Ablauf des des DHCP Lease

Beim Starten ist kein DHPC-Server vorhanden

Sollte beim Starten des IP-Router Secures kein DHCP-Server vorhanden sein, vergibt sich das Gerät selbst eine AutoIP-Adresse. Der Router sucht dann zyklisch (drei Telegramme im Abstand von 3 Sekunden, anschließend 20 Sekunden Pause) nach einem DHCP-Server. Sobald wieder ein Server vorhanden ist, wird die vom DHCP-Server zugeteilte Adresse verwendet.

DHCP-Server fällt aus (Gerät hat IP-Adresse bereits von DHCP bezogen)

Bis zum Ende der Lease-Zeit (Gültigkeitsdauer der IP-Adresse, wird bei der Vergabe der IP-Adresse vom DHCP-Server festgelegt) laufen die Anfragen zur Verlängerung der Nutzungsrechte dieser IP-Adresse ins Leere. Die IP-Adresse wird weiter verwendet.

Am Ende der Lease-Zeit oder nach einem Download suchen sich die Geräte eine AutoIP-Adresse.

Feste IP-Adresse

Soll die IP-Adresse des IPsecure Router KNX fest zugeordnet sein, so kann in der ETS eine feste IP-Adresse (sowie Subnet-Maske und Standard Gateway) eingestellt werden, siehe Parameterfenster IP-Einstellungen.

4.1.2 KNX-Telegramme im Netzwerk (Routing)

Hinweis
Bei der Auslegung des KNX-Systems ist zu beachten, dass die Anzahl der übertragenen Telegramme auch beim Einsatz des IPsecure Router KNX s begrenzt ist. Durch die hohe Übertragungsrate auf IP-Seite (10/100 MBit/s) können bei hohem Datenaufkommen systembedingt in der TP1-Linie (9,6kBit/s) Telegramme verloren gehen.

Multicast

Multicast bezeichnet die Kommunikation eines Senders mit einer Gruppe von Empfängern.

Der IPsecure Router sendet die KNX-Telegramme verpackt als UDP/IP-Telegramme auf das IP-Netzwerk und alle IPsecure Router, bei denen die gleiche Multicast-Adresse parametriert ist, empfangen dieses Telegramm und werten es aus. Sofern ein Telegramm für die entsprechende Sublinie bestimmt ist, routet der IPsecure Router das Telegramm in die Linie, ansonsten wird es verworfen.

Der IPsecure Router sendet Telegramme von KNX auf das IP-Netzwerk gemäß KNXnet/IP-Protokollspezifikation. Diese Telegramme werden in der Standardeinstellung als Multicast-Telegramme auf die Multicast-IP-Adresse 224.0.23.12 Port 3671 gesendet. Diese Multicast-IP-Adresse ist die für KNXnet/IP von der KNX Association zusammen mit der IANA festgelegte Adresse für KNX IP-Geräte. Diese Adresse sollte so beibehalten und nur geändert werden, wenn durch das vorhandene Netzwerk die Notwendwendigkeit besteht, eine andere Adresse zu verwenden.

Damit mehrere IPsecure Router im Netzwerk miteinander kommunizieren können, muss zwischen den Geräten eine Multicast-Kommunikation möglich sein. Je nach Art des Netzwerks und der Einstellung der verwendeten Netzwerkkomponenten, z.B. Router, Switch oder Firewall, muss die Multicast-IP-Adresse 224.0.23.12 eventuell erst noch explizit freigeschaltet werden. Bitte sprechen Sie dazu mit dem Netzwerkadministrator.

Weitere Informationen siehe:
Kap. 3.2.3: Parameterfenster IP-Einstellungen

Unicast

Unicast bezeichnet allgemein die Kommunikation zwischen einem Sender und einem Empfänger. Der Router stellt also zu jedem IP-Router innerhalb der Unicast-Gruppe eine Kommunikationsverbindung her.

Falls in einem Netzwerk keine Multicast-Kommunikation möglich ist, können die IP-Router auch über Unicast miteinander kommunizieren. Bis zu zehn IP-Router können zu einer Unicast-Gruppe zusammengefasst werden. Die Unicast-Gruppe kann aus IP Router KNX und/oder IPsecure Router KNX bestehen, auch gemischt. Jedem Router werden dann neun IP-Adressen zugewiesen, an die er seine Telegramme versendet.

Die Konfiguration dieser Unicast-Gruppe erfolgt einfach und automatisch mit dem IP- Tool.

Es ist auch möglich, einen Client (z. B. eine Visualisierung) mit dieser Unicast-Gruppe zu verknüpfen. In diesem Fall ist eine der zehn Unicast-Adressen vom Client belegt, und es können noch neun IPsecure Router verknüpft werden.

Die genaue Beschreibung, wie die Konfiguration mit dem IP-Tool funktioniert, ist in der Hilfe des IP-Tools zu finden (siehe Kap. Das Theben IP-Tool).

Hinweis
Sobald in der ETS unter IP-Kommunikationsart der Parameter auf *Unicast* umgestellt wird, ist die Funktion *Multicast* deaktiviert. Die Geräte können dann nicht mehr über Multicast Routing, sondern nur noch über einen der integrierten Tunneling Server oder eine separate Programmierschnittstelle programmiert werden. Allerdings wird die Kommunikationsart erst umgestellt, wenn die Parametrierung mit dem i-bus® Tool erfolgt ist. Bis dahin steht der Router intern noch auf Multicast. Dies hat den Vorteil, dass der Router über Multicast programmiert werden kann.

Weitere Informationen siehe:
Parameterfenster IP-Einstellungen

Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

Hinweis
• Bei der Verwendung der Kommunikationsart Unicast muss sicher gestellt sein, dass sich die IP-Adresse des Routers im laufenden Betrieb nicht ändert. Dazu sollte entweder eine feste IP-Adresse vergeben werden, oder eine entsprechende Einstellung beim DHCP-Server erfolgen. • Durch die ETS werden bei Änderung der physikalischen Adresse alle IP-Parameter ebenfalls aktualisiert. D.h. auch wenn nur die Option Programmieren physikalische Adresse in der ETS ausgewählt wird, werden der Gerätename, die Multicast-Adresse, IP-Kommunikationsart (DHCP, AutoIP, fest), IP-Adresse, Subnetzmaske, Standard-Gateway und alle Tunneling-Adressen neu geladen. Sofern sich die IP-Adresse dabei ändert, muss die Unicast-Konfiguration mit dem IP-Tool erneut durchgeführt werden.

Die Kommunikationsart *Unicast* kann nicht verwendet werden, wenn das Gerät im KNX Secure-Modus betrieben wird. Wird bei aktiviertem KNX Secure-Modus *Unicast* ausgewählt, schaltet die ETS auf *Multicast* um. Die Parametrierung *Unicast* in der Applikation wird dann ignoriert.

Um die Kommunikationsart *Unicast* zu verwenden, muss der KNX Secure-Modus in der ETS abgeschaltet werden.

4.1.3 Überwachung

Die Überwachung einer Tunnelingverbindung sollte bei einer aktiven Tunneling Verbindung über ein „CONNECTIONSTATE_REQUEST" erfolgen, ist aber auch per T-Connect möglich. Die Überwachung eines Gerätes über T-Connect kann Nachteile haben, z. B. bei Überwachungs-, Programmier- oder Scanvorgängen in der Linie.

System Broadcast

In einer KNX-Anlage müssen alle IP-Geräte, die miteinander kommunizieren wollen, die gleiche Multicast-Adresse nutzen. Standardmäßig wird die Adresse 224.0.23.12 Port 3671 verwendet, siehe Multicast.

Wird in einer Anlage die Multicastadresse geändert, kann es bei der Inbetriebnahme u. U. zu Problemen kommen. Wird z. B. zuerst die Multicastadresse des nächstgelegenen Routers geändert, schaltet dieser nach der Programmierung auf die neue Mulicast-Adresse um. Der Rest der Anlage ist für ihn dann nicht mehr erreichbar und die übrigen Router in der Anlage können nicht mehr programmiert werden.

Diese Geräte kann die ETS dann über die „System Broadcast Adresse" erreichen. Über die System Broadcast Adresse kann die Multicast-Adresse aller KNX IP-Geräte und auch der Backbone Schlüssel geändert werden. Dies funktioniert nur, wenn das Gerät entweder nicht im KNX Secure-Modus betrieben wird oder der ETS der Backbone Schlüssel bekannt ist.

Das Weiterleiten von System Broadcast Telegramme kann über den Parameter *Broadcast-Telegramme weiterleiten/sperren* eingestellt werden, d.h. dieser Parameter ist für (Standard) Broadcast-Telegramme und System Broadcast-Telegramme wirksam.

4.1.4 IGMP

Das Gerät unterstützt IGMP snooping in der Version V3.

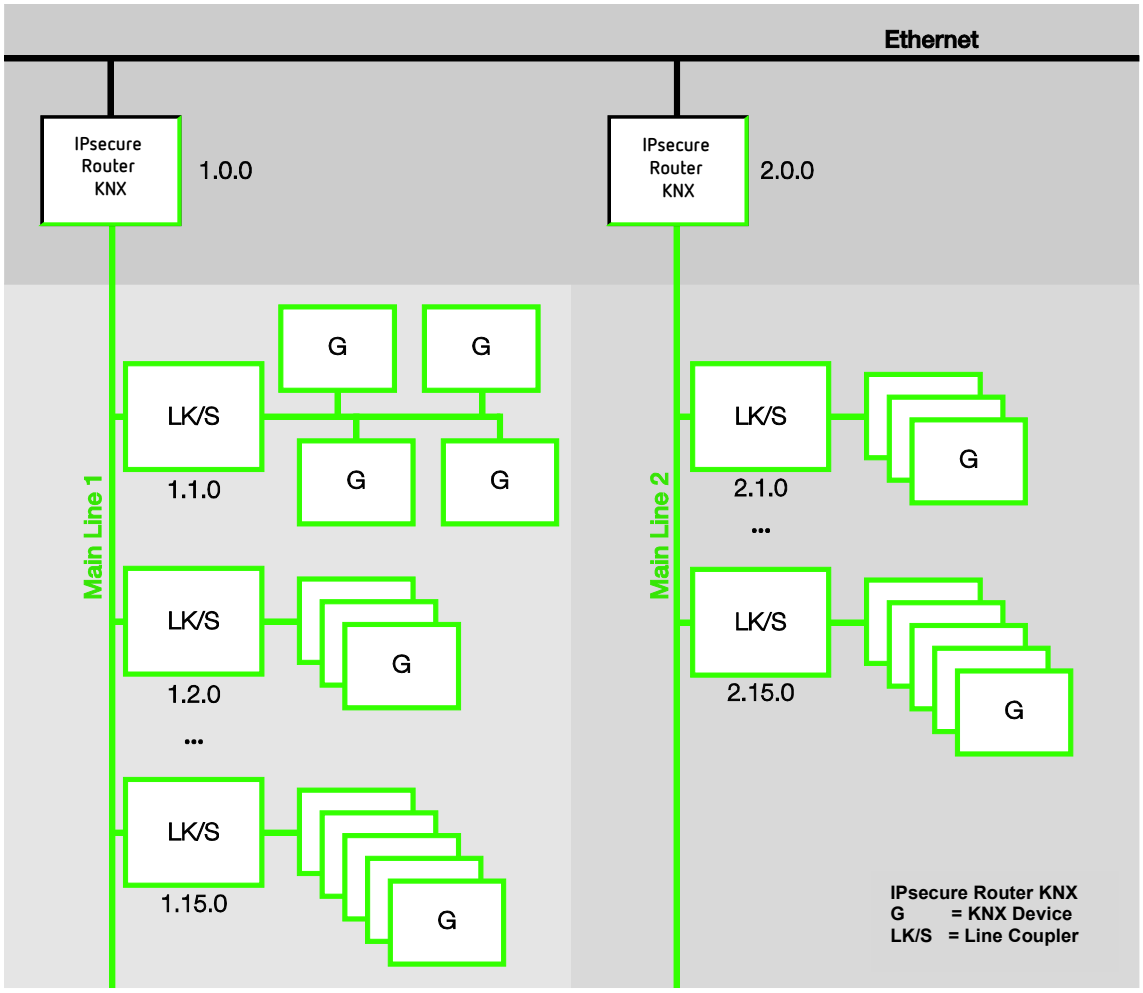
IGMP snooping ist die Fähigkeit Multicast Routing Verkehr nur dorthin weiterzuleiten, wo er auch benötigt wird. Die IT-Infrastruktur und das Gerät müssen die gleiche IGMP-Version benutzen, sonst funktioniert der IGMP-Mechanismus nicht.

Zum Freischalten einer Multicast Adresse meldet sich das Gerät mit einem Membership Report bei dieser Multicast Adresse an.

4.1.5 IPsecure Router KNX als Bereichskoppler

Der IPsecure Router KNX kann in KNX-Anlagen die Funktion eines Bereichskopplers übernehmen. Dafür muss er die physikalische Adresse eines Bereichskopplers (1.0.0...15.0.0) erhalten. In einem ETS-Projekt können bis zu 15 Bereiche mit Bereichskopplern angelegt werden.

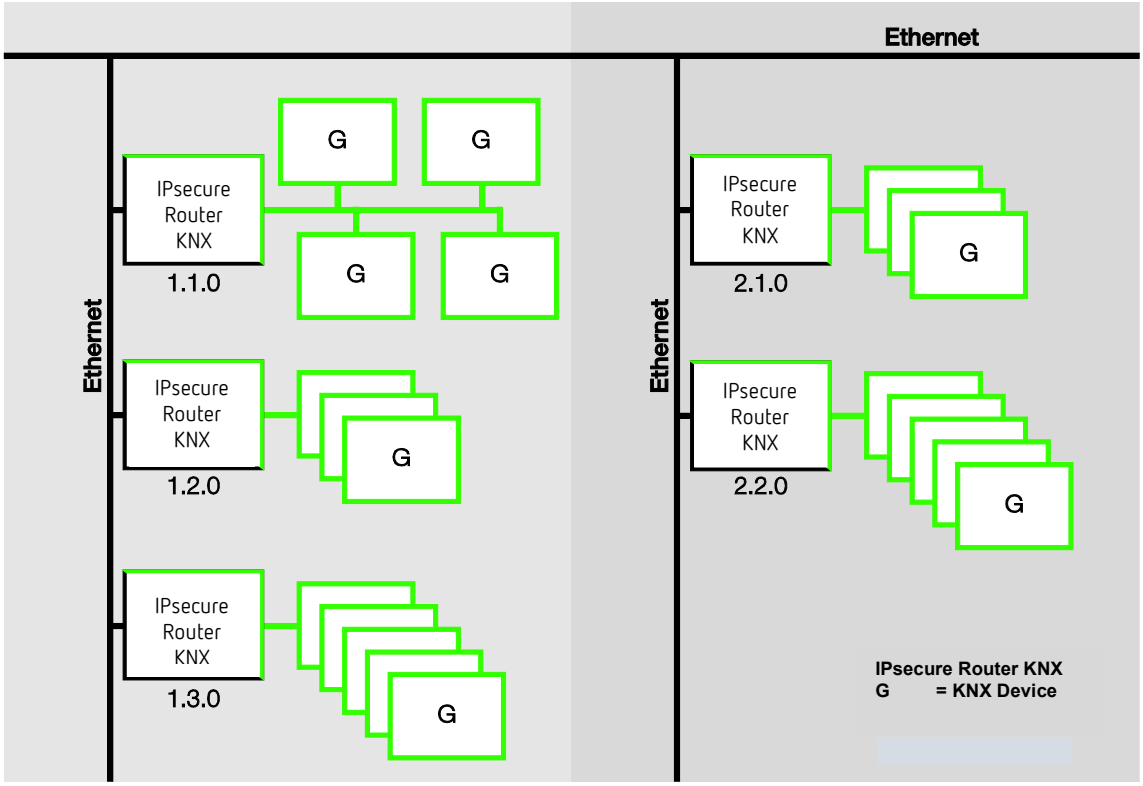
Das folgende Bild zeigt diese Topologie mit IPsecure Router KNX n als Bereichskoppler und KNX-Linienkoppler (LK/S).



4.1.6 IPsecure Router KNX als Linienkoppler

Der IPsecure Router KNX kann in KNX-Anlagen die Funktion eines Linienkopplers übernehmen. Dafür muss er die physikalische Adresse eines Linienkopplers (1.1.0...15.15.0) erhalten.

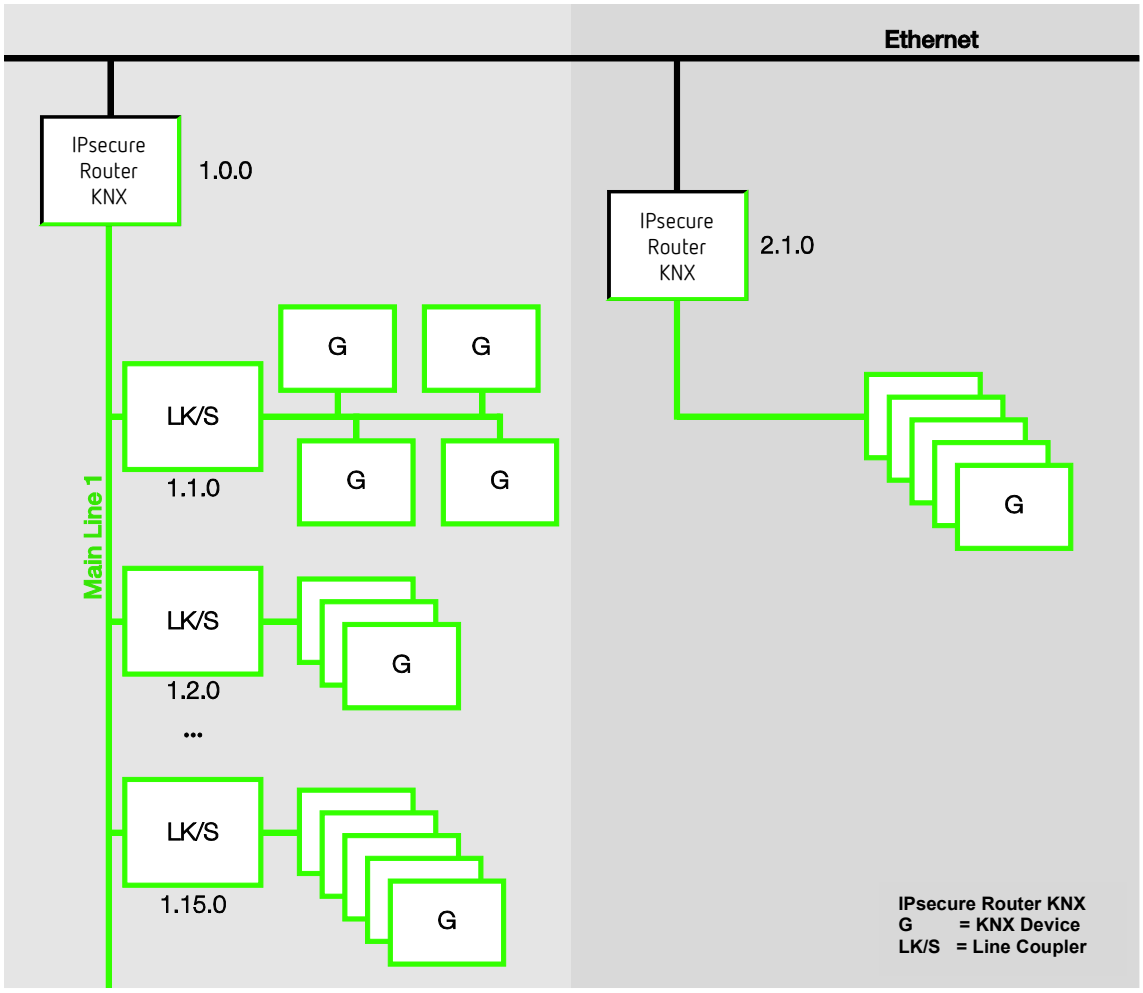
Das folgende Bild zeigt diese Topologie mit IPsecure Router KNX als Linienkoppler.



4.1.7 Gemischte Topologie

Ist es innerhalb einer KNX-.Anlage nötig, den IPsecure Router KNX an einer Stelle, z. B. Bürohaus, als Bereichskoppler und an anderer Stelle, z. B. entfernte Tiefgarage, als Linienkoppler einzusetzen, so ist dies möglich.

Dabei muss nur beachtet werden, dass der IPsecure Router KNX als Linienkoppler die Linienkoppleradresse aus einem freien Bereich verwendet, z. B. hier im Bild.

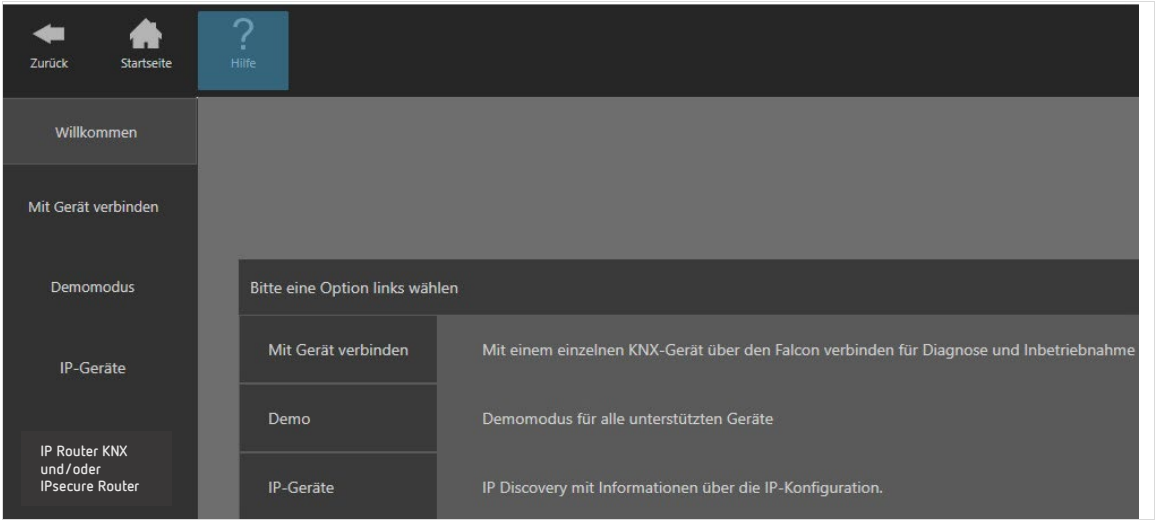


4.2 Das Theben IP-Tool

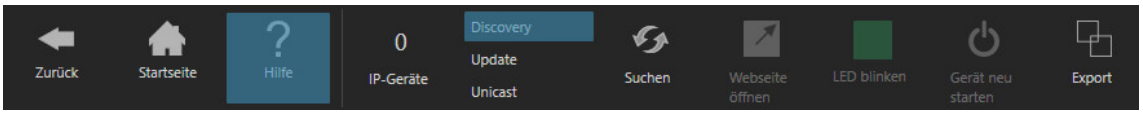
Das Theben IP-Tool wird benötigt, um bestimmte Funktionen bei den Theben IP-Geräten einzustellen.

Es erleichtert die Inbetriebnahme auf IP-Seite.

Klicken Sie auf der Startseite des IP- Tools auf *Verbinden*, in dem danach erscheinenden Fenster dann auf *IP-Geräte*.



Multifunktionsleiste: Umschalten zwischen Discovery, Firmware Update und Unicast



Klicken Sie auf die entsprechende Schaltfläche, um den Modus *Discovery*, *Update* oder *Unicast* auszuwählen.

Discovery

Wählen Sie in der Multifunktionsleiste den Modus *Discovery*. Diese Funktion dient zum Auffinden und Anzeigen von IP-Geräten im Netzwerk. Wird der IP-Router im KNX Secure-Modus betrieben, werden nicht alle Informationen angezeigt.

Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

Firmware Update

Im KNX Secure-Modus kann das Gerät nicht mit dem IP- Tool aktualisiert werden.
Das Firmwareupdate kann in diesem Fall nur mit der ETS App „KNX Bus Update" erfolgen,
die kostenlos im KNX Online Shop geladen werden kann.

Wählen Sie in der Multifunktionsleiste den Modus *Update*. Sofern es einmal notwendig sein sollte, kann mit dieser Funktion die Firmware aktualisiert werden.

Wichtig
Die Firmware muss vorab aus dem Internet geladen werden (*www.theben.de/downloads)*. Dazu verbindet sich das IP-Tool **bei bestehender Internetverbindung** mit einem Server. Für die Aktualisierung der Geräte auf der Anlage ist dann keine Internetverbindung mehr notwendig.

Wichtig
Während des Updatevorgangs muss zusätzlich zum IP-Netzwerk (LAN) auch der KNX-Bus (TP) angeschlossen sein, damit die KNX-Parameter korrekt wiederhergestellt werden können. Andernfalls schlägt der Updatevorgang fehl. Es muss sichergestellt werden, dass während des Updatevorgangs kein Spannungsausfall (KNX oder IP) auftritt, da ansonsten das Gerät zerstört werden kann.

Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

Hinweis
Für den Updatevorgang muss das IP-Tool mit Administratorrechten ausgeführt werden.

Unicast

Wählen Sie in der Multifunktionsleiste den Modus *Unicast*.

Diese Funktion ist nur verfügbar, wenn vorher in der ETS-Applikation der Parameter IP-Kommunikationsart auf *Unicast* umgestellt worden ist. Parametrierung, siehe Parameterfenster IP-Kommunikationsart

Hinweis
Eine Beschreibung der Funktionen ist in der Onlinehilfe des IP-Tools zu finden.

IPsecure Router KNX
Third party software components, notices and OSS license texts

A Anhang

Third party software components, notices and OSS license texts

Inhaltsverzeichnis

IPsecure Router KNX
Third party software components, notices and OSS license texts

Software components structured by applicable license:

This document contains Open Source Software ("OSS") subject to OSS license terms as follows:

The software contains **open source software components** as follows:

under General Public Licence (GPL):

- buildroot (v2012-05)

- memstat (v0.8)

under General Public License (GPL) Version 2:

- abbpower (v1.0)

- at91ft12 (v1.0)

- at91gpbr (v1.0)

- buildroot (v2012-05)

- busybox (v1.20.1)

- gdbserver (v7.2.50.20100908-cvs)

- glibc (v2.11.1)

- kmod (v8)

- libgcc (v4.5.1)

- Linux (v3.2.26)

- mtd-utils (v1.4.9)

- U-Boot (v2010.09)

- udev (v058)

- util Linux (v2.20.1)

under General Public License (GPL) Version 3:

- binutils (v2.21)

- gdbserver (v7.2.50.20100908-cvs)

- gzip (v1.5)

- libgcc (v4.5.1)

- tar (v1.17)

- util Linux (v2.20.1)

under Library General Public License (LGPL) Version 2:

- binutils (v2.21)

- gdbserver (v7.2.50.20100908-cvs)

- glib (libglib2) (v2.30.2)

- util Linux (v2.20.1)

under Lesser General Public License (LGPL) Version 2.1:

- glibc (v2.11.1)

- libgcc (v4.5.1)

- libkmod (v8)

- uclibc (v0.9.31.1)

- XML-RPC++ (v0.7)

under Lesser General Public License (LGPL) Version 3:

- binutils (v2.21)

- gdbserver (v7.2.50.20100908-cvs)

- gmplib (v5.0.4)

- libgcc (v4.5.1)

- uclibc (v0.9.31.1)

under GNU FDLv1.3

- popt (v1.16)

under GNU GCC Runtime Library Exception

- libgcc (v4.5.1)

under BSD and BSD/MIT style licenses:

- AT91Bootstrap 1.9 (v1.9)

- dropbear sshd (v2012.55)

- libpcap (v1.2.1)

- libxml2 (v2.7.8)

- lsof (v4.85)

- OpenSSL (v1.0.1g)

- Pcre (v8.30)

- util Linux (v2.20.1)

- zlib (v1.2.6)

under MIT licenses:

- expat (v2.1.0)

under Public Domain licenses:

- util Linux (2.20.1)

under ncurses license:

- ncurses (v5.7)

under libffi license

- libffi (v3.0.11)

under jpeg license

- libjpeg (v9a)

under CodeSourcery G++ Lite license

- codeSourcery G++ Lite (v2010.09-50)

under ROM-Bootloader license

- ROM-Bootloader

Copyright notices:

AT91 Bootstrap (v1.9):

* Copyright (c) 2008, Atmel Corporation

*

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions are met:

*

* - Redistributions of source code must retain the above copyright notice,

* this list of conditions and the disclaimer below.

*

* Atmel's name may not be used to endorse or promote products derived from

* this software without specific prior written permission.

*

* DISCLAIMER: THIS SOFTWARE IS PROVIDED BY ATMEL "AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF

* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT ARE

* DISCLAIMED. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT

* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA,

* OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF

* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING

* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,

* EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Buildroot (v2012-05):

Buildroot is a simple, efficient and easy-to-use tool to generate embedded

Linux systems through cross-compilation.

The documentation can be found in docs/manual. You can generate a text

document with 'make manual-text' and read output/docs/manual/manual.text.

Online documentation can be found at http://buildroot.org/docs.html

To build and use the buildroot stuff, do the following:

1) run 'make menuconfig'

2) select the target architecture and the packages you wish to compile

3) run 'make'

4) wait while it compiles

5) find the kernel, bootloader, root filesystem, etc. in output/images

You do not need to be root to build or run buildroot. Have fun!

Buildroot comes with a basic configuration for a number of boards. Run

'make list-defconfigs' to view the list of provided configurations.

Please feed suggestions, bug reports, insults, and bribes back to the

buildroot mailing list: buildroot@buildroot.org

You can also find us on #buildroot on Freenode IRC.

IPsecure Router KNX
Third party software components, notices and OSS license texts

Busybox (v1.20.1):

--- A note on GPL versions

BusyBox is distributed under version 2 of the General Public License (included in its entirety, below). Version 2 is the only version of this license which this version of BusyBox (or modified versions derived from this one) may be distributed under.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software

Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an

announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a

special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this

when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it

under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate

parts of the General Public License. Of course, the commands you use may

be called something other than `show w' and `show c'; they could even be

mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your

school, if any, to sign a "copyright disclaimer" for the program, if

necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program

`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CodeSourcery G++ Lite (v2010.09-50):

CODESOURCERY, INC. ("CODESOURCERY") IS WILLING TO LICENSE THE SOFTWARE ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS SOFTWARE LICENSE AGREEMENT. PLEASE READ THE TERMS CAREFULLY. BY CLICKING ON "I accept the terms of the License Agreement", YOU WILL INDICATE YOUR AGREEMENT WITH THEM. IF YOU ARE ENTERING INTO THIS AGREEMENT ON BEHALF OF A COMPANY OR OTHER LEGAL ENTITY, YOUR ACCEPTANCE REPRESENTS THAT YOU HAVE THE AUTHORITY TO BIND SUCH ENTITY TO THESE TERMS, IN WHICH CASE "YOU" OR "YOUR" SHALL REFER TO YOUR ENTITY. IF YOU DO NOT AGREE WITH THESE TERMS, OR IF YOU DO NOT HAVE THE AUTHORITY TO BIND YOUR ENTITY, THEN CODESOURCERY IS UNWILLING TO LICENSE THE SOFTWARE.

Sourcery G++ Software License Agreement

1. Parties.

   The parties to this Agreement are you, the licensee (You or Licensee) and CodeSourcery. If You are not acting on behalf of Yourself as an individual, then You means Your company or organization.

2. The Software

   The Software licensed under this Agreement consists of computer programs and documentation referred to as Sourcery G++ Lite Edition (the Software).

3. Definitions.

   1. CodeSourcery Proprietary Components.

      The components of the Software that are owned and/or licensed by CodeSourcery and are not subject to a free software or open source license, such as the GNU Public License. The CodeSourcery Proprietary Components of the Software include, without limitation, the Sourcery G++ Installer, any Sourcery G++ Eclipse plug-ins, the CodeSourcery C Library (CSLIBC), and any Sourcery G++ Debug Sprite. For a complete list, refer to the Getting Started Guide included with the distribution.

   2. Open Source Software Components.

      The components of the Software that are subject to a free software or open source license, such as the GNU Public License.

3. Proprietary Rights.

All rights in and to copyrights, rights to register copyrights, trade secrets, inventions, patents, patent rights, trademarks, trademark rights, confidential and proprietary information protected under contract or otherwise under law, and other similar rights or interests in intellectual or industrial property.

4. Redistributable Components.

The CodeSourcery Proprietary Components that are intended to be incorporated or linked into Licensee object code developed with the Software. The Redistributable Components of the Software include, without limitation, CSLIBC and the CodeSourcery Common Startup Code Sequence (CS3). For a complete list, refer to the Getting Started Guide included with the distribution.

4. License Grant to Proprietary Components of the Software.

You are granted a non-exclusive, royalty-free license (a) to install and use the CodeSourcery Proprietary Components of the Software, (b) to transmit the CodeSourcery Proprietary Components over an internal computer network, (c) to copy the CodeSourcery Proprietary Components for Your internal use only, and (d) to distribute the Redistributable Component(s) in binary form only and only as part of Licensee object code developed with the Software that provides substantially

different functionality than the Redistributable Component(s).

5. Restrictions.

   You may not: (i) copy or permit others to use the CodeSourcery Proprietary Components of the Software, except as expressly provided above; (ii) distribute the CodeSourcery Proprietary Components of the Software to any third party, except as expressly provided above; or (iii) reverse engineer, decompile, or disassemble the CodeSourcery Proprietary Components of the Software, except to the extent this restriction is expressly prohibited by applicable law.

   1. ARM Keil ULINK2 Drivers.

      Sourcery G++ may include ULINK2 drivers from ARM, Ltd. If these drivers are included, the following additional terms and conditions apply:

      1. You may use the ULINK2 drivers only in conjunction with a compatible ARM Keil ULINK2 hardware unit manufactured by or under license from ARM and purchased from CodeSourcery, ARM, or a distributor authorized by ARM.
      2. You may use the ULINK2 drivers only to connect to the GNU Debugger included in Sourcery G++.
      3. The ULINK2 drivers are not supported by ARM, Ltd.; You should contact CodeSourcery for any support regarding the ULINK2 drivers.

4. You may not redistribute or transfer the ULINK2 drivers.
5. You may not translate, adapt, arrange or otherwise alter the object code of the ULINK2 drivers (including without limitation copying, adapting or reverse compiling the object code of the ULINK2 drivers for the purpose of error correction) except as allowed by applicable law.
6. You may not remove or obstruct any notice or marker incorporated into the ULINK2 drivers to protect ARM's or third parties' intellectual property or Proprietary Rights.
7. The ULINK2 drivers are licensed, not sold; all right, title and interest therein is reserved to CodeSourcery or its licensors, and You acquire no right, title or interest therein.

2. SEGGER J-Link Devices.

Sourcery G++ Lite includes proprietary software from SEGGER Microcontroller GmbH & Co.KG that allows the use of SEGGER J-Link debug devices with the Sourcery G++ Lite Debug Sprite. You may use software from SEGGER only under the SEGGER J-Link software terms of use and license agreement.

3. Sourcery G++ Debug Sprite with ARM SWD.

The Sourcery G++ Debug Sprite for ARM and Stellaris processors includes software for ARM SWD support

from ARM, Ltd. You may use the ARM SWD software only in conjunction with a Cortex-M1 or Cortex-M3 microprocessor manufactured under license from ARM.

6. Free Software or Open Source License to Certain Components of the Software

This Agreement does not limit Your rights under, or grant You rights that supersede, the license terms of any Open Source Software Component delivered to You by CodeSourcery. Sourcery G++ includes components provided under various different licenses. The Getting Started Guide provides an overview of which license applies to different components, and, for components subject to the Eclipse Public License, contains information on how to obtain the source code. Definitive licensing information for each free software or open source component is available in the relevant source file.

7. CodeSourcery Trademarks.

Notwithstanding any provision in a free software or open source license agreement applicable to a component of the Software that permits You to distribute such component to a third party in source or binary form, You may not use any CodeSourcery trademark, whether registered or unregistered, including without limitation, CodeSourcery, Sourcery G++, the CodeSourcery crystal ball logo, or the Sourcery G++ splash screen, or any confusingly similar mark, in connection with such distribution, and You may not recompile the Open Source Software Components with

the --with-pkgversion or --with-bugurl configuration options that embed CodeSourcery trademarks in the resulting binary.

8. Term and Termination.

   This Agreement shall remain in effect unless terminated pursuant to this provision. CodeSourcery may terminate this Agreement upon seven (7) days written notice of a material breach of this Agreement if such breach is not cured; provided that the unauthorized use, copying, or distribution of the CodeSourcery Proprietary Components of the Software will be deemed a material breach that cannot be cured.

9. Transfers.

   You may not transfer any rights under this Agreement without the prior written consent of CodeSourcery, which consent shall not be unreasonably withheld. A condition to any transfer or assignment shall be that the recipient agrees to the terms of this Agreement. Any attempted transfer or assignment in violation of this provision shall be null and void.

10. Ownership.

    CodeSourcery owns and/or has licensed the CodeSourcery Proprietary Components of the Software and all intellectual property rights embodied therein, including copyrights and valuable trade secrets embodied in its design and coding methodology. The CodeSourcery Proprietary Components of the Software are protected by

United States copyright laws and international treaty provisions. CodeSourcery also owns all rights, title and interest in and with respect to its trade names, domain names, trade dress, logos, trademarks, service marks, and other similar rights or interests in intellectual property. This Agreement provides You only a limited use license, and no ownership of any intellectual property.

11. Warranty Disclaimer; Limitation of Liability.

CODESOURCERY AND ITS LICENSORS PROVIDE THE SOFTWARE AS-IS AND PROVIDED WITH ALL FAULTS. CODESOURCERY DOES NOT MAKE ANY WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. CODESOURCERY SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, SYSTEM INTEGRATION, AND DATA ACCURACY. THERE IS NO WARRANTY OR GUARANTEE THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED, ERROR-FREE, OR VIRUS-FREE, OR THAT THE SOFTWARE WILL MEET ANY PARTICULAR CRITERIA OF PERFORMANCE, QUALITY, ACCURACY, PURPOSE, OR NEED. YOU ASSUME THE ENTIRE RISK OF SELECTION, INSTALLATION, AND USE OF THE SOFTWARE. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS AGREEMENT. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

12. Local Law.

If implied warranties may not be disclaimed under applicable law, then ANY IMPLIED WARRANTIES ARE LIMITED IN DURATION TO THE PERIOD REQUIRED BY APPLICABLE LAW.

13. Limitation of Liability.

INDEPENDENT OF THE FORGOING PROVISIONS, IN NO EVENT AND UNDER NO LEGAL THEORY, INCLUDING WITHOUT LIMITATION, TORT, CONTRACT, OR STRICT PRODUCTS LIABILITY, SHALL CODESOURCERY BE LIABLE TO YOU OR ANY OTHER PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND, INCLUDING WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER MALFUNCTION, OR ANY OTHER KIND OF COMMERCIAL DAMAGE, EVEN IF CODESOURCERY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY TO THE EXTENT PROHIBITED BY APPLICABLE LAW. IN NO EVENT SHALL CODESOURCERY'S LIABILITY FOR ACTUAL DAMAGES FOR ANY CAUSE WHATSOEVER, AND REGARDLESS OF THE FORM OF ACTION, EXCEED THE AMOUNT PAID BY YOU IN FEES UNDER THIS AGREEMENT DURING THE PREVIOUS ONE YEAR PERIOD.

14. Export Controls.

You agree to comply with all export laws and restrictions and regulations of the United States or foreign agencies or authorities, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. As applicable, each party shall obtain and bear all expenses relating to any necessary licenses and/or exemptions with respect to its own export of the Software from the U.S. Neither the Software nor

the underlying information or technology may be electronically transmitted or otherwise exported or re-exported (i) into Cuba, Iran, Iraq, Libya, North Korea, Sudan, Syria or any other country subject to U.S. trade sanctions covering the Software, to individuals or entities controlled by such countries, or to nationals or residents of such countries other than nationals who are lawfully admitted permanent residents of countries not subject to such sanctions; or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals and Blocked Persons or the U.S. Commerce Department's Table of Denial Orders. By downloading or using the Software, Licensee agrees to the foregoing and represents and warrants that it complies with these conditions.

15. U.S. Government End-Users.

The Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software and commercial computer software documentation, as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire the Software with only those rights set forth herein.

16. Licensee Outside The U.S.

If You are located outside the U.S., then the following provisions shall apply: (i) Les parties aux presentes confirment leur volonte que cette convention de meme que tous les documents y compris tout avis qui siy rattache, soient rediges en langue anglaise (translation: The parties confirm that this Agreement and all related documentation is and will be in the English language.); and (ii) You are responsible for complying with any local laws in your jurisdiction which might impact your right to import, export or use the Software, and You represent that You have complied with any regulations or registration procedures required by applicable law to make this license enforceable.

17. Severability.

If any provision of this Agreement is declared invalid or unenforceable, such provision shall be deemed modified to the extent necessary and possible to render it valid and enforceable. In any event, the unenforceability or invalidity of any provision shall not affect any other provision of this Agreement, and this Agreement shall continue in full force and effect, and be construed and enforced, as if such provision had not been included, or had been modified as above provided, as the case may be.

18. Arbitration.

Except for actions to protect intellectual property rights and to enforce an arbitrator's decision hereunder, all disputes, controversies, or claims arising out of or relating to this Agreement or a breach thereof shall be submitted to and finally resolved by arbitration under the rules of the American Arbitration Association (AAA) then in effect. There shall be one arbitrator, and such arbitrator shall be chosen by mutual agreement of the parties in accordance with AAA rules. The arbitration shall take place in Granite Bay, California, and may be conducted by telephone or online. The arbitrator shall apply the laws of the State of California, USA to all issues in dispute. The controversy or claim shall be arbitrated on an individual basis, and shall not be consolidated in any arbitration with any claim or controversy of any other party. The findings of the arbitrator shall be final and binding on the parties, and may be entered in any court of competent jurisdiction for enforcement. Enforcements of any award or judgment shall be governed by the United Nations Convention on the Recognition and Enforcement of Foreign Arbitral Awards. Should either party file an action contrary to this provision, the other party may recover attorney's fees and costs up to $1000.00.

19. Jurisdiction And Venue.

The courts of Placer County in the State of California, USA and the nearest U.S. District Court shall be the exclusive jurisdiction and venue for all legal proceedings that are not arbitrated under this Agreement.

20. Independent Contractors.

The relationship of the parties is that of independent contractor, and nothing herein shall be construed to create a partnership, joint venture, franchise, employment, or agency relationship between the parties. Licensee shall have no authority to enter into agreements of any kind on behalf of CodeSourcery and shall not have the power or authority to bind or obligate CodeSourcery in any manner to any third party.

21. Force Majeure.

Neither CodeSourcery nor Licensee shall be liable for damages for any delay or failure of delivery arising out of causes beyond their reasonable control and without their fault or negligence, including, but not limited to, Acts of God, acts of civil or military authority, fires, riots, wars, embargoes, or communications failure.

22. Miscellaneous.

This Agreement constitutes the entire understanding of the parties with respect to the subject matter of this Agreement and merges all prior communications, representations, and agreements. This Agreement may be modified only by a written agreement signed by the

parties. If any provision of this Agreement is held to be unenforceable for any reason, such provision shall be reformed only to the extent necessary to make it enforceable. This Agreement shall be construed under the laws of the State of California, USA, excluding rules regarding conflicts of law. The application of the United Nations Convention of Contracts for the International Sale of Goods is expressly excluded. This license is written in English, and English is its controlling language.

Dropbear sshd (v2012.55):

Dropbear contains a number of components from different sources, hence there are a few licenses and authors involved. All licenses are fairly non-restrictive.

The majority of code is written by Matt Johnston, under the license below.

Portions of the client-mode work are (c) 2004 Mihnea Stoenescu, under the same license:

Copyright (c) 2002-2008 Matt Johnston

Portions copyright (c) 2004 Mihnea Stoenescu

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

=====

LibTomCrypt and LibTomMath are written by Tom St Denis, and are Public Domain.

=====

sshpty.c is taken from OpenSSH 3.5p1,

Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland

All rights reserved

"As far as I am concerned, the code I have written for this software

can be used freely for any purpose. Any derived versions of this

software must be clearly marked as such, and if the derived work is

incompatible with the protocol description in the RFC file, it must be

called by a name other than "ssh" or "Secure Shell". "

=====

loginrec.c

loginrec.h

atomicio.h

atomicio.c

and strlcat() (included in util.c) are from OpenSSH 3.6.1p2, and are licensed

under the 2 point BSD license.

loginrec is written primarily by Andre Lucas, atomicio.c by Theo de Raadt.

strlcat() is (c) Todd C. Miller

=====

Import code in keyimport.c is modified from PuTTY's import.c, licensed as

follows:

PuTTY is copyright 1997-2003 Simon Tatham.

Portions copyright Robert de Bath, Joris van Rantwijk, Delian

Delchev, Andreas Schultz, Jeroen Massar, Wez Furlong, Nicolas Barry,

Justin Bradford, and CORE SDI S.A.

Permission is hereby granted, free of charge, to any person

obtaining a copy of this software and associated documentation files

(the "Software"), to deal in the Software without restriction,

including without limitation the rights to use, copy, modify, merge,

publish, distribute, sublicense, and/or sell copies of the Software,

and to permit persons to whom the Software is furnished to do so,

subject to the following conditions:

The above copyright notice and this permission notice shall be

included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

NONINFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE

FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION

WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Expat (v2.1.0):

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd

and Clark Cooper

Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006 Expat maintainers.

Permission is hereby granted, free of charge, to any person obtaining

a copy of this software and associated documentation files (the

"Software"), to deal in the Software without restriction, including

without limitation the rights to use, copy, modify, merge, publish,

distribute, sublicense, and/or sell copies of the Software, and to

permit persons to whom the Software is furnished to do so, subject to

the following conditions:

The above copyright notice and this permission notice shall be included

in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY

CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,

TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE

SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Gdbserver (v7.2.50.20100908-cvs):

README for GNU development tools

This directory contains various GNU compilers, assemblers, linkers,

debuggers, etc., plus their support routines, definitions, and documentation.

If you are receiving this as part of a GDB release, see the file gdb/README.

If with a binutils release, see binutils/README; if with a libg++ release,

see libg++/README, etc. That'll give you info about this

package -- supported targets, how to use it, how to report bugs, etc.

It is now possible to automatically configure and build a variety of

tools with one command. To build all of the tools contained herein,

run the ``configure" script here, e.g.:

```
./configure
make
```

To install them (by default in /usr/local/bin, /usr/local/lib, etc),

then do:

```
make install
```

(If the configure script can't determine your type of computer, give it

the name as an argument, for instance ``./configure sun4". You can

use the script ``config.sub" to test whether a name is recognized; if

it is, config.sub translates it to a triplet specifying CPU, vendor,

and OS.)

If you have more than one compiler on your system, it is often best to

explicitly set CC in the environment before running configure, and to

also set CC when running make. For example (assuming sh/bash/ksh):

```
CC=gcc ./configure
make
```

A similar example using csh:

```
setenv CC gcc
./configure
make
```

Much of the code and documentation enclosed is copyright by

the Free Software Foundation, Inc. See the file COPYING or

COPYING.LIB in the various directories, for a description of the

GNU General Public License terms under which you can copy the files.

REPORTING BUGS: Again, see gdb/README, binutils/README, etc., for info

on where and how to report problems.

Glib (v2.30.2):

General Information

==========

This is GLib version 2.30.2. GLib is the low-level core
library that forms the basis for projects such as GTK+ and GNOME. It
provides data structure handling for C, portability wrappers, and
interfaces for such runtime functionality as an event loop, threads,
dynamic loading, and an object system.

The official ftp site is:

ftp://ftp.gtk.org/pub/glib

The official web site is:

http://www.gtk.org/

Information about mailing lists can be found at

http://www.gtk.org/mailing-lists.html

To subscribe: mail -s subscribe gtk-list-request@gnome.org < /dev/null

(Send mail to gtk-list-request@gnome.org with the subject "subscribe")

Installation

==========

See the file 'INSTALL'

Notes about GLib 2.30

==========

* GObject includes a generic marshaller, g_cclosure_marshal_generic.
  To use it, simply specify NULL as the marshaller in g_signal_new().
  The generic marshaller is implemented with libffi, and consequently
  GObject depends on libffi now.

Notes about GLib 2.28

==========

* The GApplication API has changed compared to the version that was
  included in the 2.25 development snapshots. Existing users will need
  adjustments.

Notes about GLib 2.26

==========

* Nothing noteworthy.

Notes about GLib 2.24

==========

* It is now allowed to call g_thread_init(NULL) multiple times, and
  to call glib functions before g_thread_init(NULL) is called
  (although the later is mainly a change in docs as this worked before

too). See the GThread reference documentation for the details.

* GObject now links to GThread and threads are enabled automatically
when g_type_init() is called.

* GObject no longer allows to call g_object_set() on construct-only properties
while an object is being initialized. If this behavior is needed, setting a
custom constructor that just chains up will re-enable this functionality.

* GMappedFile on an empty file now returns NULL for the contents instead of
returning an empty string. The documentation specifically states that code
may not rely on nul-termination here so any breakage caused by this change
is a bug in application code.

Notes about GLib 2.22

==========

* Repeated calls to g_simple_async_result_set_op_res_gpointer used
to leak the data. This has been fixed to always call the provided
destroy notify.

Notes about GLib 2.20

==========

* The functions for launching applications (e.g. g_app_info_launch() +
friends) now passes a FUSE file:// URI if possible (requires gvfs
with the FUSE daemon to be running and operational). With gvfs 2.26,

FUSE file:// URIs will be mapped back to gio URIs in the GFile
constructors. The intent of this change is to better integrate
POSIX-only applications, see bug #528670 for the rationale. The
only user-visible change is when an application needs to examine an
URI passed to it (e.g. as a positional parameter). Instead of
looking at the given URI, the application will now need to look at
the result of g_file_get_uri() after having constructed a GFile
object with the given URI.

Notes about GLib 2.18

==========

* The recommended way of using GLib has always been to only include the
toplevel headers glib.h, glib-object.h and gio.h. GLib enforces this by
generating an error when individual headers are directly included.
To help with the transition, the enforcement is not turned on by
default for GLib headers (it is turned on for GObject and GIO).
To turn it on, define the preprocessor symbol G_DISABLE_SINGLE_INCLUDES.

Notes about GLib 2.16

==========

* GLib now includes GIO, which adds optional dependencies against libattr
and libselinux for extended attribute and SELinux support. Use
--disable-xattr and --disable-selinux to build without these.

Notes about GLib 2.10

==========

* The functions g_snprintf() and g_vsnprintf() have been removed from
  the gprintf.h header, since they are already declared in glib.h. This
  doesn't break documented use of gprintf.h, but people have been known
  to include gprintf.h without including glib.h.

* The Unicode support has been updated to Unicode 4.1. This adds several
  new members to the GUnicodeBreakType enumeration.

* The support for Solaris threads has been retired. Solaris has provided
  POSIX threads for long enough now to have them available on every
  Solaris platform.

* 'make check' has been changed to validate translations by calling
  msgfmt with the -c option. As a result, it may fail on systems with
  older gettext implementations (GNU gettext < 0.14.1, or Solaris gettext).
  'make check' will also fail on systems where the C compiler does not
  support ELF visibility attributes.

* The GMemChunk API has been deprecated in favour of a new 'slice
  allocator'. See the g_slice documentation for more details.

* A new type, GInitiallyUnowned, has been introduced, which is
  intended to serve as a common implementation of the 'floating reference'
  concept that is e.g. used by GtkObject. Note that changing the

inheritance hierarchy of a type can cause problems for language bindings and other code which needs to work closely with the type system. Therefore, switching to GInitiallyUnowned should be done carefully. g_object_compat_control() has been added to GLib 2.8.5 to help with the transition.

Notes about GLib 2.6.0

==========

* GLib 2.6 introduces the concept of 'GLib filename encoding', which is the on-disk encoding on Unix, but UTF-8 on Windows. All GLib functions returning or accepting pathnames have been changed to expect filenames in this encoding, and the common POSIX functions dealing with pathnames have been wrapped. These wrappers are declared in the header <glib/gstdio.h> which must be included explicitly; it is not included through <glib.h>.

On current (NT-based) Windows versions, where the on-disk file names are Unicode, these wrappers use the wide-character API in the C library. Thus applications can handle file names containing any Unicode characters through GLib's own API and its POSIX wrappers, not just file names restricted to characters in the system codepage.

To keep binary compatibility with applications compiled against older versions of GLib, the Windows DLL still provides entry points with the old semantics using the old names, and applications compiled against GLib 2.6 will actually use new names for the

functions. This is transparent to the programmer.

When compiling against GLib 2.6, applications intended to be portable to Windows must take the UTF-8 file name encoding into consideration, and use the gstdio wrappers to access files whose names have been constructed from strings returned from GLib.

* Likewise, g_get_user_name() and g_get_real_name() have been changed to return UTF-8 on Windows, while keeping the old semantics for applications compiled against older versions of GLib.

* The GLib uses an '_' prefix to indicate private symbols that must not be used by applications. On some platforms, symbols beginning with prefixes such as _g will be exported from the library, on others not. In no case can applications use these private symbols. In addition to that, GLib+ 2.6 makes several symbols private which were not in any installed header files and were never intended to be exported.

* To reduce code size and improve efficiency, GLib, when compiled with the GNU toolchain, has separate internal and external entry points for exported functions. The internal names, which begin with IA__, may be seen when debugging a GLib program.

* On Windows, GLib no longer opens a console window when printing warning messages if stdout or stderr are invalid, as they are in "Windows subsystem" (GUI) applications. Simply redirect stdout or stderr if you need to see them.

* The child watch functionality tends to reveal a bug in many
  thread implementations (in particular the older LinuxThreads
  implementation on Linux) where it's not possible to call waitpid()
  for a child created in a different thread. For this reason, for
  maximum portability, you should structure your code to fork all
  child processes that you want to wait for from the main thread.

* A problem was recently discovered with g_signal_connect_object();
  it doesn't actually disconnect the signal handler once the object being
  connected to dies, just disables it. See the API docs for the function
  for further details and the correct workaround that will continue to
  work with future versions of GLib.

How to report bugs

==========

Bugs should be reported to the GNOME bug tracking system.
(http://bugzilla.gnome.org, product glib.) You will need
to create an account for yourself.

In the bug report please include:

* Information about your system. For instance:

  - What operating system and version
  - For Linux, what version of the C library

And anything else you think is relevant.

* How to reproduce the bug.

If you can reproduce it with one of the test programs that are built
in the tests/ subdirectory, that will be most convenient. Otherwise,
please include a short test program that exhibits the behavior.
As a last resort, you can also provide a pointer to a larger piece
of software that can be downloaded.

* If the bug was a crash, the exact text that was printed out
when the crash occured.

* Further information such as stack traces may be useful, but
is not necessary.

Patches

=======

Patches should also be submitted to bugzilla.gnome.org. If the
patch fixes an existing bug, add the patch as an attachment
to that bug report.

Otherwise, enter a new bug report that describes the patch,
and attach the patch to that bug report.
Patches should be in unified diff form. (The -up option to GNUdiff.)

Gzip (v1.5):

This is the file README for the gzip distribution.

The GNU gzip home page is http://www.gnu.org/software/gzip.

gzip (GNU zip) is a compression utility designed to be a replacement for 'compress'. Its main advantages over compress are much better compression and freedom from patented algorithms. The GNU Project uses it as the standard compression program for its system.

gzip currently uses by default the LZ77 algorithm used in zip 1.9 (the portable pkzip compatible archiver). The gzip format was however designed to accommodate several compression algorithms. See below for a comparison of zip and gzip.

gunzip can currently decompress files created by gzip, compress or pack. The detection of the input format is automatic. For the gzip format, gunzip checks a 32 bit CRC. For pack, gunzip checks the uncompressed length. The 'compress' format was not designed to allow consistency checks. However gunzip is sometimes able to detect a bad .Z file because there is some redundancy in the .Z compression format. If you get an error when uncompressing a .Z file, do not assume that the .Z file is correct simply because the standard uncompress does not complain. This generally means that the standard uncompress does not check its input, and happily generates garbage output.

gzip produces files with a .gz extension. Previous versions of gzip
used the .z extension, which was already used by the 'pack'
Huffman encoder. gunzip is able to decompress .z files (packed
or gzip'ed).

Several planned features are not yet supported (see the file TODO).
See the file NEWS for a summary of changes since the last release.
See the file INSTALL for installation instructions.

WARNING: gzip is sensitive to compiler bugs, particularly when
optimizing. Use "make check" to check that gzip was compiled
correctly. Try compiling gzip without any optimization if you have a
problem.

Please send all comments and bug reports by electronic mail to
<bug-gzip@gnu.org>.

Bug reports should ideally include:

* The complete output of "gzip -V" (or the contents of revision.h
  if you can't get gzip to compile)
* The hardware and operating system (try "uname -a")
* The compiler used to compile (if it is gcc, use "gcc -v")
* A description of the bug behavior
* The input to gzip, that triggered the bug

If you send me patches for machines I don't have access to, please test them
very carefully. gzip is used for backups, it must be extremely reliable.

The znew and gzexe shell scripts provided with gzip benefit from
(but do not require) the (non-GNU) cpmod utility to transfer file attributes.

The sample programs zread.c, sub.c and add.c in subdirectory sample
are provided as examples of useful complements to gzip. Read the
comments inside each source file. The perl script ztouch is also
provided as example (not installed by default since it relies on perl).

gzip is free software, you can redistribute it and/or modify it under
the terms of the GNU General Public License, a copy of which is
provided under the name COPYING. The latest version of gzip are always
available from ftp://ftp.gnu.org/gnu/gzip or in any of the gnu
mirror sites.

- sources in gzip-*.tar (or .shar or .tar.gz).
- MSDOS lha self-extracting exe in gzip-msdos-*.exe. Once extracted,
  copy gzip.exe to gunzip.exe and zcat.exe, or use "gzip -d" to decompress.
  gzip386.exe runs much faster but only on 386 and above; it was compiled with
  djgpp 1.10 available in directory omnigate.clarkson.edu:/pub/msdos/djgpp.

A VMS executable is in ftp://ftp.spc.edu/[.macro32.savesets]gzip-1-*.zip
(use [.macro32]unzip.exe to extract). A PRIMOS executable is available
in ftp://ftp.lysator.liu.se/pub/primos/run/gzip.run.

Some ftp servers can automatically make a tar.Z from a tar file. If
you are getting gzip for the first time, you can ask for a tar.Z file
instead of the much larger tar file.

Many thanks to those who provided me with bug reports and feedback.
See the files THANKS and ChangeLog for more details.

Note about zip vs. gzip:

The name 'gzip' was a very unfortunate choice, because zip and gzip
are two really different programs, although the actual compression and
decompression sources were written by the same persons. A different
name should have been used for gzip, but it is too late to change now.

zip is an archiver: it compresses several files into a single archive
file. gzip is a simple compressor: each file is compressed separately.
Both share the same compression and decompression code for the
'deflate' method. unzip can also decompress old zip archives
(implode, shrink and reduce methods). gunzip can also decompress files
created by compress and pack. zip 1.9 and gzip do not support
compression methods other than deflation. (zip 1.0 supports shrink and
implode). Better compression methods may be added in future versions
of gzip. zip will always stick to absolute compatibility with pkzip,
it is thus constrained by PKWare, which is a commercial company. The
gzip header format is deliberately different from that of pkzip to
avoid such a constraint.

On Unix, gzip is mostly useful in combination with tar. GNU tar
1.11.2 and later has a -z option to invoke gzip automatically. "tar -z"
compresses better than zip, since gzip can then take advantage of
redundancy between distinct files. The drawback is that you must
scan the whole tar.gz file in order to extract a single file near
the end; unzip can directly seek to the end of the zip file. There
is no overhead when you extract the whole archive anyway.
If a member of a .zip archive is damaged, other files can still
be recovered. If a .tar.gz file is damaged, files beyond the failure
point cannot be recovered. (Future versions of gzip will have
error recovery features.)

gzip and gunzip are distributed as a single program. zip and unzip
are, for historical reasons, two separate programs, although the
authors of these two programs work closely together in the Info-ZIP
team. zip and unzip are not associated with the GNU project.
See http://info-zip.org/ for more about zip and unzip.

For any copyright year range specified as YYYY-ZZZZ in this package
note that the range specifies every single year in that closed interval.

==========

Copyright (C) 1999, 2001-2002, 2006-2007, 2009-2012 Free Software Foundation,

Inc.

Copyright (C) 1992, 1993 Jean-loup Gailly

Permission is granted to copy, distribute and/or modify this document

under the terms of the GNU Free Documentation License, Version 1.3 or

any later version published by the Free Software Foundation; with no

Invariant Sections, with no Front-Cover Texts, and with no Back-Cover

Texts. A copy of the license is included in the ``GNU Free

Documentation License'' file as part of this distribution.

Kmod (v8):

kmod - Linux kernel module handling

OVERVIEW

========

kmod is a set of tools to handle common tasks with Linux kernel modules like

insert, remove, list, check properties, resolve dependencies and aliases.

These tools are designed on top of libkmod, a library that is shipped with

kmod. See libkmod/README for more details on this library and how to use it.

The aim is to be compatible with tools, configurations and indexes from

module-init-tools project.

Compilation and installation

==========

In order to compiler the source code you need following software packages:

- GCC compiler

- GNU C library

Optional dependencies:

- ZLIB library

- LZMA library

Typical configuration:

```
./configure CFLAGS="-g -O2" --prefix=/usr \
            --sysconfdir=/etc --libdir=/usr/lib
```

Configure automatically searches for all required components and packages.

To compile and install run:

```
make && make install
```

Hacking

=======

Run 'bootstrap' script before configure. If you want to accept the recommended

flags, you just need to run 'bootstrap-configure'.

Make sure to read the CODING-STYLE file and the other READMEs: libkmod/README

and testsuite/README.

Information

==========

Signed packages:

http://www.kernel.org/pub/linux/utils/kernel/kmod/

Mailing list:

linux-modules@vger.kernel.org

Git:

git://git.kernel.org/pub/scm/utils/kernel/kmod/kmod.git

http://git.kernel.org/pub/scm/utils/kernel/kmod/kmod.git

https://git.kernel.org/pub/scm/utils/kernel/kmod/kmod.git

Gitweb:

http://git.kernel.org/?p=utils/kernel/kmod/kmod.git

Irc:

#kmod on irc.freenode.org

Libffi (3.0.11):

Status

=======

libffi-3.0.11 was released on April 11, 2012. Check the libffi web

page for updates: <URL:http://sourceware.org/libffi/>.

What is libffi?

===============

Compilers for high level languages generate code that follow certain

conventions. These conventions are necessary, in part, for separate

compilation to work. One such convention is the "calling

convention". The "calling convention" is essentially a set of

assumptions made by the compiler about where function arguments will

be found on entry to a function. A "calling convention" also specifies

where the return value for a function is found.

Some programs may not know at the time of compilation what arguments

are to be passed to a function. For instance, an interpreter may be

told at run-time about the number and types of arguments used to call

a given function. Libffi can be used in such programs to provide a

bridge from the interpreter program to compiled code.

The libffi library provides a portable, high level programming interface to various calling conventions. This allows a programmer to call any function specified by a call interface description at run time.

FFI stands for Foreign Function Interface. A foreign function interface is the popular name for the interface that allows code written in one language to call code written in another language. The libffi library really only provides the lowest, machine dependent layer of a fully featured foreign function interface. A layer must exist above libffi that handles type conversions for values passed between the two languages.

Supported Platforms

==========

Libffi has been ported to many different platforms.
For specific configuration details and testing status, please refer to the wiki page here:

http://www.moxielogic.org/wiki/index.php?title=Libffi_3.0.11

At the time of release, the following basic configurations have been

tested:

Architecture	Operating System
Alpha	Linux
Alpha	Tru64
ARM	Linux
ARM	iOS
AVR32	Linux
HPPA	HPUX
IA-64	Linux
M68K	FreeMiNT
M68K	RTEMS
MIPS	IRIX
MIPS	Linux
MIPS	RTEMS
MIPS64	Linux
PowerPC	AMIGA
PowerPC	Linux
PowerPC	Mac OSX
PowerPC	FreeBSD
PowerPC64	Linux
S390	Linux
S390X	Linux
SPARC	Linux

| SPARC | Solaris |

| SPARC64 | Linux |

| SPARC64 | FreeBSD |

| X86 | FreeBSD |

| X86 | Interix |

| X86 | kFreeBSD |

| X86 | Linux |

| X86 | Mac OSX |

| X86 | OpenBSD |

| X86 | OS/2 |

| X86 | Solaris |

| X86 | Windows/Cygwin |

| X86 | Windows/MingW |

| X86-64 | FreeBSD |

| X86-64 | Linux |

| X86-64 | Linux/x32 |

| X86-64 | OpenBSD |

| X86-64 | Windows/MingW |

|--------------+------------------|

Please send additional platform test results to

libffi-discuss@sourceware.org and feel free to update the wiki page

above.

Installing libffi

==========

First you must configure the distribution for your particular
system. Go to the directory you wish to build libffi in and run the
"configure" program found in the root directory of the libffi source
distribution.

You may want to tell configure where to install the libffi library and
header files. To do that, use the --prefix configure switch. Libffi
will install under /usr/local by default.

If you want to enable extra run-time debugging checks use the the
--enable-debug configure switch. This is useful when your program dies
mysteriously while using libffi.

Another useful configure switch is --enable-purify-safety. Using this
will add some extra code which will suppress certain warnings when you
are using Purify with libffi. Only use this switch when using
Purify, as it will slow down the library.

It's also possible to build libffi on Windows platforms with
Microsoft's Visual C++ compiler. In this case, use the msvcc.sh
wrapper script during configuration like so:

path/to/configure CC=path/to/msvcc.sh LD=link CPP=\"cl -nologo -EP\"

For 64-bit Windows builds, use CC="path/to/msvcc.sh -m64".

You may also need to specify --build appropriately. When building with MSVC

under a MingW environment, you may need to remove the line in configure

that sets 'fix_srcfile_path' to a 'cygpath' command. ('cygpath' is not

present in MingW, and is not required when using MingW-style paths.)

For iOS builds, run generate-ios-source-and-headers.py and then

libffi.xcodeproj should work.

Configure has many other options. Use "configure --help" to see them all.

Once configure has finished, type "make". Note that you must be using

GNU make. You can ftp GNU make from prep.ai.mit.edu:/pub/gnu.

To ensure that libffi is working as advertised, type "make check".

This will require that you have DejaGNU installed.

To install the library and header files, type "make install".

History

=======

See the ChangeLog files for details.

3.0.11 Apr-11-12

Add support for variadic functions (ffi_prep_cif_var).

Add Linux/x32 support.

Add thiscall, fastcall and MSVC cdecl support on Windows.

Add Amiga and newer MacOS support.

Add m68k FreeMiNT support.

Integration with iOS' xcode build tools.

Fix Octeon and MC68881 support.

Fix code pessimizations.

Lots of build fixes.

3.0.10 Aug-23-11

Add support for Apple's iOS.

Add support for ARM VFP ABI.

Add RTEMS support for MIPS and M68K.

Fix instruction cache clearing problems on

ARM and SPARC.

Fix the N64 build on mips-sgi-irix6.5.

Enable builds with Microsoft's compiler.

Enable x86 builds with Oracle's Solaris compiler.

Fix support for calling code compiled with Oracle's Sparc

Solaris compiler.

Testsuite fixes for Tru64 Unix.

Additional platform support.

3.0.9 Dec-31-09

Add AVR32 and win64 ports. Add ARM softfp support.

Many fixes for AIX, Solaris, HP-UX, *BSD.

Several PowerPC and x86-64 bug fixes.

Build DLL for windows.

3.0.8 Dec-19-08

Add *BSD, BeOS, and PA-Linux support.

3.0.7 Nov-11-08

Fix for ppc FreeBSD.

(thanks to Andreas Tobler)

3.0.6 Jul-17-08

Fix for closures on sh.

Mark the sh/sh64 stack as non-executable.

(both thanks to Kaz Kojima)

3.0.5 Apr-3-08

Fix libffi.pc file.

Fix #define ARM for IcedTea users.

Fix x86 closure bug.

3.0.4 Feb-24-08

Fix x86 OpenBSD configury.

3.0.3 Feb-22-08

Enable x86 OpenBSD thanks to Thomas Heller, and

x86-64 FreeBSD thanks to Björn König and Andreas Tobler.

Clean up test instruction in README.

3.0.2 Feb-21-08

Improved x86 FreeBSD support.

Thanks to Björn König.

3.0.1 Feb-15-08

Fix instruction cache flushing bug on MIPS.

Thanks to David Daney.

3.0.0 Feb-15-08

Many changes, mostly thanks to the GCC project.

Cygnus Solutions is now Red Hat.

[10 years go by...]

1.20 Oct-5-98

Raffaele Sena produces ARM port.

1.19 Oct-5-98

Fixed x86 long double and long long return support.

m68k bug fixes from Andreas Schwab.

Patch for DU assembler compatibility for the Alpha from Richard Henderson.

1.18 Apr-17-98

Bug fixes and MIPS configuration changes.

1.17 Feb-24-98

Bug fixes and m68k port from Andreas Schwab. PowerPC port from Geoffrey Keating. Various bug x86, Sparc and MIPS bug fixes.

1.16 Feb-11-98

Richard Henderson produces Alpha port.

1.15 Dec-4-97

Fixed an n32 ABI bug. New libtool, auto* support.

1.14 May-13-97

libtool is now used to generate shared and static libraries.

Fixed a minor portability problem reported by Russ McManus <mcmanr@eq.gs.com>.

1.13 Dec-2-96

Added --enable-purify-safety to keep Purify from complaining

about certain low level code.

Sparc fix for calling functions with < 6 args.

Linux x86 a.out fix.

1.12 Nov-22-96

Added missing ffi_type_void, needed for supporting void return

types. Fixed test case for non MIPS machines. Cygnus Support

is now Cygnus Solutions.

1.11 Oct-30-96

Added notes about GNU make.

1.10 Oct-29-96

Added configuration fix for non GNU compilers.

1.09 Oct-29-96

Added --enable-debug configure switch. Clean-ups based on LCLint

feedback. ffi_mips.h is always installed. Many configuration

fixes. Fixed ffitest.c for sparc builds.

1.08 Oct-15-96

Fixed n32 problem. Many clean-ups.

1.07 Oct-14-96

Gordon Irlam rewrites v8.S again. Bug fixes.

1.06 Oct-14-96

Gordon Irlam improved the sparc port.

1.05 Oct-14-96

Interface changes based on feedback.

1.04 Oct-11-96

Sparc port complete (modulo struct passing bug).

1.03 Oct-10-96

Passing struct args, and returning struct values works for

all architectures/calling conventions. Expanded tests.

1.02 Oct-9-96

Added SGI n32 support. Fixed bugs in both o32 and Linux support.

Added "make test".

1.01 Oct-8-96

Fixed float passing bug in mips version. Restructured some

of the code. Builds cleanly with SGI tools.

1.00 Oct-7-96

First release. No public announcement.

Authors & Credits

==========

libffi was originally written by Anthony Green <green@moxielogic.com>.

The developers of the GNU Compiler Collection project have made innumerable valuable contributions. See the ChangeLog file for details.

Some of the ideas behind libffi were inspired by Gianni Mariani's free gencall library for Silicon Graphics machines.

The closure mechanism was designed and implemented by Kresten Krab Thorup.

Major processor architecture ports were contributed by the following developers:

alpha	Richard Henderson
arm	Raffaele Sena
cris	Simon Posnjak, Hans-Peter Nilsson
frv	Anthony Green
ia64	Hans Boehm
m32r	Kazuhiro Inaoka
m68k	Andreas Schwab
mips	Anthony Green, Casey Marshall

mips64	David Daney
pa	Randolph Chung, Dave Anglin, Andreas Tobler
powerpc	Geoffrey Keating, Andreas Tobler, David Edelsohn, John Hornkvist
powerpc64	Jakub Jelinek
s390	Gerhard Tonn, Ulrich Weigand
sh	Kaz Kojima
sh64	Kaz Kojima
sparc	Anthony Green, Gordon Irlam
x86	Anthony Green, Jon Beniston
x86-64	Bo Thorsen

Jesper Skov and Andrew Haley both did more than their fair share of stepping through the code and tracking down bugs.

Thanks also to Tom Tromey for bug fixes, documentation and configuration help.

Thanks to Jim Blandy, who provided some useful feedback on the libffi interface.

Andreas Tobler has done a tremendous amount of work on the testsuite.

Alex Oliva solved the executable page problem for SElinux.

The list above is almost certainly incomplete and inaccurate. I'm happy to make corrections or additions upon request.

If you have a problem, or have found a bug, please send a note to the

author at green@moxielogic.com, or the project mailing list at

libffi-discuss@sourceware.org.

Libjpeg (v9a):

The Independent JPEG Group's JPEG software

==========

README for release 8d of 15-Jan-2012

==========

This distribution contains the eighth public release of the Independent JPEG

Group's free JPEG software. You are welcome to redistribute this software and

to use it for any purpose, subject to the conditions under LEGAL ISSUES, below.

This software is the work of Tom Lane, Guido Vollbeding, Philip Gladstone,

Bill Allombert, Jim Boucher, Lee Crocker, Bob Friesenhahn, Ben Jackson,

Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Ge' Weijers,

and other members of the Independent JPEG Group.

IJG is not affiliated with the ISO/IEC JTC1/SC29/WG1 standards committee

(also known as JPEG, together with ITU-T SG16).

DOCUMENTATION ROADMAP

==========

This file contains the following sections:

OVERVIEW General description of JPEG and the IJG software.

LEGAL ISSUES Copyright, lack of warranty, terms of distribution.

REFERENCES Where to learn more about JPEG.

ARCHIVE LOCATIONS Where to find newer versions of this software.

ACKNOWLEDGMENTS Special thanks.

FILE FORMAT WARS Software *not* to get.

TO DO Plans for future IJG releases.

Other documentation files in the distribution are:

User documentation:

install.txt How to configure and install the IJG software.

usage.txt Usage instructions for cjpeg, djpeg, jpegtran, rdjpgcom, and wrjpgcom.

*.1 Unix-style man pages for programs (same info as usage.txt).

wizard.txt Advanced usage instructions for JPEG wizards only.

change.log Version-to-version change highlights.

Programmer and internal documentation:

libjpeg.txt How to use the JPEG library in your own programs.

example.c Sample code for calling the JPEG library.

structure.txt Overview of the JPEG library's internal structure.

filelist.txt Road map of IJG files.

coderules.txt Coding style rules --- please read if you contribute code.

Please read at least the files install.txt and usage.txt. Some information
can also be found in the JPEG FAQ (Frequently Asked Questions) article. See
ARCHIVE LOCATIONS below to find out where to obtain the FAQ article.

If you want to understand how the JPEG code works, we suggest reading one or
more of the REFERENCES, then looking at the documentation files (in roughly
the order listed) before diving into the code.

OVERVIEW

========

This package contains C software to implement JPEG image encoding, decoding,
and transcoding. JPEG (pronounced "jay-peg") is a standardized compression
method for full-color and gray-scale images.

This software implements JPEG baseline, extended-sequential, and progressive
compression processes. Provision is made for supporting all variants of these
processes, although some uncommon parameter settings aren't implemented yet.
We have made no provision for supporting the hierarchical or lossless
processes defined in the standard.

We provide a set of library routines for reading and writing JPEG image files,
plus two sample applications "cjpeg" and "djpeg", which use the library to
perform conversion between JPEG and some other popular image file formats.
The library is intended to be reused in other applications.

In order to support file conversion and viewing software, we have included considerable functionality beyond the bare JPEG coding/decoding capability; for example, the color quantization modules are not strictly part of JPEG decoding, but they are essential for output to colormapped file formats or colormapped displays. These extra functions can be compiled out of the library if not required for a particular application.

We have also included "jpegtran", a utility for lossless transcoding between different JPEG processes, and "rdjpgcom" and "wrjpgcom", two simple applications for inserting and extracting textual comments in JFIF files.

The emphasis in designing this software has been on achieving portability and flexibility, while also making it fast enough to be useful. In particular, the software is not intended to be read as a tutorial on JPEG. (See the REFERENCES section for introductory material.) Rather, it is intended to be reliable, portable, industrial-strength code. We do not claim to have achieved that goal in every aspect of the software, but we strive for it.

We welcome the use of this software as a component of commercial products. No royalty is required, but we do ask for an acknowledgement in product documentation, as described under LEGAL ISSUES.

LEGAL ISSUES

==========

In plain English:

1. We don't promise that this software works. (But if you find any bugs,
   please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a
   program, you must acknowledge somewhere in your documentation that
   you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied,
with respect to this software, its quality, accuracy, merchantability, or
fitness for a particular purpose. This software is provided "AS IS", and you,
its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-2012, Thomas G. Lane, Guido Vollbeding.
All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this
software (or portions thereof) for any purpose, without fee, subject to these
conditions:
(1) If any part of the source code for this software is distributed, then this
README file must be included, with this copyright and no-warranty notice

unaltered; and any additions, deletions, or changes to the original files
must be clearly indicated in accompanying documentation.
(2) If only executable code is distributed, then the accompanying
documentation must state that "this software is based in part on the work of
the Independent JPEG Group".
(3) Permission for use of this software is granted only if the user accepts
full responsibility for any undesirable consequences; the authors accept
NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code,
not just to the unmodified library. If you use our work, you ought to
acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name
in advertising or publicity relating to this software or products derived from
it. This software may be referred to only as "the Independent JPEG Group's
software".

We specifically permit and encourage the use of this software as the basis of
commercial products, provided that all warranty or liability claims are
assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch,
sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA.
ansi2knr.c is NOT covered by the above copyright and conditions, but instead
by the usual distribution terms of the Free Software Foundation; principally,

that you must include source code if you redistribute it. (See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part of any program generated from the IJG code, this does not limit you more than the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable. The same holds for its supporting scripts (config.guess, config.sub, ltmain.sh). Another support script, install-sh, is copyright by X Consortium but is also freely distributable.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that

> "The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

REFERENCES

==========

We recommend reading one or more of these references before trying to
understand the innards of the JPEG software.

The best short technical introduction to the JPEG compression algorithm is

Wallace, Gregory K. "The JPEG Still Picture Compression Standard",
Communications of the ACM, April 1991 (vol. 34 no. 4), pp. 30-44.

(Adjacent articles in that issue discuss MPEG motion picture compression,
applications of JPEG, and related topics.) If you don't have the CACM issue
handy, a PostScript file containing a revised version of Wallace's article is
available at http://www.ijg.org/files/wallace.ps.gz. The file (actually
a preprint for an article that appeared in IEEE Trans. Consumer Electronics)
omits the sample images that appeared in CACM, but it includes corrections
and some added material. Note: the Wallace article is copyright ACM and IEEE,
and it may not be used for commercial purposes.

A somewhat less technical, more leisurely introduction to JPEG can be found in
"The Data Compression Book" by Mark Nelson and Jean-loup Gailly, published by
M&T Books (New York), 2nd ed. 1996, ISBN 1-55851-434-1. This book provides
good explanations and example C code for a multitude of compression methods
including JPEG. It is an excellent source if you are comfortable reading C
code but don't know much about data compression in general. The book's JPEG
sample code is far from industrial-strength, but when you are ready to look
at a full implementation, you've got one here...

The best currently available description of JPEG is the textbook "JPEG Still Image Data Compression Standard" by William B. Pennebaker and Joan L. Mitchell, published by Van Nostrand Reinhold, 1993, ISBN 0-442-01272-1. Price US$59.95, 638 pp. The book includes the complete text of the ISO JPEG standards (DIS 10918-1 and draft DIS 10918-2).
Although this is by far the most detailed and comprehensive exposition of JPEG publicly available, we point out that it is still missing an explanation of the most essential properties and algorithms of the underlying DCT technology.
If you think that you know about DCT-based JPEG after reading this book, then you are in delusion. The real fundamentals and corresponding potential of DCT-based JPEG are not publicly known so far, and that is the reason for all the mistaken developments taking place in the image coding domain.

The original JPEG standard is divided into two parts, Part 1 being the actual specification, while Part 2 covers compliance testing methods. Part 1 is titled "Digital Compression and Coding of Continuous-tone Still Images, Part 1: Requirements and guidelines" and has document numbers ISO/IEC IS 10918-1, ITU-T T.81. Part 2 is titled "Digital Compression and Coding of Continuous-tone Still Images, Part 2: Compliance testing" and has document numbers ISO/IEC IS 10918-2, ITU-T T.83.
IJG JPEG 8 introduces an implementation of the JPEG SmartScale extension which is specified in two documents: A contributed document at ITU and ISO with title "ITU-T JPEG-Plus Proposal for Extending ITU-T T.81 for Advanced Image Coding", April 2006, Geneva, Switzerland. The latest version of this document is Revision 3. And a contributed document ISO/IEC JTC1/SC29/WG1 N 5799 with title "Evolution of JPEG", June/July 2011, Berlin, Germany.

The JPEG standard does not specify all details of an interchangeable file format. For the omitted details we follow the "JFIF" conventions, revision 1.02. JFIF 1.02 has been adopted as an Ecma International Technical Report and thus received a formal publication status. It is available as a free download in PDF format from http://www.ecma-international.org/publications/techreports/E-TR-098.htm. A PostScript version of the JFIF document is available at http://www.ijg.org/files/jfif.ps.gz. There is also a plain text version at http://www.ijg.org/files/jfif.txt.gz, but it is missing the figures.

The TIFF 6.0 file format specification can be obtained by FTP from ftp://ftp.sgi.com/graphics/tiff/TIFF6.ps.gz. The JPEG incorporation scheme found in the TIFF 6.0 spec of 3-June-92 has a number of serious problems. IJG does not recommend use of the TIFF 6.0 design (TIFF Compression tag 6). Instead, we recommend the JPEG design proposed by TIFF Technical Note #2 (Compression tag 7). Copies of this Note can be obtained from http://www.ijg.org/files/. It is expected that the next revision of the TIFF spec will replace the 6.0 JPEG design with the Note's design. Although IJG's own code does not support TIFF/JPEG, the free libtiff library uses our library to implement TIFF/JPEG per the Note.

ARCHIVE LOCATIONS

==========

The "official" archive site for this software is www.ijg.org.

The most recent released version can always be found there in

directory "files". This particular version will be archived as

http://www.ijg.org/files/jpegsrc.v8d.tar.gz, and in Windows-compatible

"zip" archive format as http://www.ijg.org/files/jpegsr8d.zip.

The JPEG FAQ (Frequently Asked Questions) article is a source of some

general information about JPEG.

It is available on the World Wide Web at http://www.faqs.org/faqs/jpeg-faq/

and other news.answers archive sites, including the official news.answers

archive at rtfm.mit.edu: ftp://rtfm.mit.edu/pub/usenet/news.answers/jpeg-faq/.

If you don't have Web or FTP access, send e-mail to mail-server@rtfm.mit.edu

with body

send usenet/news.answers/jpeg-faq/part1

send usenet/news.answers/jpeg-faq/part2

ACKNOWLEDGMENTS

==========

Thank to Juergen Bruder for providing me with a copy of the common DCT
algorithm article, only to find out that I had come to the same result
in a more direct and comprehensible way with a more generative approach.

Thank to Istvan Sebestyen and Joan L. Mitchell for inviting me to the
ITU JPEG (Study Group 16) meeting in Geneva, Switzerland.

Thank to Thomas Wiegand and Gary Sullivan for inviting me to the
Joint Video Team (MPEG & ITU) meeting in Geneva, Switzerland.

Thank to Thomas Richter and Daniel Lee for inviting me to the
ISO/IEC JTC1/SC29/WG1 (also known as JPEG, together with ITU-T SG16)
meeting in Berlin, Germany.

Thank to John Korejwa and Massimo Ballerini for inviting me to
fruitful consultations in Boston, MA and Milan, Italy.

Thank to Hendrik Elstner, Roland Fassauer, Simone Zuck, Guenther
Maier-Gerber, Walter Stoeber, Fred Schmitz, and Norbert Braunagel
for corresponding business development.

Thank to Nico Zschach and Dirk Stelling of the technical support team
at the Digital Images company in Halle for providing me with extra
equipment for configuration tests.

Thank to Richard F. Lyon (then of Foveon Inc.) for fruitful

communication about JPEG configuration in Sigma Photo Pro software.

Thank to Andrew Finkenstadt for hosting the ijg.org site.

Last but not least special thank to Thomas G. Lane for the original

design and development of this singular software package.

FILE FORMAT WARS

==========

The ISO/IEC JTC1/SC29/WG1 standards committee (also known as JPEG, together

with ITU-T SG16) currently promotes different formats containing the name

"JPEG" which is misleading because these formats are incompatible with

original DCT-based JPEG and are based on faulty technologies.

IJG therefore does not and will not support such momentary mistakes

(see REFERENCES).

There exist also distributions under the name "OpenJPEG" promoting such

kind of formats which is misleading because they don't support original

JPEG images.

We have no sympathy for the promotion of inferior formats. Indeed, one of

the original reasons for developing this free software was to help force

convergence on common, interoperable format standards for JPEG files.

Don't use an incompatible file format!

(In any case, our decoder will remain capable of reading existing JPEG

image files indefinitely.)

Furthermore, the ISO committee pretends to be "responsible for the popular JPEG" in their public reports which is not true because they don't respond to actual requirements for the maintenance of the original JPEG specification.

There are currently distributions in circulation containing the name "libjpeg" which claim to be a "derivative" or "fork" of the original libjpeg, but don't have the features and are incompatible with formats supported by actual IJG libjpeg distributions. Furthermore, they violate the license conditions as described under LEGAL ISSUES above. We have no sympathy for the release of misleading and illegal distributions derived from obsolete code bases.
Don't use an obsolete code base!

TO DO

=====

Version 8 is the first release of a new generation JPEG standard to overcome the limitations of the original JPEG specification.
More features are being prepared for coming releases...

Please send bug reports, offers of help, etc. to jpeg-info@jpegclub.org.

Libkmod (v8):

libkmod - linux kernel module handling library

ABSTRACT

========

libkmod was created to allow programs to easily insert, remove and

list modules, also checking its properties, dependencies and aliases.

there is no shared/global context information and it can be used by

multiple sites on a single program, also being able to be used from

threads, although it's not thread safe (you must lock explicitly).

OVERVIEW

========

Every user should create and manage it's own library context with:

```
struct kmod_ctx *ctx = kmod_new(kernel_dirname);
kmod_unref(ctx);
```

Modules can be created with by various means:

```
struct kmod_module *mod;
int err;

err = kmod_module_new_from_path(ctx, path, &mod);
if (err < 0) {
  /* code */
} else {
  /* code */
  kmod_module_unref(mod);
}

err = kmod_module_new_from_name(ctx, name, &mod);
if (err < 0) {
  /* code */
} else {
  /* code */
  kmod_module_unref(mod);
}
```

Or could be resolved from a known alias to a list of alternatives:

```
struct kmod_list *list, *itr;
int err;
err = kmod_module_new_from_lookup(ctx, alias, &list);
```

```
if (err < 0) {

  /* code */

} else {

  kmod_list_foreach(itr, list) {

    struct kmod_module *mod = kmod_module_get_module(itr);

    /* code */

  }

}
```

Libxml2 (v2.7.8):

Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar licence but with different Copyright notices) all the files are:

Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FIT- NESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

LibPcap (v1.2.1):

License: BSD

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.
3. The names of the authors may not be used to endorse or promote
   products derived from this software without specific prior
   written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

ABBpower , at91ft12, at91gpbr hardware driver (v1.0):

© Copyright [2014] ABB. All rights reserved.

This software is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version.

This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this software. If not, see <http://www.gnu.org/licenses/>.

Linux (v3.2.26):

Linux kernel release 3.x <http://kernel.org/>

These are the release notes for Linux version 3. Read them carefully,

as they tell you what this is all about, explain how to install the

kernel, and what to do if something goes wrong.

WHAT IS LINUX?

Linux is a clone of the operating system Unix, written from scratch by

Linus Torvalds with assistance from a loosely-knit team of hackers across

the Net. It aims towards POSIX and Single UNIX Specification compliance.

It has all the features you would expect in a modern fully-fledged Unix,

including true multitasking, virtual memory, shared libraries, demand

loading, shared copy-on-write executables, proper memory management,

and multistack networking including IPv4 and IPv6.

It is distributed under the GNU General Public License - see the

accompanying COPYING file for more details.

ON WHAT HARDWARE DOES IT RUN?

Although originally developed first for 32-bit x86-based PCs (386 or higher), today Linux also runs on (at least) the Compaq Alpha AXP, Sun SPARC and UltraSPARC, Motorola 68000, PowerPC, PowerPC64, ARM, Hitachi SuperH, Cell, IBM S/390, MIPS, HP PA-RISC, Intel IA-64, DEC VAX, AMD x86-64, AXIS CRIS, Xtensa, Tilera TILE, AVR32 and Renesas M32R architectures.

Linux is easily portable to most general-purpose 32- or 64-bit architectures as long as they have a paged memory management unit (PMMU) and a port of the GNU C compiler (gcc) (part of The GNU Compiler Collection, GCC). Linux has also been ported to a number of architectures without a PMMU, although functionality is then obviously somewhat limited.
Linux has also been ported to itself. You can now run the kernel as a userspace application - this is called UserMode Linux (UML).

DOCUMENTATION:

- There is a lot of documentation available both in electronic form on the Internet and in books, both Linux-specific and pertaining to general UNIX questions. I'd recommend looking into the documentation subdirectories on any Linux FTP site for the LDP (Linux Documentation Project) books. This README is not meant to be documentation on the system: there are much better sources available.

- There are various README files in the Documentation/ subdirectory: these typically contain kernel-specific installation notes for some

drivers for example. See Documentation/00-INDEX for a list of what
is contained in each file. Please read the Changes file, as it
contains information about the problems, which may result by upgrading
your kernel.

- The Documentation/DocBook/ subdirectory contains several guides for
kernel developers and users. These guides can be rendered in a
number of formats: PostScript (.ps), PDF, HTML, & man-pages, among others.
After installation, "make psdocs", "make pdfdocs", "make htmldocs",
or "make mandocs" will render the documentation in the requested format.

INSTALLING the kernel source:

- If you install the full sources, put the kernel tarball in a
directory where you have permissions (eg. your home directory) and
unpack it:

```
gzip -cd linux-3.X.tar.gz | tar xvf -
```

or

```
bzip2 -dc linux-3.X.tar.bz2 | tar xvf -
```

Replace "XX" with the version number of the latest kernel.

Do NOT use the /usr/src/linux area! This area has a (usually incomplete) set of kernel headers that are used by the library header files. They should match the library, and not get messed up by whatever the kernel-du-jour happens to be.

- You can also upgrade between 3.x releases by patching. Patches are distributed in the traditional gzip and the newer bzip2 format. To install by patching, get all the newer patch files, enter the top level directory of the kernel source (linux-3.x) and execute:

```
gzip -cd ../patch-3.x.gz | patch -p1
```

or

```
bzip2 -dc ../patch-3.x.bz2 | patch -p1
```

(repeat xx for all versions bigger than the version of your current source tree, _in_order_) and you should be ok. You may want to remove the backup files (xxx~ or xxx.orig), and make sure that there are no failed patches (xxx# or xxx.rej). If there are, either you or me has made a mistake.

Unlike patches for the 3.x kernels, patches for the 3.x.y kernels (also known as the -stable kernels) are not incremental but instead apply directly to the base 3.x kernel. Please read Documentation/applying-patches.txt for more information.

Alternatively, the script patch-kernel can be used to automate this process. It determines the current kernel version and applies any patches found.

```
linux/scripts/patch-kernel linux
```

The first argument in the command above is the location of the kernel source. Patches are applied from the current directory, but an alternative directory can be specified as the second argument.

- If you are upgrading between releases using the stable series patches (for example, patch-3.x.y), note that these "dot-releases" are not incremental and must be applied to the 3.x base tree. For example, if your base kernel is 3.0 and you want to apply the 3.0.3 patch, you do not and indeed must not first apply the 3.0.1 and 3.0.2 patches. Similarly, if you are running kernel version 3.0.2 and want to jump to 3.0.3, you must first reverse the 3.0.2 patch (that is, patch -R) _before_ applying the 3.0.3 patch.
You can read more on this in Documentation/applying-patches.txt

- Make sure you have no stale .o files and dependencies lying around:

```
cd linux
make mrproper
```

You should now have the sources correctly installed.

SOFTWARE REQUIREMENTS

Compiling and running the 3.x kernels requires up-to-date
versions of various software packages. Consult
Documentation/Changes for the minimum version numbers required
and how to get updates for these packages. Beware that using
excessively old versions of these packages can cause indirect
errors that are very difficult to track down, so don't assume that
you can just update packages when obvious problems arise during
build or operation.

BUILD directory for the kernel:

When compiling the kernel all output files will per default be
stored together with the kernel source code.
Using the option "make O=output/dir" allow you to specify an alternate
place for the output files (including .config).
Example:
kernel source code: /usr/src/linux-3.N
build directory: /home/name/build/kernel

To configure and build the kernel use:
cd /usr/src/linux-3.N
make O=/home/name/build/kernel menuconfig
make O=/home/name/build/kernel
sudo make O=/home/name/build/kernel modules_install install

Please note: If the 'O=output/dir' option is used then it must be
used for all invocations of make.

CONFIGURING the kernel:

Do not skip this step even if you are only upgrading one minor
version. New configuration options are added in each release, and
odd problems will turn up if the configuration files are not set up
as expected. If you want to carry your existing configuration to a
new version with minimal work, use "make oldconfig", which will
only ask you for the answers to new questions.

- Alternate configuration commands are:

"make config" Plain text interface.

"make menuconfig" Text based color menus, radiolists & dialogs.

"make nconfig" Enhanced text based color menus.

"make xconfig" X windows (Qt) based configuration tool.

"make gconfig" X windows (Gtk) based configuration tool.

"make oldconfig" Default all questions based on the contents of
your existing ./.config file and asking about
new config symbols.

"make silentoldconfig"
Like above, but avoids cluttering the screen
with questions already answered.
Additionally updates the dependencies.

"make defconfig" Create a ./.config file by using the default
symbol values from either arch/$ARCH/defconfig

```
                    or arch/$ARCH/configs/${PLATFORM}_defconfig,
                    depending on the architecture.
        "make ${PLATFORM}_defconfig"
                    Create a ./.config file by using the default
                    symbol values from
                    arch/$ARCH/configs/${PLATFORM}_defconfig.
                    Use "make help" to get a list of all available
                    platforms of your architecture.
        "make allyesconfig"
                    Create a ./.config file by setting symbol
                    values to 'y' as much as possible.
        "make allmodconfig"
                    Create a ./.config file by setting symbol
                    values to 'm' as much as possible.
        "make allnoconfig" Create a ./.config file by setting symbol
                    values to 'n' as much as possible.
        "make randconfig"  Create a ./.config file by setting symbol
                    values to random values.

    You can find more information on using the Linux kernel config tools
    in Documentation/kbuild/kconfig.txt.

        NOTES on "make config":
        - having unnecessary drivers will make the kernel bigger, and can
          under some circumstances lead to problems: probing for a
          nonexistent controller card may confuse your other controllers
        - compiling the kernel with "Processor type" set higher than 386
```

will result in a kernel that does NOT work on a 386. The
kernel will detect this on bootup, and give up.

- A kernel with math-emulation compiled in will still use the
coprocessor if one is present: the math emulation will just
never get used in that case. The kernel will be slightly larger,
but will work on different machines regardless of whether they
have a math coprocessor or not.

- the "kernel hacking" configuration details usually result in a
bigger or slower kernel (or both), and can even make the kernel
less stable by configuring some routines to actively try to
break bad code to find kernel problems (kmalloc()). Thus you
should probably answer 'n' to the questions for
"development", "experimental", or "debugging" features.

COMPILING the kernel:

- Make sure you have at least gcc 3.2 available.
For more information, refer to Documentation/Changes.

Please note that you can still run a.out user programs with this kernel.

- Do a "make" to create a compressed kernel image. It is also
possible to do "make install" if you have lilo installed to suit the
kernel makefiles, but you may want to check your particular lilo setup first.

To do the actual install you have to be root, but none of the normal
build should require that. Don't take the name of root in vain.

- If you configured any of the parts of the kernel as `modules', you
  will also have to do "make modules_install".

- Verbose kernel compile/build output:

  Normally the kernel build system runs in a fairly quiet mode (but not
  totally silent). However, sometimes you or other kernel developers need
  to see compile, link, or other commands exactly as they are executed.
  For this, use "verbose" build mode. This is done by inserting
  "V=1" in the "make" command. E.g.:

        make V=1 all

  To have the build system also tell the reason for the rebuild of each
  target, use "V=2". The default is "V=0".

- Keep a backup kernel handy in case something goes wrong. This is
  especially true for the development releases, since each new release
  contains new code which has not been debugged. Make sure you keep a
  backup of the modules corresponding to that kernel, as well. If you
  are installing a new kernel with the same version number as your
  working kernel, make a backup of your modules directory before you
  do a "make modules_install".
  Alternatively, before compiling, use the kernel config option
  "LOCALVERSION" to append a unique suffix to the regular kernel version.
  LOCALVERSION can be set in the "General Setup" menu.

- In order to boot your new kernel, you'll need to copy the kernel
  image (e.g. .../linux/arch/i386/boot/bzImage after compilation)
  to the place where your regular bootable kernel is found.

- Booting a kernel directly from a floppy without the assistance of a
  bootloader such as LILO, is no longer supported.

  If you boot Linux from the hard drive, chances are you use LILO which
  uses the kernel image as specified in the file /etc/lilo.conf. The
  kernel image file is usually /vmlinuz, /boot/vmlinuz, /bzImage or
  /boot/bzImage. To use the new kernel, save a copy of the old image
  and copy the new image over the old one. Then, you MUST RERUN LILO
  to update the loading map!! If you don't, you won't be able to boot
  the new kernel image.

  Reinstalling LILO is usually a matter of running /sbin/lilo.
  You may wish to edit /etc/lilo.conf to specify an entry for your
  old kernel image (say, /vmlinux.old) in case the new one does not
  work. See the LILO docs for more information.

  After reinstalling LILO, you should be all set. Shutdown the system,
  reboot, and enjoy!

If you ever need to change the default root device, video mode, ramdisk size, etc. in the kernel image, use the 'rdev' program (or alternatively the LILO boot options when appropriate). No need to recompile the kernel to change these parameters.

- Reboot with the new kernel and enjoy.

IF SOMETHING GOES WRONG:

- If you have problems that seem to be due to kernel bugs, please check the file MAINTAINERS to see if there is a particular person associated with the part of the kernel that you are having trouble with. If there isn't anyone listed there, then the second best thing is to mail them to me (torvalds@linux-foundation.org), and possibly to any other relevant mailing-list or to the newsgroup.

- In all bug-reports, *please* tell what kernel you are talking about, how to duplicate the problem, and what your setup is (use your common sense). If the problem is new, tell me so, and if the problem is old, please try to tell me when you first noticed it.

- If the bug results in a message like

    unable to handle kernel paging request at address C0000010
    Oops: 0002
    EIP: 0010:XXXXXXXX
    eax: xxxxxxxx ebx: xxxxxxxx ecx: xxxxxxxx edx: xxxxxxxx

esi: xxxxxxxx edi: xxxxxxxx ebp: xxxxxxxx

ds: xxxx es: xxxx fs: xxxx gs: xxxx

Pid: xx, process nr: xx

xx xx xx xx xx xx xx xx xx xx

or similar kernel debugging information on your screen or in your
system log, please duplicate it *exactly*. The dump may look
incomprehensible to you, but it does contain information that may
help debugging the problem. The text above the dump is also
important: it tells something about why the kernel dumped code (in
the above example it's due to a bad kernel pointer). More information
on making sense of the dump is in Documentation/oops-tracing.txt

- If you compiled the kernel with CONFIG_KALLSYMS you can send the dump
as is, otherwise you will have to use the "ksymoops" program to make
sense of the dump (but compiling with CONFIG_KALLSYMS is usually preferred).
This utility can be downloaded from
ftp://ftp.<country>.kernel.org/pub/linux/utils/kernel/ksymoops/ .
Alternately you can do the dump lookup by hand:

- In debugging dumps like the above, it helps enormously if you can
look up what the EIP value means. The hex value as such doesn't help
me or anybody else very much: it will depend on your particular
kernel setup. What you should do is take the hex value from the EIP
line (ignore the "0010:"), and look it up in the kernel namelist to
see which kernel function contains the offending address.

To find out the kernel function name, you'll need to find the system binary associated with the kernel that exhibited the symptom. This is the file 'linux/vmlinux'. To extract the namelist and match it against the EIP from the kernel crash, do:

```
nm vmlinux | sort | less
```

This will give you a list of kernel addresses sorted in ascending order, from which it is simple to find the function that contains the offending address. Note that the address given by the kernel debugging messages will not necessarily match exactly with the function addresses (in fact, that is very unlikely), so you can't just 'grep' the list: the list will, however, give you the starting point of each kernel function, so by looking for the function that has a starting address lower than the one you are searching for but is followed by a function with a higher address you will find the one you want. In fact, it may be a good idea to include a bit of "context" in your problem report, giving a few lines around the interesting one.

If you for some reason cannot do the above (you have a pre-compiled kernel image or similar), telling me as much about your setup as possible will help. Please read the REPORTING-BUGS document for details.

- Alternately, you can use gdb on a running kernel. (read-only; i.e. you
cannot change values or set break points.) To do this, first compile the
kernel with -g; edit arch/i386/Makefile appropriately, then do a "make
clean". You'll also need to enable CONFIG_PROC_FS (via "make config").

After you've rebooted with the new kernel, do "gdb vmlinux /proc/kcore".
You can now use all the usual gdb commands. The command to look up the
point where your system crashed is "l *0xXXXXXXXX". (Replace the XXXes
with the EIP value.)

gdb'ing a non-running kernel currently fails because gdb (wrongly)
disregards the starting offset for which the kernel is compiled.

Lsof (v4.85):

lsof (LiSt Open Files) version 4

(revision 4.85)

```
   ***************************************************************
   | The latest release of lsof is always available via anonymous ftp |
   | from lsof.itap.purdue.edu.  Look in pub/tools/unix/lsof.        |
   ***************************************************************


*****************************************************************************
| CHECK THE PATCHES/ SUBDIRECTORY FOR FIXES TO THE LATEST LSOF DISTRIBUTION. |
*****************************************************************************
 ****************************************************************************
 | AVOID USING PRE-BUILT LSOF BINARIES: SEE THE "PRE-BUILT LSOF BINARIES" |
 |              SECTION IN 00README FOR AN EXPLANATION.              |
 ****************************************************************************


  ***************************************************************
  | READ 00LSOF-L FOR INFORMATION ON THE LSOF-L LISTSERV MAILING LIST. |
  ***************************************************************


   ***************************************************************
   |     CHECK 00FAQ BEFORE REPORTING BUGS TO <abe@purdue.edu>.      |
   | 00FAQ ALSO AT: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/FAQ |
   ***************************************************************
```

```
**********************************************************************
| IMPORTANT!  This README file explains how the lsof tar archive   |
| is assembled -- it's a "wrapper" tar archive.  Please read the   |
| explanation of its naming and construction, immediately          |
| following the initial list of supported dialects.                |
**********************************************************************
```

Lsof version 4 lists open files for running UNIX processes. It is a
descendent of ofiles, fstat, and lsof versions 1, 2, and 3. It has
been tested recently on these UNIX dialects.

AIX 5.3

Apple Darwin 9 and Mac OS X 10.[56]

FreeBSD 4.9 and 6.4 for x86-based systems

FreeBSD 8.[02] and 9.0 for AMD64-based systems

Linux 2.1.72 and above for x86-based systems

Solaris 9, 10 and 11

Lsof 4 may work on other versions of these dialects, but hasn't been tested there recently. Lsof versions 2 and 3 are still available and may provide older dialect version support. See the notes on them in this file.

The pub/tools/unix/lsof/contrib directory on lsof.itap.purdue.edu also contains information on other ports.

Version 4 of lsof is distributed as bzip2'd, gzip'd and compressed tar archives in the files:

ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/lsof.tar.bz2
and
ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/lsof.tar.gz
and
ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/lsof.tar.Z

These files are links to the current distribution, whose name includes the revision number:

ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof_<rev>.tar.bz2

and

ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof_<rev>.tar.gz

and

ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof_<rev>.tar.Z

<rev> is the revision number -- e.g., 4.85. These archives are called wrappers, because the lsof source tar archive, its GPG certificate (lsof_<rev>_src.tar.sig), and some documentation files are wrapped together inside them. (The GPG certificate authenticates the source tar archive.) A tar archive with: a ``.bz2'' suffix has been compressed with bzip2; ``.gz'', with gzip; and ``.Z'', with compress.

When the wrapper tar is gunzip'd or uncompressed, and its tar archive contents are extracted, an lsof_4.85 subdirectory is created in the directory where the extraction was performed. The lsof_4.85 subdirectory contains these files:

00.README.FIRST	contains introductory distribution information.
README.lsof_4.85	contains instructions for the security-conscious on how to be sure that no one has tampered with the distribution.

RELEASE_SUMMARY_4.85 is this file.

lsof_4.85_src.tar is a tar archive, containing the lsof sources. When extracted with tar it creates a subdirectory named lsof_4.85_src in the directory where the extraction was performed. The lsof source files will be found in lsof_4.85_src.

lsof_4.85_src.tar.sig is a GPG certificate, authenticating the lsof_4.85_src.tar archive. See the README.lsof_4.85 file for more information on GPG authentication of lsof_4.85_src.tar.

If you've obtained this file and an lsof distribution from a mirror site, please be aware that THE LATEST VERSION OF LSOF IS AVAILABLE VIA ANONYMOUS FTP FROM LSOF.ITAP.PURDUE.EDU IN THE PUB/TOOLS/UNIX/LSOF DIRECTORY.

Patches to lsof distributions may be found in the patches/ sub-directory where you found lsof.tar.bz2, lsof.tar.gz or lsof.tar.Z. If there are any patches to the current distribution, they will be found in the patches/4.85/ branch.

(If you need a copy of gunzip, look for it at prep.ai.mit.edu in

pub/gnu/gzip*.)

* The September 27, 2011 revision (4.85): adds an automatic work-around for
  an lgrp_root conflict in some Solaris 9 and 10 versions; supports FreeBSD
  7.4 and 8.[12] (8.1 not tested); adds fixes for Solaris 11 kernel module
  path determination; picked lint for Linux; added more Linux cross
  configuration support; adds support for Mac OS X 10.6; tested on FreeBSD
  6.4; adapts to FreeBSD ZFS update; drops support for FreeBSD 7.x; adjusts
  for Solaris 10 with patch 144488-10; added Linux +|-e option support;
  adjusts for a FreeBSD 9 change; fixes a Linux AF_UNIX path reporting bug;
  adjusts for dropping of RPC headers from Linux GlibC 2.14; adds Linux
  Netlink protocol support; corrects UDP6-lite Linux path.

Read the 00.README.FIRST in the lsof distribution first.

Read the 00DIST distribution file for more details on feature additions

and bug fixes.

The 00README distribution file has build instructions, dialect

descriptions, special feature discussions, and installation hints.

The 00FAQ file contains a list of frequently asked questions and their

answers.

The 00DCACHE file explains device cache file path formation.

The 00PORTING file contains information on porting lsof to other UNIX dialects.

The 00QUICKSTART file gives a quick introduction to using lsof.

The distribution files lsof.8 (nroff source) and lsof.man (nroff formatted output) contain the manual page for lsof; it is the only other documentation besides the source code (it's included).

Version 4 Binaries

==========

Version 4 binaries for some revisions, dialects, and platforms may be found in pub/tools/unix/lsof/binaries. Check the README files for exact descriptions. Check the dialect-specific Makefiles for installation instructions. CHECKSUMS and GPG certificates are provided for authentication.

Please think very carefully before you decide to use a pre-built binary instead of making your own from the sources. Here are some points to consider:

1. Lsof must run setgid or setuid. Are you willing to trust that
   power to a binary you didn't construct yourself?

2. Lsof binaries may be generated on a system whose configuration
header files differ from yours. Under Digital UNIX (DEC OSF/1), for
example, lsof includes header files from the machine's configuration
directory, /sys/<name>. Are you willing to gamble that your
configuration directory's header files match the ones used to
compile lsof?

3. Lsof is often configured with specific options that are determined
from the configuration of the system on which it is configured --
e.g., Solaris patch level, dynamic loader libraries, etc. Are you
sure that the lsof binary you retrieve will have been configured for
your system? If you get a binary that is misconfigured for you, it
may not work at all.

If you haven't already guessed, I believe firmly that you should
retrieve sources and build your own binary. If you still want to use
the distribution binaries, please authenticate what you retrieved with
the GPG certificates; please compare checksums, too.

Version 4 Checksums

==========

Security checksums -- both MD5 and sum(1) -- for revisions of lsof
version 4 are contained in the README.lsof_<rev> files in the wrapper
tar archives of pub/tools/unix/lsof.

The CHECKSUMS file, found with the distribution archives, contains information on validating the archives with external MD5 checksums and external GPG certificates.

GPG Certificates

==========

The lsof wrapper tar archive includes a GPG certificate file in its contained lsof_4.71_src.tar.sig file.

Binary files have detached GPG certificates that may be found in their directories with ".sig" extensions.

The certificates are signed with my GPG public key, which may be found in the file:

ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/Victor_A_Abell.gpg

My key may also be available at some public key servers,

There is also authentication information in the CHECKSUMS file (a link to CHECKSUMS_<rev>), found with the lsof distribution files. CHECKSUMS contains external MD5 checksums for the distribution files and information on using the external GPG certificates, found with the lsof distribution files.

Old Dialect Support

==========

Remnants of source code and binaries for dialects for which lsof once provided support may be obtained by request. Send the request to abe@purdue.edu.

Dialects no longer supported include:

CDC EP/IX

MIPS RISC/os

Motorola V/88

Pyramid DC/OSx

Pyramid Reliant UNIX

Sequent DYNIX

SGI IRIX

SunOS 4.1.x

Ultrix

Generally I drop support for a dialect when I no longer have access to a test system.

Lsof Version 2

==========

The version 3 predecessor, revision 36 of version 2, is also available upon request. Send the request to abe@purdue.edu.

I recommend you avoid lsof version 2. It's out of date and I no longer provide support for it. (Versions 3 and 4 support more dialects, and have many enhancements, bug fixes, and improvements.)

Version 2 was tested on the following UNIX dialects:

AIX 3.2.[1234] for the IBM RISC/System 6000

DEC OSF/1 1.[23] and 2.0 for the DEC Alpha

EP/IX 1.4.3 and 2.1.1 for the CDC 4680

ETAV 1.17 for the ETA-10P*

FreeBSD 1.0e for x86-based systems

HP-UX [789].x for HP systems

IRIX 4.0.5 and 5.1.1 for SGI systems

NEXTSTEP 2.1, 3.0, 3.1 for NeXT systems

Sequent Dynix 3.0.12 for Sequent Symmetry systems

SunOS 4.1.[123] for Sun 3 and 4 systems

SunOS 5.[13] (Solaris 2.[13]) for Sun 4 systems

Ultrix 2.2 and 4.2 for DEC systems

(If you need a copy of gunzip, look for it at prep.ai.mit.edu in pub/gnu.)

Version 2 Checksums

==========

MD5:

(OLD/lsof236tar.gz) = f8a1ab3971ea2f6a3ea16752f84409e8

sum(1):

39996 106 OLD/lsof236tar.gz

The file OLD/lsof236tar.gz.asc is a detached PGP certificate that may

be used to authenticate OLD/lsof236tar.gz with my PGP public key. You

may find my PGP public key at:

ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/OLD/Victor_A_Abell.pgp

Lsof Version 3

==========

The last revision of lsof version 3, 3.88, may obtained by request.

Send the request to abe@purdue.edu.

I recommend version 4 over version 3. It is the version I actively

support.

Lsof version 3 was tested on these UNIX dialects:

AIX 3.2.5, 4.1[.[1234]], and 4.2

BSDI BSD/OS 2.0, 2.0.1, and 2.1 for x86-based systems

DC/OSx 1.1 for Pyramid systems

Digital UNIX (DEC OSF/1) 2.0, 3.0, 3.2, and 4.0

EP/IX 2.1.1 for the CDC 4680

FreeBSD 1.1.5.1, 2.0, 2.0.5, 2.1, 2.1.5 for x86-based systems

HP-UX 8.x, 9.x, 10.01, 10.10, and 10.20

IRIX 5.2, 5.3, 6.0, 6.0.1, and 6.[124]

Linux 2.0.3[01] and 2.1.57 for x86-based systems

NetBSD 1.0, 1.1, and 1.2 for x86 and SPARC-based systems

NEXTSTEP 2.1 and 3.[0123] for NEXTSTEP architectures

OpenBSD 1.2 and 2.0 for x86-based systems

Reliant UNIX 5.43 for Pyramid systems

RISC/os 4.52 for MIPS R2000-based systems

SCO OpenServer 1.1, 3.0, and 5.0.[024] for x86-based systems

SCO UnixWare 2.1 and 2.1.1 for x86-based systems

Sequent PTX 2.1.[1569], 4.0.[23], 4.1.[024], 4.2[.1], and 4.3

Solaris 2.[12345], 2.5.1, and 2.6-Beta

SunOS 4.1.x

Ultrix 4.2, 4.3, 4.4, and 4.5

Vic Abell <abe@purdue.edu>

September 27, 2011

LibXml2 (v2.7.8):

Except where otherwise noted in the source code (e.g. the files hash.c,
list.c and the trio files, which are covered by a similar licence but
with different Copyright notices) all the files are:

Copyright (C) 1998-2003 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is fur-
nished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FIT-
NESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CON-
NECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not

be used in advertising or otherwise to promote the sale, use or other deal-

ings in this Software without prior written authorization from him.

Memstat (v0.8):

This is Debian GNU/Linux's prepackaged version of Joshua M. Yelon's memstat,

for a long time maintained upstream by Bernd Eckenfels <ecki@debian.org> and

now maintained by Michael Meskes <meskes@debian.org>.

This package was put together by me, Bernd Eckenfels <ecki@debian.org>,

from the sources, which I obtained from

http://charm.cs.uiuc.edu/~jyelon/software.html

The debian/* Files are based on Ian Jacksons hello Package.

All patches by me are subject to the GPL.

Original Copyright from memstat.c:

```
 * This software copyright 1997 Joshua M. Yelon.
 * Distribution subject to the terms of the GPL.
```

On Debian GNU/Linux systems, the complete text of the GNU General

Public License can be found in `/usr/share/common-licenses/GPL'.

Ncurses (v5.7):

-- Copyright (c) 1998-2004,2006 Free Software Foundation, Inc. --

-- --

-- Permission is hereby granted, free of charge, to any person obtaining a --

-- copy of this software and associated documentation files (the --

-- "Software"), to deal in the Software without restriction, including --

-- without limitation the rights to use, copy, modify, merge, publish, --

-- distribute, distribute with modifications, sublicense, and/or sell copies --

-- of the Software, and to permit persons to whom the Software is furnished --

-- to do so, subject to the following conditions: --

-- --

-- The above copyright notice and this permission notice shall be included --

-- in all copies or substantial portions of the Software. --

-- --

-- THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS --

-- OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF --

-- MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN --

-- NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, --

-- DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR --

-- OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE --

-- USE OR OTHER DEALINGS IN THE SOFTWARE. --

-- --

-- Except as contained in this notice, the name(s) of the above copyright --

-- holders shall not be used in advertising or otherwise to promote the --

-- sale, use or other dealings in this Software without prior written --

-- authorization. --

-- $Id: README,v 1.23 2006/04/22 22:19:37 tom Exp $

README file for the ncurses package

See the file ANNOUNCE for a summary of ncurses features and ports.

See the file INSTALL for instructions on how to build and install ncurses.

See the file NEWS for a release history and bug-fix notes.

See the file TO-DO for things that still need doing, including known bugs.

Browse the file misc/ncurses-intro.html for narrative descriptions of how to use ncurses and the panel, menu, and form libraries.

Browse the file doc/html/hackguide.html for a tour of the package internals.

ROADMAP AND PACKAGE OVERVIEW:

You should be reading this file in a directory called: ncurses-d.d, where d.d is the current version number (see the dist.mk file in this directory for that). There should be a number of subdirectories, including `c++', `form', `man', `menu', `misc', `ncurses', `panel', `progs', `test', 'tack' and `Ada95'. (The 'tack' program may be distributed separately).

A full build/install of this package typically installs several libraries, a handful of utilities, and a database hierarchy. Here is an inventory of the pieces:

The libraries are:

libncurses.a (normal)

libncurses.so (shared)

libncurses_g.a (debug and trace code enabled)

libncurses_p.a (profiling enabled)

libpanel.a (normal)

libpanel.so (shared)

libpanel_g.a (debug and trace code enabled)

libmenu.a (normal)

libmenu.so (shared)

libmenu_g.a (debug enabled)

libform.a (normal)

libform.so (shared)

libform_g.a (debug enabled)

If you configure using the --enable-widec option, a "w" is appended to the library names (e.g., libncursesw.a), and the resulting libraries support wide-characters, e.g., via a UTF-8 locale. The corresponding header files are compatible with the non-wide-character configuration; wide-character features are provided by ifdef's in the header files. The wide-character library interfaces are not binary-compatible with the non-wide-character version.

The ncurses libraries implement the curses API. The panel, menu and forms libraries implement clones of the SVr4 panel, menu and forms APIs. The source code for these lives in the `ncurses', `panel', `menu', and `form' directories respectively.

In the `c++' directory, you'll find code that defines an interface to the curses, forms, menus and panels library packaged as C++ classes, and a demo program in C++ to test it. These class definition modules are not installed by the 'make install.libs' rule as libncurses++.

In the `Ada95' directory, you'll find code and documentation for an Ada95 binding of the curses API, to be used with the GNAT compiler. This binding is built by a normal top-level `make' if configure detects an usable version of GNAT (3.11 or above). It is not installed automatically. See the Ada95 directory for more build and installation instructions and for documentation of the binding.

To do its job, the ncurses code needs your terminal type to be set in the environment variable TERM (normally set by your OS; under UNIX, getty(1) typically does this, but you can override it in your .profile); and, it needs a database of terminal descriptions in which to look up your terminal type's capabilities.

In older (V7/BSD) versions of curses, the database was a flat text file, /etc/termcap; in newer (USG/USL) versions, the database is a hierarchy of fast-loading binary description blocks under /usr/lib/terminfo. These binary blocks are compiled from an improved editable text representation called

`terminfo' format (documented in man/terminfo.5). The ncurses library can use either /etc/termcap or the compiled binary terminfo blocks, but prefers the second form.

In the `misc' directory, there is a text file terminfo.src, in editable terminfo format, which can be used to generate the terminfo binaries (that's what make install.data does). If the package was built with the --enable-termcap option enabled, and the ncurses library cannot find a terminfo description for your terminal, it will fall back to the termcap file supplied with your system (which the ncurses package installation leaves strictly alone).

The utilities are as follows:

tic -- terminfo source to binary compiler

infocmp -- terminfo binary to source decompiler/comparator

clear -- emits clear-screen for current terminal

tput -- shell-script access to terminal capabilities.

toe -- table of entries utility

tset -- terminal-initialization utility

The first two (tic and infocmp) are used for manipulating terminfo descriptions; the next two (clear and tput) are for use in shell scripts. The last (tset) is provided for 4.4BSD compatibility. The source code for all of these lives in the `progs' directory.

Detailed documentation for all libraries and utilities can be found in the `man' and `doc' directories. An HTML introduction to ncurses, panels, and menus programming lives in the `doc/html' directory. Manpages in HTML format are under `doc/html/man'.

The `test' directory contains programs that can be used to verify or demonstrate the functions of the ncurses libraries. See test/README for descriptions of these programs. Notably, the `ncurses' utility is designed to help you systematically exercise the library functions.

AUTHORS:

Pavel Curtis:

wrote the original ncurses

Zeyd M. Ben-Halim:

port of original to Linux and many enhancements.

Thomas Dickey (maintainer for 1.9.9g through 4.1, resuming with FSF's 5.0):

configuration scripts, porting, mods to adhere to XSI Curses in the areas of background color, terminal modes. Also memory leak testing, the wresize, default colors and key definition extensions and numerous bug fixes (more than half of those enumerated in NEWS beginning with the internal release 1.8.9).

Florian La Roche (official maintainer for FSF's ncurses 4.2)

Beginning with release 4.2, ncurses is distributed under an MIT-style license.

Eric S. Raymond:

the man pages, infocmp(1), tput(1), clear(1), captoinfo(1), tset(1), toe(1), most of tic(1), trace levels, the HTML intro, wgetnstr() and many other entry points, the cursor-movement optimization, the scroll-pack optimizer for vertical motions, the mouse interface and xterm mouse support, and the ncurses test program.

Juergen Pfeifer

The menu and form libraries, C++ bindings for ncurses, menus, forms and panels, as well as the Ada95 binding. Ongoing support for panel.

CONTRIBUTORS:

Alexander V. Lukyanov

for numerous fixes and improvements to the optimization logic.

David MacKenzie

for first-class bug-chasing and methodical testing.

Ross Ridge

for the code that hacks termcap parameterized strings into terminfo.

Warren Tucker and Gerhard Fuernkranz,

for writing and sending the panel library.

Hellmuth Michaelis,

for many patches and testing the optimization code.

Eric Newton, Ulrich Drepper, and Anatoly Ivasyuk:

the C++ code.

Jonathan Ross,

for lessons in using sed.

Keith Bostic (maintainer of 4.4BSD curses)

for help, criticism, comments, bug-finding, and being willing to

deep-six BSD curses for this one when it grew up.

Richard Stallman,

for his commitment to making ncurses free software.

Countless other people have contributed by reporting bugs, sending fixes,

suggesting improvements, and generally whining about ncurses :-)

BUGS:

See the INSTALL file for bug and developer-list addresses.

The Hacker's Guide in the doc directory includes some guidelines

on how to report bugs in ways that will get them fixed most quickly.

OpenSSL (v1.0.1g):

LICENSE ISSUES

==========

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

/* ==

 * Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

 *

 * Redistribution and use in source and binary forms, with or without

 * modification, are permitted provided that the following conditions

 * are met:

 *

 * 1. Redistributions of source code must retain the above copyright

 *   notice, this list of conditions and the following disclaimer.

 *

 * 2. Redistributions in binary form must reproduce the above copyright

 *   notice, this list of conditions and the following disclaimer in

 *   the documentation and/or other materials provided with the

 *   distribution.

 *

 * 3. All advertising materials mentioning features or use of this

 *   software must display the following acknowledgment:

 *   "This product includes software developed by the OpenSSL Project

 *   for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

*

* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to

* endorse or promote products derived from this software without

* prior written permission. For written permission, please contact

* openssl-core@openssl.org.

*

* 5. Products derived from this software may not be called "OpenSSL"

* nor may "OpenSSL" appear in their names without prior written

* permission of the OpenSSL Project.

*

* 6. Redistributions of any form whatsoever must retain the following

* acknowledgment:

* "This product includes software developed by the OpenSSL Project

* for use in the OpenSSL Toolkit (http://www.openssl.org/)"

*

* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY

* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR

* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR

* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,

* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,

* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED

* OF THE POSSIBILITY OF SUCH DAMAGE.

* ==

*

* This product includes cryptographic software written by Eric Young

* (eay@cryptsoft.com). This product includes software written by Tim

* Hudson (tjh@cryptsoft.com).

*

*/

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)

* All rights reserved.

*

* This package is an SSL implementation written

* by Eric Young (eay@cryptsoft.com).

* The implementation was written so as to conform with Netscapes SSL.

*

* This library is free for commercial and non-commercial use as long as

* the following conditions are aheared to. The following conditions

* apply to all code found in this distribution, be it the RC4, RSA,

* lhash, DES, etc., code; not just the SSL code. The SSL documentation

* included with this distribution is covered by the same copyright terms

* except that the holder is Tim Hudson (tjh@cryptsoft.com).

*

* Copyright remains Eric Young's, and as such any Copyright notices in

* the code are not to be removed.

* If this package is used in a product, Eric Young should be given attribution

* as the author of the parts of the library used.

* This can be in the form of a textual message at program startup or

* in documentation (online or textual) provided with the package.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the copyright

*   notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

*   notice, this list of conditions and the following disclaimer in the

*   documentation and/or other materials provided with the distribution.

* 3. All advertising materials mentioning features or use of this software

*   must display the following acknowledgement:

*   "This product includes cryptographic software written by

*   Eric Young (eay@cryptsoft.com)"

*   The word 'cryptographic' can be left out if the rouines from the library

*   being used are not cryptographic related :-).

* 4. If you include any Windows specific code (or a derivative thereof) from

*   the apps directory (application code) you must include an acknowledgement:

*   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

*

* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND

* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE

* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

* SUCH DAMAGE.

*

* The licence and distribution terms for any publically available version or

* derivative of this code cannot be changed. i.e. this code cannot simply be

* copied and put under another distribution licence

* [including the GNU Public Licence.]

*/

Pcre (v8.30):

PCRE LICENCE

PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 8 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself.

The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions, and a just-in-time compiler that can be used to optimize pattern matching. These are both optional features that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS

Written by: Philip Hazel

Email local part: ph10

Email domain: cam.ac.uk

University of Cambridge Computing Service,

Cambridge, England.

Copyright (c) 1997-2014 University of Cambridge

All rights reserved.

PCRE JUST-IN-TIME COMPILATION SUPPORT

Written by: Zoltan Herczeg

Email local part: hzmester

Emain domain: freemail.hu

Copyright(c) 2010-2014 Zoltan Herczeg

All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER

Written by: Zoltan Herczeg

Email local part: hzmester

Emain domain: freemail.hu

Copyright(c) 2009-2014 Zoltan Herczeg

All rights reserved.

THE C++ WRAPPER FUNCTIONS

Contributed by: Google Inc.

Copyright (c) 2007-2012, Google Inc.

All rights reserved.

THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End

Popt (v1.16):

This is the popt(3) command line option parsing library. While it is similiar to getopt(3), it contains a number of enhancements, including:

1) popt is fully reentrant
2) popt can parse arbitrary argv[] style arrays while
   getopt(3) makes this quite difficult
3) popt allows users to alias command line arguments
4) popt provides convience functions for parsing strings
   into argv[] style arrays

Complete documentation on popt(3) is available in popt.ps (included in this tarball), which is excerpted with permission from the book "Linux Application Development" by Michael K. Johnson and Erik Troan (available from Addison Wesley in May, 1998).

Comments on popt should be addressed to popt-devel@rpm5.org.

ROM-Bootloader:

Boot strategies

AT91 chips embed a boot ROM code. It is enabled depending on BMS (Boot Mode Select) pin state on reset.

The ROM code scans the contents of different media like serial FLASH, NAND FLASH, SD/MMC Card and serial EEPROM.

If a valid application is available then it downloads this application into the chip internal SRAM and runs it.

To determine if a valid application is present the ROM code checks the eight ARM exception vectors.

If no valid application is available then SAM-BA Monitor is executed. It waits for transactions either on the USB device,

or on the DBGU serial port, then the SAM-BA tool can be used to program FLASH or EEPROM present on your board.

For more information on this topic, please check the corresponding SAM product datasheet section Boot Strategies.

GNU Tar (v1.17):

README for GNU tar

See the end of file for copying conditions.

* Introduction

Please glance through *all* sections of this

'README' file before starting configuration. Also make sure you read files

'ABOUT-NLS' and 'INSTALL' if you are not familiar with them already.

If you got the 'tar' distribution in 'shar' format, time stamps ought to be

properly restored; do not ignore such complaints at 'unshar' time.

GNU 'tar' saves many files together into a single tape or disk
archive, and can restore individual files from the archive. It includes
multivolume support, the ability to archive sparse files, automatic archive
compression/decompression, remote archives and special features that allow
'tar' to be used for incremental and full backups. This distribution
also includes 'rmt', the remote tape server. The 'mt' tape drive control
program is in the GNU 'cpio' distribution.

GNU 'tar' is derived from John Gilmore's public domain 'tar'.

See file 'ABOUT-NLS' for how to customize this program to your language.
See file 'COPYING' for copying conditions.
See file 'INSTALL' for compilation and installation instructions.
See file 'PORTS' for various ports of GNU tar to non-Unix systems.
See file 'NEWS' for a list of major changes in the current release.
See file 'THANKS' for a list of contributors.

Besides those configure options documented in files 'INSTALL' and
'ABOUT-NLS', an extra option may be accepted after './configure':

* Install

** Selecting the default archive format.

The default archive format is GNU, this can be overridden by
presetting DEFAULT_ARCHIVE_FORMAT while configuring. The allowed
values are GNU, V7, OLDGNU, USTAR and POSIX.

** Selecting the default archive device

The default archive device is now 'stdin' on read and 'stdout' on write.
The installer can still override this by presetting 'DEFAULT_ARCHIVE'
in the environment before configuring (the behavior of '-[0-7]' or
'-[0-7]lmh' options in 'tar' are then derived automatically). Similarly,
'DEFAULT_BLOCKING' can be preset to something else than 20.

** Selecting full pathname of the "rmt" binary.

Previous versions of tar always looked for "rmt" binary in the
directory "/etc/rmt". However, the "rmt" program included
in the distribution was installed under "$prefix/libexec/rmt".
To fix this discrepancy, tar now looks for "$prefix/libexec/rmt".
If you do not want this behavior, specify full path name of
"rmt" binary using DEFAULT_RMT_DIR variable, e.g.:

./configure DEFAULT_RMT_DIR=/etc

If you already have a copy of "rmt" installed and wish to use it
instead of the version supplied with the distribution, use --with-rmt
option:

./configure --with-rmt=/etc/rmt

This will also disable building the included version of rmt.

** Installing backup scripts.

This version of tar is shipped with the shell scripts for producing
incremental backups (dumps) and restoring filesystems from them.
The name of the backup script is "backup". The name of the
restore script is "restore". They are installed in "$prefix/sbin"
directory.

Use option --enable-backup-scripts to compile and install these
scripts.

** '--disable-largefile' omits support for large files, even if the
operating system supports large files. Typically, large files are
those larger than 2 GB on a 32-bit host.

* Installation hints

Here are a few hints which might help installing 'tar' on some systems.

** gzip and bzip2.

GNU tar uses the gzip and bzip2 programs to read and write compressed
archives. If you don't have these programs already, you need to
install them. Their sources can be found at:

ftp://ftp.gnu.org/gnu/gzip/

http://sourceware.cygnus.com/bzip2/

If you see the following symptoms:

```
$ tar -xzf file.tar.gz
gzip: stdin: decompression OK, trailing garbage ignored
tar: Child returned status 2
```

then you have encountered a gzip incompatibility that should be fixed
in gzip test version 1.3, which as of this writing is available at
<ftp://alpha.gnu.org/gnu/gzip/>. You can work around the
incompatibility by using a shell command like
 'gzip -d <file.tar.gz | tar -xzf -'.

** Solaris issues.

GNU tar exercises many features that can cause problems with older GCC
versions. In particular, GCC 2.8.1 (sparc, -O1 or -O2) is known to
miscompile GNU tar. No compiler-related problems have been reported
when using GCC 2.95.2 or later.

Recent versions of Solaris tar sport a new -E option to generate extended headers in an undocumented format. GNU tar does not understand these headers.

** Static linking.

Some platform will, by default, prepare a smaller 'tar' executable which depends on shared libraries. Since GNU 'tar' may be used for system-level backups and disaster recovery, installers might prefer to force static linking, making a bigger 'tar' executable maybe, but able to work standalone, in situations where shared libraries are not available. The way to achieve static linking varies between systems. Set LDFLAGS to a value from the table below, before configuration (see 'INSTALL').

Platform	Compiler	LDFLAGS
(any)	Gnu C	-static
AIX	(vendor)	-bnso -bI:/lib/syscalls.exp
HPUX	(vendor)	-Wl,-a,archive
IRIX	(vendor)	-non_shared
OSF	(vendor)	-non_shared
SCO 3.2v5	(vendor)	-dn
Solaris	(vendor)	-Bstatic
SunOS	(vendor)	-Bstatic

** Failed tests 'ignfail.sh' or 'incremen.sh'.

In an NFS environment, lack of synchronization between machine clocks
might create difficulties to any tool comparing dates and file time stamps,
like 'tar' in incremental dumps. This has been a recurrent problem with
GNU Make for the last few years. We would like a general solution.

** BSD compatibility matters.

Set LIBS to '-lbsd' before configuration (see 'INSTALL') if the linker
complains about 'bsd_ioctl' (Slackware). Also set CPPFLAGS to
'-I/usr/include/bsd' if <sgtty.h> is not found (Slackware).

** OPENStep 4.2 swap files

Tar cannot read the file /private/vm/swapfile.front (even as root).
This file is not a real file, but some kind of uncompressed view of
the real compressed swap file; there is no reason to back it up, so
the simplest workaround is to avoid tarring this file.

* Special topics

Here are a few special matters about GNU 'tar', not related to build
matters. See previous section for such.

** File attributes.

About *security*, it is probable that future releases of 'tar' will have
some behavior changed. There are many pending suggestions to choose from.

Today, extracting an archive not being 'root', 'tar' will restore suid/sgid bits on files but owned by the extracting user. 'root' automatically gets a lot of special privileges, '-p' might later become required to get them.

GNU 'tar' does not properly restore symlink attributes. Various systems implement flavors of symbolic links showing different behavior and properties. We did not successfully sorted all these out yet. Currently, the 'lchown' call will be used if available, but that's all.

** POSIX compliance.

GNU 'tar' is able to create archive in the following formats:

*** The format of UNIX version 7

*** POSIX.1-1988 format, also known as "ustar format"

*** POSIX.1-2001 format, also known as "pax format"

*** Old GNU format (described below)

In addition to those, GNU 'tar' is also able to read archives produced by 'star' archiver.

A so called 'Old GNU' format is based on an early draft of the POSIX 1003.1 'ustar' standard which is different from the final standard. It defines its extensions (such as incremental backups and handling of the long file names) in a way incompatible with any existing tar archive format, therefore the use of old GNU format is strongly discouraged.

Please read the file NEWS for more information about POSIX compliance
and new 'tar' features.

* What's next?

GNU tar will be merged into GNU paxutils: a project containing
several utilities related to creating and handling archives in
various formats. The project will include tar, cpio and pax
utilities.

* Bug reporting.

Send bug reports to <bug-tar@gnu.org>. A bug report should contain
an adequate description of the problem, your input, what you expected,
what you got, and why this is wrong. Diffs are welcome, but they only
describe a solution, from which the problem might be uneasy to infer.
If needed, submit actual data files with your report. Small data files
are preferred. Big files may sometimes be necessary, but do not send them
to the report address; rather take special arrangement with the maintainer.

Your feedback will help us to make a better and more portable package.
Consider documentation errors as bugs, and report them as such. If you
develop anything pertaining to 'tar' or have suggestions, let us know
and share your findings by writing to <bug-tar@gnu.org>.

* Copying

Copyright 1990-1992, 1994, 1997-2001, 2003-2004, 2007, 2012-2013 Free

Software Foundation, Inc.

This file is part of GNU tar.

GNU tar is free software; you can redistribute it and/or modify

it under the terms of the GNU General Public License as published by

the Free Software Foundation; either version 3 of the License, or

(at your option) any later version.

GNU tar is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

GNU General Public License for more details.

You should have received a copy of the GNU General Public License

along with this program. If not, see <http://www.gnu.org/licenses/>.

NOTE ON COPYRIGHT YEARS

In copyright notices where the copyright holder is the Free Software Foundation, then where a range of years appears, this is an inclusive range that applies to every year in the range. For example: 2005-2008 represents the years 2005, 2006, 2007, and 2008.

Local Variables:

mode: outline

paragraph-separate: "[]*$"

version-control: never

End:

---

u-boot (v2010.09):

U-Boot is Free Software. It is copyrighted by Wolfgang Denk and many others who contributed code (see the actual source code and the git commit messages for details). You can redistribute U-Boot and/or modify it under the terms of version 2 of the GNU General Public License as published by the Free Software Foundation. Most of it can also be distributed, at your option, under any later version of the GNU General Public License -- see individual files for exceptions.

NOTE! This license does not cover the so-called "standalone" applications that use U-Boot services by means of the jump table provided by U-Boot exactly for this purpose - this is merely considered normal use of U-Boot, and does not fall under the heading of "derived work" -- see file Licenses/Exceptions for details.

Also note that the GPL and the other licenses are copyrighted by the Free Software Foundation and other organizations, but the instance of code that they refer to (the U-Boot source code) is copyrighted by me and others who actually wrote it.

-- Wolfgang Denk

Like many other projects, U-Boot had a tradition of including big blocks of License headers in all files. This not only blew up the source code with mostly redundant information, but also made it very difficult to generate License Clearing Reports. An additional problem was that even the same lincenses were referred to by a number of slightly varying text blocks (full, abbreviated, different indentation, line wrapping and/or white space, with obsolete address information, ...) which made automatic processing a nightmare.

To make this easier, such license headers in the source files have been replaced with a single line reference to Unique Lincense Identifiers as defined by the Linux Foundation's SPDX project [1]. For example, in a source file the full "GPL v2.0 or later" header text was replaced by a single line:

SPDX-License-Identifier: GPL-2.0+

We use the SPDX Unique Lincense Identifiers here; these are available at [2].

[1] http://spdx.org/

[2] http://spdx.org/licenses/

uClibc (v0.9.31.1):

A C library for embedded Linux

uClibc (aka µClibc/pronounced yew-see-lib-see) is a C library for developing embedded Linux systems. It is much smaller than the GNU C Library, but nearly all applications supported by glibc also work perfectly with uClibc. Porting applications from glibc to uClibc typically involves just recompiling the source code. uClibc even supports shared libraries and threading. It currently runs on standard Linux and MMU-less (also known as µClinux) systems with support for alpha, amd64, ARM, Blackfin, cris, h8300, hppa, i386, i960, ia64, m68k, mips/mipsel, PowerPC, SH, SPARC, and v850 processors.

If you are building an embedded Linux system and you find that glibc is eating up too much space, you may want to consider using uClibc. If you are building a huge fileserver with 12 Terabytes of storage, then using glibc may make more sense. Unless, for example, that 12 Terabytes will be Network Attached Storage and you plan to burn Linux into the system's firmware...

uClibc is maintained by Erik Andersen and is licensed under the GNU LESSER GENERAL PUBLIC LICENSE . This license allows you to make closed source commercial applications using uClibc. (Please consider sharing some of the money you make ;-). You do not need to give away all your source code just because you use uClibc and/or run on Linux. See the list of Frequently Asked Questions for details.

Sponsors

Please visit our sponsors and thank them for their support! They have provided money for equipment and bandwidth. Next time you need help with a project, consider these fine companies!

Penguru Consulting
Custom development for embedded Linux systems and multimedia platforms

opensource.se
Embedded open source consulting in Europe.

Codepoet Consulting
Custom Linux, embedded Linux, BusyBox, and uClibc development.

If you wish to be a sponsor, or if you have already contributed and would like your name added here, email Erik.

Do you like uClibc? Do you need support? Do you need some features added? Then why not help out? We are happy to accept donations (such as bandwidth, mirrors sites, and hardware for the various architectures). We can also provide support contracts, and implement funded feature requests. To contribute, you can either click on the Donate image to donate using PayPal, or you can contact Erik at CodePoet Consulting (we have a credit card machine so you can avoid PayPal if you wish).

Udev (v058):

udev - a userspace implementation of devfs

For more information on the design, and structure of this project, see the

files in the docs/ directory.

To use:

- You must be running a 2.6 version of the Linux kernel.

- Your 2.6 kernel must have had CONFIG_HOTPLUG enabled when it was built.

- Make sure sysfs is mounted. udev will figure out where sysfs is mounted, but

  the traditional place for it is at /sys. You can mount it by hand by running:

        mount -t sysfs none /sys

- Make sure you have the latest version of the linux-hotplug scripts. They are

  available at linux-hotplug.sf.net or from your local kernel.org mirror at:

        kernel.org/pub/linux/utils/kernel/hotplug/

  They are required in order for udev to work properly.

  If for some reason you do not install the hotplug scripts, you must tell the

  kernel to point the hotplug binary at wherever you install udev at. This can

  be done by:

        echo "/sbin/udev" > /proc/sys/kernel/hotplug

- Build the project:

    make

Note:

There are a number of different flags that you can use when building udev. They are as follows:

prefix

set this to the default root that you want udev to be installed into. This works just like the 'configure --prefix' script does. Default value is ". Only override this if you really know what you are doing.

USE_KLIBC

if set to 'true', udev is built and linked against the included version of klibc. Default value is 'false'.

USE_LOG

if set to 'true', udev will emit messages to the syslog when it creates or removes device nodes. This is helpful to see what udev is doing. This is enabled by default. Note, if you are building udev against klibc it is recommended that you disable this option (due to klibc's syslog implementation.)

USE_SELINUX

if set to 'true', udev will be built with SELinux support enabled. This is disabled by default.

DEBUG

if set to 'true', debugging messages will be sent to the syslog as udev is run. Default value is 'false'.

KERNEL_DIR

If this is not set it will default to /lib/modules/`uname -r`/build

This is used if USE_KLIBC=true to find the kernel include directory that klibc needs to build against. This must be set if you are not building udev while running a 2.6 kernel.

So, if you want to build udev using klibc with debugging messages, you would do:

make USE_KLIBC=true DEBUG=true

- Install the project:

make install

This will put the udev binary in /sbin, create the /udev and /etc/udev directories, and place the udev configuration files in /etc/udev. You will probably want to edit the *.rules files to create custom naming rules. More info on how the config files are set up are contained in comments in the files, and is located in the documentation.

- Add and remove devices from the system and marvel as nodes are created and removed in /udev/ based on the device types.

- If you later get sick of it, uninstall it:

make uninstall

Things are still quite rough, but it should work properly. If nothing

seems to happen, make sure your build worked properly by running the

udev-test.pl script as root in the test/ subdirectory of the udev source

tree.

Development and documentation help is very much appreciated, see the TODO

file for a list of things left to be done.

Any comment/questions/concerns please let me and the other udev developers

know by sending a message to the linux-hotplug-devel mailing list at:

linux-hotplug-devel@lists.sourceforge.net

greg k-h

greg@kroah.com

Util-Linux (v2.20.1):

util-linux

util-linux is a random collection of Linux utilities

Note that in years 2006-2010 this project used the name "util-linux-ng".

WEB PAGE:

http://kernel.org/~kzak/util-linux/

MAILING LIST:

E-MAIL: util-linux@vger.kernel.org

URL: http://vger.kernel.org/vger-lists.html#util-linux

DOWNLOAD:

ftp://ftp.kernel.org/pub/linux/utils/util-linux/

SOURCE CODE:

Web interface:

http://git.kernel.org/?p=utils/util-linux/util-linux.git

Checkout:

git clone git://git.kernel.org/pub/scm/utils/util-linux/util-linux.git util-linux

NLS (PO TRANSLATIONS):

PO files are maintained by:

http://translationproject.org/domain/util-linux-ng.html

NEUTRALITY:

The stuff in util-linux should be rather distribution-neutral.

No RPMs/DEBs/... are provided - get yours from your distributor.

VERSION SCHEMA:

Standard releases:

<major>.<minor>[.<maint>[.<bugfix>]]

major = fatal and deep changes

minor = typical release with new features

maint = maintenance releases; bug fixes only

bugfix = unplanned releases for critical/security bugs

Development releases:

<major>.<minor>-rc<N>

COMPILATION:

See the INSTALL file for more details.

Notes:

* use SUID_CFLAGS and SUID_LDFLAGS when you want to define special
compiler options for typical suid programs, for example:

./configure SUID_CFLAGS="-fpie" SUID_LDFLAGS="-pie"

This feature is currently supported for chfn, chsh, newgrp,
write, mount, and umount.

STATIC LINKING:

Use --enable-static-programs[=LIST] configure option when you want to use
statically linked programs.

Note, mount(8) uses get{pw,gr}nam() and getpwuid() functions for
translation from username and groupname to UID and GID. These functions
could be implemented by dynamically loaded independent modules (NSS) in
your libc (e.g. glibc). These modules are not statically linked to
mount(8) and mount.static is still using dlopen() like dynamically
linked version.

The translation won't work in environment where NSS modules are
not installed.

For example normal system (NSS modules are available):

```
# ./mount.static -v -f -n -ouid=kzak /mnt/foo
LABEL=/mnt/foo on /mnt/foo type vfat (rw,uid=500)
                                    ^^^^^^^
```

and without NSS modules:

```
# chroot . ./mount.static -v -f -n -ouid=kzak /mnt/win
LABEL=/mnt/win on /mnt/win type vfat (rw,uid=kzak)
                                    ^^^^^^^^
```

XML-RPC++ (v0.7):

```
#ifndef _XMLRPC_H_
#define _XMLRPC_H_
//
// XmlRpc++ Copyright (c) 2002-2003 by Chris Morley
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
//
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
//
// You should have received a copy of the GNU Lesser General Public
```

```
// License along with this library; if not, write to the Free Software
// Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307
//

/* changed by ise GmbH 2013 */

#if defined(_MSC_VER)
# pragma warning(disable:4786)    // identifier was truncated in debug info
#endif

#ifndef MAKEDEPEND
# include <string>
#endif

#include "dllexport.h"
#include "XmlRpcClient.h"
#include "XmlRpcException.h"
#include "XmlRpcServer.h"
#include "XmlRpcServerProxy.h"
#include "XmlRpcServerMethod.h"
#include "XmlRpcValue.h"
#include "XmlRpcUtil.h"

namespace XmlRpc {
```

```
//! An interface allowing custom handling of error message reporting.
class XMLRPC_DLLEXPORT XmlRpcErrorHandler {
public:
  //! Returns a pointer to the currently installed error handling object.
  static XmlRpcErrorHandler* getErrorHandler()
  { return _errorHandler; }

  //! Specifies the error handler.
  static void setErrorHandler(XmlRpcErrorHandler* eh)
  { _errorHandler = eh; }

  //! Report an error. Custom error handlers should define this method.
  virtual void error(const char* msg) = 0;

  //! Define virtual destructor to avoid compiler warnings
  virtual ~XmlRpcErrorHandler(){};

protected:
  static XmlRpcErrorHandler* _errorHandler;
};
```

```
//! An interface allowing custom handling of informational message reporting.
class XMLRPC_DLLEXPORT XmlRpcLogHandler {
public:
  //! Returns a pointer to the currently installed message reporting object.
  static XmlRpcLogHandler* getLogHandler()
  { return _logHandler; }

  //! Specifies the message handler.
  static void setLogHandler(XmlRpcLogHandler* lh)
  { _logHandler = lh; }

  //! Returns the level of verbosity of informational messages. 0 is no output, 5 is very verbose.
  static int getVerbosity()
  { return _verbosity; }

  //! Specify the level of verbosity of informational messages. 0 is no output, 5 is very verbose.
  static void setVerbosity(int v)
  { _verbosity = v; }

  //! Output a message. Custom error handlers should define this method.
  virtual void log(int level, const char* msg) = 0;

  //! Define virtual destructor to avoid compiler warnings
  virtual ~XmlRpcLogHandler(){};
```

```
 protected:
   static XmlRpcLogHandler* _logHandler;
   static int _verbosity;
 };

 //! Returns log message verbosity. This is short for XmlRpcLogHandler::getVerbosity()
 int XMLRPC_DLLEXPORT getVerbosity();
 //! Sets log message verbosity. This is short for XmlRpcLogHandler::setVerbosity(level)
 void XMLRPC_DLLEXPORT setVerbosity(int level);



 //! Version identifier
 extern const char XMLRPC_VERSION[];

} // namespace XmlRpc

#endif // _XMLRPC_H_
```

Zlib (v1.2.6):

ZLIB DATA COMPRESSION LIBRARY

zlib 1.2.6 is a general purpose data compression library. All the code is thread safe. The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files http://tools.ietf.org/html/rfc1950 (zlib format), rfc1951 (deflate format) and rfc1952 (gzip format).

All functions of the compression library are documented in the file zlib.h (volunteer to write man pages welcome, contact zlib@gzip.org). A usage example of the library is given in the file test/example.c which also tests that the library is working correctly. Another example is given in the file test/minigzip.c. The compression library itself is composed of all source files in the root directory.

To compile all files and run the test program, follow the instructions given at the top of Makefile.in. In short "./configure; make test", and if that goes well, "make install" should work for most flavors of Unix. For Windows, use one of the special makefiles in win32/ or contrib/vstudio/ . For VMS, use make_vms.com.

Questions about zlib should be sent to <zlib@gzip.org>, or to Gilles Vollant <info@winimage.com> for the Windows DLL version. The zlib home page is http://zlib.net/ . Before reporting a problem, please check this site to verify that you have the latest version of zlib; otherwise get the latest version and check whether the problem still exists or not.

PLEASE read the zlib FAQ http://zlib.net/zlib_faq.html before asking for help.

Mark Nelson <markn@ieee.org> wrote an article about zlib for the Jan. 1997 issue of Dr. Dobb's Journal; a copy of the article is available at http://marknelson.us/1997/01/01/zlib-engine/ .

The changes made in version 1.2.6 are documented in the file ChangeLog.

Unsupported third party contributions are provided in directory contrib/ .

zlib is available in Java using the java.util.zip package, documented at http://java.sun.com/developer/technicalArticles/Programming/compression/ .

A Perl interface to zlib written by Paul Marquess <pmqs@cpan.org> is available

at CPAN (Comprehensive Perl Archive Network) sites, including

http://search.cpan.org/~pmqs/IO-Compress-Zlib/ .

A Python interface to zlib written by A.M. Kuchling <amk@amk.ca> is

available in Python 1.5 and later versions, see

http://docs.python.org/library/zlib.html .

zlib is built into tcl: http://wiki.tcl.tk/4610 .

An experimental package to read and write files in .zip format, written on top

of zlib by Gilles Vollant <info@winimage.com>, is available in the

contrib/minizip directory of zlib.

Notes for some targets:

- For Windows DLL versions, please see win32/DLL_FAQ.txt

- For 64-bit Irix, deflate.c must be compiled without any optimization. With

  -O, one libpng test fails. The test works in 32 bit mode (with the -n32

  compiler flag). The compiler bug has been reported to SGI.

- zlib doesn't work with gcc 2.6.3 on a DEC 3000/300LX under OSF/1 2.1 it works

  when compiled with cc.

- On Digital Unix 4.0D (formely OSF/1) on AlphaServer, the cc option -std1 is
  necessary to get gzprintf working correctly. This is done by configure.

- zlib doesn't work on HP-UX 9.05 with some versions of /bin/cc. It works with
  other compilers. Use "make test" to check your compiler.

- gzdopen is not supported on RISCOS or BEOS.

- For PalmOs, see http://palmzlib.sourceforge.net/

Acknowledgments:

  The deflate format used by zlib was defined by Phil Katz. The deflate and
  zlib specifications were written by L. Peter Deutsch. Thanks to all the
  people who reported problems and suggested various improvements in zlib; they
  are too numerous to cite here.

Copyright notice:

(C) 1995-2012 Jean-loup Gailly and Mark Adler

  This software is provided 'as-is', without any express or implied
  warranty. In no event will the authors be held liable for any damages
  arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software. If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving
lengthy legal documents to sign. The sources are provided for free but without
warranty of any kind. The library has been entirely written by Jean-loup
Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in
the file ChangeLog history information documenting your changes. Please read
the FAQ for more information on the distribution of modified source versions.

OSS License texts:

The licenses referred to above (and not set out in detail above) are available here:

GNU General Public License (GPL), Version 3.0

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/ ↗></

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems

arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not

initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

> <one line to give the program's name and a brief idea of what it does.>
>
> Copyright (C) <year> <name of author>
>
> This program is free software: you can redistribute it and/or modify
>
> it under the terms of the GNU General Public License as published by
>
> the Free Software Foundation, either version 3 of the License, or
>
> (at your option) any later version.
>
> This program is distributed in the hope that it will be useful,
>
> but WITHOUT ANY WARRANTY; without even the implied warranty of
>
> MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
>
> GNU General Public License for more details.
>
> You should have received a copy of the GNU General Public License
>
> along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

> <program> Copyright (C) <year> <name of author>
>
> This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
>
> This is free software, and you are welcome to redistribute it
>
> under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/ ↗>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html ↗>.

GNU General Public License (GPL), Version 2.0

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change. b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License. c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or, b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or, c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the program's name and an idea of what it does.

Copyright (C) yyyy name of author

This program is free software; you can redistribute it and/or

modify it under the terms of the GNU General Public License

as published by the Free Software Foundation; either version 2

of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

GNU General Public License for more details.

You should have received a copy of the GNU General Public License

along with this program; if not, write to the Free Software

Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details

type `show w'. This is free software, and you are welcome

to redistribute it under certain conditions; type `show c'

for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright

interest in the program `Gnomovision'

(which makes passes at compilers) written

by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a sub-routine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License ↗ instead of this License

---

GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is

numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you

must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License.

Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

IPsecure Router KNX

Third party software components, notices and OSS license texts

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the library's name and an idea of what it does.

Copyright (C) year name of author

This library is free software; you can redistribute it and/or

modify it under the terms of the GNU Library General Public

License as published by the Free Software Foundation; either

version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Library General Public License for more details.

You should have received a copy of the GNU Library General Public

License along with this library; if not, write to the

Free Software Foundation, Inc., 51 Franklin St, Fifth Floor,

Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in

the library `Frob' (a library for tweaking knobs) written

by James Random Hacker.

signature of Ty Coon, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

GNU General Public License (GPL), Version 1.0

GNU GENERAL PUBLIC LICENSE
Version 1, February 1989

Copyright (C) 1989 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The license agreements of most software companies try to keep users at the mercy of those companies. By contrast, our General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. The General Public License applies to the Free Software Foundation's software and to any other program whose authors commit to using it. You can use it for your programs, too.

When we speak of free software, we are referring to freedom, not price. Specifically, the General Public License is designed to make sure that you have the freedom to give away or sell copies of free software, that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of a such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any work containing the Program or a portion of it, either verbatim or with modifications. Each licensee is addressed as "you".

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this General Public License and to the absence of any warranty; and give any other recipients of the Program a copy of this General Public License along with the Program. You may charge a fee for the physical act of transferring a copy.

2. You may modify your copy or copies of the Program or any portion of it, and copy and distribute such modifications under the terms of Paragraph 1 above, provided that you also do the following:

a) cause the modified files to carry prominent notices stating that you changed the files and the date of any change; and

b) cause the whole of any work that you distribute or publish, that in whole or in part contains the Program or any part thereof, either with or without modifications, to be licensed at no charge to all third parties under the terms of this General Public License (except that you may choose to grant warranty protection to some or all third parties, at your option).

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the simplest and most usual way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this General Public License.

d) You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Mere aggregation of another independent work with the Program (or its derivative) on a volume of a storage or distribution medium does not bring the other work under the scope of these terms.

3. You may copy and distribute the Program (or a portion or derivative of it, under Paragraph 2) in object code or executable form under the terms of Paragraphs 1 and 2 above provided that you also do one of the following:

a) accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Paragraphs 1 and 2 above; or,

b) accompany it with a written offer, valid for at least three years, to give any third party free (except for a nominal charge for the cost of distribution) a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Paragraphs 1 and 2 above; or,

c) accompany it with the information you received as to where the corresponding source code may be obtained. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form alone.)

Source code for a work means the preferred form of the work for making modifications to it. For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs, or for standard header files or definitions files that accompany that operating system.

4. You may not copy, modify, sublicense, distribute or transfer the Program except as expressly provided under this General Public License. Any attempt otherwise to copy, modify, sublicense, distribute or transfer the Program is void, and will automatically terminate your rights to use the Program under this License. However, parties who have received copies, or rights to use copies, from you under this General Public License will not have their licenses terminated so long as such parties remain in full compliance.

5. By copying, distributing or modifying the Program (or any work based on the Program) you indicate your acceptance of this license to do so, and all its terms and conditions.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

7. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of the license which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the license, you may choose any version ever published by the Free Software Foundation.

8. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

9. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

10. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to humanity, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19xx name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (a program to direct compilers to make passes at assemblers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

That's all there is to it!

GNU Library General Public License (LGPL), Version 2.0

GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is

 numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Li-

brary or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the library's name and an idea of what it does.

Copyright (C) year name of author

This library is free software; you can redistribute it and/or

modify it under the terms of the GNU Library General Public

License as published by the Free Software Foundation; either

version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Library General Public License for more details.

You should have received a copy of the GNU Library General Public

License along with this library; if not, write to the

Free Software Foundation, Inc., 51 Franklin St, Fifth Floor,

Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in

the library `Frob' (a library for tweaking knobs) written

by James Random Hacker.

signature of Ty Coon, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

GNU Lesser General Public License (LGPL), Version 2.1

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts

as the successor of the GNU Library Public License, version 2, hence

the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate

your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the library's name and an idea of what it does.

Copyright (C) year name of author

This library is free software; you can redistribute it and/or

modify it under the terms of the GNU Lesser General Public

License as published by the Free Software Foundation; either

version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public

License along with this library; if not, write to the Free Software

Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in

the library `Frob' (a library for tweaking knobs) written

by James Random Hacker.

signature of Ty Coon, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

GNU Lesser General Public License (LGPL), Version 3.0

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/ ↗>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the “GNU GPL” refers to version 3 of the GNU General Public License.

“The Library” refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An “Application” is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A “Combined Work” is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the “Linked Version”.

The “Minimal Corresponding Source” for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The “Corresponding Application Code” for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following: ▪ 0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU Free Documentation License (FDL), Version 1.3

GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the

same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of

that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document

under the terms of the GNU Free Documentation License, Version 1.3

or any later version published by the Free Software Foundation;

with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU

Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with … Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the

Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

GCC Runtime Library Exception (RLE), Version 3.1

GCC RUNTIME LIBRARY EXCEPTION

Version 3.1, 31 March 2009

Copyright © 2009 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a notice placed by the copyright holder of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header files and runtime libraries with the compiled program. The purpose of this Exception is to allow compilation of non-GPL (including proprietary) programs to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for execution after a Compilation Process, or makes use of an interface provided by the Runtime Library, but is not otherwise based on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without modifications, governed by version 3 (or a specified later version) of the GNU General Public License (GPL) with the option of using any subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and use would permit combination with GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor architecture, in executable form or suitable for input to an assembler, loader, linker and/or execution phase. Notwithstanding that, Target Code does not include data in any format that is used as a compiler intermediate representation, or used for producing a compiler intermediate representation.

The "Compilation Process" transforms code entirely represented in non-intermediate languages designed for human-written code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use of source code generators and preprocessors need not be considered part of the Compilation Process, since the Compilation Process can be understood as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-compatible software, or if it is done without using any work based on GCC. For example, using non-GPL-compatible Software to optimize any GCC intermediate representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime Library with Independent Modules, even if such propagation would otherwise violate the terms of GPLv3, provided that all Target Code was generated by Eligible Compilation Processes. You may then convey such a combination under terms of your choice, consistent with the licensing of the Independent Modules.

2. No Weakening of GCC Copyleft.

The availability of this Exception does not imply any general presumption that third-party software is unaffected by the copyleft requirements of the license of GCC.

The BSD 2-Clause License (Template)

The following is a BSD 2-Clause license template. To generate your own license, change the values of OWNER and YEAR from their original values as given here, and substitute your own.

Note: see also the BSD-3-Clause *license.*

This prelude is not part of the license.

<OWNER> = Regents of the University of California
<YEAR> = 1998

In the original BSD license, both occurrences of the phrase "COPYRIGHT HOLDERS AND CONTRIBUTORS" in the disclaimer read "REGENTS AND CONTRIBUTORS".

Here is the license template:

Copyright (c) <YEAR>, <OWNER>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT) (Template)

Copyright (c) <year> <copyright holders>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Impressum

Theben AG
Hohenbergstraße 32
72401 Haigerloch
DEUTSCHLAND

Tel.: +49 7474 692-0
Fax: +49 7474 692-150
E-Mail: info@theben.de
Internet: www.theben.de

Hinweis:
Technische Änderungen der Produkte sowie Änderungen im Inhalt dieses Dokuments behalten wir uns jederzeit ohne Vorankündigung vor.
Bei Bestellungen sind die jeweils vereinbarten Beschaffenheiten maßgebend. Die Theben AG übernimmt keinerlei Verantwortung für eventuelle Fehler oder Unvollständigkeiten in diesem Dokument.

Wir behalten uns alle Rechte an diesem Dokument und den darin enthaltenen Gegenständen und Abbildungen vor. Vervielfältigung, Bekanntgabe an Dritte oder Verwertung seines Inhaltes – auch von Teilen – ist ohne vorherige schriftliche Zustimmung durch die Theben AG verboten.

© 2019 Theben AG. Alle Rechte vorbehalten.