

COMPROTware:Library

ISO C-Bibliothek/Win32-Dll/PC Linux-Shlib für Fernwirktechn. Protokolle

Offene und standardisierte Fernwirkprotokolle haben im 21. Jahrhundert ihre hersteller-spez. Vorgänger ersetzt. Heute wird die IEC 60870-5-Protokollfamilie, MODBUS oder DNP3 in der Ver- und Entsorgungsindustrie, insbesondere bei Stromversorgern, auf breiter Basis eingesetzt. Diese Protokolle haben unter harten Einsatzbedingungen gezeigt, dass sie zuverlässig und flexibel die an sie gestellten Forderungen meistern können.

Mit **COMPROTware:Library** bietet **Real Thoughts** eine herstellerunabhängige Bibliothek für Fernwirkprotokolle in diesem Umfeld an.

COMPROTware:Library (kurz **CPLB**) ist eine Bibliothek bestehend aus Komponenten mit Implementierungen verschiedener fernwirktechn. Übertragungsprotokolle sowie Modulen für sicheres Event Handling, Message Queues usw. Zielgeräte sind IEDs (Intelligent End Device), RTUs (Remote Terminal Unit), Datenkonzentratoren und Leitsystem-Vorrechner. Die Bibliothek ist multiprotokoll-fähig, so dass im Zielgerät mehrere Protokolle parallel über verschiedene Schnittstellen ablaufen können. **CPLB** baut auf unserer reichhaltigen Erfahrung rund um die Implementierung von fernwirktechn. Protokollen bei einer Vielzahl von Kundenprojekten auf. Zusammen mit unserem Testtool bieten wir damit eine integrierte, hocheffiziente Lösung.

COMPROTware:Library ist in drei Fassungen verfügbar:

- Als Win32-Dll für die Integration in bestehende Anwendungen unter MS Windows.
- Als PC Linux-Shared Library (Shlib) für verschiedene PC Linux-Distributionen.
- Oder in Form von ISO C11-Quellcode für die Integration in Geräten mit weiteren Betriebssystemen. Portierungen auf Microsoft Windows 7/8.1/10/11, PC Linux und Solaris sind verfügbar und werden mitgeliefert. Weitere Portierungen sind von Kunden bereits durchgeführt worden.

Vorteile von **CPLB**:

- Schnellere Verfügbarkeit des Endprodukts, da die Entwicklungszeit für die Protokollimplementierung wegfällt.
- Die Integrations- und Verifikationsphase wird wesentlich verkürzt, da zentrale Komponenten (Verbindungsschicht, Event Handling, ...) bereits zuverlässig funktionieren.
- Das Entwicklungsrisiko wird minimiert, da die Kommunikationsschnittstelle zum Gerät sehr schnell „steht“.
- Die Kosten werden wesentlich reduziert, da für eine korrekte und verifizierte Protokollimplementierung

normalerweise mehrere Monate Aufwand anzusetzen sind.

- Sehr hohes Niveau bzgl. Standard-Konformität und Qualität.
- Zuverlässige und vertrauenswürdige Implementierung, so dass der Fokus auf den besonderen Fähigkeiten des eigenen Produkts konzentriert werden kann. Denn nicht das herstellerunabhängige Protokoll ist ein Alleinstellungsmerkmal, sondern nur die Kombination der besonderen und einmaligen Eigenschaften des Gesamtgeräts.
- Die Implementierung wird stetig weitergepflegt und an aktuelle Änderungen angepasst.

Dadurch dass viele Anwender die Bibliothek einsetzen, sind eine Vielzahl von Einsatzszenarien abgedeckt: Das bringt für jeden einzelnen höhere Qualität.

Hauptziel von **COMPROTware:Library** ist, dass unser Kunde einfach und kosteneffizient weitere Kommunikationsprotokolle in sein Gerät integrieren kann.



Eigenschaften

- **CPLB** ist ein **Protokollstack** für fernwirktechn. Übertragungsprotokolle.
 - ▲ Die Bibliothek ist **modular** aufgebaut. Das Design folgt den Grundsätzen moderner, **objekt-orientierter** Programmierung übertragen auf die Programmiersprache ISO C11.
 - ▲ Grundsätzlich sind alle Protokollimplementierungen entsprechend dem **EPA-Schichtenmodell** mit eigener Anwendungs- und Verbindungsschicht aufgebaut.
 - ▲ Die Bibliothek ist **geräte- und betriebssystem-unabhängig** realisiert. Der resultierende Objektcode entspricht bzgl. der Größe den Erfordernissen für den Einsatz in Embedded Systems.
 - ▲ **CPLB** steht als **ISO C11-Bibliothek**, als **Win32-Dll** oder als **PC Linux-Shlib** für die Integration bereit.
 - ▲ Master und Slave sind getrennt implementiert.
 - ▲ Alle Protokolle der Bibliothek können **gleichzeitig und mehrfach** in einem Gerät eingesetzt werden und über verschiedene Schnittstellen die Kommunikation abwickeln.
 - ▲ Für einzelne Protokolle ist eine „**API for Java**“ und eine „**API for .NET**“ realisiert.
- Implementierungsschnittstelle:
 - ▲ Initialisierung mit Konfiguration.
 - ▲ Serielle/netzwerk-basierte Schnittstelle: Der empfangene Zeichenstrom wird verarbeitet, aufbereitete Telegramme werden für die Übertragung bereitgestellt. Das Erkennen von Telegrammen auf der einen und das Erzeugen von Antwortnachrichten auf der anderen Seite übernimmt die Bibliothek.
 - ▲ Zugriff auf aktuellen Objektzustand.
 - ▲ Benachrichtigung über Zustandsveränderungen.
 - ▲ Befehlsweitergabe, -verarbeitung und -rückmeldung.
 - ▲ Zählwertverarbeitung.
- Umgebung auf dem Zielgerät:
 - ▲ Thread-basierte Laufzeitumgebung erforderlich.
 - ▲ Einfache Datenstrukturen (wie Semaphoren, Counter und Speicherverwaltung) müssen vorhanden sein.
 - ▲ Protokoll- und Hardware-Integrationsschicht umrahmen die Bibliothek.
- Verfügbare Protokolle:
 - ▲ Implementierung Slave:
 - IEC 60870-5-101 Ed 2.1, -103 und -104 Ed 2.1.
 - DNP3/IEEE 1815 serial und DNP3/IEEE 1815 over LAN/WAN.
 - MODBUS serial und MODBUS TCP/IP.
 - ▲ Implementierung Master:
 - IEC 60870-5-101 Ed. 2.1, -103 und -104 Ed. 2.1.
 - DNP3/IEEE 1815 serial und DNP3/IEEE 1815 over LAN/WAN.
 - MODBUS serial und MODBUS TCP/IP.
 - ▲ IEC 62351-3 bei IEC 60870-5-104, DNP3 over LAN/WAN und MODBUS TCP/IP.
 - ▲ Weitere Implementierungen sind in der Entwicklung. Fragen Sie nach!
- Win32-Dll/PC Linux-Shlib:
 - ▲ Dll für MS Windows 7/8.1/10/11.
 - ▲ Shlib für PC Linux-Distributionen.
 - ▲ Beispielapplikation im Quellcode.
 - ▲ Die Techn. Dokumentation beschreibt die Bibliothek und die Integration.
- ISO C11-Bibliothek:
 - ▲ Verständlicher und kommentierter ISO C11-Quellcode. Es werden sprechende Variablen- und Funktionsnamen verwendet. Der Quellcode ist konform zu ISO C11.
 - ▲ Beispielapplikation, ebenfalls im Quellcode, wird mitgeliefert.
 - ▲ Die Techn. Dokumentation beschreibt die Bibliothek und die Integration.
 - ▲ Die Bibliothek basiert auf ISO C11 und kann damit auf allen Plattformen eingesetzt werden, für die ein ISO C-Compiler (mit Standardbibl.) existiert.
 - ▲ Die Bibliothek ist vorbereitet für die Integration in eine C++-Umgebung.
- Support und Integrationsunterstützung:
 - ▲ Kostenloser telefonischer Support zur Bibliothek für ein Jahr gegenüber einer Kontaktperson mit guten Kenntnissen der Aufgabenstellung und des Zielprodukts.
 - ▲ Konzeption, Durchführung und Verifikation (Zertifizierung) der Integration in das Zielgerät nach Aufwand.
 - ▲ Software Update für ein Jahr inklusive.

Kontakt zu uns ...

Real Thoughts GmbH

Welfenstraße 35, 76137 Karlsruhe, Germany

Telefon +49 721 627 6730

Website www.realthoughts.de,

E-Mail info@realthoughts.de

Irrtum und Änderungen vorbehalten. Stand: Mai 2025

Real Thoughts GmbH

„Echtzeitapplikationen und Graphikanwendungen“