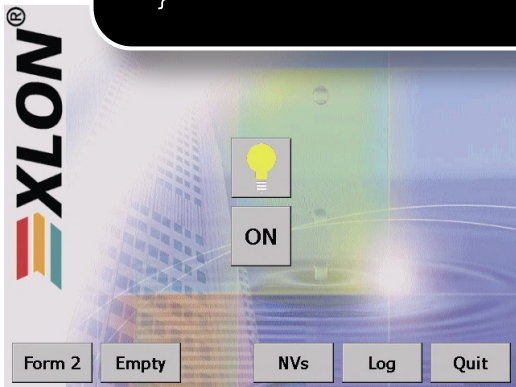


Eigenschaften

```
int main()
{
    ...
    CreateNv()
    ...
    OpenNetInterface()
    ...
    SetNv()
    ...
    WaitForNvUpdate()
    ...
    CloseNetInterface()
    ...
    return 0;
}
```



Beschreibung

Bei der **EXLON[®]HAPI** handelt es sich um eine Software-Bibliothek in Form einer DLL, welche die Erstellung LON basierter Visualisierungs-, Control-, und Web-Anwendungen stark vereinfacht und beschleunigt. Der Anwender benötigt keine tieferen Kenntnisse in der Erstellung PC basierter LON Applikationen, d.h. in der LON Host-Applikations Programmierung.

Werden für die LON-Kommunikation ausschließlich Netzwerkvariablen verwendet, so wird der Aufwand nochmals stark reduziert. Es müssen lediglich die Inhalte von Netzwerkvariablen verarbeitet werden. In der Praxis bedeutet dies, dass Ausgangs-Variablen gesetzt werden können und Änderungen von Eingangs-Variablen sig-

- Einfache Erstellung leistungsfähiger Visualisierungs-, Control- und Web-Anwendungen für LON
- Keine Kenntnisse in der Erstellung PC basierter LON Anwendungen nötig
- Kompatibel mit **EXLON[®]PC**, **EXLON[®]PC xsl**, **EXLON[®]** LonTalk Adaptern und LonTalk Adaptern von Drittanbietern
- Unterstützung aller Microsoft Desktop-, Embedded- und Server-Betriebssysteme
- Programmierung unter allen gängigen Programmiersprachen und -Umgebungen möglich, z.B. C, C++, C#, VisualBasic, Java, J#, Delphi, LabView, Python, ASP.NET usw.
- Dynamisch ladbare Bibliothek (DLL)
- Bis zu 4096 Netzwerkvariablen
- Volle Unterstützung der LonMark[®] SNVT Master List
- Self-Installation des Applikationsknotens möglich
- Empfangen und Senden beliebiger Explizit Nachrichten
- Verwendung aller LonTalk Servicetypen möglich
- Verwendung aller LonTalk Adressierungsarten möglich
- Unterstützung von Netzwerk-Management und Netzwerk-Diagnose
- Applikations-Download aus APB und NXE Dateien auf LON-Knoten im Netzwerk
- Dateitransfer via LON-FTP auf LON-Knoten im Netzwerk

nalisiert werden. Selbstverständlich werden alle Standard Network Variable Types (SNVTs) der aktuellen LonMark[®] SNVT Master List unterstützt bzw. ständig ergänzt. Es können bis zu 4096 NVs definiert werden, Self Documentation Strings von Knoten und NVs können angegeben werden.

Doch auch wenn die Anwendungen komplexere Techniken wie die Unterstützung von Explizit Nachrichten, Netzwerk-Management oder Netzwerk-Diagnose erfordern, ist die **EXLON[®]HAPI** das Mittel der Wahl. Selbst der Applikations- und FTP-Download auf andere Knoten im Netzwerk ist möglich. Fortgeschrittene Anwender können alle im LonTalk Protokoll bekannten Servicetypen wie z.B. Unacknowledged, Unacknowledged Repeated, Acknowledged, Request/Response usw. verwenden. Ebenfalls werden alle im LonTalk Protokoll bekannten Adressierungsarten wie z.B. Subnet/Node-, Neuron-, Group-, Broadcast- und Local-Adressierung unterstützt. extent of delivery. ➡

Der nebenstehende Ablaufplan verdeutlicht den schematischen Aufbau einer Visualisierungs-Applikation ähnlich nebenstehender Abbildung, mit der Netzwerkvariablen (NVs) visualisiert und gesetzt werden können. Diese bereits sehr mächtige Beispielanwendung, die nur einen kleinen Teil der Möglichkeiten der **XLON[®]HAPI** nutzt, kann grob in 3 Teile untergliedert werden:

Initialisieren und Öffnen der LON-Schnittstelle

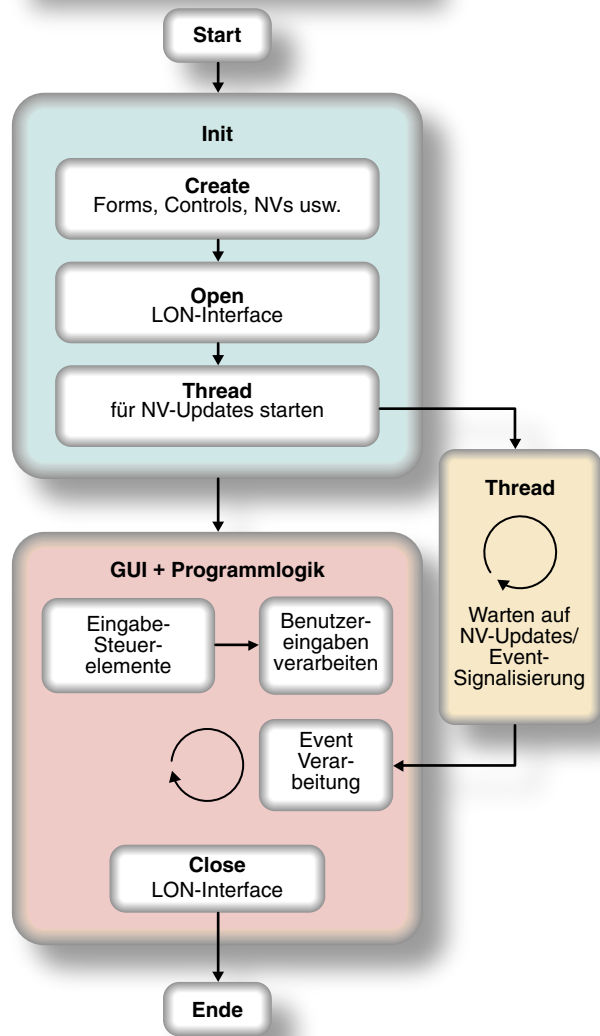
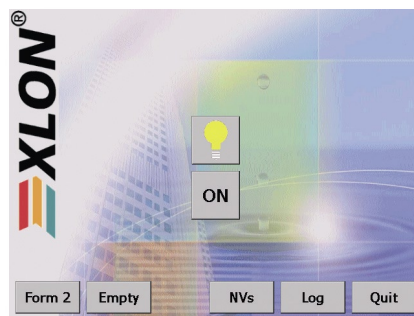
Beim Programmstart müssen alle NVs angelegt und initialisiert werden, dies geschieht mittels der Funktion „CreateNv()“. Erst dann darf das LON-Interface mittels der Funktion „OpenNetInterface()“ geöffnet werden. Vor dem Beenden des Programms sollte das LON-Interface mittels der Funktion „CloseNetInterface()“ wieder geschlossen werden.

Signalisierung von Netzwerkvariablen-Updates

Um Updates von NVs zu erhalten, ist es sinnvoll einen eigenen Thread zu starten. In der Thread-Schleife wird auf NV-Updates gewartet, was durch die Funktion „WaitForNvUpdate()“ realisiert ist. Wird ein NV-Update erkannt, wird dies mittels eines Events an andere Objekte signalisiert.

Grafische Benutzerschnittstelle (GUI) und Programmlogik

Mittels der GUI werden NVs als Reaktion auf Benutzer-Eingaben gesetzt, dazu existiert die Funktion „SetNv()“. Diese Funktion kann natürlich auch verwendet werden, um NVs auf Grund der Programmlogik oder als Reaktion auf NV-Updates zu setzen.



Spezifikation

LON Netzwerkschnittstelle

XLON[®]PC, **XLON[®]PC XSI**, **XLON[®]PCI**, **XLON[®]USB**, **XLON[®]PC/104**, **XLON[®]DONGLE**, LonTalk Adapter von Drittanbietern

Betriebssystemvoraussetzung

Windows CE 4.2, Windows CE 5.0, Windows ME, Windows 2000, Windows XP, Windows Server 2003, Windows Vista

Programmierungsumgebungen

Visual Studio 6.0, Visual Studio 2002/2003 .NET, Visual Studio 2005, Microsoft embedded Visual C++ 4.0, IDEs von Drittanbietern wie z.B. Borland Delphi oder Borland C++ Builder und Tools zur Erstellung von Webanwendungen

Dokumentation

Die Programmieranleitung ist zweisprachig in deutsch/englisch und ist im Lieferumfang enthalten. Demoapplikationen für Visual Studio 2003/2005 sind ebenfalls im Lieferumfang enthalten.