



**LONMARK®**  
**SCHWEIZ**

# **Technische Grundlagen zur LonWorks® Technologie**

LonMark Schweiz, Genossenschaft  
Schwendi 329  
Postfach 27  
3625 Heiligenschwendi

☎ +41 33 223 00 85 / 📠 +41 33 223 00 86  
[www.lonmark.ch](http://www.lonmark.ch) / [info@lonmark.ch](mailto:info@lonmark.ch)

Verfasser: Christoph Brönnimann, El. Ing. FH  
IBT Ingenieurbüro Brönnimann Thun

Copyright by LonMark Schweiz, Genossenschaft, Schwendi 329, Postfach 27, CH-3625 Heiligenschwendi, 033 223 00 85

Die Dokumentation ist Eigentum der Genossenschaft LonMark Schweiz. Der Inhalt dieses Dokumentes wurde sorgfältig geprüft, jede aus der Verwendung der Unterlagen abgeleiteten Haftungsansprüche werden jedoch abgelehnt. Der Inhalt der Dokumentation darf unter Berücksichtigung der Haftungsablehnung, unter Nennung der Autoren und der Genossenschaft LonMark Schweiz unverändert frei kopiert und weitergegeben werden; die kommerzielle Nutzung ist ausdrücklich untersagt. Autorenrechte vorbehalten.

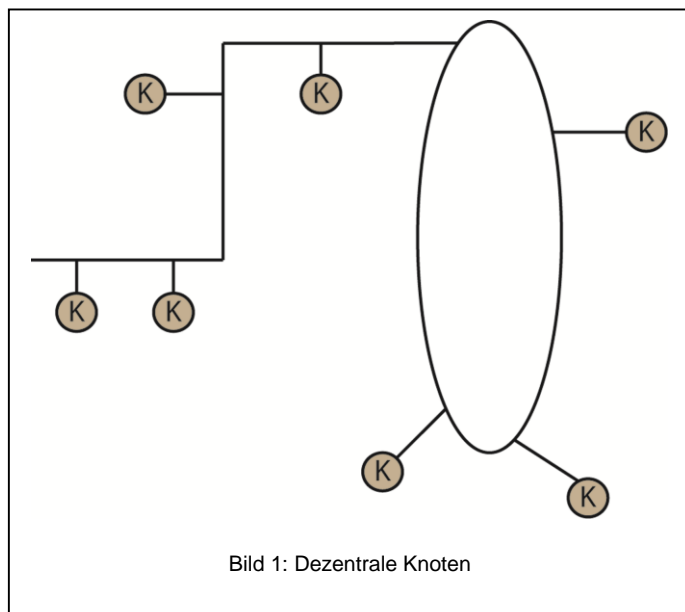
<b>1 GRUNDLAGEN .....</b>	<b>5</b>
<b>1.1 Die Idee hinter LON .....</b>	<b>5</b>
1.1.1 Philosophie .....	5
<b>2 DIE VIER ELEMENTE VON LON .....</b>	<b>7</b>
<b>2.1 Das LonTalk Protokoll .....</b>	<b>7</b>
2.1.1 Grundsätzlicher Aufbau.....	7
2.1.2 Was ist CSMA.....	9
2.1.3 Die OSI-Layer .....	10
2.1.4 Die Adresszuweisung .....	10
2.1.5 Adressierungsarten .....	11
2.1.6 Explizit Messages .....	12
2.1.7 Netzwerkvariablen.....	12
2.1.8 Die Konfiguration und das Netzwerkmanagement.....	13
<b>2.2 Die Knoten .....</b>	<b>14</b>
2.2.1 Die Neuron-Chip Familie von Echelon .....	14
2.2.2 MIP (Micro Processor Interface Program) .....	15
2.2.3 Shortstack .....	15
2.2.4 Offene Protokollimplementationen.....	15
2.2.5 Der LC3020 Chip von LOYTEC.....	15
2.2.6 Begriffe für alle LON Chips .....	16
<b>2.3 LonWorks Transceivers .....</b>	<b>18</b>
2.3.1 Twisted Pair TP 78 .....	18
2.3.2 Free Topology FTT-10 .....	18
2.3.3 RS-485.....	19
2.3.4 Link Power .....	19
2.3.5 Power Line.....	19
2.3.6 Weitere Transceiver.....	21
<b>2.4 LonWorks Tools.....</b>	<b>21</b>
<b>3 DER LONMARK-STANDARD .....</b>	<b>22</b>
<b>3.1 Der physikalische Layer (Layer 1) .....</b>	<b>22</b>
<b>3.2 Layer 2-6.....</b>	<b>22</b>
<b>3.3 Der Application Layer (Layer 7) .....</b>	<b>22</b>
<b>3.4 LonMark-Objekte.....</b>	<b>23</b>
3.4.1 Die Struktur eines LonMark Objektes .....	23
3.4.2 Das Node-Objekt .....	24

<b>4 DIE BAUSTEINE EINES NETZES.....</b>	<b>25</b>
<b>4.1 Knoten.....</b>	<b>25</b>
<b>4.2 Netzorganisationsbausteine.....</b>	<b>25</b>
4.2.1 Repeater.....	25
4.2.2 Bridges.....	25
4.2.3 Learning Router.....	25
4.2.4 Configured Router.....	25
4.2.5 Warum Router einsetzen?.....	25
<b>5 ANHANG .....</b>	<b>26</b>
<b>5.1 Begriffe.....</b>	<b>26</b>
<b>5.2 Abkürzungen .....</b>	<b>32</b>
<b>5.3 Quellenverzeichnis .....</b>	<b>33</b>

# 1 Grundlagen

Dieser Anhang zum Produkthandbuch enthält eine Einführung in die LonWorks-Technologie. Sie zeigt auf, was hinter dem LonTalk Standard steckt und macht Möglichkeiten und Grenzen transparent. Anhand eines Begriffsregisters ist ein kleines Nachschlagewerk beigelegt, welches beim Studium weiterer Literatur zum Thema LonWorks dienlich sein wird. Die Dokumentation verfolgt das Ziel, einen Überblick über die LonWorks-Technologie zu vermitteln.

## 1.1 Die Idee hinter LON



### 1.1.1 Philosophie

LON, das Lokal Operierende Netzwerk bringt das Computernetzwerk auf das Chip, das ist die Vision der Echelon Gründer. Die Technologie strebt an, dass Netzwerke aus einer grossen Anzahl kostengünstiger, so genannter Knoten aufgebaut werden können. Diese Knoten können durch verschiedene Hersteller hergestellt werden und mittels LonTalk-Protokoll untereinander kommunizieren.

Die Knoten besitzen alle ihre eigene Intelligenz und können Daten untereinander ereignisgesteuert austauschen. Sie messen, steuern, regeln und kommunizieren. Dies ergibt ein äusserst flexibles Netz von

Funktionen mit beinahe beliebigem Vernetzungs- und Komplexitätsgrad.

Das Herzstück in Form einer offenen Kommunikationssprache, des LonTalk Protokolls, ist unter IEC/ISO, CEN, GBZ und ANSI normiert und kann auf beliebigen Mikroprozessoren implementiert werden. In zahlreichen weiteren Normierungen hat LonWorks Eingang gefunden. So z.B. in BACnet (ASHRAE American Society of Heating and Air-Conditioning Engineers), ISFS (International Forecourt Standard Forum, d.h. alle grossen Ölfirmen), CEN TC-247, SEMI (Massen-Durchfluss-Meter), CELECT (UK für Heizungen) und weiteren.

Den wichtigsten Standard bildet LonMark, eine durch die Anwender von LON-Komponenten geführte Organisation.

LonTalk kann über Zweidrahtleitungen, 230 V Netze, Fiberoptik, Funk- und Ethernet-Netzwerke nahtlos übermittelt werden.

Die offene Technologie hat dazu geführt, dass bereits 1994 erste Gewerke übergreifende Systemintegrationen in Gebäuden mit über 3000 Geräten durchgeführt werden konnten. Damals wie heute steht Komfort und Energieeffizienz im Vordergrund.

LonMark ist nach wie vor der Standard, welche die grösste Modularität und Flexibilität für die aufwendigen Energieeffizienzsysteme erlaubt. LonWorks ist die einzige Technologie, welche über 30 Mio Powerline Geräte im Feld im Einsatz hat. Bezüglich neuer Chip-Generationen ist LonWorks meistens den entscheidenden kleinen Schritt voraus, so gibt es heute kostengünstigste Lösungen, welche keinen Vergleich scheuen müssen.

Was LonTalk aber ganz besonders auszeichnet ist die Nachhaltigkeit: auch heute werden die Installationen von 1994 mit den aktuellen Tools unterstützt. Die Hardware kann mit neuer Firmware geladen werden während das System mit den aktuellsten Komponenten mit den neuesten Chips ergänzt werden kann.

LonWorks ist überall die aktuell beste Wahl, wo sich Nachhaltigkeitsbedürfnisse mit aktueller Technik treffen müssen. Und wird auch in 20 Jahren topaktuell die gleichen Vorzüge anbieten können.

## 2 Die vier Elemente von LON

Die LonWorks-Technologie basiert grundsätzlich auf vier Elementen:

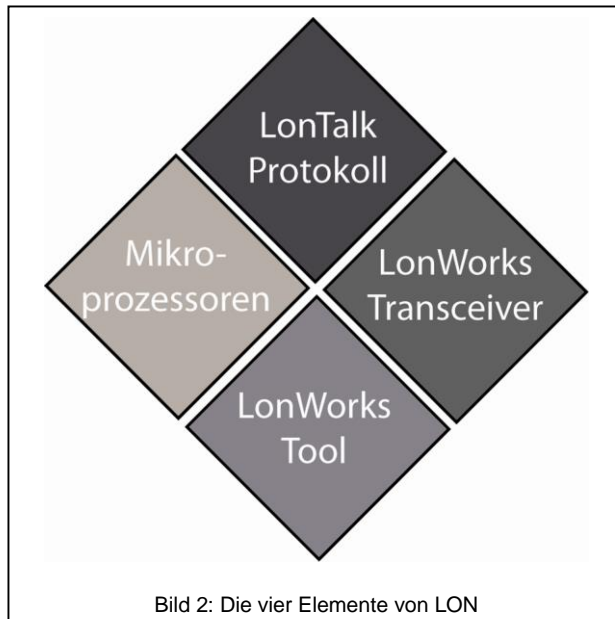


Bild 2: Die vier Elemente von LON

- Das LonTalk-Protokoll definiert die Sprache, welche auf dem Medium gesprochen wird.
- Mikroprozessoren auf Kontrollgeräten können diese Sprache interpretieren und bilden sogenannte Knoten, welche mittels der LonTalk-Sprache vernetzte Funktionen ausführen können.
- Die LonWorks Transceiver können LonTalk auf verschiedenen physikalischen Medien abbilden, so dass die Sprache über verschiedenste Kommunikationskanäle übertragen werden kann.
- Schlussendlich bilden die Tools das Rückgrat für die Entwicklung von Produkten und die Planung und Durchführung von Installationen. Dementsprechend wird zwischen Entwicklungstools (NodeBuilder, Mikroprozessor Workbench) und Installationstools (LonMaker, NL220, NL-Facilities) unterschieden.

### 2.1 Das LonTalk Protokoll

Ein LON-Chip „spricht“ LonTalk, d.h. er sendet und empfängt kurze Telegramme, in denen die eigentlichen Nutzdaten (variabel von 0 bis 228 Byte) eingebettet sind. Damit dies effizient und zuverlässig geschieht, auch wenn das Übertragungsmedium extremen Störeinflüssen unterliegt, wie zum Beispiel das 230 V Stromnetz – hat man sich an bewährte Verfahren aus der Computerwelt angelehnt und das LonTalk-Protokoll nach dem 7-schichtigen ISO/OSI Referenzmodell mit reichhaltigen Diensten ausgestattet.

#### 2.1.1 Grundsätzlicher Aufbau

##### 2.1.1.1 Übertragungsverfahren

Die Übertragung erfolgt Paketweise. Das Zusammenstellen und Versenden dieser Pakete wird von der Firmware übernommen; der Anwender muss sich also nicht mit „low-level“ Funktionen beschäftigen.

Im LON-Protokoll sind 4 verschiedene Übertragungsverfahren vorgesehen (sogenannte Dienste)

unacknowledged	Das Paket wird nur einmal versandt. Eine Bestätigung vom Empfänger wird nicht erwartet.
acknowledged	Nach dem Verschicken des Paketes wird eine Bestätigung des Empfängers erwartet. Erfolgt diese nicht oder fällt sie negativ aus, wird das Paket noch einmal gesendet. Die Maximale Anzahl solcher Wiederholungen ist frei bestimmbar.
unacknowledged / repeated	Das Paket wird mehrmals hintereinander versandt. Eine Bestätigung durch den Empfänger wird nicht erwartet. Die Anzahl Wiederholungen und die Wartezeiten dazwischen sind frei bestimmbar.
request / response	Ähnlich wie bei acknowledged. Statt einem einfachen acknowledge können in der Bestätigung jedoch noch zusätzliche Daten vorhanden sein.

Der Anwender kann frei bestimmen, welches Verfahren angewandt werden soll.

Die Datenpakete werden mittels einem differentiellen Manchester-Code übermittelt, d.h. die Dateninformation entspricht einer Frequenz. Eine Periode mit hoher Frequenz entspricht einer 0, eine langsame Periode stellt eine 1 dar. Dabei wird mindestens pro Dateninhalt ein Zustandswechsel des Signals durchgeführt. Die Manchester-Decodierung ermöglicht es, Leitungen zu führen, ohne dass auf die Polung geachtet werden muss.

- Differentieller Manchester-Code ohne DC-Anteil für beliebige Medien
- Bitsynchronisation anpassbar an Übertragungsmedium
- Variable Nutzdaten von 1-228 Bytes

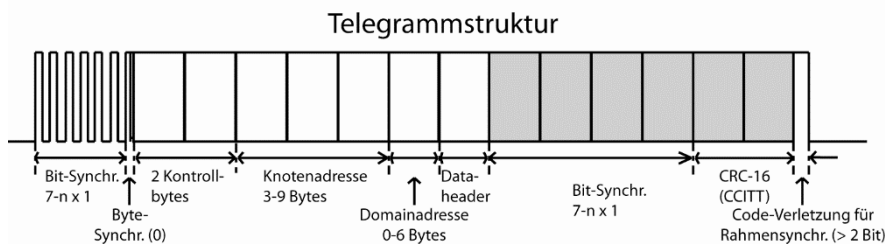
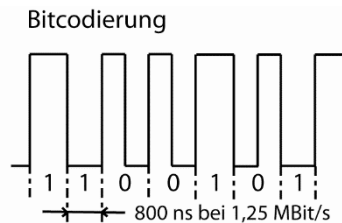


Bild 3: Datenformat

Ein Telegramm besteht immer aus den jeweiligen Transceiver anpassbaren Synchronisationsbits (Folge von „1“). Sie dienen der Transceiverschaltung, damit sich diese auf die Empfangsfrequenz einschwingen kann. Die erste 0 zeigt den Beginn der Adressdaten an, welche dem empfangenden Knoten anzeigen, ob er das ankommende Telegramm überhaupt beachten soll. Der Adresse folgen die Nutzdaten oder ACK / NACK Bytes um anzuzeigen, ob eine Meldung erfolgreich empfangen wurde.

### 2.1.1.2 Datensicherheit

Bei offenen Bussystemen kann optional zusätzliche Datensicherheit gewährt werden. In einem speziellen Übertragungsverfahren kann der Empfänger die Authentizität des Senders überprüfen. Dazu wird bei der Installation des Netzwerkes zwischen Sender und Empfänger eine 48 Bit-Codenummer vereinbart. Dieser Code ist unabhängig von der Chip-spezifischen Kennnummer. Die Codenummer wird mit einem bei jeder Übertragung wechselnden Chiffrierverfahren gesendet, was eine hohe Sicherheit gewährleistet.

Erhält ein Knoten eine authentifizierte Nachricht, dann fordert er den Absender auf, seine Autorisierung nachzuweisen. Dazu schickt er ihm eine zu verschlüsselnde Zufallszahl (64 Bit). Der Absender kodiert diese Zahl unter Verwendung seines Schlüsselworts und leitet das Ergebnis zurück. Der Empfänger vergleicht die Antwort mit seinem eigenen Verschlüsselungsergebnis. Bei Übereinstimmung akzeptiert die Netzwerk-CPU des Empfängers die ursprüngliche Nachricht und reicht sie zum Applikationsprogramm weiter. Im anderen Fall ignoriert der empfangende Knoten das Originaltelegramm und inkrementiert einen Fehlerzähler. Die Authentisierung kann für jede einzelne Netzwerkvariable und für Netzwerkmanagement-Kommandos definiert werden.

### 2.1.1.3 Prioritäten

Die verschiedenen Knoten können mit unterschiedlichen Prioritäten versehen werden. Für Mitteilungen mit hoher Priorität werden am Ende jedes Paketes spezielle Zeitabschnitte (time bins) reserviert, während denen die Übertragung eben eines dieser Pakete beginnen kann. Knoten mit tieferer Priorität können erst zu einem späteren Zeitpunkt mit der Übertragung beginnen, sofern der Übertragungskanal dann nicht schon von einem Knoten mit höherer Priorität besetzt ist. So kann bei zeitkritischen Applikationen für bestimmte Knoten eine kürzere Zugriffszeit gewährt werden.



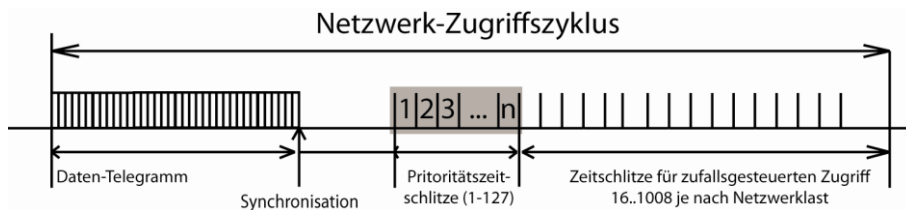


Bild 4: Prioritätszeitschlitz

Figur 4 zeigt die Telegrammfolge mit den für priorisierte Meldungen reservierten Zeitschlitz. Dadurch ermöglicht das Protokoll, einer limitierten Anzahl sehr schnell zu sendender Meldungen den Vortritt zu geben. Die Verzögerungszeit innerhalb der „priority slots“ und der normalen Zeitschlitzte wird mittels dem CSMA-Verfahren zufällig vergeben.

## 2.1.2 Was ist CSMA

CSMA heisst Carrier Sense Multiple Access. Verschiedene Teilnehmer eines Systems dürfen auf das Kommunikationsmedium zugreifen, wobei möglichst intelligenten Algorithmen zur Kollisionserkennung und –vermeidung zur Anwendung kommen.

Um die Wahrscheinlichkeit von Kollisionen möglichst gering zu halten, wurde ein raffinierter Mechanismus entwickelt. Ein Knoten, der ein Paket absetzen will, „hört“ zuerst auf den Bus um festzustellen, ob dieser nicht schon besetzt ist. Stellt er schliesslich das Ende eines fremden Pakets fest, beginnt er nicht gleich zu senden, sondern wartet eine bestimmte Anzahl Zeiteinheiten, sogenannte „time-bins“ (Zeittöpfe, die nur wenige Bits lang sind) ab. Er wird schliesslich während einem dieser „time-bins“ mit der Übertragung seines Paketes beginnen. Die ersten paar „time-bins“ sind für Knoten mit erhöhter Priorität bestimmt (siehe oben). Hat der Knoten tiefere Priorität, wartet er noch eine bestimmte Anzahl „time-bins“ weiter, bis er schliesslich zu senden beginnt. Diese Anzahl wird durch einen Zufallsgenerator bestimmt. Während diesem Warten verfolgt der Knoten weiterhin das Geschehen auf dem Bus. Kommt ihm ein anderer Knoten mit Senden zuvor, beginnt die Prozedur von neuem. Die Wahrscheinlichkeit, dass zwei Knoten während exakt dem gleichen „time-bin“ zu senden beginnen, ist dank der Steuerung mit Zufallsgenerator relativ klein. Somit kann auch bei hoher Busauslastung die Anzahl Kollisionen auf einem relativ kleinen Wert gehalten werden.

Das LonTalk-Protokoll zeichnet sich durch den an der Universität Stanford entwickelten „predictive p-persistent CSMA“ Algorithmus aus. Dieser Algorithmus ermöglicht es, bei Netzüberlast eine garantierte Datenrate zu übertragen. LonTalk ist dadurch den übrigen Feldbussystemen bezüglich Überlastverhalten überlegen.

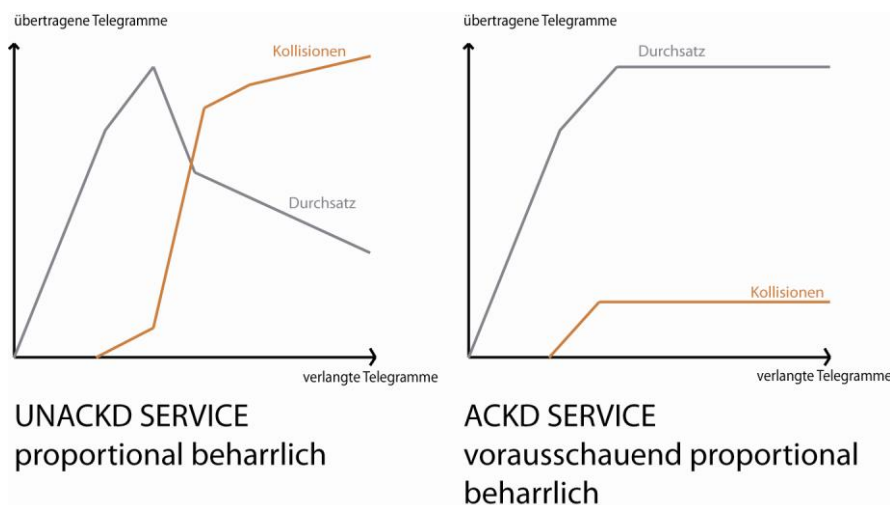


Bild 5: „predictive p-persistent CSMA“

### 2.1.3 Die OSI-Layer

Die OSI (Open System Interconnection) Definition ist die Grundlage, auf welcher die Internet / Intranet Technologie aufgebaut worden ist. LonWorks hat bezüglich der Gliederung das Rad nicht neu erfunden und das OSI-Modell angewendet.

Der damit verbundene, grössere „overhead“ führt in der Praxis zu kaum merklicher Verminderung der Transaktions- bzw. Responsezeitverhaltens, erleichtert aber die Realisierung, Inbetriebnahme und Wartung von Netzwerken enorm. Unter den genannten Diensten sind folgende hervorzuheben:

- - effizienter Zugang zum Übertragungsmedium mit Prioritätensteuerung (quasideterministische Verhalten)
- transparentes, bidirektionales Durchreichen bzw. Filtern von Telegrammen über eingebaute physikalisch-logische Trennglieder (Router)
- mehrere Adressierungsarten: Einzelknoten, Gruppe, an alle (broadcast)
- Senden und Empfangen von Telegrammen ohne / mit Quittung, Wiederholung und Berechtigungsprüfung
- gezielte Anforderung von Daten von einem oder mehreren Knoten (request-response, polling)
- Ereignisgesteuertes, priorisiertes und automatisches Senden und Empfangen von Daten über sogenannte Netzwerkvariablen
- Verwendung international standardisierter Grössen

OSI-Layer	Bedeutung	LonTalk Service
<b>7 Application</b>	Kompatibilität auf Applikationsebene	Objekt-Definition: Aktor, Sensor, Controller; Standard-Netzwerkvariablen, Netzwerk-Management, Installation, Real Time Kernel
<b>6 Presentation</b>	Interpretation	Transport von beliebigen Telegramm-Rahmen
<b>5 Session</b>	Aktion	Request-Response-Mechanismus (Polling)
<b>4 Transport</b>	Zuverlässigkeit	Übertragung mit / ohne Quittung Einzel- und Gruppen-Adressierung, Authentifizierte Meldungen (Schlüssel, PIN-Code), Duplikat-Erkennung, Überw. der Reihenfolge
<b>3 Network</b>	Ziel-Adressierung	Broadcast-Meldungen, transparente, konfigurierte und selbstlernende Router, 32385 Knoten proDomain, 2 48 Domains, 48-Bit-Code in jedem Chip
<b>2 Link</b>	Media-Zugriff und Rahmenprüfung	Rahmenprüfung, Datenendecodierung, CRC-16-Datensicherung. Predictive CSMA, Kollisionsvermeidung mit adaptiver Zuteilung von Zugriffs-Zeitschlitten, optional mit Prioritätszeitschlitten und Hardware-Kollisionsdetektion
<b>1 Physical</b>	Elektrische Verbindung	Unterstützung von div. Medien: RS485, traufgekoppelte 2-Drahtleitung, 230V-Netz, Funk, IR, LWL, Koax, Tf-Leitung etc., 610Bit/s -1,25MBit/s

### 2.1.4 Die Adresszuweisung

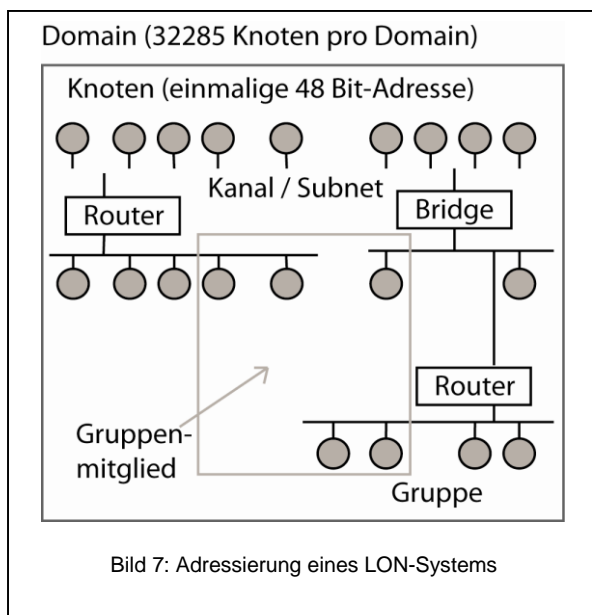


Bild 7: Adressierung eines LON-Systems

Das LonTalk-Protokoll unterstützt das Segmentieren eines LON-Systems und die Benutzung unterschiedlicher Übertragungsmedien. Die Netzwerktopologie bedient sich folgender Begriffe:

#### 2.1.4.1 Domain

Die Domain stellt eine logische Menge von Knoten auf einen oder mehreren Channels dar. Dabei kann der Datenaustausch nur zwischen Knoten innerhalb einer Domain stattfinden. Eine Domain bildet also eine virtuelle Begrenzung eines LON-Systems. Auf einem Channel können nebeneinander verschiedene Domains existieren. Dabei kann man diese nutzen, um eine gegenseitige Beeinflussung von Knoten in verschiedenen LON-Systemen auf demselben Channel zu verhindern. Kommunizieren beispielsweise die Knoten in einem Mehrfamilienhaus auf der Netzleitung, dann

sollten die LON-Systeme zweier Wohnungen unterschiedliche Domain-Adressen benutzen, damit nicht der Radiowecker morgens zusätzlich zur eigenen Kaffeemaschine auch die des Nachbarn einschaltet. Weiterhin kann die Domain-Adresse für das Servicepersonal auch als Systemseriennummer dienen. Eine Domain kann 32512 Knoten enthalten. Ein Knoten kann maximal Teilnehmer in zwei Domains sein. Eine Domain kann mittels 0, 1, 3 oder 6 Bytes definiert werden. Die Domain mit der Länge 0 dient zur Übermittlung der Servicemeldung, die Domain mit der Länge 1 und der ID 0 wird für Entwicklungswerkzeuge und LNS-Meldungen verwendet. Die Domain ist Teil der Adresse im Telegramm, d.h. eine lange Domainidentifikation generiert mehr Netzwerküberlastung.

#### 2.1.4.2 **Channel**

Ein Channel ist das physikalische Übertragungsmedium auf dem serielle Daten übertragen werden. Er kann beispielsweise ein Kabel, eine Funkfrequenz oder für die Powerline-Kommunikation ein Teil des 230 V Wechselspannungsnetz sein. Ein Channel wird von einem zweiten Channel immer durch einen Router oder einen Gateway separiert.

Channels sind frei definierbar, so können auch firmenspezifische Channels aufgebaut werden.

#### 2.1.4.3 **Subnet**

Ein Subnet ist ein logischer Zusammenschluss von maximal 127 Knoten innerhalb von einer Domain. Innerhalb einer Domain können wiederum 255 Subnets existieren. Alle Knoten eines Subnets müssen in der gleichen Domain liegen. Ein Channel kann wiederum mehrere Subnets führen, d.h. Subnets sind logische Adressierungsgruppen, welche über verschiedene physikalische Medien verwendet werden können. Ein Subnet kann aber keinen intelligenten Router überqueren, d.h. Channel überschreitende Subnets müssen mittels Bridge oder Repeater verbunden werden.

#### 2.1.4.4 **Node**

Jeder der 127 LON-Knoten innerhalb eines Subnets ist über eine sieben Bit lange Node-Nummer adressierbar. Damit errechnet sich die maximal pro Domain adressierbare Anzahl von LON-Knoten zu 32 385 (127 Knoten x 255 Subnets).

#### 2.1.4.5 **Gruppe**

Verschiedene LON-Knoten innerhalb einer Domain lassen sich zu einer Gruppe zusammenfassen, wobei die einzelnen Nodes auch in unterschiedlichen Subnets liegen dürfen. Mittels des 1 Bytes langen Gruppenadressen kann man bis zu 256 Gruppen innerhalb einer Domain definieren. Ein Neuron-Chip kann bis zu 15 Gruppen angehören. Bei einer Datenübertragung mit Bestätigung (acknowledged) darf eine Gruppe bis zu 64 Knoten umfassen. Mit einem Telegramm ohne Bestätigung (unacknowledged) können alle Knoten innerhalb einer Domain gleichzeitig angesprochen werden. Die Gruppenadressierung stellt ein probates Mittel dar, um die für eine Broadcast-Kommunikation (one-to-many – einer an viele) notwendige Telegrammanzahl zu reduzieren. In einer Kongresshalle können damit zum Beispiel mehrere Lampen eines Lampenfeldes gleichzeitig mit einem Telegramm angesteuert werden. Dadurch gibt es keinen Lauflichteffekt und der Bus wird nicht mit unnötigem Datenverkehr belastet. So kann eine Gruppe z.B. alle Lichtknoten in einer Fabrik enthalten, obwohl diese über das 220 V Netz oder mittels Zweidraht-Bus gesteuert werden.

Mit geeigneten Installationswerkzeugen (LNS basierend) kann mittels dem sogenannten „group overloading“ eine Gruppe in mehrere Untergruppen unterteilt werden. Siehe dazu Kapitel 5.4.5.

### 2.1.5 **Adressierungsarten**

Entsprechend der möglichen Adresszuweisungen können verschiedene Adressierungsarten verwendet werden. Das LonTalk-Adressfeld bezeichnet jeweils die Absender- und die Zieladresse eines LonTalk-Telegramms. Im LonTalk-Protokoll ist eine hierarchische Adressierung mit Domain-, Subnet- und Knotenadressen definiert. Für das gleichzeitige Ansprechen mehrerer LON-Knoten gibt es ausserdem die Domain- und die Gruppenadressierung. Ein LON-Knoten lässt sich also unter verschiedenen Adressen ansprechen.

Insgesamt gibt es fünf Adressierungsarten: Das vollständige Adressfeld besteht aus der Domain-Adresse (0, 1, 3 oder 6 Byte), der Zieladresse und der Absenderadresse. Die Zieladresse enthält

dabei je nach Adressierungsart die Neuron-ID (6 Byte), die Gruppenadresse (1 Byte) oder die Subnet- und Knotenadresse (zusammen 2 Byte). Die Absenderadresse besteht immer aus der Subnet- und Knotenadresse des sendenden Node.

Über seine Neuron-ID lässt sich ein LON-Knoten jederzeit gezielt ansprechen. Im Gegensatz dazu kann sich die während der Installationsphase vergebene Adresse im Laufe der Existenz eines Knotens ändern. Wegen der Länge der Neuron-ID (6 Byte) sollte sie nur während der Installation und Konfigurierung eines LON-Netzwerkes benutzt werden. Muss ein Knoten ausgetauscht werden, dann erhält der neu eingesetzte Knoten einfach dieselben Adressinformationen wie der alte. Seine Kommunikationspartner im Netzwerk bleiben hingegen unverändert.

Eine Domain wird durch die Domain-ID (0, 1, 3 oder 6 Byte) gekennzeichnet. Verwendet man für eine 6 Byte lange Domain-ID die Neuron-ID eines der zur Domain gehörigen LON-Knotens, so ist die Einzigartigkeit der Domain-ID gewährleistet. In einem LON-System in dem es keine Überschneidungsmöglichkeiten zwischen verschiedenen Bereichen geben kann, sollte man zugunsten einer kurzen Telegrammlänge auf die Domain-ID verzichten.

Je nach Adressierungsart bewegt sich die Länge einer LonTalk-Adresse zwischen 3 Byte und 9 Byte. Hinzu kommt die Länge der Domain-ID (0...6 Byte). Die in einem LonTalk-Telegramm enthaltenen Adressinformationen bewegen sich daher zwischen drei Byte für Gruppenadressierung und fünfzehn Byte für Adressierung über die Neuron-ID mit 6-Byte-Domain-Adresse.

### 2.1.6 Explizit Messages

Alle LON-Telegramme sind „explizit messages“. Sie können verglichen werden mit einem Datenzug, der sich seinen Weg durch das Netz zum richtigen Zielknoten sucht. Die Lokomotive enthält als Führer die Adresse, welche automatisch die Weichenstellungen im Netz verursacht. Ähnlich wie im Internet können somit Daten beliebiger Form übermittelt werden (Layer 6). Explizit messages werden von vielen Herstellern verwendet, um ihre proprietären Systeme zu steuern. Die Adresse des Empfängers kann entweder vom Programmierer vorgegeben oder im EPROM konfiguriert werden.

Vorteil:	Effizienter als Netzwerkvariablen
Nachteile:	Ohne genaue Kenntnisse des Meldungsaufbaus ist keine Verbindung möglich (d.h. Anschluss an Knoten von Fremdherstellern ist nur schwer möglich); benötigt grösseren Programmaufwand -> mehr Code.

LON bietet aber auf Layer 7 eine spezielle „explizit message“ an, welche die direkte Verknüpfung von Programmvariablen mit dem Netzwerk ermöglicht. Das nachfolgende Kapitel behandelt diese Meldungsform.

### 2.1.7 Netzwerkvariablen

Die Netzwerkvariablen bilden die Grundlage für eine wichtige und in dieser Form einzigartige Eigenschaft von LonWorks: die sogenannte Interoperabilität. Darunter versteht man das problemlose und nach einfachen Spielregeln funktionierende Zusammenwirken von auf LonWorks basierenden Produkten unterschiedlicher Hersteller. Interoperabilität ist wegen der vielfältigen, produktions- und installationstechnischen Verflechtungen von Herstellern, Systemplanern und Installationsfirmen eine wichtige Voraussetzung für die Verbreitung von LonWorks in der Industrie und in der Gebäudeautomation. Man könnte es auch anders ausdrücken: mit LonWorks lassen sich komplexe Systeme so aufbauen, als ob sie aus einer Hand wären.

#### 2.1.7.1 Kommunikationsprinzip

*Netzwerkvariablen (NV):*

Variablen, die zwischen zwei oder mehr Knoten Verbindungen schaffen. Die Verknüpfung der Variablen erfolgt wahlweise im Programmieren der Applikation, beim Endtest des Gerätes, vor Ort bei der Installation oder während dem Betrieb des Netzes.

*SNVT / SCPTS*

Um zwischen Knoten von verschiedenen Herstellern Verbindungen zu erstellen, werden sogenannte Standard Netzwerkvariablen (SNVT) und Standard Konfigurationsdaten verwendet (SCPTS).

SNVT's lassen sich „binden“, d.h. durch einen Eintrag im lokalen Speicher weiss dadurch ein SNVT, welche Knoten von ihr Daten erwarten. Diese Daten werden in der Folge immer übermittelt, wenn sich deren Wert ändert.

In Ergänzung dazu gibt es sogenannte UNVTS / UCPTS (User definierte Datentypen), welche die zusätzlich vom Hersteller beigestellten Format-Dateien definiert.

### **2.1.8 Die Konfiguration und das Netzwerkmanagement**

Auf logischer Ebene können mit Hilfe der Netzwerkvariablen zwischen den einzelnen Neuron-Knoten eine Vielzahl von Kommunikationsverbindungen aufgebaut werden (sogenannte Binding). Das wird in der Regel mit Hilfe eines Installationswerkzeuges (Handheld-Gerät, PC unter DOS und Windows) im Feld durchgeführt, wobei entsprechende Einträge im EEPROM der einzelnen Knoten vorgenommen werden. Es gibt aber auch Fälle, wie z.B. in einer Maschinensteuerung, wo alle Knoten bereits mit sämtlichen Kommunikationsbeziehungen vordefiniert werden.

Für die Inbetriebnahme eines LON-Systems bieten sich mehrere Szenarien. Je nach Zustand der zu installierenden LON-Knoten müssen die Kommunikationsbeziehungen und das Applikationsprogramm in den Knoten übertragen werden.

#### **Einfachste Variante**

Die einfachste Variante bei kleinen Systemen stellt die Plug-and-Play Installation von vorkonfigurierten Knoten durch den Benutzer dar.

#### **Hilfsgeräte**

Grössere Systeme nimmt man mit Hilfe eines Netzwerkmanagementknotens (NMK, Handgerät oder PC) in Betrieb. Ein NMK kann ein LON-System nach neu hinzugekommenen Knoten absuchen und konfigurieren. Ein Applikationsprogramm auf den Knoten laden, starten stoppen und zurücksetzen (Reset). Ausserdem kann er die von den Knoten geführte Kommunikationsstatistik auslesen. Router konfigurieren und die Struktur eines laufenden LON-Systems feststellen. Während der Installation muss eine Zuordnung zwischen der physikalischen Position jedes LON-Knotes hergestellt werden. Der Installateur kann dazu mit dem WINK-Kommando einen Knoten auffordern, eine spezielle Funktion auszuführen (z.B. Lampe 1 blinkt einmal), um ihn zu identifizieren oder zu finden. Daraufhin stellt er mit dem NMK die logischen Verbindungen zu anderen Nodes her. Die populärste Methode mit der unbestritten besten Interoperabilitätsunterstützung bietet LNS, der LonWorks Network Server.

#### **Erstellen einer Liste**

Ein anderes Szenario sieht das Erstellen einer Liste der Neuron-IDs und der physikalischen Positionen (und damit Funktionen) der LON-Knoten vor. Der NMK ordnet daraufhin den Nodes die gewünschten Kommunikationsbeziehungen zu und versieht sie eventuell mit dem noch fehlenden Applikationsprogramm. Zur Vereinfachung der Installation bietet LonTalk einen Knoten-Identifikations-String von acht Byte Länge.

## 2.2 Die Knoten

### 2.2.1 Die Neuron-Chip Familie von Echelon

#### 2.2.1.1 Single Chip Prozessor 3120

Der Single Chip 3120 wird für LowCost Module mit limitierten Funktionen verwendet, da seine Datenspeicher sehr limitiert sind. Programme können über den Bus in das EEPROM geladen werden.

CPUs:	3
EEPROM bytes:	512
RAM bytes:	1024
ROM bytes (Firmware):	10240
External Memory Interface:	nein
16-bit Timer / Counter:	2
Watchdog-Timer:	ja
Package:	SOIC
Pins:	32
Netzwerkvariablen:	62
Adresstabellen:	15

#### 2.2.1.2 Multiple Chip Prozessor 3150

Der 3150 ermöglicht die Steuerung eines externen Datenbusses und ist somit für komplizierte Aufgaben geeignet. Der 3150 ist bezüglich seiner für die Applikation verfügbaren Prozessorleistung mit einem 68HC11 oder 80C535 vergleichbar.

CPUs:	3
EEPROM bytes	512
RAM bytes	2048
ROM bytes (firmware):	0
External Memory Interface:	ja
16-bit Timer / Counter:	2
Watchdog-Timer	ja
Package:	PQFP
Pins:	64
Netzwerkvariablen:	62
Adresstabellen:	15

#### 2.2.1.3 Smart Transceiver Chip

Die Smart Transceiver Chips sind Neuron Chips mit einem eingebetteten Signalprozessor für den FTT oder Powerline Transceiver. Es gibt Smart Transceiver für alle Neuron-Derivate.

#### 2.2.1.4 Neuron 500 er Chip

CPUs:	4 (Internet)
EEPROM/Flash:	extern
RAM:	64 kB
ROM:	16 kB
Watchdog:	ja
Package:	7 x 7 mm QFN
Pins:	48
Netzwerkvariablen:	254
Versorgungsspannung:	3,3 V

### **2.2.2 MIP (Micro Processor Interface Program)**

Damit LonTalk auf leistungsfähigeren Prozessoren abgebildet werden kann, wurde eine parallele Schnittstelle auf andere Prozessorsysteme auf den NEURON-Chip implementiert. Die Schnittstelle wird mittels einem Link-Layer und einem Application-Message Layer Protokoll gesteuert und ermöglicht den vollen Zugriff auf das LonTalk Protokoll durch den angekoppelten Mikroprozessor. MIP-Knoten sind bezüglich Prozessorleistung nicht mehr begrenzt. Ein MIP kann 4096 Selektoreinträge bearbeiten, doch bleibt die Limitierung bezüglich der 15 Adress- und 2 Domaintabellen aufrechterhalten (legacy mode).

Für den Systemintegrator verhält sich ein MIP-basierter Knoten nicht wesentlich anders. Er bietet lediglich mehr Variablen und eine höhere Leistung an.

### **2.2.3 Shortstack**

Ähnlich wie das MIP ist der Shortstack eine spezielle Firmware für die Neuron-Chips, welche aber über die SCI oder SPI Schnittstelle an einen Mikroprozessor angebunden ist. Ein Shortstack-Knoten kann bis zu 255 Variablen / Selektoren bearbeiten. Die Begrenzung auf den „legacy mode“ bleibt.

### **2.2.4 Offene Protokollimplementationen**

Für die LonTalk-Implementationen auf leistungsfähige Prozessoren gibt es Stacks von mehreren Herstellern. In der Regel müssen diese Stacks auf Echtzeitbetriebssystemen aufgebaut werden, welche Timer in Millisekundaauflösungen unterstützen. Die bekanntesten Anbieter sind Loytec, Adept und Echelon.

Solche Stacks ermöglichen auch die Anbindung von Feldgeräten, direkt über Ethernet (z.B. L-Vis von Loytec, PCD von SaiaBurgess, InfraDALI von Infranet Partners, i.LON von Echelon).

### **2.2.5 Der LC3020 Chip von LOYTEC**

Der LC3020 Chip ist ein ARM7 Mikroprozessor von Loytec / NEC. Er unterstützt Ethernet und FTT Kanäle und arbeitet mit dem Open Source Betriebssystem RTEMS. Dadurch lassen sich beliebig komplexe Embedded Lösungen realisieren.

### **2.2.6 Layer 2-MIP von ECHELON**

Das Layer 2 MIP ist eine spezielle Firmware, welche alle ECHELON Chips befähigt, mit dem ECHELON Stack (vgl. Kapitel 2.4.5) LonTalk auf beliebigen Mikroprozessoren zu implementieren. Die Verbindung mit dem jeweiligen physikalischen Layer erfolgt über SPI. Der Echelon Stack bietet auch IP852 Unterstützung.

## 2.2.7 Begriffe für alle LON Chips

### 2.2.7.1 Firmware, EEPROM, PROM, Flash-PROM, RAM

#### Firmware

Unter Firmware versteht man das im Chip ablaufende Programm.

#### EEPROM

Elektronisch löschbare Speicherplätze, welche begrenzt auch Firmware halten können. In der Regel wird EEPROM zur Speicherung der Konfigurationsdaten verwendet. Ein EEPROM kann über das Netz geladen werden.

#### FLASH-EPROM

Ein FLASH-EPROM kann mittels eines im Chip eingebauten UV-Blitzlichtes gelöscht werden und kann einige tausendmal neu programmiert werden. Ein Flash kann über das Netz geladen werden und ermöglicht Funktionsanpassungen in bereits installierten Geräten.

#### RAM

RAM ist flüchtiger Speicher, welcher entweder mittels Batterie zwischengespeichert werden kann oder den Inhalt nach dem Ausschalten verliert.

### 2.2.7.2 Service

Der sogenannte Service-Pin ist ein spezieller Anschluss des Neuron-Chips. Er stellt ein natürliches Hilfsmittel bei der Konfiguration, Inbetriebnahme und Wartung des Netzwerkknotens zu dem der Neuron-Chip gehört, dar. Schliesst man einen Taster an und legt damit den Service-Pin auf Masse, sendet er (besser die Neuron-Firmware) ein spezielles Netzwerk-Management-Telegramm aus, in dem er u.a. seine einmalige 48-Bit-Seriennummer (Neuron-Chip-ID) allen Knoten im Netz mitteilt. Diese Information kann von einem Netzwerkmanager für das Vergeben der logischen Netzwerkadresse des Knotens bei der Installation und für die folgende Konfiguration genutzt werden. Verbindet man den Service-Pin mit einer Lumineszenz-Diode (LED), kann diese über verschiedene Blickmuster den aktuellen Betriebszustand des Netzwerkknotens signalisieren.

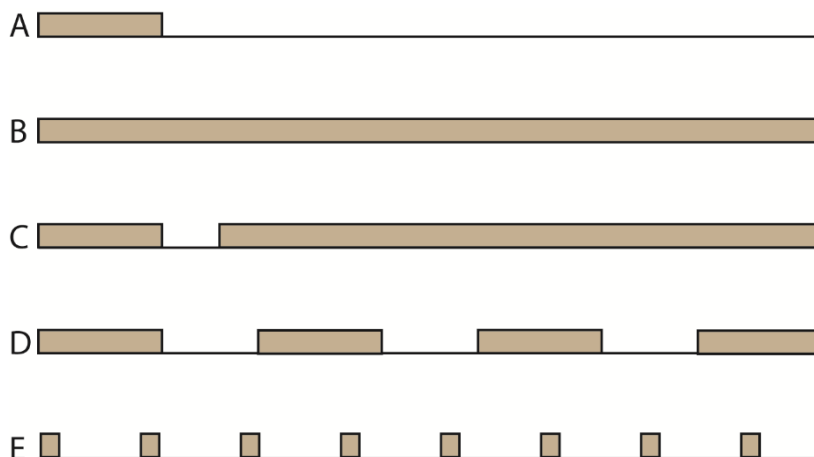


Bild 8: Blinkmuster der Service-LED

Die Bedeutung der LED-Anzeige ist wie folgt:

A) NORMAL OPERATION: Beim Starten leuchtet die Diode kurz (<1 Sek) auf und erlischt dann für immer. Der NEURON-Chip ist konfiguriert und arbeitet korrekt.

B) FATAL ERROR: der Neuron-Chip konnte nicht starten (Clock, CPU-Bus, Reset oder Firmwareproblem). In der Regel wurde die Printplatte oder deren Komponenten beschädigt.

C) APPLICATIONLESS: Im „applicationless“ Status konnte der NEURON-Chip starten, hat aber eine nicht mit der Hardware übereinstimmende Applikation gefunden. In diesem Fall muss eine neue Firmware geladen werden. Die LED zeigt beim Starten zuerst „normal operation“ an, um dann nach 3



Sekunden die LED dauernd einzuschalten.

D) UNCONFIGURED: Ein unkonfigurierter Knoten blinkt die LED mit einer Frequenz von 1 Hz. Die Hardware arbeitet richtig, hat aber das Anwenderprogramm noch nicht gestartet. Der Knoten muss nun konfiguriert werden (Zuweisung einer logischen Adresse), um in den „normal operation mode“ übergeführt zu werden.

E) WATCHDOGING: Der interne Watchdog des NEURON-Chips startet den Chip alle 750 mS neu, was mit einem kurzen Aufblinken der LED angezeigt wird. Der Knoten möchte eigentlich normal starten, findet aber einen Laufzeitfehler. Fehlerursache können nicht funktionierende Parallel Ports oder nicht synchronisierte bitserielle Schnittstellen sein.

Die Firmware des Chips wird beim Aktivieren des Service-Pin in jedem Falle gestartet, unabhängig davon, ob der Knoten bereits ein Anwendungsprogramm trägt und ob die Netzwerkkonfiguration bereits erfolgt ist.

Der Service-Pin unterliegt der Kontrolle durch die Software (Firmware), wenn er mit einem I/O-Pin verbunden wird. Das Hauptprogramm des Netzwerk-Prozessors fragt den Service-Pin regelmässig nach jedem abgeschickten oder empfangenen Telegramm ab. Auch vom Anwendungsprogramm aus kann auf den Service-Pin zugegriffen werden. Vom Programmierer sind beim Schreiben des Anwendungsprogramms jedoch gewisse Unterschiede in der logischen Einordnung des Service-Pin, die vom Prozessortyp und der Firmwareversion abhängen, zu beachten.

### 2.2.7.3 **Konfigurierbarkeit mit Legacy-Knoten**

Legacy-Knoten verfügen über eine Datenstruktur, welche die Verknüpfung zu ihren Netzwerkpartnern zulässt. Diese Datenstruktur wird in der Regel von einem Installationstool verwaltet, welches die Kontroller über die Systemfunktionen übernimmt. Zwei Domaintabellen dienen zur Speicherung der Domainzugehörigkeit. Weiter können 64 Selektoren für Netzwerkvariablen eingetragen werden, welche den Eintrag der Bindings ermöglichen. Damit der Knoten weiss, wohin er abgehende Daten senden kann, stehen ihm 14 Adresstabellen zur Verfügung.

Wenn nun eine Ausgangsvariable einen neuen Wert erhält, schaut das Programm in der nv\_tab nach, welcher Selektor eingetragen ist und mit welcher Adresstabelle gearbeitet werden muss. Die Adresstabelle wiederum enthält die Information, welche Domain zu benutzen ist. Auf diese Weise wird die Adresse des Telegramms zusammengestellt. Ein Legacy Chip kann also maximal 15 andere Knoten direkt adressieren. Wenn Gruppenadressen verwendet werden, können maximal 15 Gruppen bedient werden, wobei auch ankommende Gruppenmeldungen in der Adresstabelle eingetragen werden müssen. Die Gruppentabellen können aber mehrere Selektoren verwenden, so dass ein Knoten auf mehr als 15 Empfänger verknüpft werden kann.

### 2.2.7.4 **ECS Knoten**

Die ECS (Enhanced Command Set) Knoten unterstützen eine grössere Anzahl Adresstabellen und weitere flexiblere Konfiguration.

ECS Knoten verwenden, wie der Begriff sagt, zusätzliche Netzwerkkommandos. ECS Knoten lassen sich ohne Kompatibilitätsprobleme über LNS-Tools in die Netzwerke integrieren.

### 2.2.7.5 **Alias Tabellen**

Alias Tabellen ermöglichen eine flexiblere Integration der Geräte in komplexe Netzwerkstrukturen. Diese sogenannten „alias-bindings“ werden von LNS-Tools automatisch verwaltet.

## 2.3 LonWorks Transceivers

Die Transceiver bilden den grossen Vorteil der LonWorks-Technologie. Durch diese Bauteile ist es den Produzenten möglich, auf verschiedenste Medien effizient zugreifen zu können. Aufgrund der verschiedenen Transceiver-Technologien können entsprechende Bustopologien gebildet werden.

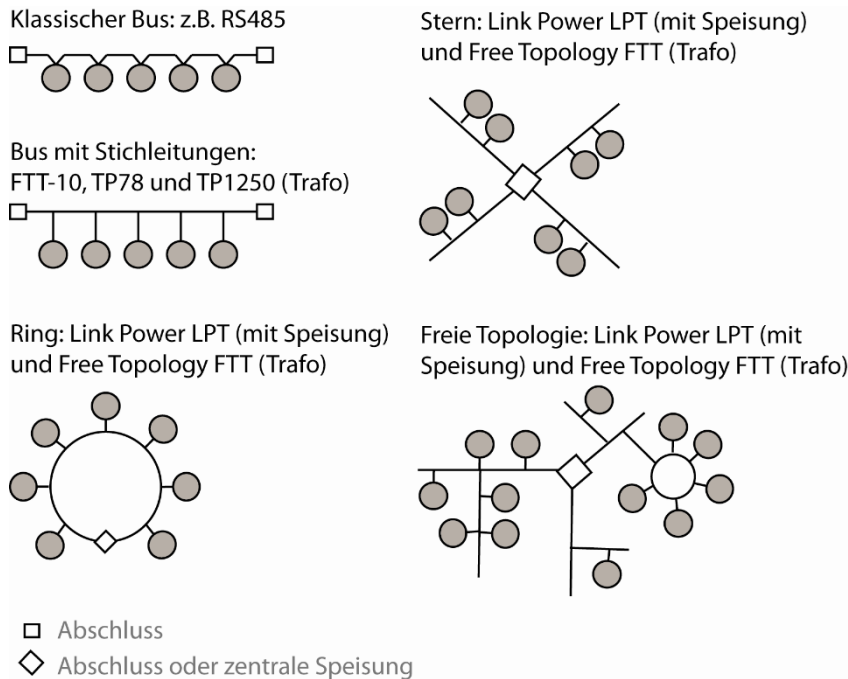


Bild 9: Mögliche Bustopologien

### 2.3.1 Twisted Pair TP 78

Für die konventionelle Bus-Topologie kann mit dem Twisted Pair Transceiver für 78.1 kBit/s oder 1.25 MBit/s gearbeitet werden. Der mittels Transformator getrennte Bus garantiert eine hohe Störfestigkeit.

#### TP-78

Strecke: 1400 m, beidseitig abgeschlossen  
 Knoten pro Channel: 64  
 Stub: maximal 3 M  
 Spezielles: bei Minustemperaturen nur 44 Knoten pro Channel  
 Zero Voltage Bereich: +220 V..220 V rms

### 2.3.2 Free Topology FTT-10

Der FTT-10 ist zweifellos der beliebteste Transceiver, welcher sich als Standard durchsetzen wird. Die Führung eines Feldbusses in wilder Topologie ist zum heutigen Zeitpunkt nach wie vor eine technologische Spitzenleistung. Besonders herausragend ist die einfache Integration dieser Komponente in Produkte, wobei die Richtlinien bezüglich Design eine erfolgreiche CE-Zertifizierung praktisch garantieren.

#### FTT-10

Strecke: 2700 m, beidseitig abgeschlossen und in Bustopologie  
 400 m in freier Topologie und einseitig abgeschlossen  
 Knoten pro Channel: 64  
 Zero Voltage Bereich: +220 V .. 220 V rms

### 2.3.3 RS-485

Der RS-485 ist nach wie vor die billigste Lösung, bietet aber (je nach Spezifikationstyp) nur einen Zero Voltage Bereich von -7 bis +12 V. Eignet sich speziell für kleinere Installationen.

Typ	Medium	kBit/s	Länge / Topologie / Bem.	Anz. Knoten
<b>TP- RS485</b>	Verdrillte 2 Drahtleit.	39 bis 625	1200m bei 39kBit/s, Bus, mit oder ohne galv. Trennung	32 pro Bus Segment
<b>TPT/XF 78 Trafo</b>	Verdrillte 2 Drahtleit.	78	1400m, Bus mit 3m Stichleitungen, Isolation 277V RMS	64 pro Bus Segment
<b>TPT/XF1250 Trafo</b>	Verdrillte 2 Drahtleit.	1250	130m, Bus mit 0,3m Stichleitungen, Isolation 277V RMS	64 pro Bus Segment
<b>FTT10 Trafo</b>	Verdrillte 2 Drahtleit.	78	2700m als Bus, 500m bei freier Topologie, Isolation 277V RMS	64 pro Bus Segment
<b>LPT10 Link Power</b>	Verdrillte 2 Drahtleit.	78	500m, freie Topologie, 42V DC, 5V / 100mA pro Knoten	32 - 128 pro Bus Segment
<b>PLT20 Power Line</b>	230 VAC oder DC	4,8	50 m – 5 km, BPSK Modulation Cenelec Band C, 132.5kHz	je nach Netz
<b>PLT30 Power Line</b>	230 VAC oder DC	2	50m - 5km, Spread Spectrum Cenelec Band A, 9-95kHz	je nach Netz
<b>IP-852</b>	Tunneling über IP		Alle IP Kanäle	

### 2.3.4 Link Power

Beim Einsatz von Link-Power-Transceivern fließen Daten und Versorgungsenergie (48 V) gemeinsam und verpolungssicher über eine verdrillte Zweidrahtleitung. Ein im Transceiver integriertes Schaltnetzteil kann den LON-Knoten inklusive Applikationsschaltung mit bis zu 100 mA bei +5 V versorgen. Dabei speist ein zentrales Netzteil ein bis zu 320 m langes Bussegment. Die Busausdehnung lässt sich durch Verknüpfen mehrere Link-Power-Segmente vergrößern. Beim Verlegen der Busleitung muss der Installateur nicht auf etwaige Maximallängen von Busabzweigen oder andere topologische Einschränkungen achten, da der LPT-10-Transceiver die freie Wahl der Topologie (Stern, Ring, Multidrop) erlaubt. Der gleiche Gedanke war Auslöser für die Entwicklung des FTT-10. Im Gegensatz zum LPT-10 besitzt dabei jeder LON-Knoten seine eigene Spannungsversorgung. Beide Varianten können auch gemischt werden.

### 2.3.5 Power Line

Mit der Thematik "Datenübertragung über die Netzleitung" haben sich schon Generationen von Entwicklungsingenieuren befasst. Das Medium Netzleitung besitzt einen enormen Vorteil: Es ist in Wohngebäuden wie in Zweckbauten bereits vorhanden, und das Aufreißen der Wände zum Verlegen von Busleitungen kann daher entfallen. Gleichzeitig hat die für die Energieübertragung gedachte Netzleitung als Datenübertragungsmedium einen ebensogrossen Nachteil: Die Leitungscharakteristik ist von Ort zu Ort verschieden und kann sich ausserdem je nach Art und Anzahl der angeschlossenen Verbraucher von einem zum nächsten Moment ändern.

Schaltnetzteile, Elektromotoren oder Dimmer sind dabei weit verbreitete Störquellen, die die auf die Netzleitung aufmodulierten Datensignale teilweise bis zur Unkenntlichkeit verfälschen. Dank Ausnutzung der zur Verfügung stehenden Übertragungsbandbreite, durch die Wahl passender Modulationsverfahren und mit geeigneter Signalfilterung kann man dennoch die Netzleitung zur Übertragung von Informationen nutzbar machen. LonWorks bietet hierzu drei Power-Line-Transceiver-Module an.

Die von den jeweiligen Behörden genehmigten Frequenzbänder für die Datenübertragung auf der Netzleitung unterscheiden sich zwischen Nordamerika, Japan und Europa. In Amerika und Japan ist

dazu der Frequenzbereich von 0 bis 500 kHz freigegeben. Diese grosse Bandbreite erlaubt den Einsatz eines Spreizband-Modulationsverfahrens (Spread-Spectrum-Modulation). Hierbei werden die Informationen breitbandig in einem grossen Frequenzbereich übertragen. Störungen, die vielfach in ihrer Bandbreite begrenzt sind, können die Datenübertragung daher nicht im gesamten Frequenzband beeinträchtigen. Der nur in den USA zugelassene Power-Line-Transceiver PLT-10 arbeitet nach diesem Verfahren im Bereich von 100 kHz bis 450 kHz und erreicht dabei eine Netto-Datenrate von 10 kBit/s.

In Europa hat das CENELEC (Comité Européen de Normalisation Electrotechnique; Europäisches Komitee für Elektrotechnische Normung) nur den Frequenzbereich bis 150 kHz (Anfang des Langwellenfunks) für die Kommunikation auf der Netzleitung freigegeben. Dieser Bereich ist zudem in verschiedene Bänder unterteilt. Das CENELEC-A-Band (9 kHz bis 95 kHz) ist für den Datenaustausch der Netzbetreiber (Energieversorgungsunternehmen und Verteiler) reserviert. CENELEC-B-Band (95 kHz bis 125 kHz) dient der Kommunikation ohne Zugriffsprotokoll für Endkundenanwendungen. Im CENELEC-C-Band (125 kHz bis 140 kHz) findet protokollgesteuerte Datenkommunikation für Customer-Applikationen statt. Der A-Band-Transceiver PLT-30 benutzt ebenfalls das Spread-Spectrum-Verfahren und erreicht damit in diesem Frequenzband eine Datenrate von 2 kBit/s. Das schmale C-Band erfordert ein anderes Modulationsverfahren. Beim PLT-20 kommt BPSK (Binary Phase Shift Keying) zur Anwendung. Damit erreicht dieser Transceiver eine Datenrate von 4 kBit/s.

Zur Untersuchung von vorhandenen Niederspannungsnetzen (230 V) auf ihre Tauglichkeit für den Einsatz als Datenkommunikationsmedium stellt Echelon den Power Line Communications Analyzer (PCLA) bereit. Dieses Gerät ermöglicht eine Reihe von Tests, die neben der Telegrammfehlerrate auch Aufschluss über die analogen Übertragungsparameter (Dämpfung, Störungen und Signalverzerrungen) der Netzleitung geben. Zusätzlich gibt es ein PC-basiertes Testkit (PLE-30), mit dessen Hilfe eine Kommunikationsverbindung zwischen zwei oder mehreren PCs aufgebaut und das Senden und Empfangen von Telegrammen unter veränderbaren Übertragungsparametern erprobt werden kann.

### 2.3.6 Weitere Transceiver

Weiter sind folgende Transceiver auf dem Markt verfügbar:

- Eigensichere Transceiver 78 kBit/s
- Funk 432 MHz
- Lichtwellenleiter
- Infrarot
- Koax
- Tf-Leitung
- Microwave

### 2.4 LonWorks Tools

Das vierte Element, die LonWorks Tools, umfassen die Entwicklungs- und die Installationswerkzeuge. Sie dienen dazu, Knoten zu entwickeln oder Installationen zu planen und durchzuführen.

Im Rahmen dieser Einführung wird nur eine Liste der gängigsten Tools beigefügt, da Tools im Rahmen eines Developerkurses oder eines Systemintegrationskurses abgehandelt werden. Weitere Tools, welche vor allem für die Entwickler von Bedeutung sind, sind Entwicklungswerkzeuge für Neuron-C und solche für Host-Applikationen. Es ist möglich, Anlagen so aufzubauen, dass mittels Feldcomilern jeder Knoten mit der zugehörigen Quellcodesoftware unterstützt und über das Netz mit neuen Programmen erweitert werden kann. Diese Fähigkeit ist für Feldbussysteme einmalig, wird aber in der Regel nur auf besonderen Wunsch verfügbar gemacht (Offenlegung der Firmware-Quellcodes). Auf Stufe „runtime-library“ ist aber eine auf alle Knoten transparente Softwarewartung durchaus üblich.

Installationswerkzeuge:

- LonMaker
- NL-220
- NL-Facilities

Alle gängigen Tools bauen auf den Standards für Windows Workstations auf und ermöglichen eine objektorientierte Gestaltung (Active-X OXC-Komponenten) der Leitsoftware sowie deren knotenspezifischen Funktionen. Bei der Wahl eines Installationswerkzeuges muss darauf geachtet werden, dass sogenannte „Device Plug Ins“ für die ausgewählte Hardware zur Verfügung stehen. Ein solches Plug-In stellt dem Systemintegrator eine grafische Oberfläche für die einfache Parametrierung des Knotens zur Verfügung, welche in das Installationswerkzeug eingebettet wird. Mittels einem Doppelklick auf das Knotenabbild wird das entsprechende Plug-In Fenster geöffnet.

Tools werden in der Regel so vermarktet, dass pro installierten Knoten eine Gebühr entfällt. Dadurch werden die Werkzeuge für kleinere Installationen innerhalb einem verträglichen Preisrahmen erhältlich. Der Aufwand einer Systemkonfiguration bezüglich Planung und Zeit wird vielfach unterschätzt. Während bei konventionellen Installationen einzelne Datenpunkte mittels Kabel verbunden werden mussten, muss die Verbindung bei LonWorks mit dem Tool erfolgen. Der Aufwand zur Verarbeitung der Information bleibt der Gleiche. Er ist aber auf den ersten Blick nicht auf die gleiche Weise offensichtlich wie dies bei mit Elektroschemas gefüllten Ordnern der Fall ist.

## 3 Der LonMark-Standard

### 3.1 Der physikalische Layer (Layer 1)

Der physikalische Layer von LonMark übernimmt die Transceiverspezifikation und ist für folgende Transceiver definiert:

- TP-RS 485-39
- TP/XF-78
- TP/XF-1250
- TP/FT-10
- PL-10 (L-E)
- PL-20 (L-N)
- PL-20 (L-E)
- PL-30 (L-N)
- IP-852

### 3.2 Layer 2-6

LonMark definiert nur in Layer 2 und 4 minimale Zusatzbedingungen:

- Layer 2: minimale Quarzfrequenzen bezogen auf Transceiver
- Layer 4: festsetzen der minimalen Grösse der Transactionbuffer auf 66 Bytes

### 3.3 Der Application Layer (Layer 7)

Um die Interoperabilität sicherzustellen, werden Netzwerkvariablen zu Objekten zusammengefasst, die, logisch betrachtet, Sensoren-, Aktoren und Controllerfunktionen darstellen.

Hier setzt LonMark den Hebel an und hat bereits über hundert SNVTs (Standard Network Variable Type) und SCPTs (Standard Configuration Parameter Types) definiert, welche die Interoperabilität von Variablen in Bedeutung, Wertigkeit und Bereich garantieren.

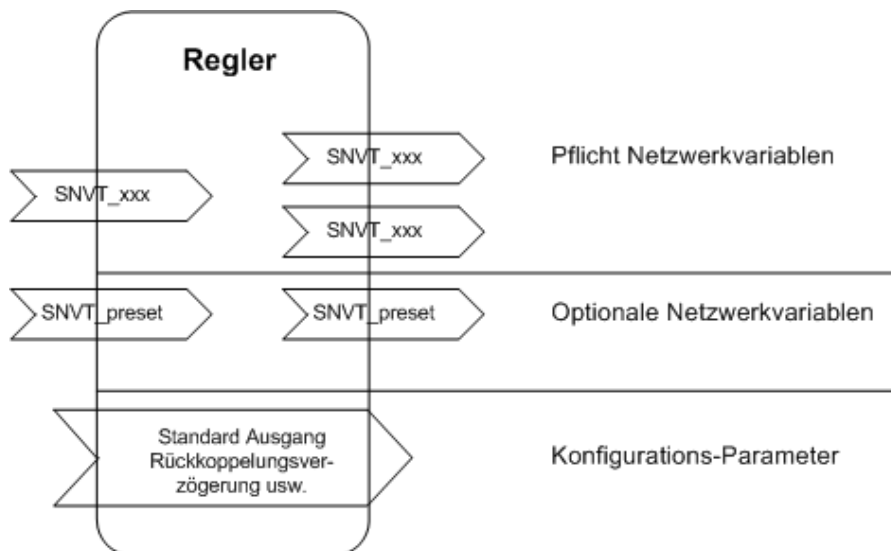


Bild 10: Dokumentation eines Knotens mit LonMark Objekten

Eine SNVT wird mit einer Nummer versehen, welche den Typ definiert. Weiter wird im Knoten eine auf die SNVT bezogene Information abgelegt, welche mit den Installationstools aus dem Knoten gelesen werden kann. Diese Textinformation enthält in der Regel den Variablennamen, so dass daraus die Funktion verstanden werden kann.

Die folgende Tabelle zeigt einen Auszug aus der SNVT Definition von LonMark.

Messung	Name	Bereich	Nr.
Speed	SNVT_speed	0..6553.5 m/ sec in 0.1 m/s	34
	SNVT_speed_f	-1E38..+1E38 m/s	39
Sound level	SNVT_sound	-327.68...327.67 dB (0.01 dB)	33

SNVTs können ganze Strukturen enthalten, so enthält z.B. der SNVT\_time\_stamp eine komplette Zeitinformation in Jahr, Monat, Tag, Stunde, Minute und Sekunde.

### 3.4 LonMark-Objekte

#### 3.4.1 Die Struktur eines LonMark Objektes

Für die Anwendung auf einem interoperablen Netzwerkknoten existiert das LonTalk-Protokoll und damit der Zugriff auf das Netz nur in Form von LonMark-Objekten. Diese sind durch ihren Typ (über eine von der LonMark-Organisation vergebene Nummer), einen Satz von Netzwerk-Input- und -Outputvariablen und einen Satz von Konfigurationsparametern gekennzeichnet. Sie stellen aus der Sicht der angestrebten Interoperabilität im Netzwerk Standards dar, bezüglich Form und Bedeutung. Die Objektnummer kennzeichnet den Typ, der Name dient nur zum Verständnis. Das Objekt besitzt immer eine oder mehrere unentbehrliche (mandatory) Netzwerkvariablen, und es kann wahlfreie (optional) NVs tragen. Die Eingangsvariablen werden links, die Ausgangsvariablen rechts eingezeichnet. Beide Arten zusammen werden von 1 bis n durchnummeriert, wobei nicht zwischen Input- und Outputvariablen unterschieden werden muss. Verwendet werden nur SNVTs.

Die Konfiguration dagegen, die auch über NVs eingetragen werden können, tragen die Nummer des zugehörigen Konfigurationsparameters (SCPT) aus der SCPT-Liste. Die Namen der NVs dürfen nicht länger als 11 Zeichen sein, sollten keinen Unterstrich enthalten und bis auf den ersten Buchstaben eines Wortes klein geschrieben werden. Beispiele sind in den folgenden Abschnitten zu finden.

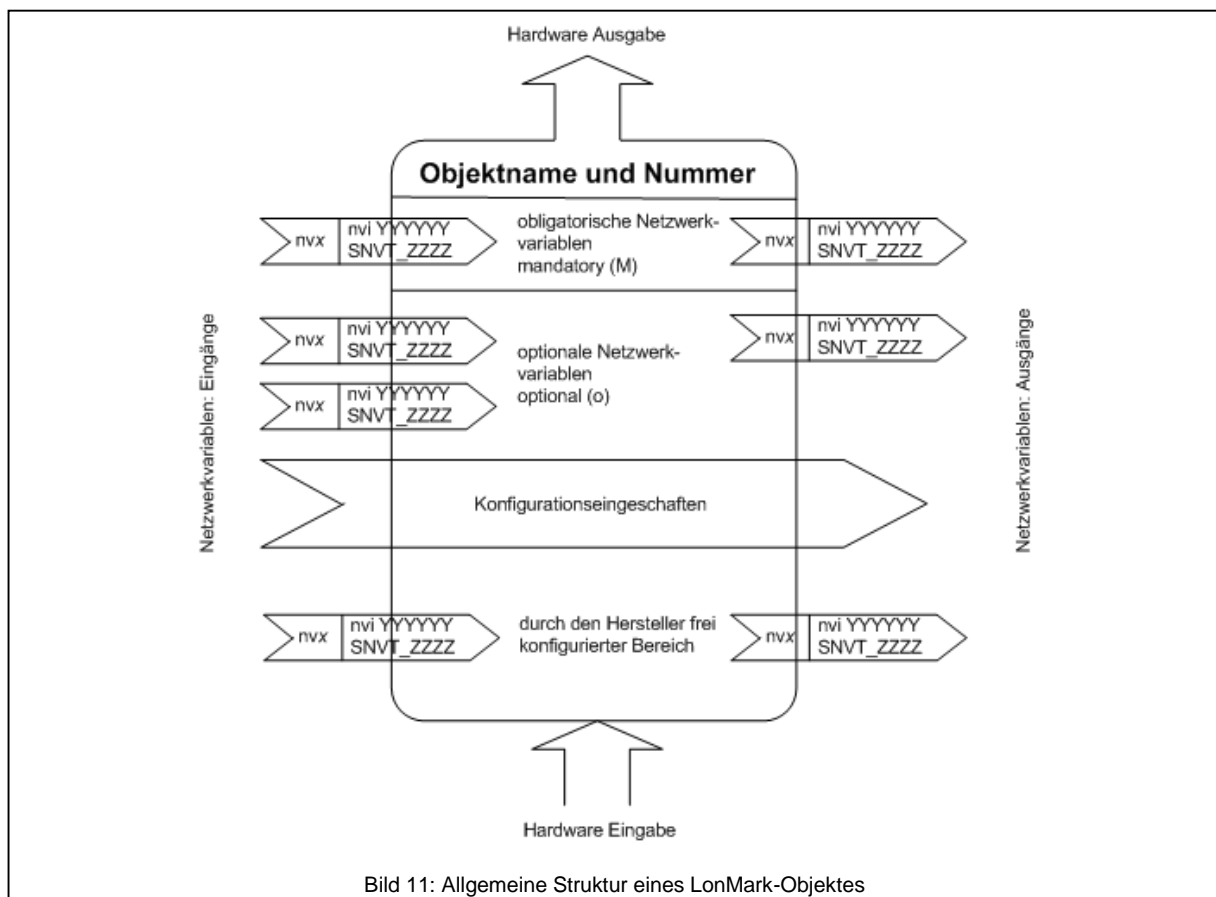


Bild 11: Allgemeine Struktur eines LonMark-Objektes

Die Namen erhalten einen Vorspann, der ihre Speicherklasse und die Übertragungsrichtung angibt:

- nvi ~ Eingangsvariable, gespeichert im RAM
- nvo ~ Ausgangsvariable, gespeichert im RAM
- nci ~ Konfigurationsvariable, gespeichert im EEPROM
- nro ~ (nur lesbare) Ausgangsvariable, gespeichert im ROM

Der Bezug zum Anwendungsprozess wird als Hardwareausgang oberhalb bzw. als Hardwareeingang unterhalb durch einen Pfeil dargestellt. Durch Präzisierung dieser allgemeinen Angaben entstehen die verschiedenen Objekttypen, wie Node-Objekt Sensor-Objekt, HVAC-Objekt usw.

### 3.4.2 Das Node-Objekt

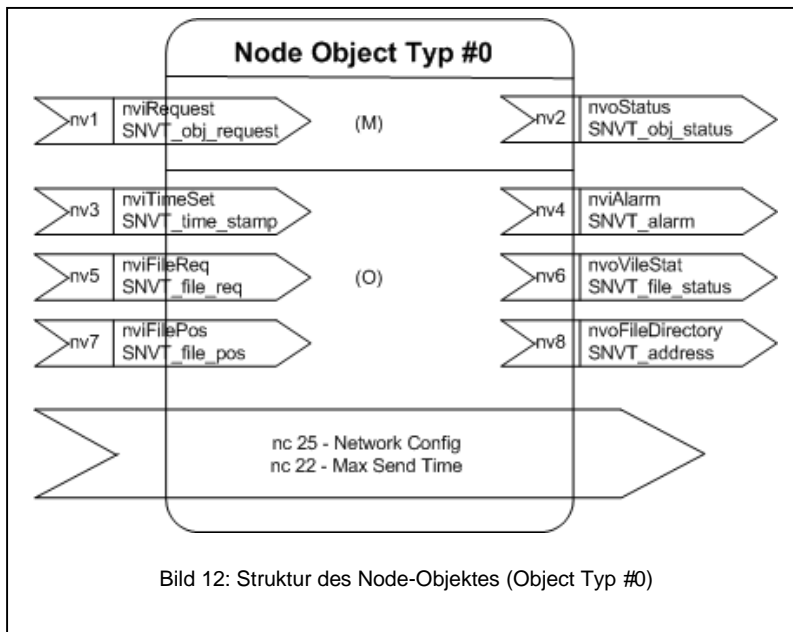


Bild 12: Struktur des Node-Objektes (Object Typ #0)

Das Node-Objekt (node object), dem die Objekttypnummer 0 fest zugeordnet ist, dient der Überwachung und Beeinflussung der Funktionen aller Objekte im Netzwerkknoten. Das wird mit Hilfe der beiden unverzichtbaren NV's nviRequest vom Typ SNVT\_obj\_request und nvoStatus vom Typ SNVT\_obj\_status realisiert.

Die Netzwerkvariable nviRequest enthält ein 2-Bytes-Feld für die Nummer des Objektes auf dem Knoten und ein 1-Byte-Feld für das als Zahl kodierte Kommando, z.B.:

- 0 ~ Rq\_Normal,
- 2 ~ RQ\_Update\_Status oder
- 3 ~ RQ\_Self\_Test

Mit dem Kommando "0" wird das adressierte z.B. aus dem inaktiven Zustand zur normalen Arbeitsweise zurückgesetzt. Wird das Kommando "2" an ein bestimmtes Objekt gegeben, sendet dieses Objekt über die Outputvariable SNVT\_obj\_status des Node-Objektes seinen aktuellen Status aus. Mit dem Kommando "3" kann ein Objekt auf dem Knoten zu einem Selbsttest veranlasst werden. Wird jedoch das Kommando "0" an das Node-Objekt selbst gerichtet, werden alle Objekte auf dem Netzwerkknoten in den Normalzustand gebracht.

In welchem Status sich ein Objekt befinden kann, zeigt die folgende (nicht vollständige) Aufzählung: disabled, out\_of\_limits, mechanical fault, electrical fault, unable\_to\_measure, comm\_failure, in\_alarm und andere. Elektrische und mechanische Fehler können aber nur dann gemeldet werden, wenn dafür die Hardwarevoraussetzungen bei der Knotenentwicklung geschaffen wurden.

Eine ausführliche Beschreibung dieser Zusammenhänge wird in den LonMark Application Layer Interoperability Guidelines gegeben. Das gilt auch für die optionalen NVs und die Konfigurationsparameter.

Der Konfigurationsparameter Max Send Time bestimmt die maximale Wartezeit, nach deren Ablauf das Objekt von selbst seinen Status über die Netzwerkvariable NVT\_obj\_status meldet, ohne dass ein NV-Update vorangegangen ist. Diese Funktion wird als "Herzschlag" bezeichnet und zeigt wie dieser an, dass das Objekt "noch am Leben" ist. Der Konfigurationsparameter Max Send Time trägt die Nummer 22 aus der Master-List der SCPTs.



## 4 Die Bausteine eines Netzes

### 4.1 Knoten

Die Knoten wurden in Kapitel 2.2 behandelt. In diesem Kapitel wird auf die durch den Systemintegrator benötigte Information verwiesen, um die Sichtweise des Systemintegrators zu dokumentieren.

Der Systemintegrator braucht für seinen Knoten mindestens folgende Angaben:

- einen guten und vollständigen Funktionsbeschreibung
- ein sogenanntes XIF-File, welches das Netzwerkinterface beschreibt
- die Beschreibung des elektrischen Interfaces
- mögliche Konfigurationsbeschreibungen
- mögliche Programmanpassung und Firmwareversionen

### 4.2 Netzorganisationsbausteine

Verschiedene Channels werden über Router logisch miteinander verknüpft, wobei die beiden Busschnittstellen des Routers unterschiedlicher oder gleicher physikalischer Natur sein können. Auf diese Weise verbindet man beispielsweise einen Funkkanal mit einer Zweidrahtstrecke.

Unter dem Oberbegriff Router fallen Kopplungsmöglichkeiten mit unterschiedlichen Vermittlungsmethoden (Router Algorithms):

#### 4.2.1 Repeater

Den einfachsten Router stellt ein Repeater dar, er leitet alle Telegramme von einem Channel zum anderen weiter. Neben der Umsetzung zwischen unterschiedlichen Übertragungsmedien kann ein Repeater auch zur analogen Signalregenerierung und damit zur Verlängerung des Busses eingesetzt werden.

#### 4.2.2 Bridges

Die nächste Stufe in der Routerhierarchie ist die Bridge. Eine Bridge ist ein Router mit lokaler Intelligenz. Sie vermittelt nur Telegramme innerhalb derselben Domain, wobei zwei Domains übertragen werden können.

#### 4.2.3 Learning Router

„Learning Router“ beobachten den Datenverkehr auf den beiden angeschlossenen Netzwerkbereichen und erschliessen sich daraus den Netzaufbau auf Domain- und Subnet-Ebene. Dieses Wissen benutzt der Learning Router dann, um die Telegramme auszuwählen, die er von einem Channel zum anderen weiterleitet. Da ein Learning Router aus dem Telegrammverkehr nicht auf bestehende Gruppentopologien schliessen kann, werden stets alle Telegramme mit Gruppenadressen weitergeleitet.

#### 4.2.4 Configured Router

„Configured Router“ setzen dagegen nur ausgewählte, in einer Vermittlungstabelle (Routing Table) eingetragene Telegramme zwischen Channels um. Die Vermittlungstabelle erstellt man mit Hilfe eines Netzwerkmanagement-Tools. Da dieses Tool auch über die Vergabe von Gruppenadressen bestimmt, kann ein Configured Router auch für das selektive Vermitteln von Gruppentelegrammen programmiert werden.

#### 4.2.5 Warum Router einsetzen?

Configured Router und Learning Router gehören zur Klasse der intelligenten Router. Sie sind nicht nur ein Mittel zum Verbinden von physikalisch unterschiedlichen Übertragungsmedien. Dank ihrer Programmierung können sie auch als Telegrammfilter zwischen physikalisch gleichartigen Channels eingesetzt werden. Indem sie nur ausgewählte Telegramme in andere Bereiche weitervermitteln, beschränken sie den Telegrammannahmeverkehr auf den örtlichen Bereich. Der Rest des LON-Systems bleibt so von dem für ihn uninteressanten Datenverkehr verschont.

# 5 Anhang

## 5.1 Begriffe

### 3120

NEURON-Chip 3120. Chip von MOTOROLA / TOSHIBA mit internem EEPROM, RAM und integriertem LON-Interface für Netzwerkkommunikation auf OSI Layer 7.

### 3150

NEURON-Chip 3150. Chip von MOTOROLA / TOSHIBA mit internem EEPROM und externem EPROM und integriertem LON-Interface für Netzwerkkommunikation auf OSI Layer 7.

### Address table

Eine Tabelle in einem Neuron Chip, welche die Gruppenmitgliedschaft eines Knotens und die Sendeadresse einer gebundenen Netzwerkvariable definiert. Auf einem Neuron Chip können 15 verschiedene Adresstabellen definiert werden.

### Alias Netzwerkvariable

Ein sekundärer Platz in einer Netzvariablen-tabelle, welcher eine „primary netvar“ referenziert. Eine Alias Netzwerkvariable wird parallel zur primary NV bedient und ermöglicht es, Daten mehrfach zu verknüpfen (z.B. Reset-Kdo über Group-Address, normale Kdos über Subnet/Node Address).

### Application Image

Das Applikationsprogramm, welches auf einem Neuron Chip lauffähig ist.

### Application Layer

Übertragungslayer, welcher die Applikations-Level Kompatibilität sicherstellt. Siehe auch unter OSI-Layer 1-7.

### Application message

Eine explicit message mit einem Messagecode zwischen 0x00 und 0x3e (62d). Die Interpretation des Codes ist der Applikation überlassen.

### Binder

Ein Softwarewerkzeug, welches Netzvariablen oder msg\_tags verbinden kann.

### Binding

Der Prozess, welcher die Verbindung zwischen

Knoten definiert.

### Bridge

Router mit zwei NEURON-Chips welcher die Meldungen von max. 2 Domains auf beiden Seiten abbildet.

### Broadcast

Adressierungsart, welche innerhalb eines Subnets oder einer Domain alle Knoten gleichzeitig erreicht.

### Channel

Physikalischer Lon-Bus-Teil, z.B. zwischen zwei Routern

### cloned\_domain

Die Domain mehrerer Knoten, deren *must\_be\_one* Bit auf 0 gesetzt worden ist. Eine cloned\_domain wird nur in Ausnahmefällen benutzt und entspricht nicht den „interoperability guidelines“ nach LONMARK. In einer cloned\_domain kann die Subnet/Node Adressierung nicht mehr angewendet werden. In einer solchen Domain wird mit Broadcast und NeuronID Adressierung gearbeitet.

### cloned\_node

Ein Knoten, dessen *must\_be\_one* Bit auf 0 gesetzt wird. Ist in der Lage, Meldungen von Knoten zu empfangen, welche mit der gleichen Subnet/Node Adresse arbeiten. Wird beim Export des MIP auf dem LON-Builder oder durch die Funktion *update\_clone\_domain* gesetzt.

### Configuration network variable

Eine spezielle Netzwerkvariablenklasse, welche die Speicherung von Applikationskonfigurationsdaten ermöglicht. Konfigurationsdaten sind immer Inputvariablen, welche im EEPROM gespeichert werden. Bei Host based nodes muss der Host sicherstellen, dass die Daten in einem nicht flüchtigen Speicherbereich abgelegt werden.

### Configured Router

Router mit zwei NEURON-Chips, welcher mittels Konfigurationsdaten weiss, welche Telegramme zu übermitteln sind

### Connection

Die implizite Adressierung, welche durch das Binding installiert wird. Eine Connection besteht zwischen zwei oder mehreren teilnehmenden Knoten.

#### **Declared msg\_tag**

Im Applikationsknoten definierte msg\_tags. Deklarierte msg\_tags sind immer bidirektional.

#### **Differential-LON-Interface**

Mit Trenntrafo galvanisch getrenntes LON Interface auf 2-Draht Leitung. Die Übertragungsrate beträgt in der Mehrzahl der Applikationen 78,1 kbps.

#### **Domain**

Eine logische Verbindung mehrerer Knoten auf einem oder mehreren *channels*.

Kommunikation kann nur zwischen Knoten einer gleichen DomainID stattfinden, es sei denn, ein Router verbinde zwei *Domains*.

#### **DomainID**

Der oberste Level der LON-Bus Adresshierarchie. Die ID kann eine Länge von 0,1,3 oder 6 Byte aufweisen. Die 0-Länge ist für NSS-10 Knoten reserviert, um Installationsaufgaben zu koordinieren und sollte nicht von Applikationsknoten verwendet werden.

#### **Downlink**

Datenübertragung von einem Host in ein Neuron Chip, in der Regel über die parallele Schnittstelle.

#### **Explicit address**

Durch die Applikation (z.B. MIP) erstellte und verwaltete in der Meldung enthaltene Adresse.

#### **Explicit message**

Durch eine NEURON oder Host-Applikation explizit ausgelöste Meldung, bei welcher der Inhalt sowie der Zeitpunkt der Übertragung durch den Applikationscode definiert wird.

#### **Flush**

Der Flush-Status eines MIP-Interface bewirkt, dass die auf dem LON-Bus übermittelten Meldungen nicht mitgeschnitten werden. Nach einem Reset befindet sich das MIP im Flush-Status, damit die Host-Applikation genügend Boot-Zeit zur Verfügung hat.

#### **Flush cancel**

Damit das MIP-Interface die LON-Messages mitschneidet, muss nach einem Reset der Befehl „Flush Cancel“ über das Parallelinterface gesendet werden. Wenn der Neuron-Chip „Flush complete“ meldet, ist die Hostapplikation mit dem LON-Bus verbunden.

#### **Free Topology Transceiver**

Aktiver Transceiver mit 78,1 kbps, welcher eine freie Bustopologie zulässt. Ein LON-Bus mit FTT Technologie kann über eine maximale Strecke von 400m betrieben werden. Nach jedem 400m Segment muss ein Physical Layer Repeater (2- oder 4-Weg, pro Weg ein FTT) installiert werden. Auf diese Weise kann eine praktisch unbegrenzte totale Netzlänge erreicht werden.

#### **Gateway**

Datenbrücke, welche Daten auf dem Application-Layer austauscht. Kann zwischen zwei Domains oder verschiedenen Netzwerkprotokollen verwendet werden.

#### **Group**

Möglichkeit, logische Gruppen über die Subnetzgrenze hinaus zu bilden. Es sind maximal 256 verschiedene Gruppen möglich.

#### **Group address**

Möglichkeit, logische Gruppen oder einzelne Gruppenmitglieder über die Subnetzgrenze hinaus zu adressieren.

#### **Group ID**

Eine Nummer zur Identifikation einer Gruppe. Jede Gruppe ist mit einer (einmaligen) Gruppennummer zwischen 0 und 255 definiert. Die Nummer 0 gilt für „huge groups“, d.h. eine Gruppe mit einer unlimitierten Anzahl von Mitgliedern.

#### **Group member**

Mitgliednummer in einer Gruppe. Es sind maximal 64 einzeln adressierbare oder eine unbegrenzte Anzahl, nicht über die Member-Identifikation adressierbare Gruppenmitgliedern möglich.

#### **Host**

Ein Mikroprozessor, welcher den Layer 7 des LON-Protokolls integriert hat. Kann ein an den Neuron Chip angekoppelter Mikroprozessor oder ein Neuron Chip sein.

#### **Host application**

Das in einem Host integrierte Applikationsprogramm.

#### **Host based node**

Ein Knoten, in welchem der Layer 7 des LonTalk Protokolls in einem nicht Neuron Chip Mikroprozessor lauffähig ist.

#### **Hub**

Das Zentrum einer Verbindung. Der Hub (Angelpunkt) hat entweder einen Eingang und mehrere Ausgänge oder mehrere Ausgänge und nur einen Eingang.

**Implicit address**

Implizit im NEURON-EEPROM enthaltene Adresse, welche verwendet wird, wenn auf eine Netzwerkvariable oder eine msg\_tag zugegriffen wird. Die Applikation referenziert die Adresse über den Netzwerkvariablenselector oder den msg\_tag.

**Implicit message**

Durch den NEURON-Core ausgelöste Meldung, wenn die Applikation einer Netzwerkvariablen Daten zuweist. Wird beim ersten Durchlauf des NEURON-Schedulers nach der Daten-Zuweisung übermittelt.

**Interoperability guidelines**

Verbindliche Richtlinien, nach welchem eine Zertifizierung erlangt werden kann. Ein nach diesen Regeln zertifiziertes Produkt ist berechtigt, das LONMARK Logo zu tragen.

**Interoperability, interoperable node**

Eine Produktklassifizierung, welche garantiert, das verschiedene Knoten verschiedener Hersteller in ein Netzwerk integriert werden können. Damit diese Installation vollzogen werden kann braucht es keine kundenspezifischen Werkzeuge oder Spezialentwicklungen. Interoperability wird durch die LonMark-Zertifizierung sichergestellt.

**Intersecting connections**

Ein Set von Verbindungen, welche mehr als eine globale Verbindung teilen (Mehrfachverknüpfungen von Variablen).

**Knoten**

Ist ein Node, wie er in der LON-Bus Technologie definiert ist: Eine Applikation mit einer LON-Schnittstelle.

**Learning Router**

Router mit zwei NEURON-Chips, welcher aus dem ankommenden Netzverkehr lernt, welche Meldungen zu übermitteln sind.

**Link Layer**

Übertragungslayer, welcher den Zugriff auf das Übertragungsmedium und das Übertragungsformat definiert. Siehe auch unter OSI-Layer 1-7.

**LON-Bus**

Durch die Firma Echelon definierter Feldbus, welcher mittels der NEURON-Chips angesteuert werden kann. Der LON-Bus ist ein Standardbus, welcher ein normiertes Protokoll über verschiedenste Medien wie 2-Draht-Leitung, Faseroptik, Mikrowellenstrecken, Funkstrecken, Netzübertragung u.s.w übertragen kann.

**LonBuilder**

Entwicklungswerkzeug mit Emulatoren und Routern, welche die Entwicklung einzelner Knoten sowie ganzer Netze ermöglichen.

**LON-Manager**

Ein Set von Hardware und Softwarewerkzeugen, welche die Installation, Konfiguration, Wartung, Überwachung und Kontrolle eines LONWORKS Netzwerkes ermöglichen.

**LonMark**

Ein Zertifizierungsprogramm, welches die Kompatibilität der Produkte verschiedener Hersteller garantiert.

**LonTalk®**

Das auf LONWORKS-Netzwerken verwendete Protokoll, welches die Kommunikation standardisiert. Es definiert den Standard, unter welchem die einzelnen Knoten Information austauschen.

**LonTalk file transfer protocol**

Ein definierter Weg, zwischen Knoten Datenfiles auszutauschen. Die Filetypen 0 und 1 sind durch LonMark als Konfigurationsdatenfiles definiert.

**LONWORKS**

Ein Set von Werkzeugen und Komponenten, um ein neuronales Netz von Sensoren, Aktuatoren und Kontrollgeräten zu erstellen.

**Mapper**

Knoten, welcher auf Explizit Messages basierende Daten in SNVT nach LonMark Standard abbildet.

**Message code**

Ein Feld in einer expliziten Meldung, welches den Typ der Meldung definiert.

**Microprocessor interface programm**

Firmware, welche die auf dem Bus erhaltenen Telegramme in einem Application Buffer abbildet. Auf diese Weise können die LonTalk Layer 4-7 in einem leistungsfähigen Mikrorechner implementiert werden.

**msg\_in**

Ein msg\_tag, welcher auf allen Knoten „defaultmässig“ existiert, um ankommende Meldungen entgegenzunehmen. Msg\_in kann nicht für abgehende Meldungen verwendet werden.

**msg\_tag**

Variable im EEPROM, welche das Einbinden von expliziten Meldungen in die EEPROM-

Adressinformation ermöglicht. Dient der impliziten Adressierung expliziter Meldungen und funktioniert im Prinzip wie eine „Netzwerkvariable“ für Meldungen. Ist immer bidirektional für Input und Output.

### **Network**

Ein Sub-System

### **Network address**

Die logische Adresse eines Knotens (Domain/Subnet/Node).

### **Network driver**

Software, welche auf einem (nicht Neuron Chip) Host läuft, um das Netzwerkinterface (Ankopplung an Neuron Chip) zu betreiben.

### **Network image**

Eine Netzwerkadresse eines Knotens sowie dessen Verbindungsinformationen. Es besteht aus der Domain-, Address- und Network Variable configuration table. Ist im EEPROM des Neuron Chip oder bei Host Applikationen (Network Variable configuration table) auf dem Host untergebracht.

### **Network interface**

Eine Apparatur, welche den Netzwerk Layer 6 an einen Host ankopfelt (z.B. PCLTA PC LonTalk Adapter).

### **Network interface API**

Eine Softwarebibliothek (C-Source), welche Basiskommunikationsfunktionen unterstützt. Ist im NSS-10 Developers Kit enthalten.

### **Network Layer**

Layer zur Übertragung, welcher die Zieladressierung sicherstellt. Siehe auch unter OSI-Layer 1-7.

### **Network management**

Der Prozess, ein Netzwerk logisch zu definieren, zu installieren und Instand zu halten.

### **Network services API**

Eine Softwarebibliothek (C-Source), welche Basis- Servicefunktionen unterstützt. Ist im NSS-10 Developers Kit enthalten.

### **Network variable**

High-Level Objekte, welche zur Kommunikation zwischen Applikationsknoten verwendet werden. Die Typen, Funktion und Anzahl der Netzwerkvariablen werden durch den Applikationscode des Knotens definiert. Netzwerkvariablen ermöglichen eine einfach Kommunikationsform, insbesondere wenn Neuron Chip Hosted Applikationen verwendet werden.

### **Network variable configuration table**

Eine Tabelle, welche einem Network variable index einen Selector zuweist. Für Downlinkvariablen wird zusätzlich eine Adresstabelle zugewiesen und mitverknüpft. Bei Neuron-Chip hosted Knoten befindet sich die Tabelle im Neuron-Chip EEPROM. Bei Host Applikationen wird die Tabelle im Host gespeichert, falls das MIP mit dem Pragma *netvar\_processing\_off* erstellt worden ist.

### **Network variable index**

Eine Nummer, welche zur Identifikation der Netzwerkvariable verwendet wird. Die Indexnummern werden durch den Neuron-C Compiler aufgrund der Position der Variable im Deklarationsteil zugewiesen. Die erste Variable entspricht dem Index 0. Neuron Chip Hosted Knoten können maximal den Index 61 verarbeiten, Host Applikationen können bis zum Index 4095 erweitert werden.

### **Network variable selector**

Eine 14 Bit Nummer zur Identifikation der Verbindung zwischen Netzwerkvariablen. Die Selektornummern werden vom für die Installation zuständigen Knoten zugewiesen.

### **Neuron Chip-hosted node**

Ein Knoten, bei welchem der Layer 7 des LonTalk Protokolls in einem Neuron Chip implementiert ist.

### **NEURON-Chip**

Von "Neuron" (die Zelle) abgeleitete Bezeichnung einer integrierten Schaltung, welche eine LON-Schnittstelle enthält und die Implementierung einer Applikation zulässt.

### **NeuronID**

Bei der Fabrikation eingebrannte, 48 Bit lange Identifikationsnummer eines jeden NEURON-Chip. Jede Nummer ist ein garantiertes Unikat.

### **Node**

Knoten. Eine Apparatur, welche die Layer 1 bis 6 des LonTalk Protokolls sowie einen Neuron Chip, Lon Transceiver, Memory und Trägerhardware enthält.

### **NodeID**

Der unterste Level der LonTalk Adresshierarchie, bestehend aus Domain/Subnet und Node. Während der Installation erhält jeder Knoten eine nur einmal vorkommende Subnet / Node Kombination zugewiesen. Ausnahme: *cloned\_node*. Es können 127 verschiedene NodeIDs definiert werden (1..127). Die NodeID 0 wird für einen noch nicht installierten Knoten verwendet.

## OSI-Layer 1-7

Layer 7: **Application Layer**. *Applikationslevel-Kompatibilität*: Standard Network Variable Types

Layer 6: **Presentation Layer**. *Data Interpretation*: Netzwerkvariablen, Foreign Frame Übertragungen.

Layer 5: **Session Layer**. *Remote Actions*: Request-Response, Authentication, Netzwerk Management, Netzwerk Interface.

Layer 4: **Transport Layer**. *Punkt zu Punkt Zuverlässigkeit*: Ackd / Unackd Service, Unicast/Multicast Authentication, Adresszuweisung und Kontrolle von Doppeleintragungen.

Layer 3: **Network Layer**. *Zieladressierung*: Adressierende Router

Layer 2: **Link Layer**. *Zugriff auf das Übertragungsmedium und Übertragungsformat*:

Framing, Data Encodeing, CRC Error Checking, CSMA, Vermeidung von Kollisionen, Prioritäts und Kollisionserkennung (optional)

Layer 1: **Physical Layer**. *Elektrische Verbindung*: twisted pair, power line, radio frequency, coaxial cable, infrared, fiber optic, RS-485 usw.

## Physical Layer

Layer zur Übertragung, welcher die elektrische Verbindung definiert. Siehe auch unter OSI-Layer 1-7.

## Poll

Ein explicit Request an einen Knoten, den Wert einer Variablen mit dem entsprechenden Selector zu senden.

## Polled network variable

Eine Output Netzwerkvariable, welche ihren Inhalt nur aufgrund von Pollingfragen sendet. Netzwerkvariablen senden normalerweise automatisch ihren Inhalt, wenn sich dieser verändert hat (d.h. wenn die Variable durch die Applikation beschrieben wurde).

## Polling network variable

Eine Input Netzwerkvariable, welche ihren Inhalt nur aufgrund von Pollingfragen an eine Outputvariable aufdatiert.

## Presentation Layer

Übertragungslayer, welcher die Datenpräsentation festlegt. Siehe auch unter OSI-Layer 1-7.

## Priority

Ein durch das LonTalk Protokoll unterstützter Mechanismus, um priorisierte Meldungen zu Übermitteln. Prioritymeldungen werden innerhalb eines reservierten Slots vor den

normalen Meldungen übermittelt. Speziell für die Übermittlung deterministischer Information (Zeitstempel, zeitkritische Daten) geeignet.

## Processed netvar

Adressierung der Netzvariablen mittels implicit address, d.h. mit im NEURON-Chip-EEPROM enthaltener Adressinformation.

## Program ID

Ein Identifikationsstring, welcher im EEPROM des Neuron Chip gespeichert wird. Der String wird zur Identifikation des

Applikationsprogramms benutzt, alle Knoten mit gleicher Programm-ID müssen über das gleiche externe Interface verfügen, da sonst Probleme mit Installationswerkzeugen auftreten. Interoperable Knoten, welche nach LonMark zertifiziert werden, enthalten eine *Standard programm ID*.

## Property

Ein Attribut eines Objektes, z.B. der Standort des Knotens.

## Repeater

Router mit zwei NEURON-Chips oder Physical Repeater, welcher alle Meldungen eines Channels auf dem nächsten Channel abbildet.

## Self-documentation

Ein Mechanismus welcher dem Applikationsknoten erlaubt, beschreibende Information im EPROM unterzubringen.

## Self-identification

Ein Mechanismus, welcher die Dokumentation von SNVT Variablen im PROM des Applikationsknotens (SNVT ID) ermöglicht. Diese Information kann bei der Installation mittels einem dazu geeigneten Softwaretool abgefragt werden.

## Serial LonTalk Adapter

Ein auf einer EIA-232 Schnittstelle basierendes Netzwerkinterface. Diese Information kann bei der Installation mittels einem dazu geeigneten Softwaretool abgefragt werden.

## Session Layer

Übertragungslayer, welcher die Externzugriffe (Remote Actions) definiert. Siehe auch unter OSI-Layer 1-7.

## SMX-compatible transceiver

Jeder Transceiver, welcher den Standard Modular Transceiver Erkennungscode verwendet.

## Standard network object

Eine Kollektion von Netzwerkvariablen mit zugehörigem Verhalten gemäss den

Anforderungen der LonMark Interoperability Guidelines.

### **Standard Network Variable Type**

Standard Netzwerk Variablentypen sind durch LonMark normierte Variablen, welche es ermöglichen, Daten aus Knoten verschiedener Hersteller auf einfache Art und Weise auszutauschen.

### **Standard Network Variable Type ID**

Ein normierter Code, welcher einem entsprechenden Variablentyp zugewiesen ist. Wird in ECHELON-Dokumenten gelegentlich auch als SNVT index benannt. Ein SNVT ID ist immer eine Zahl ungleich 0, 0 bedeutet, dass es sich bei der Variablen nicht um eine SNVT-Variable handelt.

### **Standard programm ID**

Eine Programm ID eines nach LonMark Interoperability Guidelines zertifizierten Knotens, welche Rückschluss auf Hersteller, Applikation und Softwareversion zulässt.

### **Subsystem**

Zwei oder mehrere Knoten, welche gemeinsam eine Funktion erfüllen. Die Konfiguration aller Knoten eines Subsystems wird durch ein einzelnes Installationswerkzeug durchgeführt.

### **Subnet**

Logisches Unternetz innerhalb einer Domain. Es kann maximal 127 Knoten enthalten, eine Domain kann 255 Subnets enthalten.

### **subnet / node address**

Standardadresse eines LON-Knotens. Total sind 32385 Kombinationen möglich.

### **Subnet ID**

Der zweite Level einer Subnet / Nodes Adressierungshierarchie. Gültige Subnetnummern sind 1..255. Die Subnetnummer 0 wird für einen nicht installierten Knoten verwendet.

### **System**

Ein oder mehrere unabhängig verwaltete(s) Sub-System(e). Ein System kann eine oder mehrere Domain(s) verwenden.

### **Transceiver**

Eine Apparatur, welche den Neuron Chip physikalisch an das Übertragungsmedium bindet.

### **Transceiver ID**

Eine 5 Bit Nummer, welche eine hardwaremässige Dekodierung des Transceivertyps zulässt.

### **Transport Layer**

Übertragungslayer, welcher die Pt. zu Pt. Übertragung sicherstellt. Siehe auch unter OSI-Layer 1-7.

### **Turnaround network variable connection**

Eine Netzvariablenverbindung, bei welcher sich Input und Output auf dem gleichen Knoten befinden.

### **Typeless network variable**

Eine Netzwerkvariable, bei welcher weder der Typ noch die Datenlänge bekannt sind. Für die Übertragung solcher Variablen ist die Host-Applikation verantwortlich.

### **Unprocessed netvar**

Adressierung der Netzvariablen mittels explicit address, d.h. mit an den Host-Applikationscode delegierter Adressinformation.

### **Uplink**

Datenübertragung von einem Neuron Chip in einen Host-Mikrocomputer, in der Regel über die parallele Schnittstelle.

### **Variable Fetch**

Ein Request an einen Knoten, den Inhalt der Variablen mit einem entsprechenden Index zu senden.

## **5.2 Abkürzungen**

### **CRC**

Übertragungskontrolle und Fehlerkorrektur

### **CSMA**

Kollisionsfähiges Netzwerkprotokoll, d.h. jeder Teilnehmer darf bei freiem Medium aktiv senden.

### **ECS**

Enhanced Command Set

### **FTT**

Free Topology Transceiver

### **IP**

Internet Protocol

### **IP-852**

IP Tunneling Standard für Feldbusse (u.a. LonTalk)

### **ISO**

International Standard Organisation

### **kbps**

kilobyte per second 1 kbps = 1000byte / sek = 1kHz

### **LNS**

Lon Network Services

### **LON**

Local Operating Network

### **LPA**

Lon Protocol Analyzer

### **MIP**

Microprocessor Interface Program

### **NIC**

Network Interface Card

### **OSI**

Open Systems Interconnection

### **SCPT**

Standard Configuration Parameter Type

### **SLTA**

Serial LonTalk Adapter

### **SNVT**

Standard Network Variable Type

### **TP**

Twisted Pair



### **5.3 Quellenverzeichnis**

LONWORKS Engineering Bulletin, April 1993: „LONTALK PROTOCOL“

ECHELON Engineering Bulletin, 1991: „NEURON Chipbased Installation of LONWORKS Networks“

LONWORKS Engineering Bulletin, Januar 1995: „Installation Overview“

LONWORKS Engineering Bulletin, Januar 1995: „Enhanced Media Access Control with LONTALK Protocol“

LONWORKS Users Guide, 1994 : „FTT-10 Free Topology Transceiver“, Version 1.2, Dokument Echelon 078-0114-01B

LONWORKS Host Application Programmers Guide, Revision 2, 078-0016-01B

ECHELON Data Book, Januar 1995: „Neuron Chip Data Book“

MOTOROLA Data Book, 1994 Rev 3: „Neuron Chip Distributed Communications and Control Processors“

LonMark, 1995 V 2.0: „Application Layer Interoperability Guidelines“

LonMark, 1994 V 1.3: „Layers 1-6 Interoperability Guidelines“ Ludwig Brackmann, „Local Operating Network“, ELRAD Heft 12/1994,1/1995

Nils Meinert, „Offene Kommunikation mit LON und BACNET“, LNO Info 1996

ANSI / ASHRAE 135-1995, BACNET specification 1995, ISSN 1041-2336

Fritz Kurt, EBV Elektronik, Grundlagenpräsentation zur LonWorks Technologie, Jan 1997

Dietrich Loy Schweinzer, "LON-Technologie", Hüthig Verlag, ISBN 3-7785-2581-6/1998. 1998

Tiersch F., LonTech® Thüringen e. V., "Die LonWorks®-Technologie", ISBN 3-932875-03-6, 1998

GNI-Handbuch der Raumautomation, AZ Verlag1. Auflage, ISBN 3-905214-33-4

LonWorks-Installationshandbuch, LON Nutzer Organisation e.V., ISBN 3-8007-2575-4

LonWorks Gewerkübergreifende Systeme, LON Nutzer Organisation e.V., ISBN 3-8007-2669-6