

Mini-Web-Server

fernsteuern per Internet und TCP/IP

Von Dipl.-Ing. Peter Stuhlmüller

Was im Großen schon länger möglich ist, lässt sich nun auch im Kleinen realisieren: Die Mikrocontroller sind Internet-fähig geworden, denn sie "sprechen" inzwischen TCP/IP.

Der Autor beschreibt hier zunächst einige wichtige Grundlagen der Internet-Kommunikation, die für diesen Einsatzzweck relevant sind.

Embedded Internet

Mit diesem Beitrag startet Elektor einen 3-Teiler, der über die Grundlagen der Internetkommunikation in dieser und die Programmierung eines TCP/IP-Stacks in der nächsten Ausgabe zu einem ersten „Elektor-Internet-Projekt“ im Halbleiterheft (Doppelheft Juli/August) führt. Bei diesem Elektor-Bauprojekt handelt es sich um einen universellen **Mini-WEB-Server** (WEB-IO) mit folgenden bemerkenswerten Eigenschaften:

- 80186-Prozessorkern
- Integriertes Ethernet-Interface
- 2 serielle Ports
- I²C-Interface
- 16 digitale Eingänge
- 16 digitale Ausgänge
- LC-Display
- Erweiterungsanschluss für eigene Hardware-Entwicklungen
- DOS-alike Multitasking-Betriebssystem
- TCP/IP-Stack mit PPP/HTTP/FTP/SMTP/telnet
- zahlreiche kostenlose Beispielprogramme im Internet
- Einfach zu programmieren mit verschiedenen DOS-Compilern

Lassen Sie sich überraschen!

Das Steuern und Überwachen komplexer Systeme und Prozesse an zentralen Schaltwarten ist seit Jahrzehnten gängige und bewährte Praxis. Dass dabei die räumlichen Entfernungen immer stärker in den Hintergrund treten, kann im Zeitalter der globalen

Hochgeschwindigkeits-Vernetzung nur als folgerichtig bezeichnet werden. Die allumfassende Daten-Vernetzung auf lokaler und globaler Ebene gehört heute zur Infrastruktur jeder modernen Industriegesell-

schaft. Schon lange sind Reisebüros mit Fluggesellschaften, Konzertkassen mit Veranstaltern und Auskunftssysteme mit ihren Betreibern über Datenleitungen verbunden, ohne dass dies nach außen immer in Erscheinung tritt. Natürlich werden auch Nachrichten-Satelliten und terrestrische Sender aus der Ferne gesteuert, und manche Industrie- oder Forschungs-Anlage wäre wegen ihrer für den Menschen gefährlichen Emissionen ohne Fernsteuerung und -wartung nicht denkbar. Die Liste der Systeme, die auf den Datenaustausch über größere Entfernungen angewiesen sind, ließe sich nahezu beliebig fortsetzen. Die Methoden und Verfahren der Datenkommunikation sind in der Regel an ihren Einsatzzweck angepasst, für die Realisierung der Übertragungswege ist die Übertragungssicherheit häufig ein wichtiger Aspekt.

Der Gedanke, dass sich das weltumspannende Daten-Übertragungsmedium Internet auch für viele Aufgaben der Fernsteuerung und -überwachung eignet, ist durchaus nicht neu; daran wurde bereits seit den

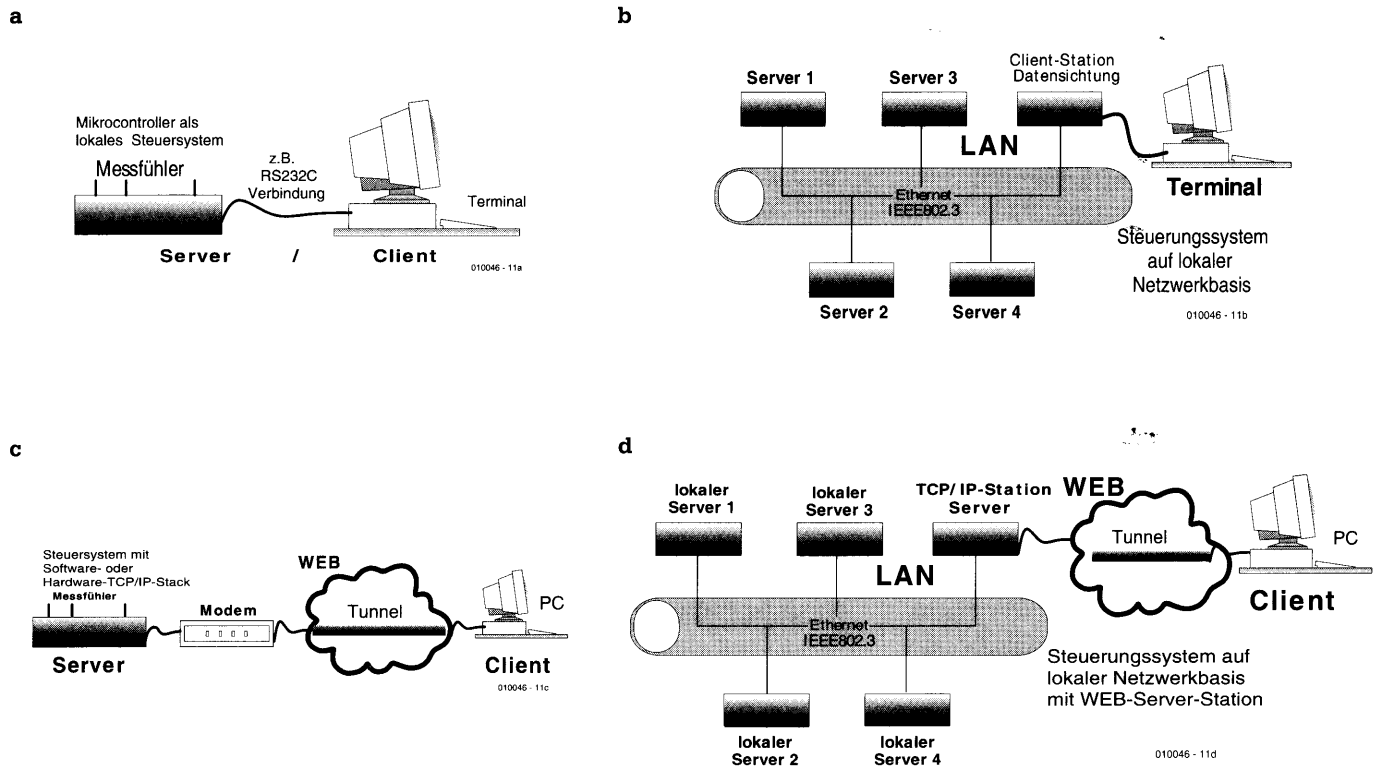


Bild 1. Client-Server-System als singuläres System (a), als Teil eines LAN (b), als WEB-fähiges Einzelsystem (c) und als Teil eines WEB-fähigen LAN (d).

ersten Anfängen gedacht. Die Systeme, die über das Internet miteinander kommunizieren sollen, müssen lediglich die Sprache des Internet sprechen: sie müssen einen "Stapel" (Stack) von Protokollen beherrschen, der unter dem Namen "TCP/IP" zusammen gefasst ist. Die Teil-Aspekte von TCP/IP, die hier von Bedeutung sind, sollen im Folgenden etwas näher betrachtet werden.

Client-Server-Modell

Ein Beispiel aus der Praxis macht schnell deutlich, was unter diesem aus der Netzwerk-Technik stammenden Begriff zu verstehen ist. Nehmen wir an, dass ein Messsystem von einem mehr oder weniger entfernten Ort bedient werden muss und die Messwerte (= Daten) an dem entfernten Ort ausgewertet werden sollen. Hier ist das Messsystem der Server (= Daten-Lieferant) und die Bedien- und Auswerte-Einrichtung der Client (= Daten-Abnehmer). In **Bild 1** sind vier verschiedene Fälle von Client-Server-Konfigurationen skizziert. Bild 1a stellt die einfachste Form dar, die eigentlich keiner wei-

teren Erläuterung bedarf. In Bild 1b sind bereits mehrere Server über einen lokalen Verbund (LAN) untereinander vernetzt; gemeinsam sind sie mit der Steuer- und Auswertezentrale (mit dem Client) verbunden. Die Bilder 1c und 1d entsprechen den Konfigurationen in Bild 1a bzw. 1b mit dem entscheidenden Unterschied, dass hier die Übertragungsstrecken in Form von "Tunneln" durch das Internet führen.

Ein Messdaten-System, das über das Internet kommuniziert und bei dem die Internet-Fähigkeit bereits in dem System selbst steckt (im System "embedded" ist), hat folgende Vorteile:

- Der Zugriff auf das System ist über das WEB von jedem Ort der Erde (und sogar aus dem Weltraum) möglich,
- das System kann ohne Rücksicht auf Ort und Zeit von Spezialisten überwacht und gesteuert werden, die sich ausschließlich auf diese Aufgabe konzentrieren müssen,
- Sicherungs-Mechanismen wie Daten-Verschlüsselungen und Passwörter verhindern unbefugte Eingriffe und missbräuchliche Ver-

wertung der Daten,

- auf der Ebene der Bedien-Oberfläche verhält sich das System so, als ob es vollständig vor Ort installiert wäre.

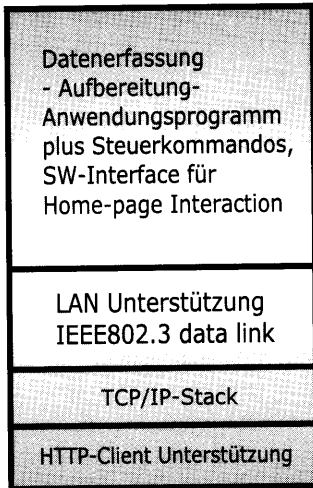
Natürlich stehen dem auch einige Nachteile gegenüber:

- Der Software-Aufwand für das System ist wesentlich höher als bei der System-Installation vor Ort,
- die Realisierung der Internet-Fähigkeit ist in der Regel nur mit leistungsfähigerer Hardware möglich, was die Kosten nicht unwesentlich erhöht,
- der Datentransport über das Internet verursacht laufende Provider-Kosten, die bei der Installation vor Ort nicht anfallen.

Wege durchs Internet

Um Messdaten oder andere beliebige Daten über das Internet transportieren zu können, müssen die Daten in eine dafür geeignete "Hülle" verpackt werden. Das ist schon allein deshalb notwendig, weil die Transportwege im Internet vorwiegend aus ökonomischen Gründen von vielen Servern und Clients *gemeinsam* benutzt werden. Die Daten gelangen deshalb nicht als Ganzes, sondern in kleinen Teil-Paketen vom Server über mehr oder weniger viele Zwischenstationen zum

OS Aufbau f. WEB-Server
* >> relative Verteilung



010046 - 12

Bild 2. Software-Module eines Mini-WEB-Servers.

Client. Das Vorschriften-Werk, nach der die Nutzdaten in Teilmengen zerlegt und verpackt werden müssen, heißt TCP/IP, oder ausgeschrieben: Transport Control Protocol / Internet Protocol.

In Bild 2 ist schematisch dargestellt, wie die einzelnen Aufgaben bei einem Client-Server-Messwert-Erfassungssystem verteilt sind, das über das Internet kommuniziert. Außer der Hard- und Software für die Gewinnung der Messdaten und deren Aufbereitung ist eine LAN-Unterstützung notwendig, die hauptsächlich die lokale Vernetzung der Messeinrichtungen koordiniert, eine TCP/IP-

Unterstützung, die den Transport über das Internet möglich macht, sowie eine HTTP-Unterstützung, die weitere notwendige oder wünschenswerte Funktionen übernimmt. Da die Messdaten in Pakete zerstückelt zusammen mit vielen anderen fremden Paketen auf dem Daten-Highway unterwegs sind, muss jedes Paket durch eine eindeutige und nur ein einziges Mal vorkommende Empfänger-Adresse gekennzeichnet sein. Die Adressen-Behandlung und -Verarbeitung bei Client und Server ist eine zentrale Aufgabe, die von jedem über das Internet kommunizierenden System beherrscht werden muss.

Internet-Adressen

Jede Internet-Adresse besteht aus vier Ziffernblöcken, in denen die Zahlen 0 bis 255 vorkommen können, was einer Wort-Breite von 32 bit entspricht. Die Adresse setzt sich aus zwei Teilen zusammen: der Netzwerk-Adresse, die für alle am gleichen Netzwerk angeschlossenen Stationen gleich ist, und aus der individuellen Host-Adresse. Die Adressen sind in die fünf Klassen A bis E eingeteilt, bei denen die Aufteilung der verfügbaren 32 bit auf Netzwerk- und Host-Adresse unterschiedlich ist. Tabelle 1 gibt hierüber einen Überblick.

Für lokale Netze, die ständig mit

dem Internet verbunden sind, muss die Adresse beim InterNIC (Internet Network Information Center) beantragt werden. Anderen Internet-Nutzern, die nur temporär über das Internet kommunizieren (z. B. über Modem-Verbindungen), kann eine Adresse dynamisch, also nur für die Dauer der Verbindung zugewiesen werden.

TCP/IP-Service-Ebenen

Die Gestaltung der Software-Struktur eines Client-Server-WEB-Modells läuft auf die Gliederung in so genannte Service-Ebenen hinaus. Das Referenz-Modell, für das man den Begriff "Open System Interconnection" (OSI) gefunden hat, ist eine Abbildung der Software-Module, die ein Client-Server-WEB-Modell unbedingt enthalten muss. Im Fall eines Mini-WEB-Servers spiegelt das Referenz-Modell alle Aktivitäten wider, die der Mikrocontroller zu leisten hat, um die Nutzdaten in Internet-gerechter Form über ein Modem oder eine andere Daten-Verbindung ins Internet zu geben. In Bild 3 ist diese Struktur im Detail dargestellt. Die für den Transport über das Internet wichtigste Funktion hat das TCP/IP-Modul. Es fügt den zu sendenden Daten-Paketen außer der Empfänger-Adresse insgesamt 26 Werte hinzu, die in vorgeschriebener Reihenfolge diverse Versendungs-Merkmale repräsentieren. Die Struktur in Bild 3 ist vertikal orientiert, sie verläuft von oben nach unten.

Ein weiteres wichtiges Detail des Client-Server-WEB-Modells ist das Kommunikations-Management, das zum Beispiel regeln muss, ob der Client oder der Server die Kommunikation einleitet. Normalerweise fällt die Rolle der "rufenden" Seite dem Client zu, aus bestimmten Gründen (z. B. Schutz vor Manipulation), kann die Rollenverteilung aber auch umgekehrt werden. In der Software muss dies natürlich implementiert sein.

Hinein und durch das Netz

Die am Internet beteiligten Computer und ihre Betriebssysteme gehören den unterschiedlichsten Welten an: Außer DOS- und Win-

OSI reference WEB-Server

OSI reference WEB-Client

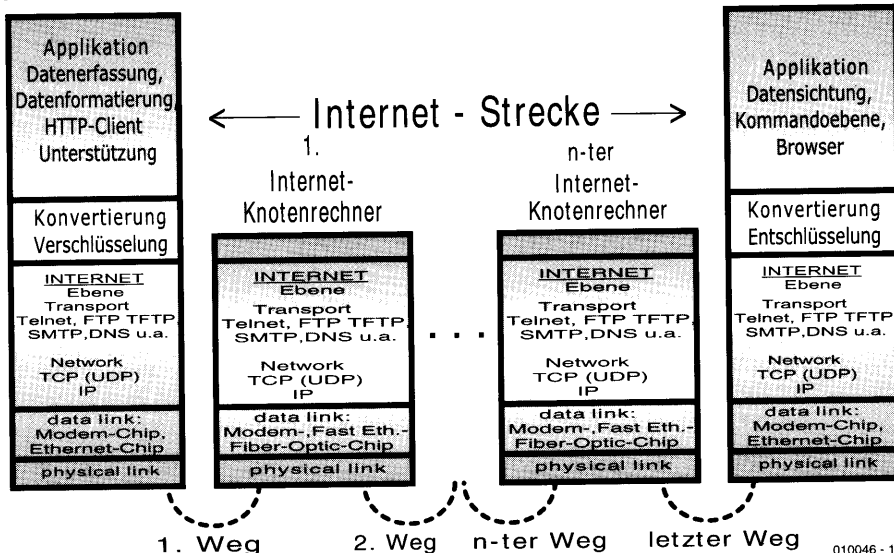


Bild 3. OSI-Referenz-Modell für WEB-basierte Steuer-Systeme.

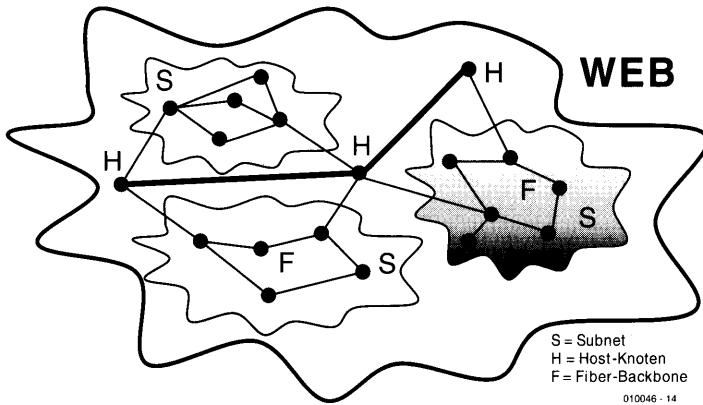


Bild 4. Schematische Darstellung des Daten-Transfers im Internet.

Tabelle I. Einteilung der Internet-Adressen in die Gruppen A bis E

Klasse	Netzwerke	Hosts	Adressenbereich
A	1 – 126	16 777 214	0.0.0.1 bis 127.255.255.255
B	128 – 191	65 534	128.0.0.0 bis 191.255.255.255
C	192 – 223	254	192.0.0.0 bis 223.255.255.255
D	224 – 239		Reserviert für Multicast-Adressen
E	240 – 254		Reserviert, zur Zeit unbenutzt

dows-Rechnern, UNIX-, SUN-OS- und AS400-Systemen, um nur einige zu nennen, sind Großrechner-Systeme vertreten, deren Namen manchmal nur Spezialisten kennen. Unabhängig von der System-Art und -Größe sind jedoch viele Prozeduren

für alle gleich. Ein Mikrocontroller-basierter Mini-WEB-Server mit eigenem Betriebssystem findet in genau gleicher Weise Anschluss an das Netz wie ein Großrechner, sofern die *Eingangsbedingungen* gleich sind. Beide müssen die gleiche Anmelde-

Prozedur durchlaufen, wenn sie zum Beispiel über den S0-Bus eines ISDN-Anschlusses mit der Daten-Außenwelt verbunden sind. Die Zugangs-Prozedur über den ISDN-S0-Bus ist standardisiert:

- Verbindungsaufbau zum Knoten-Rechner des Providers mit dem Netzwerk-Protokoll Telnet. Diese Verbindung hat zunächst den Charakter einer Terminal-Verbindung.
- Auf der Basis des Plattform-unabhängigen Protokolls FTP (File Transfer Protocol) wird das Daten-Paket zum Knoten-Rechner gesendet. Die Verbindung ist eine Punkt-zu-Punkt-Verbindung.
- Von diesem Zeitpunkt an ist TCP/IP oder genauer gesagt, IP an der Daten-Kommunikation beteiligt. Der Knoten-Rechner des Providers wertet den Datei-Kopf aus und legt den Weg zum nächsten Knoten-Rechner fest. Dieses "Routing" übernimmt eine Software, die mit so genannten Routing-Tabellen (routing tables) arbeitet. Die Routing-Tabellen können sowohl manuell erstellt worden sein (statisches Routing) als auch laufend automatisch an die aktuellen Gegebenheiten angepasst werden (dynamisches Routing). Außerdem ist auch ein Routing möglich, das ausschließlich auf Hardware basiert. Für den Datenpaket-Transport von Knoten-Rechner zu Knoten-Rechner wird die Methode des "IP-Tunneling" angewendet.

Spätestens nach der Übergabe des Daten-Pakets an den ersten Knoten-Rechner wird eine Rückmeldung an den sendenden Rechner gegeben. Die Abläufe im Knoten-Rechner sind ziemlich komplex, deshalb an dieser Stelle nur das Wichtigste: Für jeden möglichen Weg zum Empfänger wird eine maximale Datenpaket-Länge ermittelt, sie ist Bestandteil der Eingangsbedingungen für die betreffende Teilstrecke im Netz. Jede Strecke kann nämlich nur eine bestimmte maximale Datenpaket-Länge übertragen, so dass das Datenpaket, wenn es zu lang ist, in Teilpakete zerlegt werden muss. Da die Übertragungs-Geschwindigkeiten der benutzten Wege im Netz sehr unterschiedlich sein können, ist auch die richtige Reihenfolge auf der Empfänger-Seite nicht gewährleistet. Es ist Aufgabe der Empfänger-Software, die Original-Datei aus den Teilpaketen zu rekonstruieren. Zur Veranschaulichung ist in **Bild 4** ein Ausschnitt aus dem Internet skizziert.

TCP/IP-Header

Aus den **Tabellen 2a und 2b** geht hervor, welche Parameter im TCP-Teil und IP-Teil des Datei-Kopfes enthalten sind. Sämtliche Kopf-Wörter haben eine Breite von 32 bit; sie sind im Datei-Kopf so angeordnet, dass sie mit

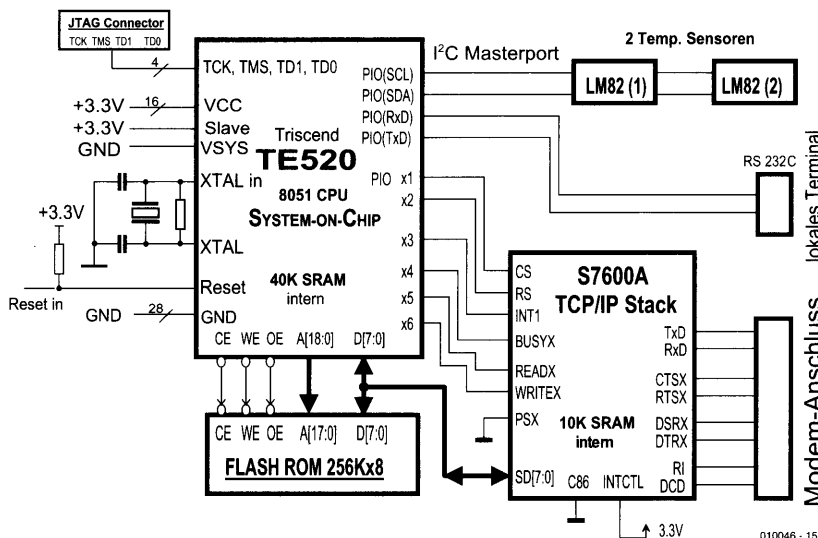


Bild 5. Basis-Schaltung eines singulären Mini-WEB-Servers mit dem S7600A von Seiko als TCP/IP-Hardware-Stack.

Tabelle 2a. TCP-Einträge

Feld-Bezeichnung	Länge (bit)	Bedeutung
Source Port	16	Absender-Adresse des Prozesses/Dienst der Daten eines höheren Protokolls. Mit IP-Adresse Zeiger auf Prozedur "Sockets". Daten-Paket an Telnet-Port des Empfängers. Telnet-Port Kommunikation Sender/Empfänger ist n i c h t Bestandteil der Sender-IP-Adresse.
Destination Port	16	"Socket"-Prozess/Dienst der Daten des Empfänger-Rechners. Die "Sockets" bleiben auf der Sende- und Empfangsseite so lange aktiv, wie die Verbindung besteht.
Sequence Number	32	Datenfluss; Sequenz-Nummer für erstes Byte im Daten-Paket.
Acknowledge Number	32	Datenfluss; Bestätigung des Empfangs aller gesendeten Bytes.
Data-Offset	5 - 15 norm. 5	Anzahl der 32-bit-Worte im TCP, ab diesem Wort inklusive des Datenteils (immer Vielfaches von 32 bit). Reserved 6 Für zukünftige Anwendungen.
Condition-Control-Flags	6	URG, Urgent-Zeiger; ACK, die Acknowledge-Nr. bedeutet wichtig/unwichtig (1/0); PSH, Übergabe an höheres Protokoll ja/nein; RST, bestehende Verbindung beenden ja/nein; SYN, an Empfänger: Verbindungsaufbau ja/nein; FIN, Sender zeigt Ende der Verbindung an.
Windows-Limit/Size	16	Flusskontrolle zwischen Sender und Empfänger des Datenblocks im ursprünglichen Datagramm.
Checksum	16	Prüfsumme aus 96-bit-Pseudo-Kopf (32+32+8+8+16 bit) IPSA + IPDA + Leer-Protoident + TCP-Segment.
Urgent-Pointer *	16	Positiver Offset der Sequence Number.
Service-Options **	Var.	Art, Länge und Optionsdaten im Datagramm.
Padding	8	Füll-Infos zur Sicherstellung für 32-bit-Format des Headers.
Data Set	32	

* Zeigt an, dass TCP-Daten hohe Dringlichkeit haben. Der Zeiger zeigt immer auf das Ende voran gehender Daten. Durch Addition des Zeiger-Werts mit der Sequenz-Nummer wird die Sequenz-Nummer des letzten dringenden TCP-Datenteils ermittelt. Daten dieser Art sind Interrupts oder Breaks, die von einer Gegenstelle übertragen werden.

** Feldlänge, abhängig von Art, Länge und speziellen Daten. Jede TCP-Implementierung muss sämtliche definierten TCP-Optionen unterstützen.

Tabelle 2b. IP-Einträge

Feld-Bezeichnung	Länge (bit)	Bedeutung
Version	4	Version des IP Headers, aktuell: 4.
IHL	4	Internet Header Length.
Service Type	8	Dienste IP-Datagramm, Verzögerung Durchsatz, Zuverlässigkeit.
Total Length	16	Länge Datagramm, IP-Header + Datenteil.
Identification	16	Kennwert Zuordnung der Fragmente im Datagramm.
Flags	3	DT: Fragment, MF: last Fragment / no Fragment. Fragment Offset 13 Lage der Fragment-Daten relativ zum Anfang des Datenblocks im ursprünglichen Datagramm.
Time to Live	16	Verbleibende Lebensdauer des Datagramms.
Protocol *	16	ICMP, IGMP, TCP, EGP, UDP, XNS-IDP, ISO-IP, OSPFIGP.
IP Header Checksum	16	Prüfsumme für den IP-Header (Einer-Komplement).
IP Source Address	32	Quell-Adresse des Servers (Sender).
IP Destination Address	32	Ziel-Adresse des Client (Empfänger).
Options **	7 min.	Klassen: Kontrolle und Debugging, IT-Time-Stamp.
Padding	8 - 24	Füll-Infos zur Sicherstellung für 32-bit-Format des Headers.

* ISO-IP (ISO Internet-Protocol), UDP (User Datagram Protocol).

** Kopier-Flag bei Fragmentierungs-Prozess.

möglichst hoher Geschwindigkeit ausgewertet werden können. Der Programmierer eines Client-Server-WEB-Systems muss im Dateikopf unbedingt die Werte einsetzen, die für

das von ihm entwickelte System zutreffen! Wenn das Client-Server-WEB-System Bestandteil eines lokalen Netzes (LAN) ist, braucht nor-

malerweise nur die Station TCP/IP auszuführen, die die Verbindung zum Internet aufbaut. Für den unmittelbaren Zugriff aller Stationen

auf das Internet wäre der Software-Aufwand beträchtlich.

TCP/IP als Chip

Was alles für einen typischen Mini-WEB-Server notwendig ist, ging bereits aus Bild 2 hervor. Bei einem singulären Mini-WEB-Server können die notwendigen Funktionen grundsätzlich mit einem geeigneten Mikrocontroller und der dazugehörigen Software realisiert werden. Eine interessante Alternative zu dieser klassischen Lösung bietet Seiko Instruments mit einem IC an, der das Programmieren des TCP/IP-Stacks überflüssig macht: Im S7600A ist der TCP/IP-Stack bereits gebrauchsfertig in Hardware "gegossen". Seiko Instruments hat hier die aktuelle Version 4 der TCP/IP-Protokoll-Suite mit UDP (User Datagram Protocol) und PPP (Point to

Point Protocol) in Chip-Form realisiert. Einschränkend ist dazu anzumerken, dass UDP etwas einfacher und nicht ganz so zuverlässig wie TCP ist, die Arbeitsgeschwindigkeit von UDP ist jedoch höher.

Im SA7600A ist eine serielle Schnittstelle (UART) integriert, an die ein Modem über einen RS232C-Leitungstreiber angeschlossen werden kann. Der 8-bit-Parallelport des SA7600A ist zu den gängigen Mikrocontrollern kompatibel, gegebenenfalls lässt er sich leicht anpassen.

Bild 5 zeigt die Prinzip-Schaltung eines mit dem SA7600A aufgebauten Mini-WEB-Servers. Der TE520 ist ein konfigurierbarer Mikrocontroller mit 8051-Kern, 40 KB SRAM und 2048 frei programmierbaren Logik-Zellen. An den I²C-Bus sind als Beispiel zwei Temperatur-Sensoren vom Typ LM85 angeschlossen. Was sie messen, kann per Internet über Kontinente

hinweg abgefragt werden!

Eine weitere Lösung basiert auf dem bekannten Mikroprozessor Z80 von Zilog, der inzwischen mit Taktfrequenzen bis 80 MHz arbeitet, Speicher und Peripherie bis 16 MB adressieren kann und auch DSP-Funktionen auf dem Chip bietet. Zusammen mit der dazugehörigen "Embedded WEB-Server Software Suite" und dem Ein-Chip-Modem Z02915 erhält man einen Mini-WEB-Server, der Erstaunliches leistet.

Von den weiteren aktuellen Hardware-Lösungen soll noch SA1110 von Intel erwähnt werden, der mit reichhaltiger Peripherie auf dem Chip zusammen mit einem V.90-Modem-Chipset eine leistungsstarke Alternative darstellt.

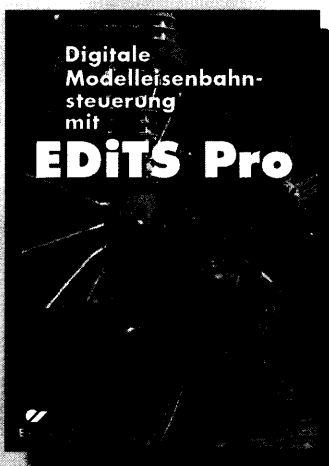
Im nächsten Teil dieses Beitrags liegt der Schwerpunkt auf der Programmierung eines typischen Mini-WEB-Servers, und im dritten Teil wird dann der Mini-WEB-Server als neues Elektor-Bauprojekt vorgestellt.

(010046-1)gd

Anzeige

Anzeige

NEU!



302 Seiten
Format 17,5 x 24 cm
Hardcover
ISBN 3-89576-107-9
DM 59,00 sFr 53,50 öS 431

Das EDiTS-System hat einen Nachfolger: **EDiTS Pro**

Steffen van de Vries

Die Bahn im Miniaturformat ist für Millionen Menschen faszinierend. Der Selbstbau eines elektronischen Modellbahn-Steuersystems mit der dazu gehörenden Hard- und Software ist ein erreichbares und sicher auch lohnendes Ziel. Das Buch zeigt, wie man diesem Ziel näher kommt: EDiTS Pro wurde für den modellbahn-begeisterten Hobby-Elektroniker maßgeschneidert, Schaltungen und Software haben ihre Bewährungsprobe in der Praxis bestanden, und Vieles ist so universell konzipiert, dass es auch unabhängig von EDiTS Pro Verwendung finden kann.

Kompatibilität war bei der Entwicklung von EDiTS Pro oberstes Gebot. Neben den EDiTS-Weichen- und Signaldecodern, den Erkennungsmodulen und dem Booster arbeitet EDiTS Pro auch gut mit den meisten Märklin-Modulen zusammen. EDiTS Pro unterstützt sowohl das alte als auch das neue Motorola-Format. Die Steuerung der modernen 4-Funktionen-Lokdecoder ist somit problemlos möglich, und zusammen mit der benutzerfreundlichen Controllersoftware lassen sie (fast) naturgetreuen Bahnverkehr zu. Eine Demoversion ist auf der zum Buch gehörenden CD-ROM enthalten

Jetzt direkt beim Verlag ordern; per Internet oder mit der Bestellkarte am Heftende

ELEKTOR ELEKTRONIK & COMPUTERTECHNIK

Elektor-Verlag GmbH
Tel.: 02 41/8 89 09-66
Internet: www.elektor.de

Süsterfeldstraße 25 52072 Aachen
Fax: 02 41/8 89 09-77
e-Mail: vertrieb@elektor.de