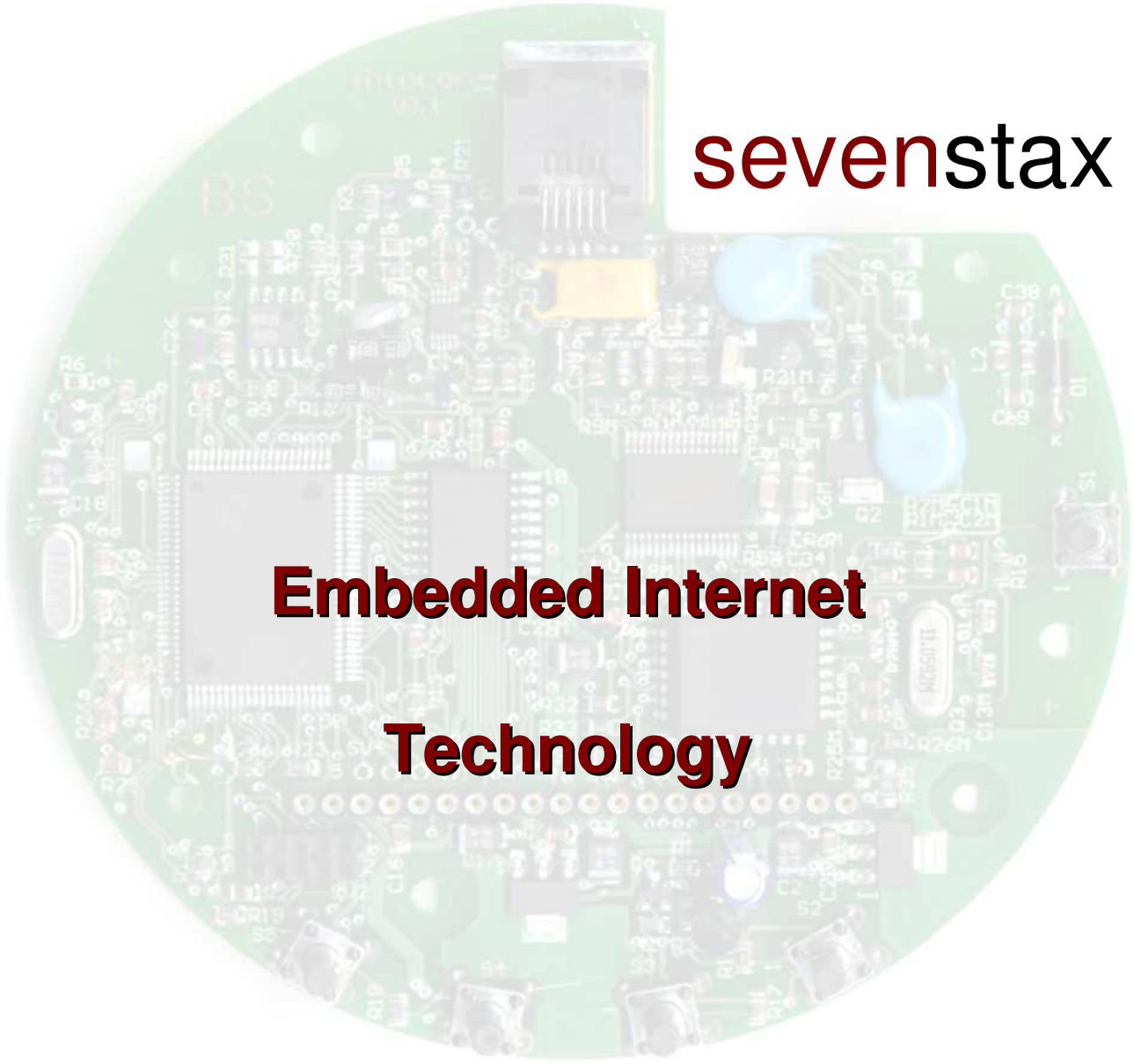


sevenstax

Embedded Internet Technology

sevenstax GmbH
info@sevenstax.de
www.sevenstax.de

Revision 4.3

Embedded Internet

Content

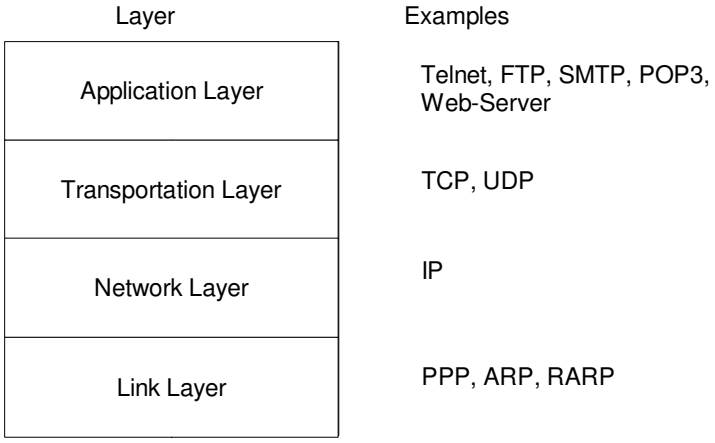
The networking of different integrated systems is an important request for new device generations currently both for new developed devices as well as for new releases of existing devices. The networking is standard for PCs and similar powerful architectures currently but the integration of embedded systems is not common.
The extension of embedded systems with network components seems often as too expensive , too complex or the requirements on resources like memory and CPU seems to be too high. The next pages show solutions how to connect embedded systems reliable to the Internet with reasonable micro controllers and low development effort.

Internet Communication

The Internet is an infrastructure to transfer data world wide. The most important advantage is the economical use for everybody. An user pays only low mostly time dependent rates, whereas the location of the sender and receiver are irrelevant. If a device is connected to the Internet a data communication is possible all over the world. The connection to the Internet itself is possible in different ways, e.g. phone connections or networks like the intranet are common. Today wireless connections like GPRS and for the future UMTS are getting more and more popular. All these methods are used to connect different devices together and to transfer data between devices and servers.

Base Protocols

The most known protocol for Internet communication is the TCP/IP protocol. Often the term TCP/IP is used for all methods and protocols integrated in a TCP/IP stack. In principle TCP/IP stacks are subdivided in different layers for different parts of the communication. The elements of these layers can be combined according to the special purpose. Following the OSI reference model 4 out of 7 possible layers have to be implemented in a TCP/IP stack.



The layer on top is the application layer. All programs responsible for data transfer and the responding protocols are situated in the application layer. Typical standard applications are telnet, the file transfers (via FTP protocol), email programs (SMTP and POP3 protocol) and web servers (http protocol). Amongst these applications and protocols user interfaces are included in this layer too like programs to configure devices or programs to transfer readings to the server.

The transportation layer consist of two protocols - TCP and UDP. Both provide a special communication method for the application layer. TCP provides a secure packet oriented data transfer including checksums and sequence numbers to assure the reliability of data transfer. TCP is the most common used protocol because it ensures an integrity of data by itself. UDP does not offer this security but the speed of data processing is higher. Due to the insecurity of data transfers UDP is mostly used for server communications with low data volume. This server communication can be lost without any impact on the actual data transfer. One example for a

typical UDP communication is the query on domain names with the help of the domain name service (DNS). If UDP is used as transfer protocol the application layer must supervise the transfer. Otherwise the transmitter can not assure a successful data delivering at the receiver.

The network layer contains the Internet protocol. Its principle task is to address and deliver/accept the data. Therefore every device connected to the Internet in order to send and/or receive data has an own IP address(e.g. 192.168.130.0). This address is a fixed number for one device or it is given by an Internet service provider as soon as a device is connected.

The link layer realizes the connection to the transmission medium. In this layer protocols are implemented to adapt TCP/IP data to the transfer medium like a modem or an Ethernet network.

TCP/IP

The transmission control protocol and the Internet protocol are the most common used protocols for Internet communication. They take care for a propper data transfer between two communication partners. Yet the data transfer is independent from the kind of data because the data will be interpreted by the application layer. There are some standard application protocols like HTTP, SMTP or FTP which will be described below. Sometimes it might be useful to have an own protocol, e.g. to connect devices with an already existing propriety protocol or to connect devices transferring only low data volume.

The big advantage of TCP/IP is the high distribution. It is used as standard for data transfer between different computers for many years mostly to transfer data between powerful machines like PCs and work stations. Special TCP/IP stack implementations for microcontrollers are required for the use in embedded systems due to the limited resources to run protocols on such systems. And there are even more requirements for such implementations: They should be integrated in existing software systems easily, they should be compatible with existing stack implementations on other operating systems and the implementations should achieve an appropriate data volume on small controllers with low RAM, ROM and CPU load.

PPP / Ethernet

Below the TCP/IP layer the link layer is situated. This layer adapts the given data to the transmission medium. In practice two different mediums are used mainly, either an Ethernet network or a serial connection via modem. Depending on the used transmission medium different protocols are used in the link layer. For a serial data connection the point to point protocol (PPP) is used. Its tasks are the encapsulation of data before transmitting and the control and configuration of connections. In addition the PPP gives different possibilities for authentication at the counterpart which is used by the most Internet providers like PAP or CHAP. With an Ethernet network the data transfer is done by a separate chip. This chip divides TCP/IP data in separate Ethernet packets, addresses them and handles collisions of packets on the transport medium. For Ethernet connections a different kind of address (MAC address) is used as for TCP/IP (IP address). The applications initiating a data transfer know the IP address of their counterpart only but not the required MAC address. Therefore two additional protocols are necessary - one to transform IP addresses in MAC addresses and the second vice versa. These protocols are the address resolution protocol (ARP) and the reverse address resolution protocol (RARP).

Application Protocols

The application protocols facilitate the use of standard applications which are common for Internet communications. To give some examples emails are sent and received using SMTP and POP3, files transferred via FTP and web pages are sent and received by the http protocol.
The application protocols use the transportation layer below to transfer the data. The protocols described below make use of TCP.

HTTP

The hyper text transfer protocol is mainly used for communications with web servers and Internet browsers. Internet browsers can request HTML-formatted pages from a server and can display them in order to send information or to offer input frames to an user. Therefore forms are used which allow the input of words or of special values which can be chosen. The browser sends the values/words back where they are analysed by the server. Typical for the use of http is a client -server-architecture where a client requests data from the server and sends answers back where appropriate. It should be taken into consideration that only a small data volume flows between client and server because the data are transferred in one block and the server has to analyze the data costly. HTTP is the best solution for embedded systems to configure the devices by a browser especially in combination with HTML. If configurations should be interchanged between devices a combination of HTTP and XML (or SOAP) would be provided.

SMTP / POP3

The mail transfer protocol (SMTP) and post office protocol version 3 (POP3) are used to send emails. SMTP is responsible for sending and POP3 is the receiving protocol. Besides the common email applications emails can be used for status messages from devices in particular if these messages do not have to be processed immediately.

XML / SOAP

The extensible markup language (XML) is a text based language similar to HTML. There are two important differences to HTML why XML can be used especially for machine-machine communication:

- Data content and representation are strictly separated in XML. So a formatted output of data is possible for man-machine-communications whereas the representation can be ignored for machine-machine-communications
- XML can be expanded, own tags and attributes can be chosen. Therefore an adaptation for special systems is easy.

Due to this two points XML based transfer standards are defined like the simple object access protocol (SOAP) mentioned above. This standard define a mechanism which allows all computers to transfer data by there own independent of different computer architectures. SOAP is supported by all important software companies like SUN, IBM and Microsoft.

FTP

The file transfer protocol (FTP) is used to transfer files via the Internet i.e. to copy files to other devices. Typical for the use of FTP is the availability of a file system to handle files and the use of two TCP connections simultaneously (one connection for controlling and one connection for the data transfer). Due to this characteristics FTP can be recommended for the use in embedded systems only limited because a file system and the additional memory requirements for a second TCP connection put a high pressure on the available resources. If bigger data volume should be transferred via embedded systems FTP is the standard solution.

Applications

An additional logic is necessary for most applications in addition to the described application protocols, e.g. to access a server the implementation of the email protocols SMTP and POP3 are sufficient, but for the presentation and the management of emails an additional software is necessary. It is similar for the combination of http and web servers. If a browser should be able to access an embedded system the management and the dynamical construction of HTML pages is required. This functionality is not part of the HTTP protocol but part of the web server application

E-Mail

Beside the standard protocols SMTP and POP3 there are some additional international standards (RFCs) for emails regarding the interpretation of the content. The standards are concerning the email header (e.g. the way the sender, the sending time, and the subject are included) and concerning the coding of special characters and different languages. The corresponding RFCs must be kept for writing and receiving emails in order to secure a faultless email presentation on any mail client. An email application has to handle more functionalities like a folder to list all received emails, to mark the emails as read or new, to save a copy of the email to a hard disk or to delete emails from the server. To define an email application own requirements have to be clarified. A mail client only for messaging about operating states of an embedded system has lower requirements concerning the complexity than an email program including a user interface for writing and managing emails. If the need of system resources is also under consideration it is obvious that no universal solution exists. An application can be developed under consideration of the requirements and the existing RFCs which fulfills the given task.
For the selection of the required functionality the following aspects should be considered:

- Is the sending and receiving of emails necessary?
- What is the use of the email (only messaging or writing too)
- Which languages will be used?
- Will binary data (like appendices) be used in addition to mail texts?
- How much system resources (RAM/ROM) are available?
- Which requirements are given by additional software?

On the base of these answers a requirement profile can be drawn up so that an evaluation of the software's complexity and the resulting solution is possible.

Embedded Web-Server

An embedded web server enables any client supporting a browser to access the system by using the HTTP protocol. The data's format is in general HTML which allows to present device states and data on nearly all computers connected to the Internet and an installed Internet Browser. It should also be taken into consideration that only authorized users should be allowed to access the device.
XML is growing used as new standardised data transfer format . The first browser generations using both XML and HTML are available. Currently there are differences in the presentation of graphical outputs from XML so that problems can occur when XML is used in the communication of man and machine. But XML offers many advantages in machine-machine communication like described above.

Embedded Client for Web Services

Web services are a new kind services offered by servers in the Internet. They are not limited to provide Internet pages like HTTP or to provide files like FTP but they can be adapted to special customer requirements, e.g. a server may offer a weather service. Any Client which is able to connect to this server may invoke queries and may use the results in any way it wants. It is for instance as well possible to represent the data graphically as to use them for further processing. This functionality is in general independent of the hardware platform but the integration of web services into embedded devices is quite new and offers a lot of new possibilities.

Utilization of Internet Protocols

There is a wide range in which every of the described protocols may be used. The examples described below are meant as solution to some typical problems.

Configuration of Single Devices

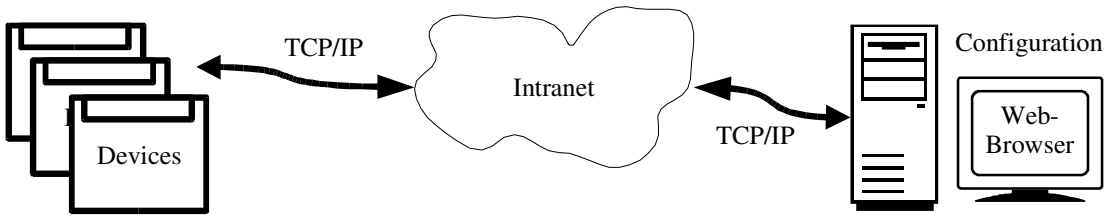
Single devices are easily configurable using a web-interface. To enable the remote control of a particular device an embedded web-server is additionally integrated into the existing software. The web-server creates a form, using common elements like edit fields and select boxes, which is displayed at a web-browser. Any user who has access to this system can now control or modify current settings of the device. As this example shows an online communication between a user and one machine takes place. Therefore this solution is mainly suitable for single devices.

Advantages

- Remote devices can be configured using a common web-browser (Internet Explorer, Netscape, Opera etc.) from any computer which supports TCP/IP and a web-browser
- No software has to be installed on these computers.
- There are no dependencies to the computer's operating system.
- Applications to configure the device can be easily written in HTML.
- Low hardware requirements for sevenstax software.

Disadvantages

- The user and the device have to be online i.e.
    1. a permanent connection exists e.g. over an Ethernet network or
    2. the PC establishes a connection to the device using a dial-in or
    3. a mechanism is integrated in the device to connect to the PC
- Web-Pages are unsuitable for real-time communication. This disadvantage may be eliminated by using a JAVA applet.



Realisation

The device's firmware is extended by the basic Internet protocols and a small embedded web-server. The information (e.g. status, settings, errors) is displayed in hierarchically structured HTML pages which are sent to the browser by the web-server. The access to the server can be password protected to exclude unauthorized users. Furthermore so called forms can be dynamically created to manipulate the device settings.

Typical Usage

printers, copiers, heating systems, telephone systems, home automation

Real time Communication of Devices

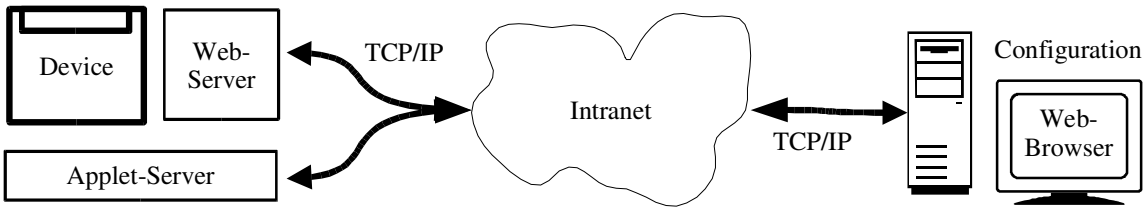
As described above a web-interface is very suitable for configuring single devices without real time demands on the communication. But if a real time communication is required this problem is quite unsolvable using HTML formatted pages. One good solution is to integrate JAVA applets in the HTML pages. The applets may be used for evaluating data collected by the device or even for their graphical representation. As the PC does the drawing of the data and the device only has to send the measured values even small devices can be enhanced to represent data graphically.

Advantages

- Real time representation of device data
- Evaluation of data at the PC which is more powerful than the device
- Larger flexibility and possible interaction with the user.
- Works on many different clients (e.g. PC, PDA, mobile)
- No additional software required at the device's side
- Applets are secure

Disadvantages

- The Java virtual machine must be installed at the browser
- The applets are loaded from the client.
- The creation of JAVA applets is more time consuming than writing HTML pages.



Realisation

The HTML page contains a link to the JAVA applet which is stored at the devices ROM in the same way as the web-pages. When the page is loaded by a browser it will be recognized and executed by the PC. The applet itself establishes a connection to the device which will start sending the required data to the PC where they will be interpreted. sevenstax offers all modules to realize a JAVA applet based communication.

Proprietary Application Protocols in IP-Networks

Even if the Internet is associated with well known application protocols like HTTP, FTP or POP3 also own application protocols may be used for data transfer. Neither TCP nor IP care about the data they are carrying - their only purpose is to guarantee the consistency of the data. By this way devices that already use communication protocols can be enhanced to be Internet compatible. The existing communication protocol is embedded in TCP packets and delivered to the receiver via the Internet. But some additional aspects of the Internet have to taken into consideration:

- The runtime of any TCP packet is unpredictable and depending of the network itself e.g. how many data is currently transferred or of the amount of network nodes a packet passes.
- Some packets may get lost or may be delivered out of order. Although TCP corrects these errors the retransmission of packets is possible and causes delays.
- There is hardware like some firewalls or routers which do not support the usage of “unknown” protocols.

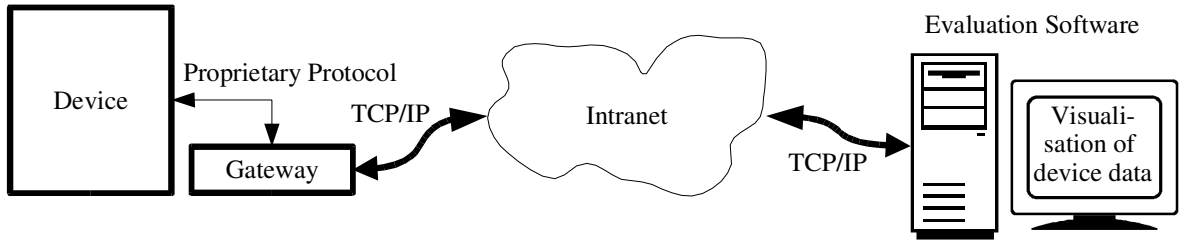
To integrate the Internet functionality in the device it is enhanced by the protocols and methods to access the network, for instance by a modem or an Ethernet network. At the receiver's side the evaluation software has be enhanced to use the PC's TCP/IP software stack.

Advantages

- The existing software at the client's and the PC's side can be used.

Disadvantages

- A proprietary software is always required at the PC's side
- Not suitable for all combinations of application protocols and network infrastructure



Protocol Data and Problem Reporting of Shared Systems

A cheap and reliable solution to control the state of embedded systems is the automatic and apparent dispatch of pre-formatted emails. An embedded email software is integrated into the device. Its purpose is e.g. to collect data for a predefined time and to send the information to a mail server.

The receiver may either be a machine or a human.

Advantages

- existing software (mail clients)
- in case the receiver is a machine the content can be evaluated easily by common scripting languages or interpreters like Perl or PHP
- gateways for voice output or SMS (Short Message Service) notification are available
- low cost as the device is only a short time online

Disadvantages

- no confirmation whether the email has been received within a certain time
- not suitable for large amounts of devices

Maintenance and Diagnosis of Device Pools

Many automations, for instance vending machines are arranged where they are accessible easily by many people.

A maintenance at regular intervals of these machine pools is a time consuming and expensive

task. If any problems occur even more costs arise e.g. The loss of gains. To reduce or avoid these problems a remote maintenance via the Internet is an effective solution.

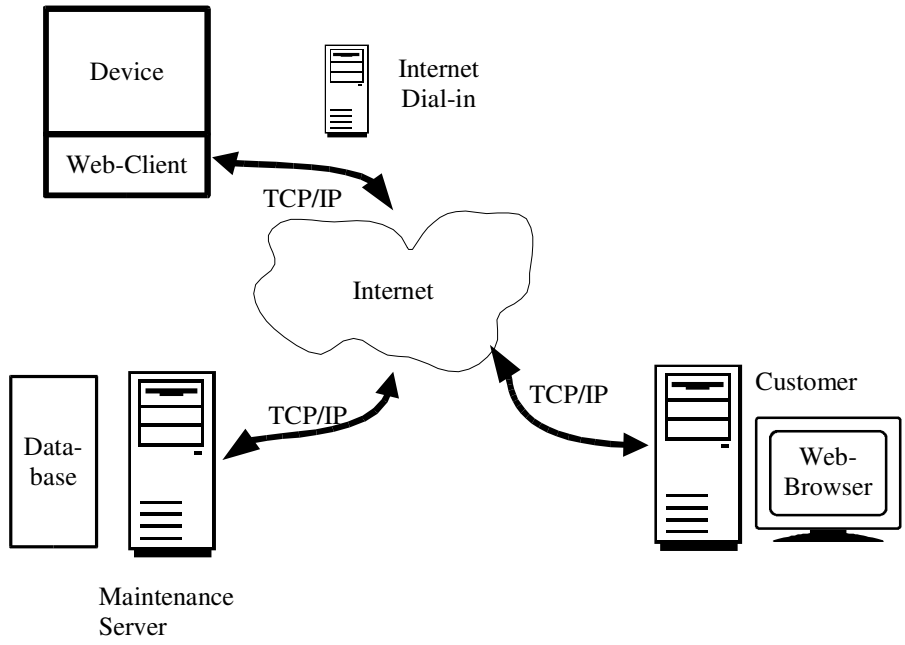
The device dials-in the Internet and reports errors or maintenance data to a dedicated maintenance server which forwards for example error messages and stores maintenance data for analyses. A customer may connect the maintenance server and gets the collected data of his devices. To assure no direct influence from any user to a device the user does not directly communicate with the device but with the maintenance server instead.

Advantages

- the administration of large amount of devices is simple
- a structured administration for instance grouped by device types or subgroups is possible
- access rights of user can be easily managed
- additional information are available for instance for statistical purposes
- smooth integration in existing infrastructure
- all accesses are can be journalized

Disadvantages

- an additional server is always required
- increased initial cost compared to the other solutions



Realisation

A XML/SOAP module is integrated in the device either by enhancing the existing software or by a separate communication hardware. This hardware contains the complete hard- and software to communicate with the device and the maintenance server. If wanted the device is enabled by the software to report relevant data immediately to the server e.g. in case an error has occurred. .

As two machines are exchanging information the SOAP protocol is used which combines:

- data is interpretable by machines in both directions
- common web-server may be used to avoid own developments
- XML parsers are available for most platforms
- no proprietary solutions as only standards are used

sevenstax Protocol - Implementations

The sevenstax Internet protocol stacks are specific implementations for the use in embedded systems with 8..32 bit microcontrollers and minimal resources. A high value was set to ease the integration of the communication stacks into existing software.
All sevenstax protocol implementations are free of blocking states to guarantee an undisturbed operation between your software and the communication enhancements.
The programmer's interface is small but supports all required actions to communicate with servers all over the world. The feedback informations of any protocol is reported to your software by callback-functions, called by the sevenstax modules to inform about the occurred events. These callback-routines integrate the protocols into the existing software without requiring costly modifications. The thin application programming interface (API) is of course also an advantage when new software is developed as ambiguity is obviated and the test of the software becomes less difficult.

As result of the modest needs of resources the enhancements of Internet communication is possible without changing the existing hardware, especially with regard to the microcontroller, the ROM and RAM.
The protocols are designed to work in systems with or without an operating system.

sevenstaxTCP/IP

Operating Environment

The sevenstax TCP/IP stack was conceived for units which require only small amounts of data (in an order of a few hundred kilobytes) such as the download of parameters, web pages, software, or configuration data via Internet, remote service and control.

Advantages

The sevenstax TCP/IP stack is completely new implemented with the result of a lean, fast and efficient software, containing all the functionality required for an efficient and problem-free operation.
In short, when unit cost is of direct influence and a high level of system security is required, that is when the advantages of sevenstax TCP Stack are most evident.

Characteristics

- Minimal development timeframe required. Only those functions necessary for operation are integrated
- Easy integration into the existing software with only a few existing interfaces
- High security through clearly defined functional structures
- Suitable for use in micro controllers from 8...32 Bits
- Asynchronous performance also without OS multithreading capability
- Operates with CMX-RTOS from CMX Systems and embOS from Segger Microcontroller Systeme
- Needs no send/receive buffer, using the capacity of the lower network protocol layers.
- Portability - ANSI C
- Tested in Germany, USA, Asia and South America

Optional (increases code size)

- UDP and ICMP (ping) support
- Server functionality
- Multi-channel connect

Memory Requirements

- ROM: **2.6...8.2 kbytes code** (16-bit CPU), depending on the chosen options
- RAM: **95...250 bytes** for variables
- 32-bit timer/counter with an resolution of 1ms
- Typical CPU load less than 5 % (16-bit CPU, 10 MHz, data transmission via modem, 14400 bd connect)

sevenstaxPPP

Operating Environment

The sevenstax PPP stack is used in units with serial data transfer i.e. via analogue modem, GSM, RS232, ISDN or DECT. sevenstax PPP was designed for running on units which are in use as clients on a network e.g. Web-Browsers or units which pull or send data autonomously. In short, when unit cost is of direct influence and a high level of system security is required, that is when the advantages of sevenstax PPP are most evident.

Characteristics

- Fast integration and short development timeframe as only functions necessary for operation need to be integrated and tested.
- The stack has only minimal interface requirements and can be easily integrated into existing software.
- minimal processor and memory requirements
- Send and receive buffers can be used for data processing by the upper protocol layers
- For use with micro-controllers from 8...32 bit
- Integrated sub-protocols LCP, PAP and IPCP
- Automatic authenticity check with PAP
- "On the fly" HDLC encoding/decoding
- Asynchronous performance without OS multithreading capability
- Portability - ANSI C
- Optimized for the connection to Internet Service Providers (ISPs)
- Tested in Germany, USA, Asia and South America

Optional

- Authentication with CHAP
- Supports ACCM (Asynchronous Control Character Map)
- Header compression
- Protocol compression

Memory requirements

- ROM: **3.5...6.0 kbytes** code (16-bit CPU), depending on the chosen options
- RAM: 600 bytes receive- and 600 bytes send-buffer. Both values are adjustable.
  160...280 bytes variables
- 32-bit timer/counter with a resolution of 1ms
- Typical CPU load less than 5 % (16-bit CPU, 10 MHz, data transmission via modem, 14400 baud connect)

sevenstax XML Parser/Generator

Operating Environment

The sevenstax XML-module enables embedded systems to make use of XML-based languages (e.g. WML and SOAP). The module consists of a parser to evaluate received XML data and a generator to create XML code. A high value was set to ease the integration of the communications stacks into existing software.

Characteristics

- Adaption to every XML based language possible by defining appropriate name spaces (tag-names, attribute-names, etc.)
- Streaming parser, i.e. received data is evaluated during receipt to minimize the required buffer
- Support of all XML entities
- Syntax check up to a user definable interlocking
- Integration of the parser into existing software by registering of Callback-routines which are called upon recognized events (e.g. after XML-tag) or errors

The parser requires the statement of supported tag- and attribute-names. These names and the XML name-spaces can be freely defined by the user.

Memory requirements

- ROM: **11.0 kbytes** code (16-bit CPU)
- RAM: **350 bytes** variables

sevenstax SOAP-Client

Operating Environment

The sevenstax SOAP-client was developed to make the data exchange between embedded systems and existing SOAP servers as easy as possible. The SOAP-client is based upon the sevenstax XML-Module, which was enhanced by the SOAP specific name-spaces (tags, attributes, etc.) and by the inquiry/response mechanisms. Furthermore the SOAP error handling has been implemented. Besides the XML-module the SOAP-client also supports the required HTTP methods to communicate with standardized web servers.
An application which integrates the sevenstax Soap-client does not have to know anything about SOAP mechanisms, the only thing to do is to add the application specific part of the XML requests and the appropriate handler functions that are called when the XML responses have been received.

Characteristics

- Enables embedded systems to use all web services which are working with SOAP technology
- Integration of the client module in existing software by registering of the SOAP parameters and the application specific XML properties (i.e. name spaces, structures of the request, handler functions for responses)
- Modular structure of the client. The XML, HTTP and TCP/IP functionality is separated in independent modules

Memory requirements

- ROM: **14 kbyte code** (16-Bit CPU)
- RAM: **0.7 kbyte** variables

sevenstax Web-Server

Operating Environment

The sevenstax Web-server enables embedded systems to be accessed from any system that supports a common web browser.
The web server has been designed to realize minimal processor and memory requirements. Nevertheless the restrictions resulting from this requirements do not affect the functionality of the server itself, e.g. the request of HTML formatted pages or HTML forms.
The server supports the HTTP GET- and POST methods and is capable to create HTML pages dynamically by using special HTML tags. To create a HTML form, the servers reads a template of this page from the ROM and enables the application to add or modify parameters of the web page. In case the application does not fill in these parameters the default values stored in the template are sent. Two typical examples for the use of the web server are the configuration of embedded systems via the Internet and the representation of measures values in a web browser.

Instead of exchanging HTML formatted pages the server is also capable of exchanging XML formatted data e.g. when using the SOAP protocol.

Characteristics

- For use with micro-controllers from 8...32 Bits
- Support of parallel connections at one time
- Capable to create pages dynamically (HTML/XML-forms, HTML/XML-pages)
- Authentication (Basic HTTP Authentication) to avoid unauthorized access to the embedded system
- Own send buffer (in case the TCP/IP stack has its own buffer this buffer can be used instead)
- All resources (pages etc.) may be stored in the ROM / flash-memory
- Independent of an operating system
- Does not require a file system
- No socket interface required to save memory (sockets optionally usable)
- Tested with the most common web browsers (Mozilla 1.x, Netscape 4.7, MSIE 5.x, MSIE 6, StarOffice 5, Opera 5.x, Opera 6.x)
- Portability - ANSI C

Memory requirements

- ROM: **1.4 ... 4.5 kbyte code** depending on chosen options
  465...690 bytes for constants
- RAM: **160...280 bytes**
- TCP/IP stack required

sevenstax Mail-Client

Operating Environment

The sevenstax Mail-Client enables embedded systems to send and receive emails by supporting all required protocols, e.g SMTP, POP3 and the MIME encoding/decoding of the content. The management of the mails is not part of the mail clients tasks but must be done by a separate email application. The client software uses six sub modules

- the POP3 generator to generate communication sequences depending on the requested action, e.g. receive one certain email
- the POP3 parser to evaluate the server's responses
- he SMTP generator to generate communication sequences depending on the requested action, e.g. send an email
- the SMTP parser to evaluate the server's responses
- the IMF/MIME generator to create a mail according to the Internet mail format and to encode the content
- the IMF/MIME parser to decode the received mail and to evaluate the mail header and content

Characteristics

The sevenstax embedded mail client enables embedded systems to create, send and receive emails according to the appropriate RFCs. These emails may be read by every mail client over the world. No knowledge of the protocols is required, the API (application programming interface) offers the embedded software all the interfaces to communicate with mail servers.

- Software can be easily configured for versions supporting the sending or receiving of mails only
- Streaming parser, i.e. received data is evaluated during receipt to minimize the required buffer
- Integration of client into existing software by registering handler functions that will be called upon occurred events e.g. after the recognition of a senders address
- Modular structure through using separate POP3-, SMTP, IMF/MIME and TCP/IP modules
- Integration into existing software by registering handler-functions

Memory Requirements

- ROM: **23.0 kbyte** (16-Bit CPU)
- RAM: **2.0 kbyte**

sevenstax – The Company

sevenstax GmbH is specialized in developing systems with high requirements regarding to operating security but reasonable hardware costs.
We integrate communication technology, especially the Internet, with all kind of systems and appliances. Together with our customers we create new applications – the maintenance or diagnosis of systems via the Internet as well as new generations of consumer products.

If you have systems which could be more effective when using network technology sevenstax has got the know-how of the data transfer. We will support you to manage your project efficiently and of prime quality.
Our strengths lie in the communications software for embedded systems. Commissioned by renowned clients to develop embedded systems for mass production, we deliver on-time and in-budget.

Our Team

Experience, creativity and versatility are our basics. The sevenstax team is working together for more than 5 years and has accumulated over 40 years of experience in the area of development of telecommunications, audio/video processing, measuring and monitor control technology. We have carried out and successfully completed tasks ranging from the integration of hardware related drivers, operating systems, protocol stacks and development of "Man Machine Interfaces" on embedded systems up to the interfacing of the units into networks.
The best given conditions are a goal-oriented and qualified team. With the knowledge of past experience, our team has built-up a high level of hardware and software system knowledge and our designers have the subject-specific know how from which you can benefit.

Quality management and the customer's satisfaction are our main goal. Customers who are not satisfied by erroneous products, overdrawn budgets and deadlines are as counterproductive as overburdened developers after having worked for 80 hours ore even more per week.
If you are looking for a service enterprise that delivers on-time and in-budget then sevenstax will be the perfect solution.

Our services

We offer all development phases from the conception through to production and when required, handle the total development module or in co-operation as a "Joint Team".

The development services can be calculated either at a fixed price or according to time and work involved. You have full control over all phases in the development process.

Contact

sevenstax GmbH
Hamburger Allee 43
30161 Hannover
Germany

Tel. +49(0)511/ 47386 04
Fax +49(0)721/ 151 505 622
Internet: www.sevenstax.de
E-Mail: info@sevenstax.de

Glossary

Term	*Explanation*
ARP, RARP	Address Resolution Protocol and Reverse Address Resolution Protocol: These protocols assign MAC-addresses as used in an Ethernet network to the corresponding IP-addresses and vice versa.
CHAP	Challenge Handshake Authentication Protocol: Protocol to authenticate an user at the ISP. CHAP is part of the PPP mechanism.
DECT	Digital Enhanced Cordless Telecommunication: Wireless communication standard which is used to transfer speech and data for a short distance.
DNS	Domain Name Service: DNS assigns a domain-name (z.B. www.google.de) to the corresponding IP-address (216.239.33.100) which is used in the Internet.
FTP	File Transfer Protocol: Protocol to transfer files via the Internet
GPRS	General Packet Radio Service: GPRS represents the first implementation of packet switching within the GSM, which is essentially a circuit-switched technology. Using GPRS will enable users to send and receive data at speeds up to 115kbit/s. GPRS is very efficient in its use of scarce spectrum resources.
HDLC	High-Level Data Link Control: Bit oriented, synchronous protocol used in the link layer of the OSI reference model. HDLC is normed by the ISO.
HTTP	Hyper Text Transfer Protocol: Protocol to transfer web pages
IPCP	IP Configuration Protocol: Sub-Protocol of PPP to configure the Internet protocol
LCP	Link Control Protocol: Sub-Protocol of PPP to configure the connection
PAP	Password Authentication Protocol: Protocol to authenticate an user at the ISP. CHAP is part of the PPP mechanism.
POP3	Post Office Protocol Version 3: Protocol to get emails from a mail server
PPP	Point To Point Protocol: Protocol to transfer data via a serial connection or a modem. The TCP/IP data will be embedded into special PPP frames, provided with a checksum and afterward transmitted. At the receiver the integrity of the frame is checked and the TCP/IP data extracted before being passed to the TCP/IP stack.
RFC	Request For Comment: Obliging standards of all Internet related topics to assure the interoperability of different implementations of protocols and stacks. There are e.g. a RFC for SMTP, a RFC for POP3 or the format of email.
SMTP	Simple Mail Transfer Protocol: Protocol to send emails

Term	*Explanation*
SOAP	Simple Object Access Protocol: SOAP offers a simple mechanism to exchange structured and typed data between computers in a distributed environment. Soap is based on XML. To transport the data HTTP is used generally.
TCP/IP	Transmission Control Protocol / Internet Protocol. The combination of these two protocols is mostly used in the Internet as it guarantees the integrity of the transferred data. This is done for instance by assuring the correct sequence of data and adding checksums to avoid erroneous content.
UDP	User Datagram Protocol: Protocol for an unreliable data transfer between two communication devices. There are a few applications that make use of UDP instead of TCP because UDP causes less overhead and therefore increases the transfer speed. Comparing UDP and TCP shows the main disadvantage of UDP. As UDP does not assure the integrity of the data or even its delivery an application which uses UDP has to take care of this tasks itself.
UMTS	Universal Mobile Telecommunication System: UMTS is a so-called "third-generation (3G)," broadband, packet-based transmission of text, digitized voice, video, and multimedia at data rates up to 2 megabits per second (Mbps).
XML	Extensible Markup Language: Markup language to create, describe, share or modify richly structured documents over the web. As XML describes the underlying information and its structure, content can be separated from its presentation. This overcomes a severe limitation of HTML, which merely describes content presentation for a particular set of HTML-compliant applications (like Web browsers). Furthermore XML is base for languages like WML and SOAP.