Design Notes

Embedded TCP/IP Implementation

Ethernet and TCP/IP communications are the networking backbone of PCs. In recent times however, a new market has emerged, networking embedded devices to extract information in real time. Applications include building automation, home automation, telemetry, industrial control, and Point Of Sale systems (POS). The ability to gather, collate and process data quickly and easily, provides tremendous advantages. Information is gathered (virtually in real time) allowing decisions to be made with minimal latency.

Data can be obtained using LANs, WANs and the Internet, providing a very cost effective solution when compared to competing technologies such as dial-up lines, GSM and GPRS. There is no doubt the use of TCP/IP over Ethernet networks (Intranet and Internet), is revolutionising how we obtain information. Dumb devices which previously had slow forms of networking such as RS485, or no networking at all, can now easily communicate with the outside world using a LAN connected to the Internet.

One asks, how can a simple embedded product communicate over a network using Ethernet and TCP/IP, when previously this required the power of a Pentium processor? The answer is simple. The processing power of a Pentium is only required when large volumes of data are to be processed. Most embedded products transmit and receive very little data, hence high processing power is not required. This allows microcontrollers to implement TCP/IP.

Ethernet Solutions

There are a variety of Ethernet solutions in the marketplace, but they all fall into two categories.

<u>Processor and Controller</u>

The lowest cost solution is a processor (microcontroller or microprocessor), combined with an Ethernet controller, as shown in figure 1.

The processor communicates to the controller using an 8 or 16 bit parallel bus. The processor implements the upper layers of the ISO 7 layer model such as processing the IP packet, TCP packet and the upper layer application data. It also programs the 48 bit MAC address in the Ethernet controller.

The Ethernet controller is a 10Base-T device, and has a built-in transceiver, which manages the lowest level of the ISO 7 layer model, i.e. the physical layer interfacing which is the electrical interface. It also has a Media Access Controller (MAC) which handles layer 2 of the ISO model, i.e. the data layer which is the Ethernet data frame.

The choice of processor is application dependant, as it will probably be required to perform many other tasks in addition to Ethernet and TCP/IP. Over the last few years, semiconductor vendors have recognised the evolving new market in embedded TCP/IP communications. Consequently, they are now offering turnkey Ethernet and TCP/IP solutions, including schematic diagrams which show how to interface their processor to a recommended Ethernet controller.

They also provide TCP/IP software stacks, either as C source codes or binary library files. The TCP/IP code can be linked with the user's source code at compile

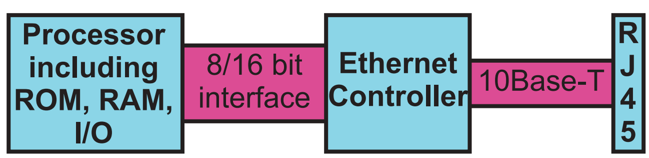

Figure 1: Processor and Ethernet Controller

<u>**Electronics By Design**</u>

Suite 25, 1-7 Jordan St Gladesville 2111
Phone : (02) 9816-3965
Fax: (02) 9816-3967
Web: www.electronicsbydesign.com.au

time. Most offer the software for free, or it's included in the cost of the compiler or development system the engineer is required to purchase. The TCP/IP software stack is usually available royalty free. This makes TCP/IP implementation much simpler than it used to be, however it still requires a steep learning curve to understand and implement the protocol, and have the capability to debug a new design.

One word of warning however. Vendors who provide source codes instead of library files, write the code in C but with lots of assembler. This makes it extremely difficult to port the code to a different vendor's processor!

Luckily for us embedded designers, the phasing out of ISA bus in PCs, has left quite a range of Ethernet controllers available for use in embedded applications. ISA controllers are limited to a speed of 10Mbit/s, which is ample for embedded Ethernet. Controllers can be interfaced to the processor, and most of them have an integrated MAC and transceiver.

Single Chip TCP/IP Bridges

Single chip TCP/IP solutions offer the benefit of ease of implementation. These solutions usually consist of a single chip which has Ethernet interfacing on one side, and a simple interface on the other side. The simple interface can be I2C or SPI bus, serial port (TTL levels), RS232 or a parallel bus. The topology of a bridge is illustrated in figure 2.

The designer adds an RJ45 connector (and in some cases an Ethernet transceiver to handle the electrical interfacing), and viola, the design is done. The microprocessor communicates to the bridge using the simple interface to receive and transmit data. The bridge implements Ethernet and TCP/IP communications and handles everything, including the entire TCP/IP communications protocol stack. The host processor can program various parameters into the bridge including the IP address and various timeouts.

These devices are the simplest available implementation of TCP/IP. They require no TCP/IP software to be written. The bridge does everything.

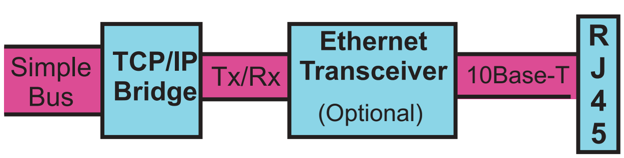

Figure 2: TCP/IP Bridge Solution

Now for the downside. These devices are very expensive, and are cost effective only for low product volumes. If volumes are medium to high, it's more cost effective to use a processor, and interface it to an Ethernet controller as described above.

Other Requirements

There are other requirements to be aware off when developing an Ethernet and TCP/IP product.

Protocol Analyser

At some stage during development, a protocol analyser will probably be required to view the contents of packets transmitted on the Ethernet cable. This includes the lowest level data, i.e. the Ethernet frame, all the way to the upper layer application protocol. The analyser breaks the packets down into their defined fields, and they can be viewed and checked.

Analysers are available as hardware or software. Hardware analysers physically connect to the Ethernet cable, and can be standalone devices with a colour screen to display relevant information. Software analysers run on a PC, and use the Ethernet (NIC) card of the PC to extract the data. Software analysers are much cheaper than hardware analysers and work very well.

Analysers can also transmit test packets prepared by the user. These packets can be used to test the response of the target design to anomalies, and test the operation and response to specific packets, which would otherwise be impossible to generate.

MAC Addressing

An Ethernet product requires a unique 48 bit Ethernet address for every product released by a vendor. MAC addresses can be obtained at a reasonable cost from the IEEE. They are programmed into the target product at production time. The numbers also need to be carefully controlled so they are not duplicated.

Electronics By Design has implemented a number of embedded TCP/IP solutions. We can assist a client with the design and implementation of a TCP/IP product, and minimise the learning curve, resulting in a fast development path, and a quicker time to market. By bringing the product to market in a quicker time frame, more revenue is generated for the vendor, which offsets the cost of using a third party design house.

Author

Written by Ef Misoyannis