Application Note #2

Embedded Modems – Are we communicating yet?

September 1999

SCOPE.

There is a growing requirement to add remote communication capabilities directly into embedded systems using modems, e.g. remote checking of fuel tank levels permits better inventory control and delivery. During development, engineers face a number of dilemmas:-

1. In many cases, modem development is not a core competency. As a result there are many misconceptions which yield non-ideal solutions.
2. Embedded systems have many varying requirements compared to PC based modems, therefore many off-the-shelf modems are unsuitable which in many cases lead to a bespoke design.
3. Regulatory approvals are non-trivial and ambiguous, particularly with the recent and forthcoming changes associated with CTR21 and the R&TTE directive. This represents the key risk impacting time to market for any company.

The purpose of this article is to educate embedded system engineers on the key issues related to embedded modem design with particular emphasis on the above, in order to side step the many pitfalls. It provides an analysis of the types of solutions available, the applications they are best suited and gives an indication of advantages and limitations. The article covers the following topics:-

Background

By means of introduction, describes where and why modems are used within embedded systems. The growing market is attributed to the widespread use of low cost microcontrollers in place of electro-mechanical designs, and the growth of the Internet.

Modem markets

The embedded modem market is significantly different to the PC market. The key differences include physical size, product lifetime, the amount of data to be transferred, connection times, shared telephone connectivity and power consumption etc which vary on a per application basis. This section examines the limitations of off-the-shelf PC modems in these areas.

Modem building blocks

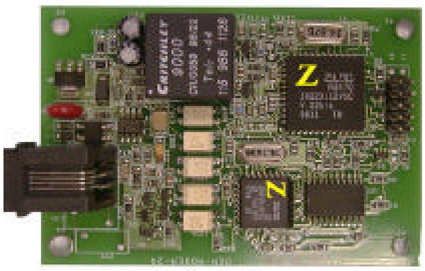
In order to understand the types of embedded modem solutions, it is important to understand the basic modem architecture. This permits the boundaries dividing host and modem to be re-defined in order to reduce cost for high volume applications. Furthermore it examines the additional features required by many embedded systems, i.e. On/Offhook parallel phone detection, serial phone detection, line disconnection and caller line indentification (CLI).

Approvals

A brief summary is provided to clarify the ambiguities associated with CTR21 Pan-European approvals and the implications of the R&TTE directive.

Types of solution

There are many types of modem solution including PC modems, socket modems, soft modems, OEM modules and bespoke designs based on a data pump and DAA. This section examines the advantages and limitations of each with respect to risk, time to market, approvals, volume and cost.



V.22Bis OEM Modem module

Upgradable modular solutions

The article concludes with the introduction to a family of footprint upgradable OEM modules designed specifically for the embedded market. In summary they provide a turn-key solution addressing the feature set required for the embedded market, approvals issues and provide a route to a lower cost via technology licensing for high volume applications.

Comtech House, 28 Manchester Road, Westhoughton, Bolton, Lancs, England BL5 3QJ
Tel: 01942 851800 Fax: 01942 851808 Email: sales@ctl-bolton.com Web: www.ctl-bolton.com

EMBEDDED MODEMS – ARE WE COMMUNICATING YET?

More and more products require remote communication capability. However, if Telecommunications is not a core competency, and considered by many as a "black art" with its Regulatory Approvals "minefield" - where do you start? Steve Whitehead, Technical Director of Complementary Technologies Ltd looks at the implications of designing modems within embedded systems.

BACKGROUND

For the computer literate, a modem is well understood as the "box of tricks" that enables his computer to communicate over a telephone line to other remote computers, servers and networks. It allows him to remotely access his office network when working from home, and the ability to connect to the Internet so that he can browse the World Wide Web, exchange emails and even order his favourite Pizza. In its simplest form, a modem can be considered as a "telephone for computers". Instead of allowing humans to exchange analogue audio, it allows computers to exchange digital data using the telephone network.

Even if we do not use a computer and a modem to communicate with remote networks, we probably use modem's everyday without realising it.

- A modem is used in a point of sale terminal when we use our credit card to purchase goods.
- A modem is sometimes used when we order a movie at home using a satellite television set top box
- Domestic electricity or water usage may be monitored by the utility company for billing purposes.
- A security system may dial a central monitoring station to notify any alarm events.

There is a growing demand to add remote communication into embedded products, the key advantages being flexibility and reduced cost of ownership. A few simple examples include

- Reduced engineering call out
- Automatic stock/inventory ordering
- Diagnosing faults from central sites
- Gathering real-time statistics
- Automatic security/event reporting.

For example, it is far cheaper for a vending machine to warn a central site that it is low on stock, rather than dispatch maintenance engineers when the machine does not need re-stocking. The cost of one call for automatic re-stocking is naturally less when compared to wasted engineering time, not to mention transportation charges such as lease hire, car tax, petrol and other consumables.

Much of the demand for adding modems into embedded products is fuelled by 2 key developments:-

1. The growth of electronic control using low cost microcontrollers in place of electro-mechanical control. Primarily, the microcontroller provides the intelligence required to communicate with local and remote devices since it is digitally controlled using firmware, i.e. dedicated software programmed into Read Only Memory (ROM). If a unit has the capability to send serial data to a local terminal for diagnostic purposes, a modem can provide the same capabilities over a telephone with minimal software changes. Furthermore, a microcontroller will have the ability to control, monitor and store a greater amount of information. For example, a vending machine can report not only the stock levels, but information such as the last maintenance visit time, peak usage etc - which may allow a maintenance visit to be scheduled outside of busy hours e.g. tea break, when the vending machine will be used most, thereby minimising lost revenue.

2. The explosive growth of the internet, which essentially provides a communication backbone between central servers and remote units. Any remote unit can connect to the internet using a local point of presence and transfer data to any server which is also attached to the internet. Calls are charged at local rates independent of the location of the central server which manages the system. Considering that a large system may have many thousands of remote units supported by one server, there are significant cost savings to be made by reducing the amount of long distance calls.

There are many applications, too numerous to mention all, that benefit from the ability to support remote communication. In addition to the applications already mentioned, others include franking machines, mailing systems, gaming machines, data loggers, fuel tank level monitoring, industrial control, energy management, control systems, medical monitoring, process control, lighting control ... The list is endless and growing, particularly as the cost of microcontrollers continues to reduce, which fuels new markets and applications.

MODEM MARKETS

One obvious way to add modem capability to an embedded system is to use a standard off-the-shelf box modem, usually to be found in your favourite PC superstore. At a first glance, it presents a least risk solution since it is fully Approved and relatively cheap, due to the competitive and volatile nature of the PC market. However, BEWARE! Considering the lifetime of an embedded system compared to PC based products, does it offer a long term solution? Their are other considerations - some obvious, but others more subtle which essentially highlight some fundamental differences between PC and embedded system modem needs ...

- **Do you physically have the space to integrate a box modem inside your product?**
  Many embedded systems do not have the space to integrate a box modem within the product enclosure. This is further hindered by the safety approvals associated with many box modems, i.e. the box modem is approved together with its own external power supply unit, since many box modems only provide a supplementary isolation barrier. In many cases, box modems will only be suitable as an external unit to the embedded system, which are prone to wiring issues and inherently less appealing than a fully integrated system.

- **How much data do you need to transfer?**
  The PC market demands speed, speed and more speed in order to keep up with the large and growing file sizes associated with modern PC applications, e.g. desktop publishing, video, databases and web browsing to name a few. Box modem manufacturers therefore use the latest state of the art chipsets offering the highest possible speed, with the result that slower speed modems and associated chipsets are quickly obsoleted. On the contrary, many embedded systems only require a small amount of data to be transferred. Think about how much data is actually transferred for credit card transactions - a credit card stores probably less than 50 characters of information! A highspeed modem, boxed or otherwise is therefore overkill in terms of both speed and price in many applications. For low data rate embedded modem applications, it is therefore important to choose a solution based on a chip set supplier that is focussed on the embedded market so that your design does not suffer from the rapid fluctuations in technology and volatile nature of the PC market.

- **Is the connection time important?**
  Consider a large network of units which remotely communicate with a central site, such as credit card transaction systems. The key requirement is to transfer the data as quickly as possible in order to minimise call charges and reduce the processing bottle neck at the central site. No problem - we could just increase the data rate using a faster modem? This is a popular misconception! Undoubtebly this is true when we are transferring large files, however when transferring small amounts of data the initial connection time prior to data transfer becomes more significant. Anybody who has ever worked in an office environment has heard the annoying exchange of squeaks and warbles that modems generate before faxes and emails are transferred! Modems are complex beasts. They need to detect and adapt to both the remote modem and telephone line in order to communicate at the same speed whilst minimising potential noise. Faster modems use more complex schemes to increase data rate, as a result the initial "handshake" is more complex and therefore takes longer to execute. A low speed modem can therefore connect quicker than a highspeed modem. For low amounts of data, a low speed modem can connect, transfer data and disconnect before a highspeed modem has completed handshaking! Figure 1 shows relative comparisons between connection times and modem standards. Considering that box modems are predominantly highspeed, they are inherently slow for small amounts of data.

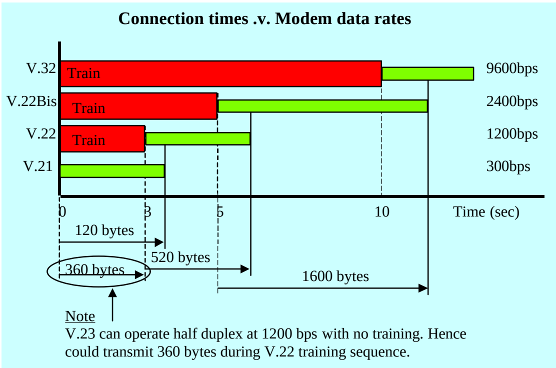


Figure 1

- **Are you connecting to the Internet?**
  V.90 is the latest state of the art modem standard, some times referred to as a 56K modem. Prior to this, V.34 supported modem speeds upto 33.6kbps. Many people are unaware that a V.90 modem only receives data at 56Kbps, but actually transmits data at V.34 rates. If you were to connect two V.90 modems together, what data rates would you expect to connect at, considering that their is no source which can transmit at 56Kbps? So, if you are connecting directly to a central site, their is no benefit to using a V.90 modem over a V.34, since both will connect at 33.6kbps in both directions! In order to support 56Kbps rates, a V.90 DIGITAL modem is required at the central site side. Typically, these are provided by Internet Service Providers (ISP), such that highspeed downloads at 56Kbps can be achieved from Internet servers. If your application does not demand that you download large files from the Internet, then there is no need to use the latest, more expensive technology. Once again, for embedded applications that require a small amount of data to be transferred, low speed modems can be used to access the Internet, e.g. File Transfer(FTP), Email in order to minimise cost.

- **Will the modem share a network connection with a telephone, if so which has priority?**
  PC based systems are designed for use by humans, and therefore can respond easily to human intervention to changing or adverse conditions. For example, if you are "Surfing the Net" and decide to call mother, you can quickly terminate the modem connection, freeing the line for your telephone call. Embedded systems do not have this luxury and must be robust to the needs of the application. For example, under alarm conditions, a security system will take priority over the shared telephone in order to dial the monitoring station, whereas a utility meter will have lower priority over the telephone user. A security system or digital set top box may need to sense if the telephone line is physically connected. Embedded modems may therefore require additional circuitry to support these features, which are not traditionally supported by box modems, or only via human intervention.

- **Is power consumption an issue?**
  Embedded systems have many varied power supply requirements, ranging from battery powered utility meters to mains powered control systems, not to mention battery backed mains powered medical systems and gas turbine generators! In its minimal form, a modem could be made of simple tone generators and filters. Granted – the speed will not be of lightning proportions, and considered laughable by the avid "Net Surfer"! At speeds above 1200bps full duplex (both directions), simple filters cannot be used to meet the modulation requirements, therefore Digital Signal Processing (DSP) technology is required. As an approximation, a V.22Bis modem (2400bps full duplex) requires 10Mips of DSP processing power and a V.34 modem requires 40-50Mips of DSP processing power. Since the power consumption of CMOS devices is directly related to speed, then power consumption is directly related to the speed of the modem. Since box modems typically use the latest state of the art technology they are inherently power hungry. On the contrary, embedded modems and related chip-sets are available across the spectrum of modem speeds which permits the engineer to trade-off speed in order to meet the power requirements.

- **Is your application high volume?**
  It goes without saying that high volume applications are extremely cost sensitive, where savings of even a few pence are significant. Definitely not the domain of the box modem with additional enclosures and power supplies! Furthermore, most typical modems will also include a microcontroller for driving the host and the main modulation chip (A.K.A Datapump) which interfaces the digital data to the analogue telephone line. Since the modem is embedded in a system that is also controlled by a microprocessor, then an obvious area for cost reduction is to eliminate the modem controller, allowing the host microprocessor to drive the datapump directly. Taking this one stage further, if the host processor is sufficiently powerful enough, their is no reason why the host controller could not also implement the datapump function - this is known as a "Soft Modem". Whilst offering the least cost solution, they naturally represent more risk since the engineer will need an extensive knowledge of modem development, firstly to design the custom modem, and secondly to take the design through approvals. Considering that modem development is highly specialised and in most cases not the core focussed competency of a company - this is a dilemma, particularly when time to market and least risk are also key issues for consumer products!

MODEM BUILDING BLOCKS

At this stage, I think it is worth spending a few moments to review the key building blocks within a modem in order decide on the level of design that the application demands. Figure 2 shows the 4 key blocks within the modem.

Modem Controller

The Modem Controller provides the key interface to the Host microprocessor using the AT-Command set. By decoding AT-Commands, it controls the datapump and DAA in order to establish a call. When connected, the Modem Controller passes data from the Host to the datapump, implementing any data compression (V.42Bis) and error correction (V.42) in order to increase bandwidth and reduce data corruption's.

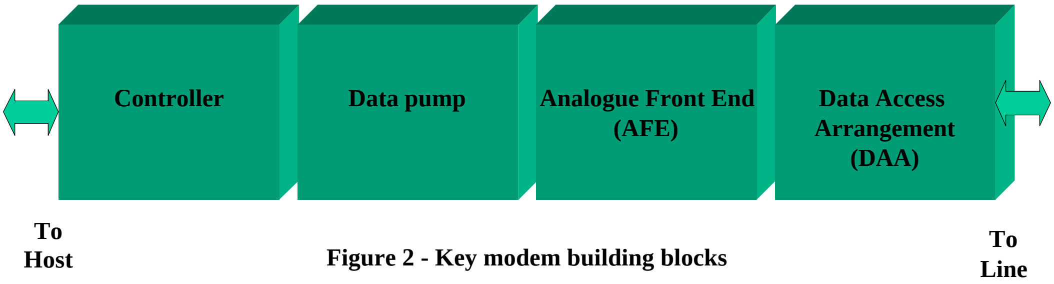


Figure 2 - Key modem building blocks

1. **Datapump**
   The datapump is the heart of the modem, which converts digital data into a format suitable for transmission over an analogue line, i.e. it modulates the data onto a carrier signal. Likewise, it converts the analogue signal from the line back into digital data, i.e. it demodulates the data from a carrier signal. Thus the derivation for the name "Modem" - MOdulator/DEModulator. We already know that DSP technology is used to implement complex modulation schemes at speeds above 1200bps full duplex in the digital domain, and therefore also ideal for performing DTMF generation and Call Progress Signalling (CPS) detection in order to establish the call.

2. **Analogue Front End (AFE)**
   The AFE performs the analog-to-digital conversion (ADC) and digital-to-analogue conversion (DAC). The AFE is sometimes called the modem codec, the precision of which may vary depending on modem speed and modulation scheme.

3. **Data Access Arrangement (DAA)**
   The DAA interfaces the modem to the phone network, and is required by BABT and network providers throughout the world to provide safety protection both to the end user and to the telephone switching lines. DAA's usually consist of discrete components and provide complex impedance matching, current limit, voltage isolation, ring detection, pulse dialling, Onhook and Offhook functions.

OTHER FEATURES

In order to provide an extra level of decision making and robustness in an embedded application, a number of other DAA features may need to be considered, particularly if the modem is to provide an automatic response without human intervention. Typically these types of functions are not provided by a box modem, since PC based applications are naturally based on user control.

1. **Onhook parallel phone detection**
   Prior to making a call, the modem will monitor the line to determine if the parallel phone is Offhook. This could be achieved by detecting dial tone, however this creates annoyance in the form of line "clicks" as the modem grabs the line. Additional circuitry can be added to the DAA to perform this function non intrusively, however care has to be taken to ensure operation with varying line feed voltages and parallel phone loadings.

2. **Offhook parallel phone detection**
   Once the modem has established a connection, it will continue to monitor the line to determine if the parallel phone subsequently goes Offhook. Once detected, the modem will hang-up the connection permitting the phone to access the line. This can only be performed using additional DAA circuitry.
3. **Line connection detection**
   As an extension to parallel phone detection, it is possible to detect if the modem is physically disconnected from the network. Offender tagging is naturally a key application for this feature. Although this may sound a trivial feature to implement, it is particularly difficult to design when considering response times, line voltage variations and parallel phone loadings.

4. **Serial phone detection**
   In applications where the modem needs to have higher priority than the attached phone, this can only be achieved by locating the modem between the line and attached phone, i.e. in serial. This permits the modem to disconnect the serial phone leaving only the modem connected. This can only be performed using additional DAA circuitry.

5. **Caller identification**
   Any embedded system that receives calls must be able to automatically answer a call. Traditionally this is achieved by detecting the RING signal, e.g. Auto-answer after 3 rings. However Onhook Caller Line Identification (CLI) can also be used as a basis for auto answer, e.g. a modem is programmed to auto-answer a call from the central station. Since the CLI information is transmitted prior to the first RING signal in many countries, this permits the modem identify the calling party, e.g. central station and answer the call without disturbance to an attached phone. CLI is therefore ideal for shared line applications such as cigarette machines within pubs, and utility meters within the home. Furthermore, CLI is ideal for providing a level of security permitting a remote embedded system to only answer valid calling party calls.

APPROVALS

Approvals are rightly regarded as a "minefield", particularly as they never appear to be static. After many years of each Country dictating specific requirements, e.g. NTR3 for BABT, then CTR21 finally became endorsed by the European community offering a single Pan European Approval with effect from August 1998. Great idea! It appears to offer a panacea of 1 design covers all, making it much easier for the "non modem expert" to develop modem based products. However if you look deeper, an attempt to ratify the European standards was made with NET4 a number of years ago. Because the task was so immense due to the variation in requirements, NET4 effectively became a huge document that described the behaviour of all of the European exchanges rather than form one common standard. CTR21 is therefore a very small subset of NET4, with the result that CTR21 essentially protects the network, but does not guarantee that the modem will actually operate! For example, CTR21 does not cover call progress tone detection of which there is huge variation across Europe. Many Countries provide advisory notes (ATAAB), which when used in conjunction with CTR21, give greater confidence that the product will actually work. However, in a number of cases the advisory notes refer you back to NET4 anyway. Since it is pointless approving a product which may not work - do we really have a Pan European standard? This is complicated even further by the forthcoming Radio and Telecommunication Terminal Equipment Directive (R&TTE) which permits self declaration without formal approval testing from March 2000 onwards. The onus is therefore on the manufacturer to guarantee that a product conforms, whilst the focus of the approvals body moves into a policing role. That sounds great - no formal approvals, no time to market delays, and reduced cost! But if you can't guarantee that your product conforms and there is a problem in the field - what kind of implications does that have for a company! On a personal note, I believe this opens the door to an influx of cheap substandard product, which combined with lack of policing will cause mayhem in the market.

TYPES OF SOLUTIONS

So, if we discard PC modems, what solutions best suit embedded applications? Naturally, you can design your own modem by bolting a datapump and DAA onto your host processor as we have already discussed. If you do not feel comfortable with designing your own DAA, there are a number of Silicon and Hybrid DAA's on the market to ease the burden - the application notes from vendors are always a help. However remember, CTR21 protects the network and does not guarantee operation - the host controller software is therefore critical to success! The same is also true using soft modems, however it really only makes sense to use these routes in high volume applications particularly if there is any licensing cost involved. When designing your own application at this level there are also a number of other considerations, e.g. achieving good performance at low signal levels, investment in test equipment and compatibility with other modems. There can be a significant variation not only in AT Command sets, but also handshaking between modems - remember that you also want to be able to talk to every body else's modem! A least risk route is to opt for a fully approved modem module, which resolves all the key issues, e.g. approvals, compatibility, performance, risk etc. These can take the form of "Socket Modems" or OEM modules. Socket modems are component size modules typically developed by modem chipset manufacturers, and often require external circuitry for additional features such as parallel phone detection, CLI etc as well as the RJ11 connector itself. OEM modem modules tend to provide a full turn-key solution incorporating many of the features discussed in this article.

MODULAR

As communication technologies rapidly expand, their is a growing need to integrate communication products in a modular upgradable format rather than dedicated on-board hardware. With the advent of Pan-European Approvals and varying uptake of new technologies such as ISDN and ADSL across Europe, there is an increasing requirement to tailor standard products prior to shipment to meet the end customers communication needs. Furthermore, a modular solution permits the approval to be associated with the module, and not the complete embedded system. This permits ongoing upgrades to the host system without affecting approvals, thereby permitting fast time to market. In addition, the host system does not need to be manufactured at a BABT Approved production facility, eliminating the need for regulatory production test and BABT audits. In the case of dedicated hardware or soft modems, approvals can be affected by minor application software modifications or PCB changes. Although socket modems provide a turn key solution, they only offer limited upgradability, typically governed by the chipset range of the manufacturer. OEM modem modules tend to offer upgradability across chipsets from numerous manufacturers and therefore a more extensive range.

OEM-MODEM-24

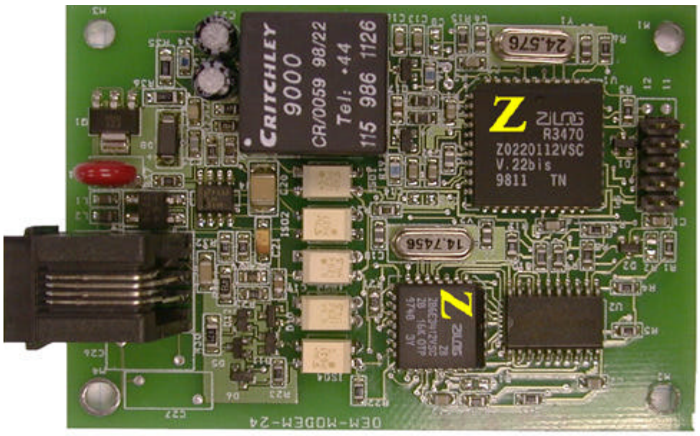
Oem-modem-24 is the first of a family of communication modules available from Comtech. It is a fully BABT Approved low speed modem for internal embedded applications and meets the modem standards for V.22Bis, V.22, V.23, V.21, Bell 202 and Bell 103. Oem-modem-24 provides a 5v TTL compatible serial interface to the host data terminal equipment (DTE), allowing the host to control the modem via the AT-Command set. It additionally supports auto-dial, auto-answer, Onhook and Offhook parallel phone detection.

PAN-EUROPEAN MODULES

A full range of CTR21 Pan-European Approved modules are currently being released, offering low speed, high speed and low power capabilities. Furthermore, a range of embedded features include Caller Line Identification (CLI), line connection detection in addition to both Onhook and Offhook parallel phone detection. For high volume applications where cost is sensitive, Comtech offer technology partnerships. This permits customers to focus on there own core competency, whilst reducing cost and risk by licensing proven technology from a partner than can offer design, approvals and production support.

UPGRADABILITY

Comtech have developed a range of foot print compatible OEM communication modules offering direct upgradability between low speed, high speed, low power modems and Basic Rate ISDN terminal adapters. These are fully approved and can be treated as a component ready for direct integration into an embedded system ready for network connection. In summary, they offer fast time to market, reduced risk, flexibility for customer needs, and provide a route to address future emerging communications technologies.



Approximate size – V.22Bis Low speed modem