XMODEM00.WS4 (= XMODEM story)

- "Hobbyist Computerized Bulletin Board"
  Ward Christensen (software) & Randy Suess (hardware)
  BYTE, November 1978, p.150

(Retyped by Emmanuel ROCHE.)

Note: This project was a collaboration of Ward Christensen and Randy Suess. Each had a particular part to uphold. The first part of this article, describing the purpose of the bulletin board, is written by Ward. The part describing the hardware details is written by Randy... RGAC

The Computerized Hobbyist Bulletin Board System is a personal computer-based system for message communication among experimenters. People with terminals or computers equipped with modems call in to leave or retrieve messages. It was conceived, designed, built, programmed, tested, and installed in a 30-day period (January 16 1978 to February 16 1978) by the two of us. In an effort to generate material for our computer's club newsletter, I first thought of the idea and discussed it with Randy on January 16 1978.

We laid out the hardware requirements: an 8080 processor with 24 K bytes of memory, single floppy disk, modem interface, and some sort of local keyboard and display. Randy scoured the computer stores and purchased a mother board and two 4 K byte memory boards at a reasonable price. I talked with Lloyd Smith and Bill Bassett, who operate DMA Inc, a manufacturer of floppy disk drive systems based on the Tarbell controller and the Innovex (now Innotronics) floppy disk drive. DMA offered to donate 40 percent of the cost of a controller and floppy disk drive to the project. I purchased the floppy disk drive, controller and CP/M license, and loaned 24 K bytes of memory to the project, pending receipt of 16 K bytes offered by DMA. Randy donated his D C Hayes modem board, PolyMorphics VTI, SwTPC keyboard, power supply, chassis, IMSAI 8080 processor card and Vector memory board.

We started with the monitor for the system, but found that it was difficult to make the VTI keyboard port work because the VTI keyboard data bus shares the bus carrying the characters being displayed. To solve this, Randy bought a Processor Technology 3P+S board and interfaced the keyboard to it. This also allowed us to have a sense switch port, using the 3P+S. Since this board has a serial port, Randy later decided to add a Teletype to the system, for logging incoming data. This completed the configuration.

Programming

In the first week of the project, I wrote a mock-up of the software using MITS 8 K BASIC. The input/output (I/O) drivers could be switched to my modem under sense switch control, so I had people call in and critique the system. Many good suggestions were made.

By now, Randy had the computer far enough along to need some programming, so a monitor was put into read-only memory. Since the system was to run under CP/M, a Teletype-compatible scroll routine was also put into the read-only memory, for both the monitor's and CP/M's use. Additions were made later to support the 3P+S board for keyboard input.

After the BASIC mock-up of the system was close to what we wanted, programming started on the assembler version of the software.

Assembler language was chosen over BASIC for the inplementation language because of size and speed efficiency, and to maintain control over such functions as Control-K to terminate (kill) the current function and return to the main menu. The program now consists of the pieces of assembler source, shown in Table 1, which are combined to produce the final assembler source program.

Table 1. Assembler routines used in the bulletin board program.

Name	Function
CE00	Mainline routine: sign on, menu, etc.
CE10	Goodbye function, comments function.
CE20	Message summary retrieval function.
CE30	Message entry (input, edit, save).
CE40	Message retrieval.
CE50	System functions (erasing messages, etc).
CE80	Disk subroutines (read, write, extend, etc).
CE90	Non-disk subroutines (modem I/O, etc).
CE99	Constants and work area.

(ROCHE> "CE" = ?)

In addition to this application program, a modified Basic Input/Output System (BIOS) for CP/M had to be written. We wanted the system to be able to be started from a "cold boot" bootstrap program and come up running the bulletin board program, not CP/M. This would make the system more crash-resistant, since the ringing of the phone would activate it, and the person calling in would not be able to gain access to CP/M itself. I studied the functions of the CP/M Console Command Processor, and modified the cold start routine to be able to load in a program and branch to it. This was done selectively under control of the sense switches. If we wanted to load CP/M instead, we just set the REMOTE mode sense switch off.

About 30 days after the project was conceived, it went on the air via a new telephone line Randy had installed in his basement. If was used for a week by our friends and was then announced at the February (1978) meeting of CACHE, the Chicago Area Computer Hobbyist's Exchange. A poll was taken to see how many people could call. We expected perhaps five or ten, but 25 hands were raised!

During the weeks that followed the CACHE meeting, many people tried the system. We informed Dennis (D C) Hayes and PCNET of it, and they both tried it. Most people were able to communicate with the system. Those who couldn't had either bad telephone lines or modems that put out weak signals, which the system seems quite intolerant of. People who left messages saying they had some information of interest, and those who said they needed information, discovered that other people using the system contacted them. We were pleased to find the system working this way, because that was one of its purposes.

Getting good feedback about the operation of the system is important to us. That is why the GOODBYE function allows you to leave comments. Many people contributed good suggestions which were soon implemented. For example, a new version of the software was being put up on the system every few days, to send nulls after carriage returns and linefeeds for the many people calling in using Texas Instruments Silent 700 terminals.

We will continue to improve the system, although we consider the project to be close to completion. We might add date and time or, if we get ambitious, a second telephone line, but we are quite satisfied with the system as it exists now. We would like to see other experimenters or clubs implement such a system.

These bulletin board systems could then become NODES in a communication network of automated message and program switching.

Hardware details

My objective for the hardware of the bulletin board system was to get the most functions at the least cost. To this end, I started with my IMSAI 8080 processor card, having replaced it with a Cromemco Z-80 in my personal system. I found an old Vector mother board for free, and two reasonably priced 4 K byte memory boards at a computer store. I also bought a PolyMorphics video terminal interface (VTI) card.

We burned a monitor for a memory board to support the video terminal interface. Adding an SwTPC keyboard completed the local console for the system.

We are now running an 8 K Vector Graphic memory board and two 8 K byte boards designed by Forrest Duston (a local hobbyist and design engineer), which were donated by Lloyd Smith and Bill Bassett of DMA Inc.

A floppy disk was necessary for storing the messages, so Ward bought a Tarbell controller board and Innovex drive from DMA, who graciously offered it at 40 percent off.

I put these components into a chassis and home-built card cage, and started testing the system. I found some incompatibilities in the hardware, such as between the new VTI and the memory board (pressing RESET would not give control to the monitor in read-only memory). Switching to an older design video terminal interface solved the problem.

I then found problems with the keyboard. The sharing of the interface's on board bus between the display data and keyboard status data caused the display to break up badly. Putting a delay loop in the software didn't help, because the non-latched data would easily be missed. I solved this by purchasing a

Processor Technology 3P+S board for the keyboard input port. This turned out to be very useful, first to allow a sense switch input port, and later to drive a logging Teletype for all incoming messages.

I had also purchased a D C Hayes modem board to be used for exchanging files and programs with other CP/M users, but decided to use it in this system because of its ability to select the transmission rate (110 or 300) and to answer the phone. Hooking this through a data access arrangement (DAA) to the phone line completed the STANDARD hardware of the system.

Ward needed some way for the system to recover from possible software and hardware crashes, so I built a simple interface between the modem board and the hard reset of the system, so that, every time the telephone rang, the system would do a cold boot to CP/M. A 555 timer was added to the reset circuitry, so that, after the first ring, subsequent rings would not interfere with the CP/M booting process. I did not use the ring detect feature of the Bell data access arrangement because the modem board's setup was easier to control and detect under software. After some preliminary problems with the modem board (Dennis Hayes was more than helpful in resolving these) and the Tarbell controller, the hardware has been functioning as expected.

To prolong the motor life and minimize power consumption, I installed a solid state relay that turns the drive motor on when the telephone rings. The 555 then provides 30 seconds during which ring detect is disabled to prevent subsequent rings from interrupting the booting, allowing CP/M to come in and answer the phone. The Tarbell controller board is designed to boot CP/M when the system is reset, so this hardware, with Ward's software, gets the system going.

That's how the hardware now stands, except that I sold the Teletype and am going to use an SwTPC PR-40 for the logging device. The system has now been operating very successfully since mid-February 1978.

Programming details

The following information is supplied for those who want the details of the structure of the files on the system, what modifications were made to CP/M to support the system, or the design of the bulleting board program itself.

File structure

CP/M is capable of storing 240 K bytes of information on a (8-inch) diskette. The directory is capable of keeping track of up to 64 files. I would like to keep about 200 to 300 active messages on the system. This means that each message cannot have its own directory entry. I decided to group messages in quantities of ten per directory entry. Thus, filename MESSAGE.00X contains messages 1 thru 9, MESSAGE.01X contains messages 10 thru 19, etc. Assuming the number of active messages to be about one half the highest message number implies that there will be five messages per file. If there are 40 active files (such as messages from 1 to 400), then there will be approximately 200 messages. If each message averages 8 lines of 50 characters, this is 400 bytes. The total disk space for the messages themselves will be 80,000 bytes. Thus, 200 messages can easily be held. The directory keeps itself clean: when the last message in a group of ten is erased, the file itself is automatically erased from the directory.

There are also message summary files. They contain the following fields for each message:

Message number.
Number of lines.
Date entered.
From (author)
To (recipient)
Subject.
Password (must be supplied to erase the message).

The summary files are grouped like the message files, but by 100s. Thus, summaries 001 thru 099 are in a file called SUMMARY.0XX, etc. This keeps the message erase function simple: it can read the entire file into memory, delete the summary, and rewrite the entire file.

There are a number of text files on the disk. They are used when predefined information is to be given, such as the welcome message. There are two types of files: straight text files, and question files.

Question files are used in such functions as Help, in which the user is asked a question: "Want help with summary function?", for example, and is then allowed to answer Y or N. If Y is typed, the text associated with the question is typed. If N is typed, the text of the message is skipped. Table 2 contains the text and question files used by the system, as well as other files not previously mentioned.

Table 2. Descriptive files to aid users of the bulleting board system.

WELCOME: This file is typed when you first connect to the system. It welcomes you, and tells of the various control characters that can be used to control the system.

BULLETIN: This file may or may not exist. If it exists, it is typed after the welcome file. It contains bulletins such as changes in the operation of the system, club meeting announcements, etc.

LOG: When you first get on the system, it asks you for your first and last name. They are written to this file, to give us a log of who uses the system. Also, if you are a first-time user, the system asks where you are calling from, and logs this information after your name.

FIRSTIME: This file is a question file, and it is interpreted if you reply Y to the question asking if this is your first time on the system. It prompts the new user about what can be done.

HELP: This is a question file. It is quite lengthy, going through the various system functions and, at each one, asking if you want help.

ENTINTRO: When you ask to enter a message into the system, you are asked if you would like to review the steps used to enter a message. If you say Y, the system types this file.

ENTRHELP: After you have entered your message, you have several options: aborting, editing, saving, or help. This file is the question file which is interpreted if you ask for help.

NEXT: This file contains the next message number to be entered into the system. It is incremented by one every time a message is saved.

COMMENTS: When you leave the system via the G (Goodbye) function, the system asks if you would like to leave any comments. If you say Y, it writes the lines you type into the COMMENTS file. This provides a technique much easier than message entry, so we will get feedback on the use of the system.

PASSWORD: Certain system functions are of such a nature that only the system operators (as opposed to users) may execute them. For example: erasing a message whose password is not known, exiting to CP/M for file maintenance, determining disk status, etc. This file contains the password which must be entered to get into this operator mode. It is a disk file, rather than being built into the program, to facilitate changes when the need arises.

CP/M modifications

When an unmodified CP/M system is initially loaded, control transfers to the "cold boot" entry in the Basic Input/Output System (BIOS). Its function is to:

- o Build the required jumps in low memory to CP/M.
- o Transfer control to the Console Command Processor.
- o Reset the disk, log in drive A, and begin communicating with the console.

Since we don't CP/M to get control when the system answers the phone, a modification was made to the cold start routine, so that the sense switches were examined to determine if the REMOTE switch was on. If not, control proceeded to normal CP/M operation. However, if the REMOTE switch was on, the system performed the reset disk system and log in drive A functions normally done by the Console Command Processor. The normal CP/M functions of OPEN and READ are performed to load the bulletin board program into memory. Note that all of this happens before the telephone is answered, so that, if the program can't be found, or has an error loading, a call is not wasted. When the program is loaded, control transfers to it, the telephone is answered, the information transfer rate is determined, and communication begins.

Bulletin board program

The functions of the program are:

o Answer the telephone.
o Determine the information transmission rate.
o Process the user's requests.
o Hang up the telephone.

The system answers the telephone by an output instruction to the D C Hayes modem board. It then has two subroutines, called SET110 and SET300, which set the transmission rate. Initially, the program gets control, calls SET110, which picks up the telephone, sends out the answer tone, and selects the 110 bps transmission rate with two stop bits and no parity.

The program then monitors the modem status for about 15 seconds to determine when a connection with another modem has been made. If no connection is made, a message is printed to the local console, the telephone is hung up, and control is transferred to the monitor. Note that the monitor does not take part in the bulletin board system, except to supply the scroll routine for the video terminal interface. The ring detect and system startup is all done through hardware, for reliability. Thus, the system is able to experience an unexpected power outage and still operate the next time the telephone rings.

If a connection is established, the system, at 110 bps, waits for a character from the line. If this character is not an ASCII carriage return, it switches to 300 bps and tries again. It tries each speed, alternately, ten times. If this technique does not result in a carriage return being seen, the system prints a message, hangs up the telephone, and jumps to the monitor.

The technique for establishing communications with the system is to:

o Dial the telephone.
o Wait for the answer tone.
o Puts the modem on the telephone line.
o Press RETURN repeatedly until the system responds.

When the data connection is established, the system types a carriage return and linefeed to indicate that it is connected. It then sends the contents of the disk file WELCOME to the caller, followed by any bulletins that may be of interest. This output sequence is shown in Listing 1. Next, it asks for your first and last name, and logs them to disk. It asks if you are a first-time user and, if so, asks where you are from, logs this to the disk, then gives you an introduction to the system.

```
          TERMINAL NEED NULLS?  TYPE CTL-N WHILE THIS TYPES:

             ***   WELCOME TO CBBS/CHICAGO   ***
      ***  WARD AND RANDY'S COMPUTERIZED BULLETIN BOARD SYSTEM  ***



     -----> CONTROL CHARACTERS ACCEPTED BY THIS SYSTEM:

          DEL/RUBOUT  ERASES LAST CHAR. TYPED (AND ECHOS IT)
          CTL-C      CANCELS CURRENT PRINTING
          CTL-K      'KILLS' CURRENT FUNCTION, RETURNS TO MENU
          CTL-N      SENDS 5 NULLS AFTER CR/LF
          CTL-R      RETYPES CURRENT INPUT LINE (AFTER DEL)
          CTL-S      STOP/START OUTPUT (FOR VIDEO TERMINAL)
          CTL-U      ERASE CURRENT INPUT LINE

      ----------------------   BULLETIN   ----------------------
              PROBLEMS WITH THE SYSTEM??
      HARDWARE: RANDY (SUESS), (312) 935-3356
      SOFTWARE: WARD (CHRISTENSEN), (312) 849-6279
      ----------------------   BULLETIN   ----------------------
      )
      ----------------------   BULLETIN   ----------------------
      --->  ALL USERS:  BE FAMILIAR WITH MESSAGES 3, 6, AND 60

                         NOTE
      -----> AS OF 4/8/78, MESSAGES PACKED AND RENUMBERED <-----
      ----------------------   BULLETIN   ----------------------
```

Listing 1. Output of the bulletin board system when a user first logs onto the system. The welcome message is printed out, followed by the important control characters. Any bulleting of general interest is also shown at this time. Bulletins can only be put on the system by the system operators. Individual users may not access this function.

The system also keeps track of how many people have used the system. This number is output at sign on. Mostly, it is just for interest, but helps to see if anyone has used the system since you last did.

The following prompt asks what you want to do:

```
      FUNCTION: B,C,D,E,G,H,K,N,P,R,S,W,X (OR ? IF NOT KNOWN)?
```

If you do not know or remember what the letters stands for, you type ?, in which case the system replies:

```
      FUNCTIONS SUPPORTED:

      B=PRINT BULLETIN
      C=CASE SWITCH (UPPER/LOWER)
      D=DUPLEX SWITCH (ECHO/NO ECHO)
      E=ENTER MSG INTO SYSTEM
      G=GOODBYE (LEAVE SYSTEM)
      H=HELP WITH FUNCTIONS
      K=KILL (ERASE) A MESSAGE
      N=NULLS: SET 0 TO 9 AS REQUIRED
      P=PROMPT SWITCH (BELL ON/OFF)
      R=RETRIEVE MSG
      S=SUMMARIZE MSGS
      W=TYPE WELCOME
      X=EXPERT USER
```

Details of these functions are given below.

B=PRINT BULLETIN

If you are using a printing terminal, you can simply look back to see what the bulletin said when you first connected to the system. If, however, you are using a video terminal, you may want to review what the bulletin said. The B function does this.

C=CASE SWITCH (UPPER/LOWER)

Most terminals are capable of handling any ASCII character that comes to them, whether it is lower or upper case, typically translating lower case to upper case if necessary. However, there are exceptions, notably the Heath H9 terminal, which displays garbage when sent lower case characters. For this reason, the system defaults to upper case only, until you use the C command to switch to lower and upper case mode. Using the command again will switch back to upper case only mode.

D=DUPLEX SWITCH (ECHO/NO ECHO)

Full duplex and half duplex refer to the ability for a medium (such as the telephone line) to have data going in both directions simultaneously (full duplex) or only in one direction (half duplex) at a time. These terms have now taken on different meanings when referring to a computer terminal. Full duplex refers to a terminal which sends a character from its keyboard, then echos what comes back from the phone line. Half duplex refers to a terminal which echos its characters locally. Thus, the D function is used to determine whether the system echos characters back to you.

If you are using a Teletype, for example, which prints every character you type, you will use the D command to go to half duplex mode, so you do not see every character twice (one from the Teletype, the other from the phone line). This does sacrifice some integrity, however, because you do not see the characters echoed back to you to verify that what you typed went in correctly.

E=ENTER MESSAGE INTO SYSTEM

You must know: the date; who the message is addressed to (up to 20 characters); and the subject of the message (up to 30 characters). You will be asked if you want to be able to erase the message; if you reply Y, you may elect to protect it with a 4-character password, or to use no password. We recommend not using passwords, so that it will be easier for the recipient to erase the message when he or she has read it. Also, you will be asked if you want to review the steps used in entering a message. This is useful the first few times you enter a message but, typically, is not used after that.

G=GOODBYE (LEAVE SYSTEM)

When you use the G function, the system asks if you want to leave any comments. If you say Y, you will be able to type in comments, which are written to disk at the end of the COMMENTS file. Typically, we find suggestions for system improvement, or congratulations, or requests to erase messages whose passwords have been forgotten, or any other information which the user wanted to send to the operators, but didn't want to leave as a message.

H=HELP WITH FUNCTIONS

The H function asks a series of questions such as: "Do you want help with the E (enter) function?" For any question which you answer Y to, you will be given some detailed information.

K=KILL (ERASE) A MESSAGE

The letter K (for kill) rather than E (for erase) was chosen, because E was already for enter. In order to keep disk utilization down, this command allows users to erase their own messages. The system asks for the password and, if correct, the message and its summary are removed. If users do not elect to be able to kill their own messages, a special system password, NONE, is put in the file. Then, only the system operators who have entered another password will be able to kill it.

N=NULLS: SET 0 TO 9 AS REQUIRED

Some terminals (notably the Texas Instruments Silent 700 series) require nulls to be sent at the beginning of each line, to allow the carriage time to return. This command sets the nulls to any value (0 to 9) necessary. Note that use of Control-N any time the system is typing something to you, will set the nulls to 5. Thus, using Control-N will allow the welcome and bulletin messages to be printed properly until the N (null) command can be used.

P=PROMPT BELL

The P command allows the prompting bell (normally sounded every time the system wants input) to be turned off or back on. It operates as a toggle and sets the bell response into whatever mode is not current.

R=RETRIEVE MESSAGE

To retrieve a message, you must know the message number, which is typically found through the use of the S (summary) command.

S=SUMMARIZE MESSAGES

The S command prints a 2-line summary of each message from a starting message number. Control-C cancels the printing of a particular summary, going on to the next. Control-K kills the printing entirely, returning to the function menu.

The summary function asks you for a starting message number. It then types the summaries for all active messages from that number on, stopping when there are no more. If you want to scan the summaries for a particular value, you can do so on the following fields: Date, From, To, or Subject. Any character string can be searched for. For example, to retrieve all summaries which contain CACHE (our local computer club) in the Subject, type:

```
        S;nnn,t=CACHE
```

which will scan the SUMMARY file, starting at message NNN, looking for the character string CACHE. All fields which can be scanned are stored in upper case only, and all scan requests are translated to upper case.

You can also use this summary scan to see if there are any messages to you. For example:

```
        S;nnn,t=ward
```

checks if I have any messages. Other typical scans are:

```
        s=pcnet  (any messages relating to PCNET).
        s=sale   (find things for sale).
        f=oper   (messages from system operators).
```

W=TYPE WELCOME

Since the WELCOME message gives a list of all control characters which can be used, you might want to review it, to check them again. The W command will retype the WELCOME message.

X=EXPERT USER

This command specifies the expert user mode to the computer. All prompt messages are shortened and, in general, the system is much more pleasant to use by the frequent user.

Accessing the system

If you would like to try out the system, all you need is a 110- or 300-bps ASCII terminal and a modem, such as the type used to communicate with a time-sharing system. Dial the system at (312) 528-7141. When you hear the answer tone, connect your modem, and press RETURN several times until the system determines your transmission rate. If you have a reasonably good telephone line, and a moderately strong modem signal, you should be able to communicate with the system. We find that, if the conditions are marginal, communications are possible, but such functions as Control-C and Control-S do not necessarily work. Feel free to hang up and try several times, if you have problems.

References

DMA Inc is located at 530 Pierce Av, Dyer IN 46311.

CACHE is at POB 52, South Holland IL 60473.

EOF