THE UO-11 DCE MESSAGE STORE-AND-FORWARD SYSTEM

Harold E. Price, NK6K
1211 Ford Ave
Redondo Beach, CA 90278

Jeff Ward, GO/K8KA
Dept. of Electrical Engineering
University of Surrey
Guildford, Surrey GU2 5XH
England

Abstract

The Digital Communications Experiment (DCE) onboard the UoSAT-Oscar-11 spacecraft recently began a new phase of regular operations. Development and installation of enhanced store-and-forward message transfer software (MSG2) - capable of 200-kbytes transatlantic data transfer per day - is the second plateau in the DCE experimental program. This program is designed to gain experience with computer-based message systems in low earth orbit.

The DCE is the first orbiting store-and-forward device to carry general amateur traffic on a continuing basis. The drafts for this paper were developed and edited by the collaborators in the USA and the UK using the spacecraft as the only means of communications.

This paper provides information on the capabilities and the design of this system as well as some background information on the UoSAT-OSCAR 11 spacecraft.

1.0 BACKGROUND

The UO-11 spacecraft, also known as UoSAT-2, was designed and built at the University of Surrey in England during the second half of 1983. It was known as UoSAT-B until its launch from Vandenberg Air Force Base near Lompoc California in March, 1984.

The possibility of flying a small store-and-forward message experiment onboard UO-11 was first discussed at a PACSAT design meeting in July 1983. Amateur groups in Dallas, Los Angeles, Ottawa, and Tucson immediately began work. A flight ready unit was turned over to the integration team at Surrey five months later, A partial account of this whirlwind development can be found in AMSAT's "Orbit" magazine number 18, March/April 84.

1.1 The Spacecraft

The uo-11 spacecraft is a cuboid with dimensions of 35.5cm x 35.5cm x 58.5cm. There are solar cells on the four long faces, generating a total of 35 watts of power, which is stored in a 6-amp-hour, NiCd battery. Included in the complement of experiments are:

- CCD camera with 384 x 256 pixel image capability with 128 levels of gray scale.
- Three particle detectors (Geiger counters) and multi-channel electron spectrometer.
- Space dust (micrometeorite) detector.
- Magnetometer - to measure magnetic field and to determine spacecraft attitude.
- 192kbytes CMOS memory for storage of CCD camera and particle/wave data.

The spacecraft has three downlinks: 145.825, 435.025, and 2401.5 MHz. The 145 MHz downlink is usually on. The 435 MHz downlink is now regularly used for DCE operations. The 2.4 Ghz downlink is rarely used.

The main spacecraft control computer - the On Board Computer (OBC) - is based on an 1802 microprocessor with 48kbytes of static RAM.

1.2 DCE Hardware

The Digital Communications Experiment (DCE) is an important experiment on uo-11 -- establishing that store-and-forward communications in low-earth orbit is realizable and thus fulfilling one of the UO-11 mission objectives. The major goal of the DCE is to provide a software and hardware testbed for PACSAT-type store-and-forward devices. To that end, it was designed to be as flexible as possible. The DCE is all CMOS and contains:

- An NSC-800 CMOS microprocessor using the 280 instruction set.
- Two 2 Harris HD-6402 UARTs.
- One 82C55 parallel port
- 14k of 2kx8 static RAM, Harris 6516.
- 16k of Harris 6564 16kx4 static RAM, using 12 bits to store 8 with single bit error detection and correction in hardware.
- 64k of 8kx8 static RAM, Hitachi 6264LP.
- 32k of 2kx8 static RAM, Hitachi 6116L.

o 512 bytes of Harris 6641 bootload PROM. This PROM has an identical, command selectable backup device.

o command selectable clock speeds of .9 and 1.8 MHz.

The DCE resides on three circuit boards which fit in a standard UoSAT module box approximately 6" x 9" x 1". It draws 120ma at +5 volts.

The total DCE memory capacity is 126kbytes, bringing the total memory aboard UO-11 to 366k -- far exceeding the total memory previously flown by amateur spacecraft.

1.3 Early DCE Operations

UO-11 DCE began its orbital operations on June 5, 1984. Initially, it supported spacecraft operations -- this unplanned activity made necessary by the post-launch failure in an uplink data communications path. UoSAT spacecraft have considerable redundancy in this area, and the problem could be bypassed by routing all VHF spacecraft communication through either the DCE computer or the spacecraft's main onboard computer (OBC). The DCE provided this bypass function in the initial months of spacecraft operations. Since that time, the software in both computers has matured sufficiently to perform the command bypass function while carrying out their other duties. The OBC carries out autonomous operational control of the spacecraft and its experiments, and also automatically determines and adjusts the satellite's attitude. The DCE is dedicated to the message store-and-forward function.

A prototype message system, developed by Hugh Pett, VE3FLL, was used for a demonstration of low earth orbit store-and-forward capabilities at the Pacific Telecommunication Conference in Hawaii in January 1985. The demonstration was done by Hugh and Larry Kayser, WA3ZIA, with support by Harold Price, NK6K and Chris Wachs, WA2KDL in Los Angeles; and Martin Sweeting, G3YJO and the UoSAT team in Surrey.

1.4 Current DCE Activities.

MSG2, the current DCE software, was developed in November 1985 by Harold Price, NK6K, and Jeff Ward, K8KA, at the University of Surrey's UoSAT laboratory. Assistance in implementing the MSG2 ground segment on the BBC micro was provided at UoS by Michael Meerman, PA3BHF. The spacecraft automated control software, DIARY, which also permits DCE ground stations to command the downlinks and multiplexors, was written by Steve Holder (UoS).

2.0 MSG2 SOFTWARE

MSG2 supports the following features:

o Stores up to 96k bytes of message data.

o A single message can be up to 16k bytes.

o Up to 128 messages may be stored at one time.

o A partially downlinked message can be continued at a later time without repeating parts of the message already received.

o A partially uplinked message can be continued at a later time without retransmitting parts of the message already sent.

o If a ground station looses positive control, the DCE will automatically revert to a known state after 2 minutes. The spacecraft DIARY program will also return the downlinks and data multiplexors to a known state after 15 minutes.

o The MSG2 protocol provides complete data transparency.

o The ground station's transmit/receive changeover time is not a factor in communications. The ground station can be full duplex, half duplex using computer controlled (fast) switching, or half duplex using human (slow) switching.

Restrictions in this version:

o Data transfer is in one direction at a time. Acknowledgments can be full duplex.

o Only one ground station can interact with the DCE at one time. The MSG2 software provides the means to keep ground stations from accidentally violating the restriction.

o The integrity of message data stored is not currently guaranteed as the message storage area of the DCE memory is not protected against externally induced errors. The program and non-message data are protected by hardware.

These restrictions may be lifted in later implementations.

MSG2 consists of three elements, a protocol specification, the MSG2 software running on the DCE, and several implementations of software for various computers which implement the MSG2 protocol for ground users.

2.1 MSG2 Protocol

The MSG2 protocol was design primarily to be easy to implement. Its only other goal was to provide the minimal message handling capability to PUT a message on the DCE, to GET one back, and to KILL a message no longer required.

As "easy to implement" dictated a single user approach, a LOGON and LOGOFF capability was added to keep two or more ground stations from starting message transfer operations simultaneously.

Experimentation with minimal ground stations is planned; the MSG2 protocol was designed to accommodate this activity. Messages are broken into small (64 byte) blocks with CRC error detection. Once a message transfer is begun, message blocks can be acknowledged at any time, and in any quantity. This allows a battery powered station to reduce its transmissions by requiring only one acknowledgment for a message of any arbitrary number of blocks. Unacknowledged blocks are retransmitted in a "round robin" fashion.

DCE blocks are acknowledged by sending a bit map frame. The bit map contains one bit for each block in a message. Bits set to 1 represent unacknowledged blocks, and 0s represent acknowledged blocks (Fig 1). The transmitting station continues to send the blocks indicated by 1 bits, until a bitmap is received with all bits set to 0.

Figure 1. -- Example of an MSG2 Bit Map

```
7 6 5 4   3 2 1 0     (1)

0 0 1 0 1 0 0 0       (2)

0 1 2 3 4 5 6 7       (3)
```

(1) numbering of bits in bit map (MSB is 7)
(2) bit map ack'ing all but blocks 2 and 4
(3) blocks represented by bit map bits

2.2 MSG2 Frame Format

This section is not meant to provide a formal MSG2 protocol specification, but to outline the structure of the protocol and the frames used by it. Frame types may be added or removed as the protocol matures.

Although there are several types of frames, they all share the following format:

<10h><03h><cmd><cmd not><data length><data><crc>

Each byte is sent as an asynchronous character with 8 data bits and no parity bit. Frames are preceded by several SYN bytes <16h> for modem and timing synchronization.

Frame breakdown:

<cmd> -- A single ASCII character specifying a DCE command.

<cmd not> -- The inverse of <cmd>. This byte can be calculated by <CMD> XOR FFh or by 255 minus <cmd>.

<data length> -- A single byte giving the length of the <data> portion, in bytes. Data length is between 0 and 128 bytes.

<data> -- <data length> bytes of data. This data can be either ASCII characters or binary bytes.

<crc> -- Two bytes of cyclic redundancy check. The CRC is a type of checksum, and it covers everything from <cmd> to the end of <data>, inclusive.

In order to assure that <10h><03h>, the beginning of frame marker, does not get transmitted in the frame, all <10h> bytes other than the one at the beginning of a frame are doubled. That is, during transmission, <10h> is converted to <10h><10h>. When receiving a frame, after the first <10h><03h> has been detected, all <10h><10h> sequences should be converted to a single <10h>. If a non-doubled <10h> is encountered in a frame, it is an error.

2.3 MSG2 CRC

Every frame transmitted by the MSG2 ends with a two-byte Cyclic Redundancy Check (CRC). The CRC is an error detection code, and if you use the CRC equation on a received frame, your two-byte answer should match the two bytes transmitted at the end of the frame. The CRC used by MSG2 is calculated using a modified CCITT CRC algorithm. A Z80 machine-language program showing how this is done is provided in the appendix.

The CRC calculation includes all bytes from <cmd> to the end of <data>. The CRC calculation is done prior to doubling <10h> bytes and, by the receiver, after removing the extra <10h>.

2.4 Title Frames

The DCE was required to "do something interesting" when it was idle, i.e. not performing a function at the specific request of a ground station. To this end, MSG2 sends the first line of each active message on the downlink when it is idle. This line is the message "title" and usually contains at least the source, destination and subject of the message. Ground stations can see if they have any waiting traffic without interacting with the DCE by simply copying these title blocks. The OBC DIARY program currently switches the DCE onto the downlink for 30 seconds at roughly 5 minute intervals.

Title frames provide a way for stations not directly involved in DCE operations to monitor DCE activity. The <cmd> byte in a title frame is "T". The contents of the <data> portion of a title frame are are as follows:

Message number, 1 byte. If the first bit of this byte is set, the message is not complete, and the message title may be invalid. Message numbers for complete messages run from 0 to 127.

Message length, 1 byte. This is the length of the message that is stored on the DCE, it is not the length of this title frame. Multiply this by 64 to get the message length in bytes.

Call sign of station using DCE, 9 bytes of ASCII. If no one is using the DCE then this will be 9 blanks.

Title of the message, the remaining <length> minus 11 bytes of the <data> field. This is taken from the first line of the message. The length referred to above is the FRAME LENGTH (which follows the inverted command). The 11 accounts for the message number, message length and call sign data.

The title for message number 0 contains MSG2 administrative and status information. It currently contains MSG2 version number, a counter from the error detection and correction (EDAC) memory, the number of free memory blocks available, the number that will be assigned to the next message, a counter that is incremented every time MSG2 receives a valid frame, an error indicator and an indication of which bank of RAM is active. Message 0 itself is used to download portions of the program variables, including a table of memory address where the EDAC circuits have corrected an error.

2.5 Other Frames

The above information and a short computer program will allow causal ground observes to monitor DCE activity. During actual DCE operations, however, several other frame types are used. The following command frames are used by DCE ground stations, and the list provides insight into the operation of the MSG2 mailbox.

LOGIN tells the DCE the call sign of the ground station.

LOGOUT frees the DCE for use by another ground station. Logout is automatic if the DCE does not hear the ground station for two minutes.

PUT is used by the ground station to store a message to the DCE.

CONTINUE allows the ground station to continue (on another orbit) a PUT operation that was interrupted by LOS.

GET is used to retrieve a message from the DCE.

KILL deletes a message.

END resets DCE software to the title-display mode, without logging out the ground station.

Thus, the DCE has all of the commands needed in a computer bulletin-board system.

3.0 DCE SOFTWARE

The DCE MSG2 software is implemented in 280 assembler code. It is in assembler for both size and speed, the DCE MSG2 runs in 2.5k and supports full duplex operations at 1200 baud on a 280 with a .9MHz clock.

The program resides in the memory protected by hardware EDAC. Messages are stored in 96k of non-protected memory. This memory is mapped into the upper 32k of the 280 memory space. There are two 32k banks and two 16k banks. The banked memory is organiied as a linked list of 256-byte blocks. All of the banked memory is used except for the block in bank three containing location A5Flh, which has a bit that went bad shortly before launch. Messages consist of a linked list of memory blocks, unused blocks are kept on a free list.

All of the link pointers for the memory blocks are kept in the memory protected by hardware EDAC. Currently, blocks from deleted messages are returned to the top of the free list where they will be the next to be re-allocated. This tends to kept bank 1 in use and bank 4 empty. This will be changed in a future version.

The banked memory is not currently protected against charged-particle induced soft errors. Algorithms are being developed to provide this memory with software EDAC in the future. In 60 days of monitoring, we have only seen three errors in the 16k of hardware-protected memory. It is hard to draw conclusions from this limited data, and the memory technology is not the same in the banked memory. Experiments are planned to gather data on soft errors in all parts of memory.

3.1 Other MSG2 Functions

The MSG2 software on the DCE supports two non-message related functions.

BYPASS - The DCE sends all characters received through the VHF uplink to the UART that leads to the spacecraft command system. This provides backup to the similar function performed by the DIARY program on the 1802 computer in case of 1802 failure or the need to completely reload 1802 software. The BYPASS software resides in the receive interrupt handler, and therefore should perform the bypass function with high reliability no matter what the other levels of MSG2 software are up to. There is no way to disable the BYPASS. The BYPASS was made necessary by a failure in the VHF uplink facility shortly after launch.

MEMWASH - This function washes any accumulated single-bit errors out of the hardware EDAC memory by reading a location and writing it back. A different memory byte is fetched and stored once on each trip through the main loop of the software, which occurs at least once every 9.2ms. All of the EDAC memory is checked at least once every 150 seconds, this number might be 10 to 100 times faster, depending on uplink activity.

A third function under development is the gathering of information on soft memory errors in the unprotected areas of memory. This is to further the DCE's role in gathering engineering data used for PACSAT development as well as to provide a high level of data integrity for messages stored onboard the DCE.

4.0 MSG2 GROUND STATION SOFTWARE

The MSG2 protocol provides a small number of basic features: logon, logoff, put, get, and kill. The ground station software supplies additional "user friendly" facilities. Such features include remembering the start point for partial messages, providing wildcard or multiple file transfer, automatic logging, and providing antenna pointing cues; these features are not part of the basic set of functions, but make the DCE easier to use. The features available at any particular ground station depend on that station's hardware.

5.0 MESSAGE FORMATS

MSG2 is a data-transparent system, i.e. messages are stored as a single string of 8 bit bytes. Message content does not effect and is not effected by communication through MSG2. Most messages, however, will follow a fixed format for their first line. The first line is defined as the text up to the first <cr>, or 116 characters. This is the part of the message that is sent on the downlink in title blocks.

5.1 Person-to-Person Messages

The following format is used for standard messages:

To:<call> De:<call> Re:<title>

The call can be up to 9 characters. There are no spaces after the colon in any field.

For example:

To:GO/K8KA De:NK6K Re:Software updates

The To: and De: fields are the call signs of DCE ground stations. A future command in MSG2 will permit messages in this format to be searched by To: field and downlinked in a group. The format is flexible, and fields may be added to it if the DCE is used for other than direct ground station to ground station data transfer.

6.0 THE FUTURE OF THE DCE

Several hundred kbytes of data have traveled between UoSAT headquarters in Surrey (UK) and NK6K in Los Angeles (USA) on the DCE. MSG2 ground station and satellite software works efficiently and reliably. While much of the future use of DCE store-and-forward capability depends on radio-regulatory matters beyond our control, the DCE has successfully proven that store-and-forward communications using a satellite in low-earth orbit can be carried out routinely. It has put us in a position to make informed design decisions while working an a proposed dedicated store-and-forward spacecraft.

The limited memory available on UO-11 and the fact that DCE activities consume bandwidth on the UoSAT-11 command uplink and general downlink dictate that only a limited number of selected ground stations will take part in future DCE communications. These ground stations will be chosen such that, regulations permitting, each can serve as a gateway -- delivering amateur radio news and and technical information to stations outside of the DCE ground station network.

Equipment for an East Coast North America gateway is in place and should be operational by the time this sees print, A station will be brought on the air soon in New Zealand. Discussions are under way for stations in Australia and Japan. The DCE is ready now to serve as an effective link between the amateur radio service's far flung packet radio networks.

7.0 SUMMARY

The DCE project was begun as an opportunity to gain experience in the design, construction, and orbital operation of space-based large-memory store-and-forward message relay satellites. The parts used in its construction are giving us data on the suitability of high density non-specialized memory devices and microprocessors (i.e. inexpensive) in low earth orbit, which is directly applicable to future amateur space projects. The coming months will give us experience scheduling gateway operations to maximize data volume in a worldwide network of ground stations.

The capability exists now to move 114k bytes per day to or from a single ground station.

The experience gained from writing MSG2

software, managing DCE operations, and dealing with regulatory issues will be of great use when the mailbox on JAS-1 becomes operational and when design teams set to work on PACSAT -- an amateur radio satellite dedicated to store-and-forward communications.

Appendix 1 - Acknowledgments

Here is a list of the North American crew who worked on the design and construction of the DCE, in no particular order:

Stan Kazmiruk, VE3JBA; Dick Atkinson, VE3JBO; Gord Scale, Bob Gillies, VE3JA; Hugh Pett, VE3FLL; Geoff Clarke, VE3JBD; Dale Ward, George Roach, VE3BNO; Grant Bechthold, VE3JBF; John Henry, VE2VQ; Ron Archer, VE3CNM; Murray Gold, VE3KHG; Larry Kayser, WA3ZIA; Dave Cheek, WA5MWD; Chuck Green, NOADI; Lyle Johnson, WA7GXD; Harold Price, NK6K; Bill Reed, WDOETZ; Jose Sancho, WB5YFU; Bob Stricklin, N5BRG. Richard MacBeth, G8VLY, assisted in the final close-out at UoS. Funding for DCE hardware was provided by Volunteers in Technical Assistance as part of their support of the PACSAT project. Gary Garriott, WA9FMQ is the VITA interface. Acknowledgment is given to AMSAT for its continued support, and to the UoS team.

Additional information on the DCE hardware, design, and construction can be found in a paper by Lyle Johnson, WA7GXD, "The OSCAR 11 Packet Experiment", Proceedings of the Third ARRL Amateur Radio Computer Networking Conference.

Appendix 2 - DCE CRC algorithm.

The routine below can be used to compute the checksum for reception of DCE frames. The HL register is cleared before the first byte is received. Each byte is acted on in turn. When all bytes have been checksummed, the result is compared against the received checksum. The L register contains the first byte received, the H register the second.

```
,
; COMPUTE CRC ON A, INTO HL
;
CKSUM:
        LD      B,8
        LD      C,A
CRC2:
        LD      A,C
        RLCA
        LD      C,A
        LD      A,L
        RLA
        LD      L,A
        LD      A,H
        RLA
        LD      H,A
        JR      NC,CRC4
        LD      A,H
        XOR     10H
        LD      H,A
        LD      A,L
        XOR     21H
        LD      L,A
CRC4:
        DEC     B
        JR      NZ,CRC2
        RET
```

In using this program on DCE frames, remember that the CRC covers all bytes from the <cmd> to the end of the <data> segment, inclusive,' It does not include the CRC itself, or the leading <10h><03h> bytes. Also, CRC calculation is done prior to doubling <10h> bytes and, by the receiver, after removing the extra <10h>. To check your CRC program, CRC check the characters "TEST MESSAGE". The result should be CRC bytes L=253 and H=223. 253 would be the byte transmitted or received first.