AAEC/E433

AAEC/E433

AUSTRALIAN ATOMIC ENERGY COMMISSION

RESEARCH ESTABLISHMENT

LUCAS HEIGHTS

SMUT - SERIAL MULTIPLE-USER TERMINALS SYSTEM

by

P.J. ELLIS

December 1977

ISBN 0 642 59634 4

AUSTRALIAN ATOMIC ENERGY COMMISSION
RESEARCH ESTABLISHMENT
LUCAS HEIGHTS

SMUT - SERIAL MULTIPLE-USER TERMINALS SYSTEM

by

P.J. ELLIS

ABSTRACT

A serial data communications network capable of supporting up to 64 send-receive data terminals is described. Each terminal is serviced on a polling basis by a control unit that is interfaced to a minicomputer which has access to the AAEC Multiple User Dataway and its facilities. This serial link was developed to relieve the Dataway of the mundane servicing of slow, byte-oriented, input-output devices.

National Library of Australia card number and ISBN 0 642 59634 4

The following descriptors have been selected from the INIS Thesaurus to describe the subject content of this report for information retrieval purposes. For further details please refer to IAEA-INIS-12 (INIS: Manual for Indexing) and IAEA-INIS-13 (INIS: Thesaurus) published in Vienna by the International Atomic Energy Agency.

DATA TRANSMISSION; COMMUNICATIONS; COMPUTER NETWORKS; COMPUTERS; EQUIPMENT INTERFACES

CONTENTS

		Page
1.	INTRODUCTION	1
2.	IMPLEMENTATION	1
	2.1 General	1
	2.2 Transmission	2
3.	OPERATION	4
	3.1 Function	4
	3.2 Polling	5
	3.3 Character Buffering	6
4.	SMUT SEQUENCES	7
	4.1 Controller Write Sequence	7
	4.2 Controller Read Sequence	8
	4.3 Bit Assignments	8
	4.4 Frame Bit Positions	11
5.	OPERATING EXPERIENCE	11
	5.1 Effects of Dropouts	11
	5.2 Controller Efficiency	12
	5.3 System Lockup	13
	5.4 Smart Polling	14
6.	PROPOSED BURST MODE	15
	6.1 Operation	16
	6.2 Definition	17
	6.3 Block Structure	17
7.	CONCLUSION	18
8.	ACKNOWLEDGEMENTS	18
9.	REFERENCES	18

GLOSSARY

baud	-	*bit per second defining information transfer rate.*
byte	-	*eight bits.*
controller	-	*the computer and its interface to the SMUT network, i.e. the network controlling station.*
frame	-	*the transmission block of start, address, status, data and stop bits.*
operator	-	*the person using the terminal.*
poll	-	*a transmitted frame from the master station to a station which governs the station's 'right to send'.*
reply frame	-	*the response by the station to a poll.*
station	-	*the terminal and its drive unit, particularly the drive unit.*
terminal	-	*the input-output device that interfaces to the user, normally a teleprinter, decwriter or video display.*

1. INTRODUCTION

The Serial Multiple User Terminals (SMUT) system has followed as a necessary expansion of the AAEC computer-computer link or Multiple User Dataway. The system was designed to relieve the Dataway of the ever-increasing traffic from single-byte oriented, input-output devices, because these consume an equivalent overhead in address, control and status facilities per character as computer-use per block.

SMUT has been implemented with the capability of operating many input-output devices using a single unlooped transmission line. The operations on this line are controlled on a polling basis by a minicomputer which is connected to the Dataway [Ellis 1970]. This connection allows the minicomputer, and consequently any of the SMUT stations, to use the larger facilities of the main IBM360 computer [Richardson 1969].

Originally, SMUT was developed to transmit data on a demand basis to transfer either words or bytes between 16-bit computers such as the PDP11. However, owing to the mushrooming of demand for the system by users of ACL-NOVA [Bennett & Sanger 1973; Sanger 1971] and the subsequent development of conversational software for the IBM360 computer [Backstrom 1975], SMUT has been constrained to operate with a NOVA820 minicomputer as the controlling station, every station operating as a unit-byte device with a continually updated status byte. This constraint was accepted as a reasonable compromise between hardware and software costs,and convenience for the terminal user.

2. IMPLEMENTATION

2.1 General

The SMUT system consists of a series of stations,each comprising a terminal, usually a teleprinter operating at 110 baud or a decwriter operating at 300 baud, and a drive unit. These stations are connected to a single coaxial cable which extends approximately 1200 m excluding stubs. The cable interconnects about ten buildings at the AAEC Research Establishment with 13 junction boxes (entry points) at convenient locations. The cable is terminated at the two extremities and stubs are unterminated. Each entry point can be extended to connect multiple stations or extended several hundred metres before reflections cause dropouts.

The SMUT system consists of a complex controller which polls a series of stations in a regular sequence. Each station has a unique 7-bit address and will respond only to that address. The station communicates to the controller with a reply frame immediately it has been polled, the reply address being the station address with the controller bit asserted. A transmission consists of address, status and data information, each using one 8-bit byte. The address byte is composed of a write bit, a controller bit and an effective 6-bit address which permits 64 stations. The status byte includes reader and writer busy, channel end and device end with two spare bits. The data byte is generally an ANSCII character with either a parity bit or a mark bit according to the input-output device.

The actual transmission frame consists of a start bit, an address byte, a status byte, a data byte, an address complemented byte and two stop bits. The address byte and the address complemented byte are compared by the receiver as a transmission safeguard. The transmission frame is 35 bits long and is sent at 100 kbaud. Each station will transmit a reply frame when a valid poll is received, thus taking approximately 700 μs for a poll and reply. The controller operates at 1 kHz which allows for some programming overlap.

2.2 Transmission

2.2.1 Cable type

The SMUT cable is nominally 70 Ω impedance which was chosen because 50 Ω was too low and the 70 Ω was available in sufficient length for testing (~ 1 km). After initial testing of the transmitter-receiver pair, the cable was regarded as satisfactory and was specified before being laid. Three cables were laid, two for use and the other for a spare in case of extensive breakdown. The spare was also intended to be used for television monitor signals when and if necessary, but a need for this has not eventuated. The cables are joined and tapped at the entry points. The entry points use standard 50 Ω connectors and no special care has been taken to keep impedances constant.

2.2.2 Terminations

The SMUT cable is terminated at the two extremities with 2 Zo (150 Ω) pullup resistors. The receivers switch at a constant threshold, so these terminators should be kept as high as possible to minimise the effective series resistance or attenuation of the cable. The terminator should ideally be kept close to Zo (70 Ω) to minimise reflections but, because the series resistance is about 70 to 100 Ω, terminating into

70 Ω results in too much attenuation at the remotest stations. The ideal of a correctly terminated system is not feasible, as a permanent stub of 200 to 300 m is used to service particular locations.

2.2.3 Cable signals

The cable signal convention is normally high for zero (space) and low for one (mark). The original design allowed for 24 V as zero and ground as one, but this was found to be unnecessary, as the 10 V signal provided by using the unregulated supply for the 5 V logic was sufficient. This resulted in a moderate cost advantage over the 24 V system with slightly less noise immunity.

The driving circuitry for the SMUT transmitters was tested under several conditions to ensure that transmission could be guaranteed over the proposed network. The final drivers are constructed from discrete components and have pull-up and pull-down capabilities. The pull-up drive is required to stop excessive deterioration of the signal due to reflections, *etc*. The pull-up drive is transient and does not apply to stations not currently sending.

2.2.4 Transmission speed

SMUT operates at 10 μs per bit (100 kbaud) which is about the maximum speed that can be achieved without a lot of care and inflexibility. The main reasons for limiting the speed to 100 kbaud are:

(i) The baud rate is sufficient to operate about 32 terminals at a fairly reasonable rate without overloading the controlling computer with frequent polling and response interrupts. The computer is still required to execute arithmetic statements for ACL-NOVA programs as well as handle Dataway activity and transfers. This is not a real problem with the NOVA820 (800 ns cycle) but it causes significant timing difficulties when a similar system (Mk II) is operated using a slower NOVA (2.6 μs cycle).

(ii) If the SMUT transmission is increased in speed to 500 kbaud, the error rate in a laboratory environment is less than 1 in 10^6. However, if 1000 m of cable is used with a 100 m stub at a central point, the reflections due to the unterminated stub degrade the signal sufficiently to make any further increase in speed unfeasible. Any increase in speed to about 1 Mbaud is easily attainable in a well-terminated environment with stubs kept to a minimum.

(iii) The signal level for the SMUT serial code is 10 V maximum, so that earth-loops and variations in zero potential can affect performance significantly. These variations can be minimised, where necessary, using standard opto-isolators which have switching risetimes of 1 to 2 μs and can be used up to about 200 kbaud without significant distortion. At present, SMUT has not required isolation at any of the entry points. However, any increase in transmission rate precludes a simple isolation technique.

3. OPERATION

3.1 Function

The types of data transfer at the AAEC Research Establishment fall into three generalised areas:

(a) Batch applications, such as remote job entry and retrieval stations, which are typified by relatively large quantities of input and output data per job. In these applications, the control overheads must be kept as low as possible but still consistent with the requirement that error detection and recovery are good enough to proceed without close attention by an operator.

(b) Conversational computing, where the traffic is balanced and operates on a character by character basis from an operator-controlled terminal. Most messages are of short duration but the response times have to be minimal. The error detection and recovery can be controlled within reason by the terminal user.

(c) Inquiry applications which consist of quite short input messages and relatively long output messages. The inquiry input has much the same constraints as the conversational area and is on a slow character basis. The output, however, can be quite large and must be as fast and error-free as possible with automatic error recovery.

The batch application area of data transfer is catered for quite adequately by the faster, parallel Dataway link. This has sufficient error checking facilities for block transfers required by remote card-readers and line-printers. When data errors or sequence errors occur, the transfer can be repeated, or recovery procedures undertaken, by the computer operating system.

The conversational area is serviced by SMUT. Here, the input messages are at slow typing speeds. Errors, when they occur, can be deleted and retyped in most instances.

The inquiry area is currently being serviced by SMUT with a limitation on output speed, or by faster video displays on the Dataway which cause high line occupancy.

3.2 Polling

Various methods were considered to enable the SMUT system to be completely asynchronous, *i.e.* to have stations sending characters whenever they are received and, likewise, requesting data whenever the printer becomes ready. Most of these methods become very intricate at the start-bit times for the serial transmission, mainly due to the large line delays involved ($\sim 5\ \mu s\ km^{-1}$), and do not effectively stop two sending stations from overlapping should they be coincident at transmission start. The problem becomes minimal if extra wires are used to establish priority, but this defeats the concept of one-line.

Polling was used to overcome the possibility of overlapped transmissions. This increases the complexity of the controller but enables a much simpler design for the station. As one controller is required for about 30 stations, an incidental economy is realised. However, polling the stations results in slightly slower responses for the input-output terminals, and increases the programming overhead for the controlling computer.

As a station will only respond to a unique address, the address byte on the reply frame is not necessary. Using one bit of the address to specify station or controller-return address, enables the transmission frame to be kept constant, which greatly simplifies receiver-transmitter design. The controller should only accept a reply frame if it receives the expected return address but, in practice, the redundancy in the code ensures that the reply address will be correct. This also makes it possible to separate the transmitter and receiver functions in the controller algorithms.

3.2.1 Terminal speeds

The maximum speed at which the terminal can operate depends upon its baud rate. This speed can be achieved only if the next character is available immediately the current character is complete. This requires instant attention from the controlling device whenever the printer becomes ready or whenever the keyboard receives a new character.

In a polled environment, a terminal will become ready but will have to idle until the next poll to the station is received. The maximum speed that the terminal can operate is governed by how often the station is polled. When the poll is at a constant rate, as for a simple control program, the minimum period per character is the smallest multiple of the poll period that is greater than the minimum period attainable by the terminal with immediate attention.

An example of the above character speeds is as follows (as at 5th April 1976):

If the polling computer polls 27 terminals at 1 kHz then a teleprinter will operate at 9.2 characters/second on a print stream. Minimum period = 27*4, *i.e.* 108 ms per character (cf. 100 ms). A decwriter will operate at 18.5 characters/second on a print stream. Minimum period = 27*2, *i.e.* 54 ms per character (cf. 36.6 ms).

3.2.2 Priority polling

In the conversational environment, an operator expects a correct character to be echoed immediately and an incorrect character to produce no output or an error message. Any delays in printing a valid character tend to entice the operator to have another try. When a station is polled sequentially, an input character is received on one poll and the output character is sent on to the next poll to the station. The added delay is minimal, but when it is grouped with the idle time, the input poll and the print time, then the teleprinter appears to be irregular, slow to respond and very difficult to type on. This problem is reduced by polling the station immediately the character has been received when it is valid and then continuing the sequential polling sequence. Similarly, when a printer ready (SMUT reader ready) is received, then an immediate poll increases the printing speed. The priority poll has very little effect on the other stations but makes a large difference in operator convenience.

3.3 Character Buffering

Because of the delay between a character being received and the station being polled, there is a possibility that another character can start to be received before the current character has been read. When this occurs, a non-buffered character can be partially cleared or garbled by the receiver accepting the incoming character or, conversely, by the incoming character being ignored or garbled. This problem can be accentuated by using the keyboard repeat feature. When a character is buffered as soon as it is received, fast access is not necessary and the polling rate can drop to about two polls per character before over-run occurs instead of requiring immediate service.

A similar situation occurs when characters are to be printed at the maximum possible rate. When an output character is not buffered, the maximum rate will depend upon the poll speed (see Section 3.2.1). However, when the output character is buffered in the station, the terminal can operate at its maximum printing speed provided sufficient polls are received to keep the secondary buffer full whenever the transmitting buffer completes a character.

With the development of Universal Asynchronous Receiver-Transmitters (UART), these buffering techniques are currently available 'on-chip', and more recent SMUT stations use such devices. This results in a significant increase in output speed but does not decrease the polling delays on input. However, the buffering enables more effective control of the keyboard,especially when the teleprinter paper-tape reader is being used.

4. SMUT SEQUENCES

The two basic sequences used by SMUT are 'write to a station' and 'read from a station'. All other status responses are covered by these two fundamental sequences because they completely define the scope of a station. The write and read sequences have been made as symmetrical as possible so that a station could communicate with other stations if required. This usage has not been implemented in the current system but was required as a feature before polling by a controller was introduced. The sequences operated on a 'hand-shake' basis wherever practicable so that a fixed response is expected for each operation. This has been used to try to overcome the effects of lost or garbled transmissions.

4.1 Controller Write Sequence

When the controller writes to a particular station, the sequence is commenced by the controller asserting the write address bit and setting the data byte to the required character on the next poll to the station. If the station is not busy, the write will be accepted, the station will reply CERO (Channel End Reader Out) on the reply frame and proceed to print the character.

On the next poll to the station, the controller will set CEWO (Channel End Writer Out) to acknowledge the channel end received on the previous poll. When the station receives the channel end, the CERO bit is dropped on the reply frame.

The write phase of the sequence is now complete. When the station reader becomes ready after printing the character, DERO (Device End Reader Out) will be sent as a return to some subsequent poll. The controller will acknowledge the device end on the next poll by replying CEWO which will drop the station DERO.

The above sequence specifies the controller write completely but in actual operation a shorter sequence has proved more satisfactory. The short sequence ignores the device end response and tries to write to the station on every poll. Whenever a CEWI (Channel End Writer In, *i.e.* CERO from the station) is received at the controller, the next character is sent on subsequent polls. The poll delays due to waiting for a ready response are avoided and approximately half the polls per character are required. The station will respond busy status to all write requests when the printer is not ready.

4.2 Controller Read Sequence

The read sequence is practically a mirror image of the write sequence except for the timing of the poll sequences. The sequence is initiated when the SMUT controller sets DERO when a character is expected or required by the operating system. When this bit is received at the station as DEWI (Device End Writer In), the papertape reader will be enabled (if present) and CEWO will be sent on the reply frame. BWO (Busy Writer Out) will be sent on all reply frames while the paper-tape reader is enabled *i.e.* until a character is received.

When the keyboard receiver has received a character from either a paper-tape reader or a keyboard, the station will idle until the next poll from the controller. The station will set the write address bit, DEWO (Device End Writer Out), and set the data byte to the received character on the reply frame.

If the controller will accept the character, CERO will be sent on the next poll to the station. When this bit is received, DEWO is dropped and the write request is cleared. This is indicated by sending CEWO on the reply frame.

4.3 Bit Assignments

4.3.1 Address

Bit 07 Set whenever the character in the data byte is to be written. This bit will generally coincide with the device-end reader in bit.

Bit 06 Set whenever the transmission is a return from the station. It may be set by the controller for diagnostic purposes to check a transmitter-receiver pair.

Bit 05 High order address bit. This bit is currently set to zero as no station above 31 is polled (as at 7th March 1977).

Bits 04-00 These low order address bits are currently used for polling the low address stations (0 to 31).

4.3.2 Status

Bit 07 Attention Writer In, Reader Out. This bit is currently not used and is unlikely to be used for this purpose. It is set to zero.

Bit 06 Busy Writer In, Reader Out. Indicator that the writer (keyboard) is receiving a character. Busy signal from the printer transmitter.

Bit 05 Channel End Writer In, Reader Out. When received by a station, this bit clears DEWO (Device End Writer Out) and sets CEWO (Channel End Writer Out). When this bit (CERO) is sent, it signals that the reader (printer) accepted the last character.

Bit 04 Device End Writer In, Reader Out. When this bit is received by the station, the next character is requested, *i.e.* the advance solenoid is actuated for a teleprinter paper-tape reader. When sent, DERO signals that the reader (printer) is ready for another character.

Bit 03 Attention Reader In, Writer Out. This bit is not used.

Bit 02 Busy Reader In, Writer Out. Indicator that the reader (printer) is printing the last character. Busy signal from the writer (keyboard receiver) which covers the period that the paper-tape reader would have been actuated.

Bit 01 Channel End Reader In, Writer Out. When received by the station this bit clears Device End Reader (printer ready) and acknowledges that channel end was received by the controller. When sent by the station, CEWO indicates that CEWI or DEWI was received from the controller.

Bit 00 Device End Reader In, Writer Out. Indicates that another character is ready for the reader (printer). DEWO indicates that a character has been received and is present in the data byte.

Bit	Write from Control to Station: Control function	Write from Control to Station: Station function	Status from Station to Control: Station function	Status from Station to Control: Control function
01	start bit = one	start bit = one	start bit = one	start bit = one
02	addr 07 = one	addr 07 = one	addr 07 = zero	addr 07 = zero
03	addr 06 = zero	addr 06 = zero	addr 06 = one	addr 06 = one
04	addr 05 = zero	addr 05 = zero	addr 05 = zero	addr 05 = zero
05	addr 04 = one	addr 04 = one	addr 04 = one	addr 04 = one
06	addr 03 = one	addr 03 = one	addr 03 = one	addr 03 = one
07	addr 02 = one	addr 02 = one	addr 02 = one	addr 02 = one
08	addr 01 = zero	addr 01 = zero	addr 01 = zero	addr 01 = zero
09	addr 00 = zero	addr 00 = zero	addr 00 = zero	addr 00 = zero
10	stat 07 = zero	stat 07 = zero	stat 07 = zero	stat 07 = zero
11	stat 06 = BRO	stat 06 = BWI	stat 06 = BRO	stat 06 = BWI
12	stat 05 = CERO	stat 05 = CEWI	stat 05 = CERO	stat 05 = CEWI
13	stat 04 = DERO	stat 04 = DEWI	stat 04 = DERO	stat 04 = DEWI
14	stat 03 = zero	stat 03 = zero	stat 03 = zero	stat 03 = zero
15	stat 02 = BWO	stat 02 = BRI	stat 02 = BWO	stat 02 = BRI
16	stat 01 = CEWO	stat 01 = CERI	stat 01 = CEWO	stat 01 = CERI
17	stat 00 = DEWO	stat 00 = DERI	stat 00 = DEWO	stat 00 = DERI
18	data 07 = one	data 07 = one	data 07 = one	data 07 = one
19	data 06 = one	data 06 = one	data 06 = one	data 06 = one
20	data 05 = zero	data 05 = zero	data 05 = zero	data 05 = zero
21	data 04 = zero	data 04 = zero	data 04 = zero	data 04 = zero
22	data 03 = zero	data 03 = zero	data 03 = zero	data 03 = zero
23	data 02 = zero	data 02 = zero	data 02 = zero	data 02 = zero
24	data 01 = zero	data 01 = zero	data 01 = zero	data 01 = zero
25	data 00 = one	data 00 = one	data 00 = one	data 00 = one
26	addc 07 = zero	addc 07 = zero	addc 07 = one	addc 07 = one
27	addc 06 = one	addc 06 = one	addc 06 = zero	addc 06 = zero
28	addc 05 = one	addc 05 = one	addc 05 = one	addc 05 = one
29	addc 04 = zero	addc 04 = zero	addc 04 = zero	addc 04 = zero
30	addc 03 = zero	addc 03 = zero	addc 03 = zero	addc 03 = zero
31	addc 02 = zero	addc 02 = zero	addc 02 = zero	addc 02 = zero
32	addc 01 = one	addc 01 = one	addc 01 = one	addc 01 = one
33	addc 00 = one	addc 00 = one	addc 00 = one	addc 00 = one
34	stop bit = zero	stop bit = zero	stop bit = zero	stop bit = zero
35	stop bit = zero	stop bit = zero	stop bit = zero	stop bit = zero

4.4 Frame Bit Positions

Two sample transmissions are listed to show bit positions within the frame and the function of each bit. The status bit functions at the controller and the station are complementary and are specified separately. The addr and addc bits are logical complements within each frame. The address used is '1C' hexadecimal and the character used is 'A' ('C1' hexadecimal).

5. OPERATING EXPERIENCE

There have been many observations made of the SMUT system, some of which have resulted in modifications to the original concept. The results of these have been integrated into the text at appropriate positions, since, from the operator's viewpoint, they are related to polling and terminal speeds. However, as the number of terminals has increased, faults due to cable length, reflections and traffic congestion have become more apparent.

5.1 Effects of Dropouts

The transmission frame consists of four basic bytes, the address byte, the status byte, the data byte and the address-complemented byte. The address bytes are compared when the transmission is received and, if the first byte does not match the complement of the second, then the complete transmission is rejected.

When SMUT was formulated, the transmission was address byte, address-complemented byte, status byte and data byte. When cable length was increased, the transmission could not be guaranteed to give reasonably error-free operation. The current format was adopted in the hope that, if the leading and trailing bytes could be guaranteed, then the central bytes would be more reliable. This proved to be the case although the original fault was traced to the controller power supply and not to the transmission mode.

The likely ways that incorrect performance can occur are either by the loss of the outward poll or reply, or by the incorrect reception of individual status or data bits. Much effort has been made to ensure correct transmission but dropouts will occur. When data bits are dropped, the echo will be incorrect. This can be deleted and retyped by the operator. When status bits are dropped the system, which operates in 'hand-shake' mode can usually cope. In a few known instances, the system may fail, particularly when the dropout is a channel end.

5.1.1 Loss of channel end on a write

When either the controller or the station sends a character using the short sequence (Section 4.1), the correct channel end response is vital,as the sender will keep sending the character until the channel end reply. If the controller write sequence is used as an example, the station will accept the write and then send channel end on the reply frame. If this reply frame is rejected owing to mismatch, or, the CEWI bit is lost, the controller will keep sending the character. On the next outward poll, when the station receives the next write, the channel end bit will not be sent, as the terminal is busy and the station will not accept the character. As the controller will keep sending the character, double printing will result. A classic example of this fault is where the station transmitter has failed and no reply frames are received. In this case, the first character sent to the station will be printed indefinitely. The other alternative is to keep the channel end active until it is cancelled but, as this can cause lost characters, it was considered the greater evil. A similar result occurs when the channel end is lost on the outward poll during the read sequence from the controller. In this case, the double print will be two valid input characters, both of which have been accepted.

5.1.2 Loss of device end

When the system is relying on the successful receipt of a device end, then a stall may result. For the operation of a terminal paper-tape reader, if the device end is dropped and then cancelled by a valid channel end, the station will not start the paper-tape reader. This particular fault has been minimised by changing the keyboard busy signal to cover the period from energising the solenoid until the next character is received. When the controller expects a character from a station that is flagged as not busy, then another device end is sent. The inverse problem occurs when the controller relies upon the device end before writing the printer character. As this is not a problem with the short write-sequence it has been ignored.

5.2 Controller Efficiency

The SMUT controller is a computer that has other operations to perform in parallel with the serial line processing. So that the computer does not waste time processing reply frames from an idle terminal, only significant status bits of the reply frame will cause a computer interrupt and subsequent processing. When a terminal is not printing or reading, it is effectively idle and need be polled only to detect input characters.

The status bits that are used to initiate a computer interrupt are the two device ends, the two channel ends, and writer (printer) busy. Originally, only the channel and device ends were used, but when writer busy was also used, the detection of lost polls and/or reply frames was possible and error anticipation and avoidance was possible.

The reader (keyboard) busy bit occurs whenever a character is requested from a terminal and will occur until a character has been received. As this covers most of the idle time for a terminal, the resultant processing was considered unnecessary. This is in conflict with the operation discussed in Section 5.1.2, which requires detection of keyboard busy, but sufficient significant reply frames are usually received because of the writer busy reponses.

5.3 System Lockup

The SMUT controller is a program controlled device and therefore subject to program delays when the computer is handling interrupt requests from direct connection teleprinters, Dataway transfers, *etc.* These requests can delay the outward poll when the clock interrupt is taken late. When delays can be greater than 300 μs, there is a finite possibility that an undelayed outward poll could be sent over a delayed return from the previous outward poll. The computer tests whether the receiver is busy before issuing an outward poll; if the receiver is busy with the reply frame, the overpoll is delayed one clock interrupt.

As a result of the above test a fault has been found in the receiver logic design. There are two common methods of detecting when the incoming transmission is finished; either start a counter and count off the incoming bits until the counter is zero, or shift the start bit through a shift register of the right length until it falls out. The Mk I receiver uses the latter technique and, very occasionally, the receiver is set busy without the start bit being set in the shift register owing to noise pickup and variation in gate switching-thresholds. When this occurs, the receiver remains busy until the next poll when another bit is detected and shifted through.

The above failure to clear busy until the next poll is incidental unless the receiver that is incorrectly set into the busy state is the controller-receiver which, because of the overpoll possibility mentioned above, will stop the next poll and prevent any chance of clearing the busy state. If the receiver had used a counter, this failure mode would not occur unless the controller-receiver clock failed. This lockup condition has only occurred when one of the remote stations is very marginal, so it has not been a problem. It does indicate that, for this application, the counter method of receiver control is superior to reliance upon the shift register, although both techniques are logically correct. The count technique is used in the Mk II system and also in the later models (and obviously, the controller) of the Mk I system.

5.4 Smart Polling

The SMUT controller is interfaced to a NOVA820 minicomputer with 32 k words of core memory. This enables the controller to operate a complex poll sequence without significantly reducing the memory available for user programs. One of the aims of intelligent polling was to permit the use of video displays on the system. These require high poll rates to operate at reasonable speeds and this cuts the polls available to service slower terminals. It is envisaged that displays will be serviced by using a line concentrator on output and sending a line or block of data on a write poll (see Section 6). However, as an interim solution, smarter poll sequences were developed in an attempt to increase specific poll rates without significantly reducing overall performance.

A relatively primitive algorithm was used to enable those stations using 300 baud decwriters to receive more polls than teleprinter stations so that polling delays for decwriters would be reduced. The resultant increase in speed was marked when the fast terminals were used in inquiry mode where output is large compared with input. This was accomplished by forming a fast poll table of decwriter addresses which was fixed and which limited the flexibility of allotting fast stations. However, the main objection to this method was the decrease in conversational performance of the slow poll stations,making these appear very spongy (see Section 3.2.2).

The primitive algorithm above improves the response of decwriters but does not increase rates sufficiently to operate video displays. The smart fast poll algorithm was developed because some stations had not been installed and all the stations were not in constant use. The basic problem was to define usage for operator convenience and still keep display operation feasible. The following criteria were used in developing the smart poll algorithm:

- All stations capable of operating must be polled.
- All proposed stations up to a sensible maximum must be polled at a slow rate to allow installation and maintenance without constant program modification.
- Only those stations that are currently in use need to be polled at a higher rate.
- A station is defined as 'in use' when it commences printing a character or message.
- A station is defined as 'not in use' at some fixed time after the last character was printed.
- The operating program is to be stable and does not have to change whenever address alterations or changes to poll rates are made.

The algorithm was implemented by creating a primary address table which uses three out of four polls and which expands or contracts according to usage. The remaining poll is used to scan the block of addresses from 25 (as at 6th June 1976) to 0 sequentially, regardless of speed of operation or expected usage. Entries were placed into the fast poll table whenever a write poll was sent to a station and were deleted 10-12 s after the last write poll.

The smart poll technique is still in use and has proved very satisfactory because, as well as increasing printing speed for all fast poll stations, the conversational response is very good, as long as input characters are echoed at better than the 10 s usage time-out. When the operator exceeds time-out, the next input character will be slow to respond. However, 'hunt and peck' operators have to be exceedingly slow to lose fast response.

The system has been monitored to try to gauge usage and it has been found that, even during peak periods, the fast table rarely has more than six entries. Video displays capable of 183 characters/second are being used on SMUT and will operate up to 150 characters/second when they are printing alone (*i.e.* one fast table entry) dropping to 125 characters/second when other entries are present in the table.

6. PROPOSED BURST MODE

Video displays can be serviced adequately, from the operator's viewpoint, at about 150 to 200 characters/second during output for most inquiry applications. A higher speed is preferable where the lines approach the full 80 characters or where an operator is scanning for keywords. When most lines consist of less than forty or so characters, 150 characters/second is quite tolerable. For display operation the line rate is more noticeable than the character rate and, if lines are written at approximately five per second, the overall operation seems reasonable. When faster line rates are used, the screen is not stable enough to read easily. The possibility of using the SMUT system to service video displays on a line-by-line basis instead of character-by-character has been investigated and modes of operation have been formulated. The implementation of burst mode stations has been delayed, partly owing to lack of effort but mainly owing to the improvements realised when smart polling was introduced. Burst mode will only be implemented when traffic increases to the point where the smart poll fast table becomes large enough to reduce display performance to below 100 characters/second for extended periods.

Each character takes at least one poll to transmit it to a station which gives a maximum rate of 1000 characters/second for the total system, regardless of terminal types. This figure becomes inadequate when three or more displays are operating simultaneously with page printers and teleprinters. If, however, the output to the displays is packed into a block format consisting of either a fixed block or a variable block controlled by carriage returns, the theoretical maximum data rate of 12 500 character/second is possible without changing the overall baud rate of the SMUT system. If the system could be run super-efficiently, then the 64 possible stations could all be displays. However, quite obviously these rates cannot be attained and operation of more than ten displays is not envisaged because the controlling NOVA computer would become overloaded.

6.1 Operation

The only completely different sequence from the standard SMUT system is the block write from the controller to a station with burst capability. For this sequence, when a line or block is to be transmitted to a video display or page printer, the separate characters are loaded into a First-In, First-Out (FIFO) buffer under program control. The block is transmitted with the station address, *etc.* (see Section 6.3) at the next poll time. As this transmission can cover up to the next ten polls, the next poll is delayed by the receiver-busy inhibit (see Section 5.3).

When the transmission is being received, the characters are loaded into the station FIFO buffer until the code is completed and error checked (see Section 6.3). When this block is verified, characters are pulled from the FIFO buffer and sent to the display/printer until the buffer is empty. The station does not require any attention until either the output buffer is empty or a character is entered at the terminal keyboard. For these cases, quite a slow poll rate is adequate as inquiry characters are normally only processed at the end of a line, such lines occurring at 5-10 s^{-1}. The conversational character by character input and echo will operate in the normal SMUT mode.

6.2 Definition

The following criteria were used as a basis for burst or blocked mode:

- Burst mode data would be available as output from the control computer only and all input data would be standard.
- Each station using burst mode must have sufficient storage for the maximum block likely to be sent, possibly 128 bytes.
- Burst mode should be selectable by setting a verified bit, *i.e.* part of the address byte (bit 05).
- The stations capable of burst mode must have particular known addresses so that the controlling NOVA can identify them.
- Burst mode must not interfere with any standard stations and should not affect their performance to any marked extent.
- Each burst mode station must be capable of being operated in a normal SMUT sequence when the burst mode bit is not selected.

6.3 Block Structure

The transmission frames for the block write from the controller to the station must take the following form which, logically, includes the standard frame as a subset:

C.A.S.W.AC.(data)(B).T

where

() = optional field

C = start bit (assertion)

A = address byte

S = status byte

W = byte count (character for standard frame)

AC = address byte complemented

B = block check of complete frame except C and T

T = trailing bits (two zeros)

7. CONCLUSION

There is little doubt that without the SMUT system, the Dataway responses would have deteriorated drastically, largely due to the mundane servicing of an ever-increasing number of remote teleprinter/terminal stations. The Dataway is currently being used to perform its primary function, namely the transfer of data in burst mode between minicomputers and, in restricted instances, handling unit-byte transfers.

At present, the only unit-byte devices on the AAEC Dataway are a 2400 baud video display and several simulated devices where a computer calls the NOVA as a terminal. All other terminals except direct connections are serviced by SMUT.

8. ACKNOWLEDGEMENTS

The SMUT software interface with DATERCOM [Sanger 1976] was written by Dr P.L. Sanger who also made numerous refinements and tests to improve overall system performance. During this developmental phase he provided an effective interface to users as well as contributing many pertinent suggestions.

The SMUT hardware was developed with the help of R.F. Thornton, J.D. Milne and P. Parton of the Digital Systems Group; Mr Thornton also carried out the installation and debugging of separate stations.

9. REFERENCES

Backstrom, R.P. [1975] - IBM360 software for the AAEC computer network. AAEC unpublished document.

Bennett, N.W. & Sanger, P.L. [1973] - The development of the ACL language and its implementation ACL-NOVA. Aust. Comput. J., 5 (3) 105-113.

Ellis, P.J. [1970] - A multiplexed computer-computer, computer-device data link. AAEC/E206.

Richardson, D.J. [1969] - A generalised computer to computer link for an IBM360 computer. Aust. Comput. J., 1 (5) 273-276.

Sanger, P.L. [1971] - ACL-NOVA - A multi-user conversational interpreter for the NOVA computer. AAEC/E221 (revised July 1972).

Sanger, P.L. [1976] - The AAEC DAtaway TERminal COMmunication system. Proc. 7th Australian Computer Conference, 30th August to 3rd September, Perth, W.A. pp. 610-622.