LA-UR 79-1241

CONF-791104--3

TITLE: A REAL-TIME INVENTORY SYSTEM FOR SPECIAL NUCLEAR MATERIAL

MASTER

AUTHOR(S): Thomas H. Kuckertz
C. Dwayne Ethridge
Nicholas Nicholson

SUBMITTED TO: First International Micro & Mini Computer Conference
Houston, Texas
November 14-16, 1979

University of California

LOS ALAMOS SCIENTIFIC LABORATORY
Post Office Box 1663 Los Alamos, New Mexico 87545
An Affirmative Action/Equal Opportunity Employer

A REAL-TIME INVENTORY SYSTEM FOR SPECIAL NUCLEAR MATERIAL

Thomas H. Kuckertz, C. Dwayne Ethridge, and Nicholas Nicholson
Los Alamos Scientific Laboratory
Los Alamos, NM 87545

ABSTRACT

Because of the damage that could occur if special nuclear material (SNM) fell into the wrong hands, it is very desirable that all such material be subject to strict access and security controls. This paper describes a special purpose peripheral device for a minicomputer that can monitor SNM in vault storage in real time and give timely indication of any tampering with the material. This device, called a shelf monitor, is designed around a single-chip microcomputer, and can be manufactured in quantity for about $100. A typical system of shelf monitors controlled by a minicomputer is described. The minicomputer is used to acquire data associated with the weight and gamma activity of the sample of SNM under observation. Significant deviations in the weight and gamma activity are cause for a tampering alarm.

Introduction

Special nuclear material (SNM) in long-term storage in a vault is an attractive target for a diverter. Physical inventories are taken periodically but, if access can be obtained, material may be taken without timely detection. To provide a back-up for access control and to reduce or eliminate the need for inventories, a special purpose peripheral device, which can interface in large numbers to most minicomputers now marketed, has been designed to monitor cans of SNM in storage. These devices, called shelf monitors, continuously observe both the weight of and the gamma radiation coming from a can of SNM placed on them. Any significant change in either the weight of the can or its gamma count rate will cause timely detection of tampering with the material. While the number of shelf monitors that can be interfaced to a typical minicomputer system is practically unlimited, a typical system would contain approximately 1000 shelf monitors. Each shelf monitor possesses a certain amount of intelligence, which is based on an Intel 8748 microcomputer; however, a minicomputer, Data General Nova 3, is required to control a system of monitors. The minicomputer acquires weight and gamma data from each shelf monitor in the system. These data are compared to previously acquired data for each shelf monitor with any significant deviation causing an alarm.

Description of a Shelf Monitor

Each shelf monitor is a smart device whose intelligence is provided by an Intel 8748 microcomputer. Because any practical shelf monitor system requires large numbers of shelf monitors, a primary design consideration was that of low cost. Two other considerations were sensing the weight of the can and sensing the amount of gamma radiation emanating from the can. It is not necessary that either sensor provide calibrated data in engineering units; however, each sensor must provide data that are indicative of weight or gamma radiation and be stable for hours at a time, because knowledge of changes in weight or gamma radiation is of primary importance rather than actual accurate values for these quantities.

It quickly became clear that a minicomputer system would be unable to control a system of shelf monitors if the minicomputer had to monitor the gamma and scale information associated with a can of SNM in its raw form. However, if the raw data could be processed in the shelf monitor and sent to the minicomputer when requested, large numbers of shelf monitors could be controlled by a minicomputer system. A microcomputer system was a logical choice to perform this data processing and control function, except that those on the market, when the shelf monitor ideas were first conceived, required many support chips that would have driven the cost of a finished shelf monitor to unacceptable levels. Then Intel introduced the 8048 microcomputer family, which provided for an entire microcomputer system on a single chip. The 8748 member of this family was chosen to be used in the early shelf monitors because its on-board read-only memory consisted of EPROM, which could be erased and reprogrammed if the need arose. The 8748 contains the following features that enabled the development of the shelf monitor:

64 bytes of read/write memory
1024 bytes of program (EPROM) memory
Event counter
External interrupt
24 lines for input and output of digital information.

All this is contained on a single chip that requires a single 5-V power supply. The latest revision of the shelf monitor requires components costing about $60. It is believed that these shelf monitors can be made in quantity for less than $100 each.

The weight sensor of a shelf monitor is a variable capacitor whose capacitance is dependent upon the weight of a can of SNM that might be placed on it. This capacitor is part of an oscillator circuit whose frequency is dependent upon the weight of an object placed upon the capacitor. The event counter, which is an integral part of the Intel 8748 microcomputer, is used to count each pulse coming out of this oscillator. This is shown schematically in Fig. 1. Tests have shown that the frequency stability of this oscillator over a 24-h period is usually better than ±3 Hz. The can of SNM sitting on a monitor is also electrically coupled to the capacitor; thus, even touching the can will cause a significant change in the frequency of the scale oscillator. The event counter of the 8748 consists of 8 bits and thus "rolls over" every 256 counts. Each such roll over causes a 16-bit software counter in the 8748 to be incremented. This allows the high order bits of the scale count to be saved in addition to freeing the 8748 CPU of the responsibility for responding to every scale pulse.

A Geiger-Müller (GM) tube is used to count gamma rays emanating from the can of SNM. This tube is appropriately shielded and is collimated so that the tube "sees" only gamma radiation from the can directly above it and the gamma count rate is restricted to low values so that the tube lifetime will be at least 3 to 5 yr of continuous operation. The voltage pulses coming from the GM tubes are conditioned so they meet TTL specifications and are then used to drive the external interrupt of the 8748. Thus, the gamma pulses are counted entirely in a 16-bit software counter. This is possible because of the relatively low count rates (~200 counts/s) seen by the GM tube as opposed to the scale frequency, which centers at 35 000 counts/s. It should be noted that the gamma activity seen by the GM tube is not sufficient to detect some kinds of tampering. This is because the bottom few centimeters of SNM in a can effectively shield the material at the top of the can from the GM tube. Thus, both the scale and gamma sensors are required to give protection in depth to the SNM.

Shelf monitors in a system communicate with their minicomputer controller over a serial bus structure consisting of a command bus and a response bus. This structure is shown in Fig. 1. As many as 223 shelf monitors can be attached to a single bus structure. Both the command and response buses carry data in bit serial fashion. Line drivers and receivers for the buses were chosen so that a large number of devices could be attached to the buses without degrading the electrical performance of the buses. The data conveyed on the command bus in bit serial format have the same information content and timing as would the Receive Data signal of a full-duplex, asynchronous EIA R232C interface; however, the electrical levels of the two are different. Likewise, the data on the response bus correspond to the Transmit Data signal in an EIA RS232C interface. The command and response buses are connected to a receiver/driver network shown in Fig. 2 that converts the signals on the command and response buses to the electrical levels of the RS232C interface specification. Thus, a string of shelf monitors can communicate with any minicomputer with an RS232C interface directly or over telephone lines from a remote site using modems. The shelf monitors are designed to communicate with the buses at one of two possible baud rates, 2400 or 9600, although other rates are possible.

A system of shelf monitors is controlled by dispatching an 8-bit command on its command bus and then reading the response that occurs on the response bus. There exists 256 different possible commands that can be transmitted on the command bus. Each command transmitted on the command bus is read by each shelf monitor on the bus and, if the command pertains to the shelf monitor, the shelf monitor performs the activity specified by the command. Certain commands are executed by all shelf monitors on a string; these commands are called broadcast commands. The remaining commands are addressed to a single shelf monitor, which then responds by transmitting its scale and gamma information to the minicomputer. These commands are called device-specific commands because a single shelf monitor will execute them. The entire command space is described in Table I.

There are 10 different broadcast commands. The two most important of these commands are (1) reset counters and start counting, and (2) stop counting and hold counts. The first of these commands is used to start a data-acquisition cycle, while the second command is used to save the acquired data until each shelf monitor can be read with a device-specific command. The command and response buses each consist of two serial buses called bus A and bus B. Normally, both of these buses are enabled and identical data are transmitted in paralled over each bus. If a malfunction occurs in one but not both, the enable bus commands can be used to isolate the bad bus without degrading the performance of the shelf monitor system. Certain diagnostic broadcast commands are used to verify the proper operation of each shelf monitor in the system. These are preset commands that allow the minicomputer to preset the counters in each shelf monitor to one of four possible hexadecimal counts: 00, AA, 55, and FF. There is an additional broadcast command that allows the counters to be incremented by the minicomputer.

While there are many different broadcast commands, there is only one type of device-specific command. This type of command is basically a command to read the information stored in a particular shelf monitor into the minicomputer. Each shelf monitor is equipped with a hardware switch register of 8 bits mounted in a dual-in-line package. The switch settings determine uniquely the address of the shelf monitor. Only those addresses from 1 through 223 are valid. When any command comes down the command bus, each monitor on the bus examines the command. If it is a broadcast command, the shelf monitor executes it. If the command is identical to the shelf monitor address, the shelf monitor transmits a 6-byte response on the response bus. If the command is not a broadcast command and it is not identical to the address of the shelf monitor, the command is ignored.

When a shelf monitor is addressed by an 8-bit command on the command bus, it responds with 6 bytes of data on the response bus. The first byte is the address of the shelf monitor (an echo of the command received from the command bus). The second and third bytes contain a 16-bit scale count, while the fourth and fifth bytes contain a 16-bit gamma count. The sixth byte contains a check sum so that all bytes of the response including the check sum total 254. The check sum is used to detect errors in data transmission on the response bus.

Description of a Timer Monitor

In order for a comparison of the most recently read data points with previously read data points, the data must be in a form that is independent of the counting time. All data comparisons are made in units of count rates. The count rates are calculated by dividing the scale and gamma counts by the time the shelf monitors had been counting. The minicomputer's knowledge of the counting time, in the case of a Nova 3 using the real-time disk operating system (RDOS), could be in error by as much as 0.1 s for count times of the order of 5 s. The scale frequencies are stable with considerably greater precision than that of this count time. Thus, a more accurate count time would yield higher accuracy scale count rates and to some extent gamma count rates. To solve this problem, a special shelf monitor was designed to measure the time between reset and start counting and hold counts commands. This monitor is called a timer monitor.

A single timer monitor resides on a command bus that contains as many as 223 additional shelf monitors. This timer monitor responds to a device-specific command 0. When this command is transmitted on the command bus, the timer monitor responds with a 6-byte response in the same manner as a shelf monitor, except that bytes 2 through 5 of the response contain a 32-bit count, which had been driven by an accurate 12.480-kHz oscillator. The timer provides a precise counting time that is used to calculate scale and gamma count rates. The timer also provides a 6-byte response to all broadcast commands. Thus, every command sent on the command bus should provoke a 6-byte response from either a shelf monitor or the timer monitor. The response to a broadcast command by the timer is as follows:

1. Byte 1 - echo of broadcast command received.
2. Bytes 2-5 - contents of 32-bit timer register after executing the broadcast command.
3. Byte 6 - check sum.

Typical Shelf Monitor System

A typical shelf monitor system is shown in Fig. 2. This system will monitor as many as 892 cans of SNM in a vault using a Data General Nova 3 minicomputer. The 892 shelf monitors are apportioned equally to four sets of command/response buses that are driven by four RS232C interfaces of the Nova 3. Note that four timer monitors are also part of this system. It should also be noted that each of the four strings of 223 shelf monitors could be in a different vault at remote sites with modems between the RS232C interfaces and the receiver/driver circuitry that converts RS232C signals to/from those used on the command and response buses.

The minicomputer controlling the entire shelf monitor system is a Data General Nova 3, although any minicomputer with four RS232C interfaces could have been used. The following peripheral devices are attached to the Nova 3:

64-k word memory
Versatec 1200A printer/plotter
9-track tape drive
10-Mbyte moving head disk.

It is believed that this configuration is near minimal for the number of shelf monitors shown in Fig. 2. The successful operation of this system required a significant software development effort for that software residing in each shelf monitor and the controlling software, which resides in the Nova 3. The Nova 3 computer system shown in Fig. 2 was used to develop the Nova 3 resident software, while the 8748 resident software was developed on an Intel microcomputer development system.

Distributed Software

While most of the intelligence of this system resides in the Nova 3, considerable intelligence resides in each shelf monitor. Thus, distributed software is used to control and acquire data from this system. The executive residing in each shelf monitor was written in Intel MCS 8048 assembly language whose macro-expansion facility was used heavily. The Nova 3 control programs were written mostly in FORTRAN IV with the exception of the RS232C interface drivers, and are subordinate to the Data General RDOS. The RS232C drivers were written in Nova 3 macro-assembly language.

Shelf Monitor Executive

An event-driven executive resides in each shelf monitor. The events that drive the executive are power-up, event counter/timer overflow, and external interrupt. When the power is first turned-on or restored, the executive initializes the counters to known, very large, values and puts the counter in a hold condition. The executive then goes into a wait loop that examines every command that comes down the command bus. If a received command is a broadcast command, it is executed; otherwise the command is compared to the address of the shelf monitor. When both are identical, the shelf monitor responds with the 6-byte response described previously. In the case of the timer monitor, a [illegible]-byte response is made to all global commands. The event counters and external interrupts are enabled by appropriate broadcast commands being executed.

Whenever a gamma pulse occurs and interrupts have been enabled, an interrupt is requested on the external interrupt input to the 8748. The executive services this interrupt by incrementing a 16-bit software counter where the gamma counts are kept. Whenever a scale pulse occurs and the counter has been enabled, the event counter is incremented. An interrupt is requested whenever this 8-bit counter overflows. The executive services this interrupt by incrementing a 16-bit software counter. Whenever a scale and gamma interrupt occur simultaneously, the gamma interrupt is serviced first while the scale interrupt is latched to prevent its being lost. The executive now uses 25 bytes of read/write memory and 400 bytes of read-only memory.

Nova 3 Resident Software

The Nova 3 minicomputer has responsibility for controlling the entire system of shelf monitors. In order to reduce the complexity of the control program, which is resident in the Nova 2, as many control and input/output ac-tivities as possible are performed by the Data General RDOS. The functions performed by RDOS include:

1. Scheduling of all activities (tasks).
2. Gross timing of shelf monitor counting.
3. Transfer of all data to/from external devices. These functions are all performed using system calls to RDOS.

The shelf monitor system control program does the following in support of the shelf monitor system:

1. Carries on a man/machine dialogue to configure the shelf monitor system.
2. Performs the routine acquisition of data from each shelf monitor in the system.
3. Processes acquired data to detect any tampering with SNM under surveillance.
4. Issues messages describing system exceptions to a human operator.

System start up first requires the bootstrapping of RDOS. The applications program that operates the shelf monitor system is then run. The applications program acquires the following information from the human operator by way of a man/machine dialogue:

Identity of each RS232C interface with shelf monitor.

Identity of each shelf monitor associated with an RS232C interface.

When a system is running, a routine monitoring sequence is performed every 5 to 80 s, depending on the number of monitors in the system. The routine monitoring sequence consists of the following:

1. Issue reset and count command.
2. Wait for response.
3. Wait for a nominal counting time (5 s).
4. Issue hold counts command.
5. Wait for a response.
6. Issue command to read timer monitor.
7. Wait for response.
8. Issue command to read shelf monitor.
9. Wait for response.

10. Calculate scale and gamma rates using time from timer.

11. Queue alarm message if calculated scale and gamma rates deviate significantly from average scale and average gamma rates.

12. Calculate approximate new average scale and gamma rates using present scale and gamma rates and old average rates.

Steps 8 through 11 are repeated until all shelf monitors on a single line have been read and their data processed. All RS232C lines in the system are serviced as described above.

Whenever a command is sent to a string of shelf monitors, the command dispatch sequence is executed. Because every command should provoke a 6-byte response, the command dispatch sequence is identical for all commands regardless of type, broadcast or device specific. A 6-byte response consists of the first byte, which is merely an echo of the command dispatched, bytes 2 through 5, which contain timer or gamma and scale information, and byte 6, which contains a check sum that forces the low-order 8 bits of a sum of the 6 bytes to equal 254.

A variety of error conditions can occur in the system response to a dispatched command as follows:

I/O error detected by RDOS
No response received
First byte not the same as command dispatched
Bytes 2 through 5 contain power-up default data counts
The low-order 8 bits of the sum of all 6 bytes does not equal 254.

Every response from a dispatched command is examined for all of the above possible error conditions. If an error is detected, an error message describing the condition is queued. Additionally, an appropriate command dispatch sequence is executed until all lost data are recovered error free. If the lost data cannot be recovered after a specified number of unsuccessful retries, the defective device (RS232C interface and/or shelf monitor) is sent no more commands appropriate to it.

A past history of each shelf monitor must be saved if current scale and gamma rates are to yield possible alarm information. It is known that the scale and gamma rates may drift significantly over long periods (days and weeks), thus it was decided that an average count rate consisting of the last

100 points read would be used as the basis for the detection of tampering. At least 175 000 16-bit words of storage would be required to save the most recent 100 gamma and scale readings of 892 monitor systems, and considerable computation time would be required to compute the average count rates each time a new data point was acquired. Thus, all averages are calculated using an approximation to the true average described as follows:

$$A_n = (1/n)P + [(n-1)/n]A_o \quad . \qquad (1)$$

A_n is the new average, A_o is the old average, P is the point being added to the average, and n is the number of points in the new average. By using this approximation, the only data that need to be saved for each shelf monitor are average gamma count rate and average scale rate. These are kept as floating point quantities. An additional 16-bit status word is associated with each shelf monitor. This status word tells the application program whether or not the monitor is operative or in use. The status word also contains initial timing sequence information and how many points are in the average count rates. It is possible to configure the system before the cans of SNM are put on each monitor. The initial timing sequence is a time period wherein no data are acquired from the monitor in the sequence. This permits a time delay between entry into the system of a shelf monitor and the acquisition of data from it. This time delay is variable but is usually less than 10 min. When a shelf monitor is first placed in service, the first scale and gamma rates read are all that is known about the monitor. While it is desirable to use 100 points as the basis of the approximate average calculation, the average will require the acquisition of at least 100 points for it to converge to a value that can be used to trigger alarms. Three bits in the status word are reserved for indicating the number of points in the old average count rates. The range of values indicating the number of points is 0 through 7. Each time a point is added to the average count rates associated with the shelf monitor, the number of points is incremented and its new value is used in calculating the average count rates according to Eq. (1). Upon reaching the value 7, the number of points is assumed to be 100 for the purpose of performing the calculations in Eq. (1) and is incremented no further.

An alarm is to be given if a significant deviation from average scale or gamma count rates is detected. The scale frequency is sufficiently stable that a deviation of '3 counts/s is considered significant. The gamma alarms proved more difficult to calculate. It was finally decided that gamma deviations greater than 3.5 to 4.0 standard deviations from the average count rate were significant. Calculating the standard deviation for the most recent 100 gamma points is conceptually easy; however, the last 100 points would have to be saved and great computational effort would have to be expended for each deviation calculated. If the gamma points are assumed to have a Poisson distribution, the standard deviation of the gamma rate can be calculated using only knowledge of the average gamma rate and the average counting time.

$$\sigma = \sqrt{\frac{G_a}{T_c}} \qquad (2)$$

σ is the standard deviation, G_a is the average gamma count rate, and T_c is the counting time. T_c is normally chosen to be the nominal count time used by RDOS to time a counting period.

Messages are occasionally sent to the operator to advise him of abnormal conditions of various kinds occurring in the system. All such messages are placed in a queue to be printed at a time during which the computer is waiting for an input/output operation or a count operation to be finished. The objective here is to not delay the routine counting activities when a group of messages stack up. It is possible that an authorized individual may remove a can of SNM from the vault and then notify the shelf monitor system of the transaction. Tampering messages are queued subject to a time delay before they are actually printed. Thus, if the shelf monitor system gains knowledge of the transaction within a reasonably short time period of the transaction, it can dequeue any tampering messages associated with the can of SNM. The message queue is examined every 2 s for ready messages.

Nova 3 Task Structure

Six tasks perform the above described functions. All tasks are written in FORTRAN IV and use its task monitor for their scheduling. Task MONCHK is active when the shelf monitor system control program first starts executing. This task performs the following functions. It receives information from a human operator on the operational status of every shelf monitor to be watched. It then creates the remaining tasks and, if desired, prints periodic reports giving the status of each monitor that is active in the system. MONCHK has the highest priority of the six tasks.

Task RDACK reads and checks all shelf monitors in the system as follows:

- Dispatches all commands to the RS232C interfaces
- Checks for errors in responses to commands
- Calculates gamma and scale rates from data read from monitors
- Checks for alarms
- Calculates the approximate average count rates.

This task executes as often as it can to acquire the count rate data it needs to check for alarms. This task is idle when it is waiting for a response to a command or is waiting for a counting time period to end. During these idle periods, the other tasks are active if their execution is desired.

Task TMTT is used to time the period required for a response to a command dispatched by RDACK. If no response is received in 4 s, task RDACK is notified that the command produced no response. This task executes every 0.5 s with a very short execution time. Task TMTU performs another timing function. This task can effect a delay in the start of calculation of the average scale and gamma rates after the time the monitor is first identified to the system. The delay is approximately 5 to 10 min. This task executes every 2 min with a very short execution time.

Task RMTT reads the 6-byte response to every command sent to the RS232C interface. This task executes when demanded by task RDACK. The execution of this task is timed by task TMTT. If this task takes too long to execute, it is terminated abnormally.

All messages to the operator are queued so that they can be printed without delaying the data-acquisition activities of task RDACK. Task MSGWR examines the message queue every 2 s or as often as possible, whichever is longer for queued messages. The messages are then printed and dequeued. Messages with a time delay have their delays updated and remain queued. Normally, this task has a very short execution time because the message queue is usually empty. However, there is room for as many as 250 messages in the queue at any one time. The set of possible messages contains [illegible] messages that have similar formats, each of which is determined by the content of the message. As many as three numeric quantities can be associated with a single message.

Conclusions

This paper describes a special purpose peripheral device that when used in large quantities, can monitor cans of SNM in storage. These devices are called shelf monitors and a system of such shelf monitors together with a minicomputer controller can keep a real-time inventory of the SNM and give timely indication of tampering with the SNM. Because the shelf monitors contain a microcomputer, the intelligence of this system is distributed among both a minicomputer and a large number of inexpensive shelf monitors.

The typical system described here has been operational since April 1, 1979, with a reduced number of shelf monitors operating in two vaults and one test stand. This system required the considerable effort of the authors over two calendar years time. Engineering time to design several different models of the present shelf monitor required about 12 man-months of effort while the software executive required 6 man-months of effort. The software development effort for the Nova 3 codes required 12 man-months of effort and the developed code occupies 18-K words of memory exclusive of RDOS, the operating system.

TABLE I

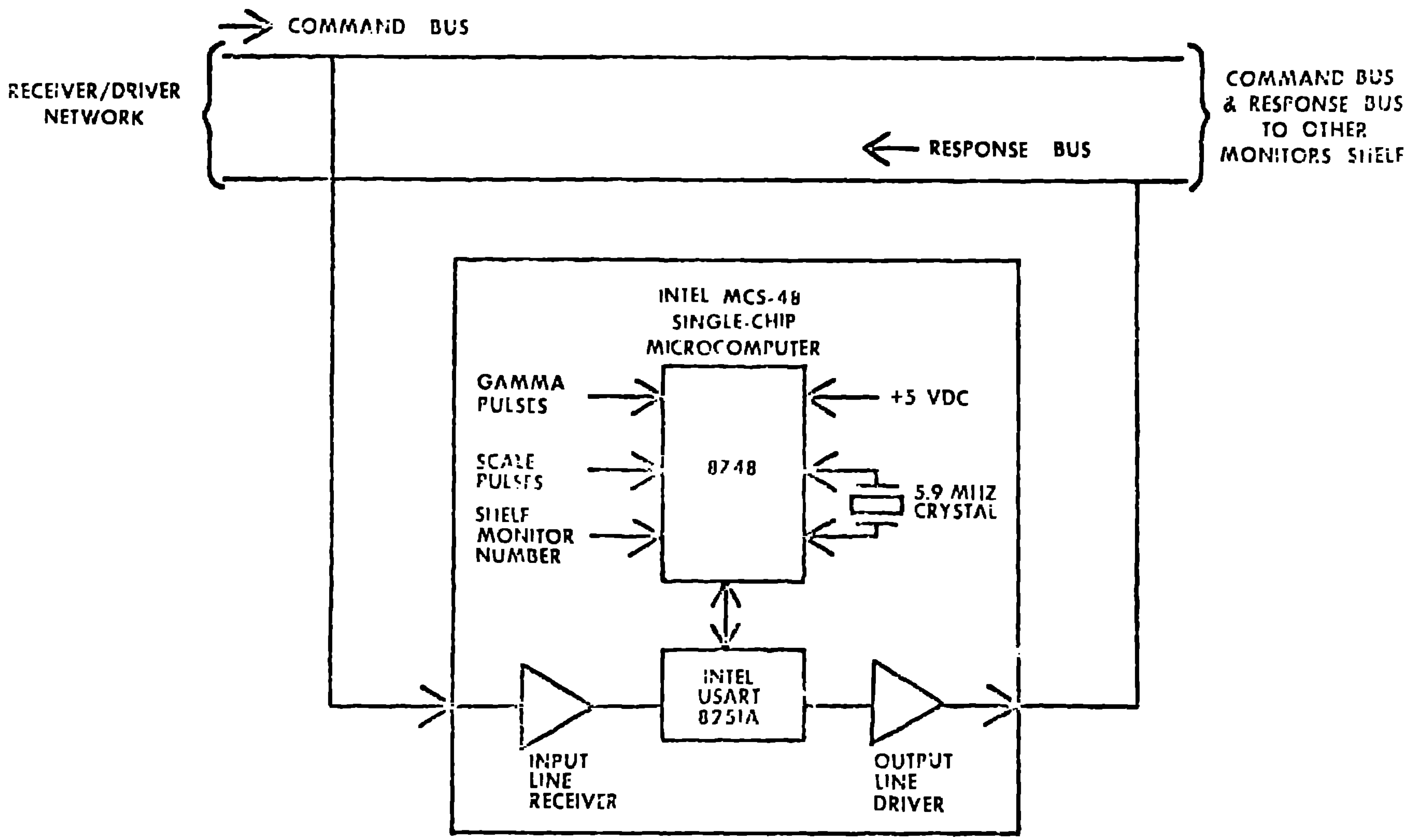
SHELF MONITOR SYSTEM COMMAND LIST

Command Hex	Command Decimal	Function Performed*	Type
00	0	Read timer monitor	DS
01	1	Read shelf monitor with	DS
Thru	Thru	address same as command	
DF	223		
EC	224		
Thru	Thru	Reserved for future use	
F5	245		
F6	246	Increment counters	B
F7	247	Preset counters to AA	B
F8	248	Preset counters to 55	B
F9	249	Preset counters to 00	B
FA	250	Preset counters to FF	B
FB	251	Disable bus A/enable bus B	B
FC	252	Disable bus B/enable bus A	B
FD	253	Continue counting	B
FE	254	Hold present counts	B
FF	255	Preset counters to 00 and start counting	B

DS - Device specific

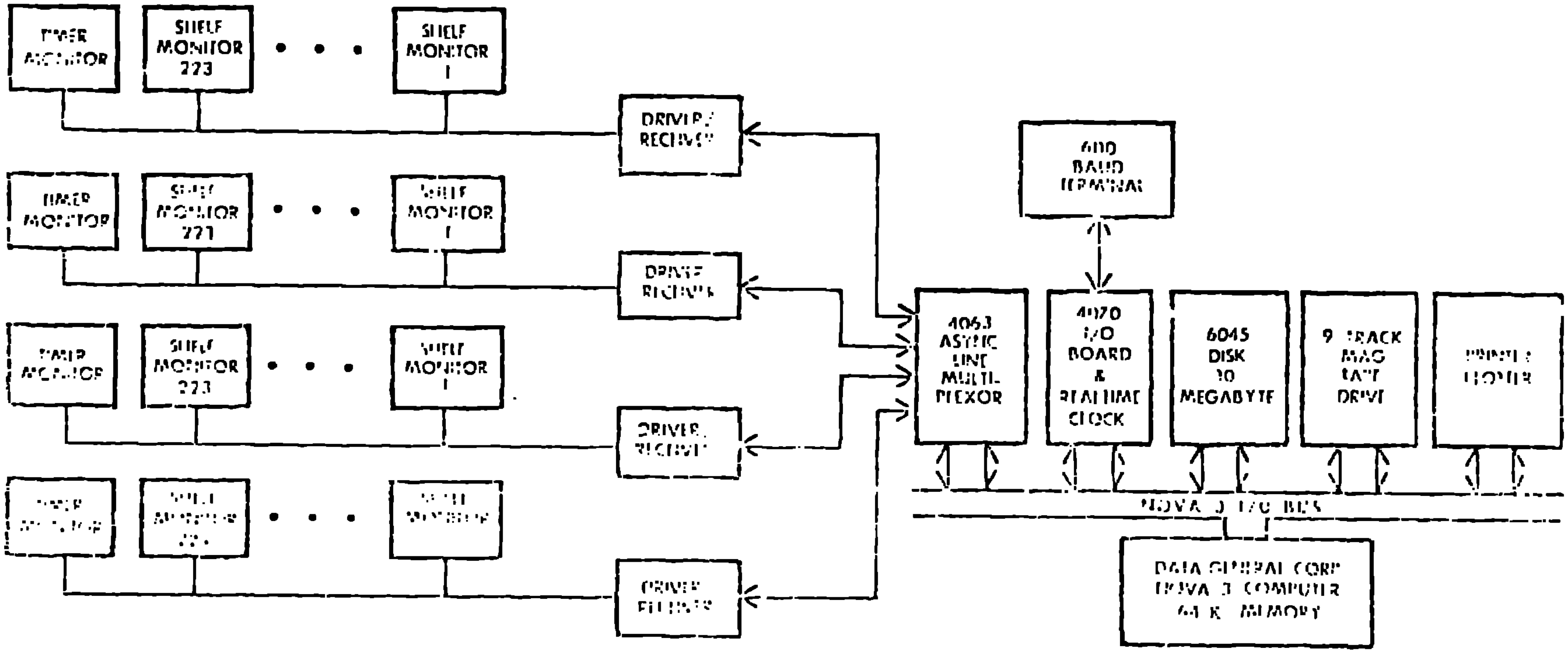
B - Broadcast

*All numbers in this column are hexadecimal.



INTELLIGENT SHELF MONITOR
FOR
SNM REAL TIME INVENTORY SYSTEM

FIGURE 1



REALTIME INVENTORY SYSTEM REAL-TIME INVENTORY SYSTEM BLOCK DIAGRAM

FIGURE 2