

512-Channel Correlator Controller

S. S. Brokl

Communications Systems Research Section

JPL and the Haystack Observatory have developed a high-speed correlator for radio and radar observations. To ensure that the host computer could operate during the extended run times of the JPL-Haystack correlator, a controller was designed so that the correlator could run automatically without computer intervention. The correlator controller assumes the role of bus master and keeps track of data and properly interrupts the computer at the end of the observation.

I. Introduction

In 1972 JPL and the Haystack Observatory developed a 1024-Channel Solid State Correlator System, which consists of a control computer plus sixteen 64-channel correlator modules. It was decided that 512 channels of this correlator design could meet current JPL requirements for on-going radar experiments.

Each correlator module may be operated as a 64-channel "real" correlator or as a 32-channel "complex" correlator. The correlator may be used in either an autocorrelation mode or crosscorrelation mode. The maximum bandwidth is 10 MHz with a 20-MHz clock rate. Each module can accept quantized data in two-, three-, or five-bit formats.

The correlator controller was designed to allow the master computer to be free to carry on other tasks while the correlator accumulates the data. Once the computer loads the operating parameters into the correlator controller, the correlator is free to operate without the

computer's central processor. The controller takes over the task of monitoring the most significant bits from each correlator channel accumulator and stores this overflow information in the computer's memory by way of a Direct Memory Access (DMA). An overflow is the process of detecting the "1" to "0" transition of the most significant bit of the accumulator. When a correlator run is complete, the controller will interrupt the computer with an "end-of-run" indication. At this time the computer takes over control and proceeds to unload the low order counts from the correlator accumulators and process the data.

II. Correlator Controller Requirements

The host computer chosen for this project is a Digital Equipment Corporation PDP 11-20. A block diagram illustrating the communications requirements of the controller is shown in Fig. 1. The controller is a moderator correctly formatting communications between the computer and the correlator modules. In addition, when the

correlator is running, the controller takes charge of the unibus and becomes bus master. This is done to take advantage of the Direct Memory Access (DMA) feature of the PDP 11-20 computer. DMA allows a peripheral device to communicate directly with the computer's memory. With DMA an external device can run at memory cycle times and does not interfere with the computer central processor operation.

During long run times the controller computes the overflow of the accumulator of each channel and stores this information by way of the DMA in the computer's memory. With this feature of the controller, the maximum accumulation is increased from 24 to 40 bits. Integration times in excess of 8 hours with a 20-MHz clock rate can be realized by this method.

The correlator is structured in groups of 64 channels each (one module or unit). There are nine modules (units) that make up the correlator. In normal operation only eight units are used (512 channels), leaving one spare. The controller can specify any number of units up to eight to be used. These are called good units.

At the end of a correlator run, the controller signals the computer it is done. This is accomplished with an interrupt. An interrupt is a message to the central processor of a computer indicating a need to communicate. The PDP 11 has four interrupt priority levels. The controller is programmable in this area and can be set to any level of interrupt priority by the user.

III. Correlator Controller Operation

Figure 2 is a block diagram of the 512-channel correlator controller. The computer address and control buffers, priority BR/BG NPR/NPG, and data buffers are Digital Equipment Corporation suggested circuits and components (Ref. 1).

The address decoder allows computer access to the controller's control and status register and internal registers. Two codes alert the controller to either a control message transfer or data transfer. The decoder consists of manually settable dip switches and an 18-bit comparator. With this method the controller may be accessible with any two adjacent addresses selected by the user, and preset into the dip switches.

The control and status register (CSR) is a 16-bit programmable register. It is a pointer and allows data entering the controller to be channeled to internal registers and to the individual correlator modules.

Additionally, it contains information on internal operations of the controller and correlator. When the end-of-run occurs, the four least-significant bits are automatically zeroed. This is done to allow the interrupt vector number contained in the interrupt vector address register to be presented to the PDP 11 central processor. The format of this register is in Fig. 3, and programming information can be obtained from Ref. 2.

The data multiplexer (MUX) receives its control information from the CSR and allows data from any of the internal controller registers or a selected correlator module to be read by the computer.

The interrupt vector register is a 16-bit programmable register that supplies the vector address during an interrupt at the end of a run.

The good units register is a 9-bit register with the format in Fig. 4. It controls the correlator unit counter by supplying information on which units are in use and allows the unit counter to stop at the proper unit during a run. The correlator unit counter controls end-of-overflow and which unit the overflow control is scanning. Either the PDP-11, by way of the control and status register, or the correlator unit counter supplies the unit address. The unit address source is selected by the PDP-11 or local address control multiplexer. When the correlator is running, the unit address is controlled by the correlator unit counter. When the correlator is being readied for a run or the data after a run is being unloaded, the unit address is supplied by the computer program through the controller control and status register.

The zero lag counter is a 40-bit-long high-speed counter which is loaded with the 1's complement of the *number* equal to the run time in seconds and multiplied by the clock frequency in hertz. Because the PDP-11 is a 16-bit word computer, the counter is segmented into three parts. It is segmented into 8-, 16-, 16-bit parts with the 8 bits being the most significant.

The start overflow sequence is initiated every 2^{18} count of the zero lag counter. Each correlator lag can store 2^{24} bits, and by running the controller overflow scanning faster ensures that no overflows in 512 lags will be missed. An additional overflow sequence is initiated when end-of-run occurs to ensure that no overflows happened since the last scan. The overflow scan time varies depending on how many accumulators overflowed. If all 512 bits overflowed, then the total scan time is approximately 1.2 ms. At the end of the zero lag count an interrupt is generated signalling the computer that a run is complete.

The overflow starting address register contains a 16-bit word, which is the starting address for the overflow array in the PDP-11 memory. This number is strobed into the 16-bit address counter at the beginning of each overflow sequence during a run. The address counter is incremented by the overflow control at the end of each lag "1" to "0" transition test.

The overflow control generates overflow read requests during a run and stores 16 overflow bits at a time in the overflow shift register. Each overflow bit is sequentially shifted from the shift register and tested for a "1" to "0" transition. This information is stored in a 512-bit scratch pad memory within the overflow controller. If an overflow is detected, the overflow controller generates a nonprocessor request (NPR) and upon receiving a nonprocessor grant (NPC) initiates a read request from the overflow array in the computer, stores these data in the overflow temporary storage register (overflow temp), increments overflow temp by one, and stores overflow temp back in the overflow array. If no overflows are detected, the computer array is not accessed.

Figure 5 is the format for the correlator bus function word. The correlator control buffer contains 12 line drivers for this bus. Additionally, the correlator control buffer contains line drivers for run control, high-speed clock, bus control strobe, connect clock, and the unit address. The bus control strobe and connect clock are signals generated during a load and control operation from the control and status register or from a read request from the overflow control.

Figure 6 is the format of the correlator data word. The correlator data register in the controller holds this 24-bit word while the computer reads data. The computer program first sets the control and status register to point to the high order 16 bits and then to the low order 8 bits.

The run control distributes the master clock to the zero lag counter and the correlator when commanded to be in run by the computer. The run control is turned off by an interrupt or a stop command from the computer. Additionally, the run control supplies both a true run and extended run signals. The true run disables the front end counters in the correlator so that additional data are suppressed; the extended run disables the master clock after the final overflow scan is completed.

The interrupt control can be programmed to act on any priority level. The interrupt control automatically zeros the four least-significant bits of the control and status register when an end-of-run occurs. This is done so that

the interrupt vector address is presented to the data bus during the interrupt. The interrupt control also sends a bus request and waits for a bus grant before executing the interrupt to the computer.

When the correlator is not in the run condition, all data transfer is under control of the computer. The controller has two addresses, one (164000) allows data to be transferred into and out of the controller control and status register (CSR). The other address (164002) allows data to be transferred from registers or modules being pointed to by the CSR. To read data from a correlator module, first set all units to the read mode by setting 13606₈ into the CSR. Follow this by a data transfer word from the correlator controller to the unit array within a program. The read request to correlator module via bit D₄ of the CSR must go to a "one" and then to a "zero" state between data transfer requests. The following is an example of a machine language program (Ref. 3). All numbers are in octal.

Example: Correlator Unit #1 Read

First set all correlator modules to read mode

```
MOV #13606 @ #164000 (sets all units
CLR @ #164002         to read mode)
```

Word

```
#1 MOV #14236 @ #164000 (send a read request to
                        unit #1)
```

```
#2 MOV #164002 @ #XXXX (XXXX = location; reads
                        data from 8 low order
                        bits into an array)
```

```
#3 MOV #14217 @ #164000 (satisfies 0 condition for
                        read request and pre-
                        pares to read high order
                        register bits)
```

```
#4 MOV #164002 @ #XXXX (XXXX = location;
                        reads data from 16
                        high order bits into an
                        array)
```

Each module contains its own address counter which is zeroed when a run is completed. To read all 64 lags, 64 read requests per module must be made. The first program word in the above example sends a read request to the module pointed to in the CSR word and the module unloads its 24-bit word into the correlator data register within the controller. The second word transfers the 8 *least*-significant bits from this register into an array within the computer. The third word sets the CSR pointer to read the 16 high order bits from the correlator data

register. The fourth word transfers the 16 *most*-significant bits from this register into an array within the computer.

IV. Summary

The correlator controller may assume the roll of master only after the computer has issued a run command. At that time the controller keeps track of overflows and ensures proper updating within the overflow array in the

computer's memory without interrupting the computer's processor. At the end of a run, the controller interrupts the computer and releases itself to operate as a slave.

The first operation of the 512-channel correlator was successfully accomplished during joint experiments in January 1976 between Aricibo and Goldstone, when Aricibo transmitted signals to Saturn and Goldstone received.

References

1. *Interfacing Handbook, PDP-11/05, 10, 35, 40*, Digital Equipment Corporation, 1974.
2. Jurgens, R. F., "Theory, Operation, and Computer Programming of the 512-Lag Correlator System," in *The Deep Space Network Progress Report 42-32*, pp. 164-177, Jet Propulsion Laboratory, Pasadena, Calif., Apr. 15, 1976.
3. *Processor Handbook, PDP-11/05, 10, 35, 40*, Digital Equipment Corporation, 1973.

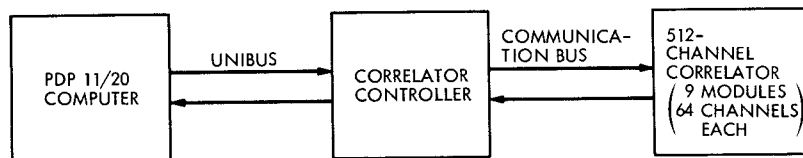


Fig. 1. Correlator system block diagram

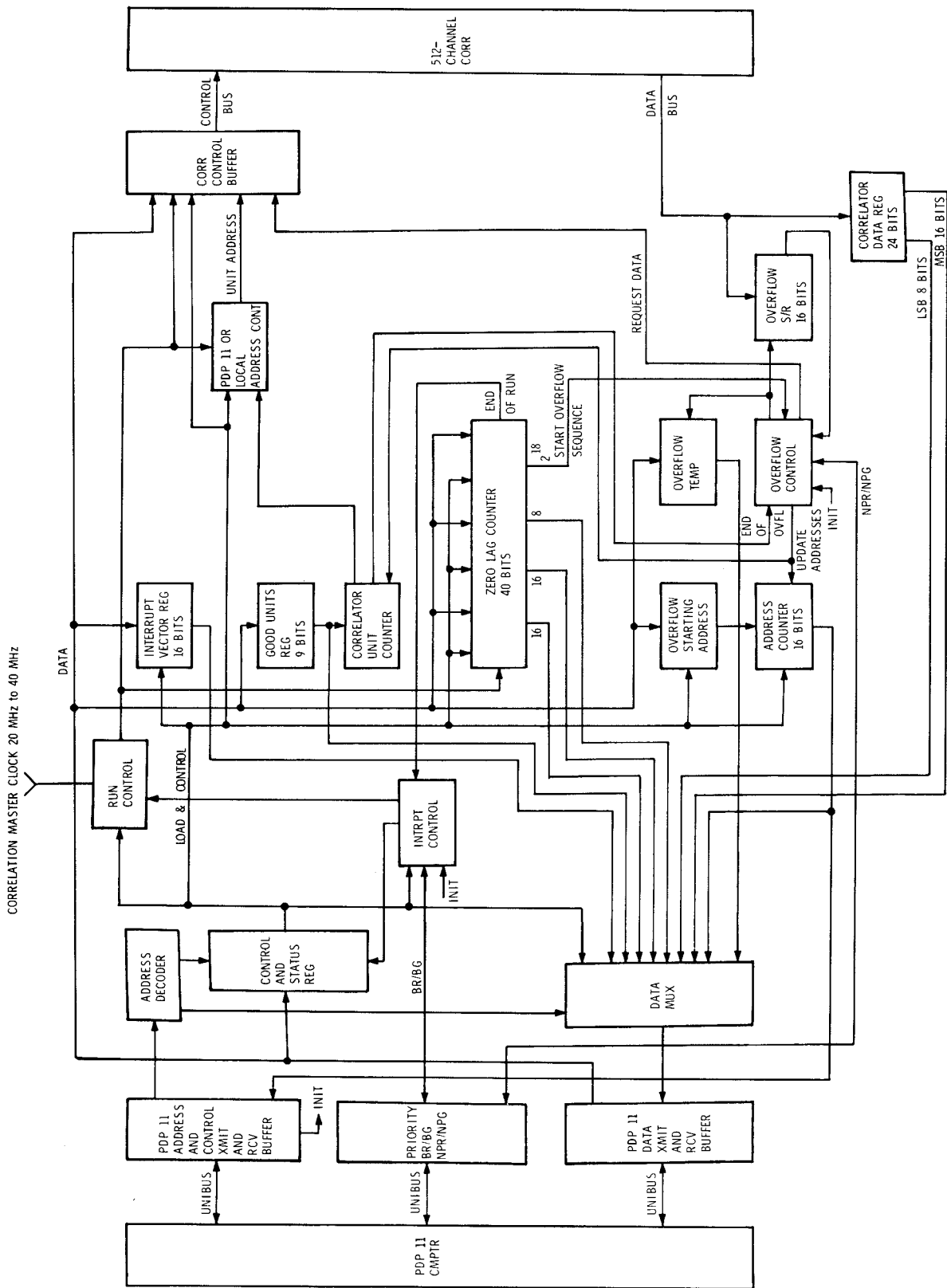
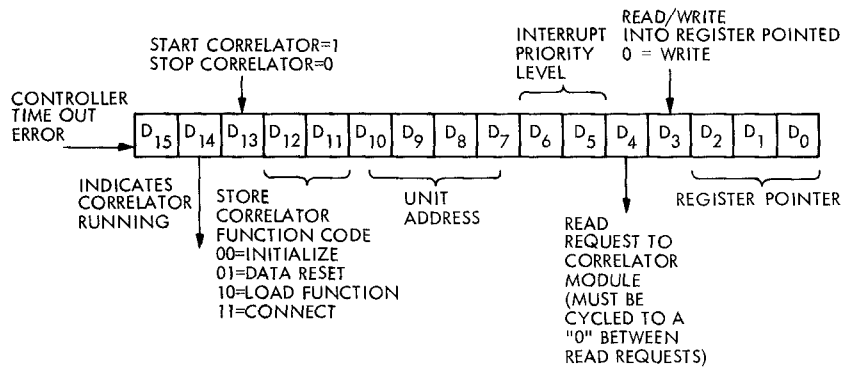


Fig. 2. 512-channel correlator controller block diagram



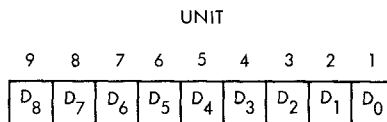
REGISTER POINTER CODES:

- 000 = INTERRUPT VECTOR 16 BITS
- 001 = GOOD UNITS 9 BITS
- 010 = OVERFLOW STARTING ADDRESS 16 BITS
- 011 = ZERO LAG COUNTER 16 LSB
- 100 = ZERO LAG COUNTER MIDDLE 16 BITS
- 101 = ZERO LAG COUNTER 8 MSB
- *110 = CORRELATOR LOW ORDER 8 BITS IN READ ONLY
- 111 = CORRELATOR HIGH ORDER 16 BITS IN READ ONLY

<u>INTERRUPT PRIORITY</u>	<u>UNIT ADDRESS</u>
00 = 4	0001 THRU 1001 = BCD UNIT 1 THRU 9
01 = 5	1111 = ADDRESS ALL UNITS
10 = 6	
11 = 7	

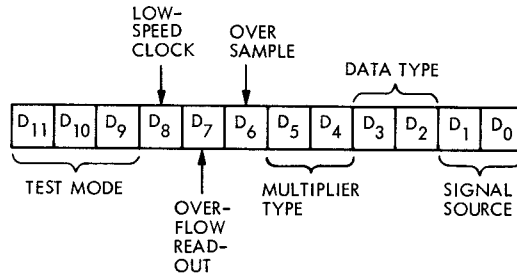
*IN THE WRITE MODE THE DATA TRANSFER PORT LOADS DATA DIRECTLY INTO THE UNIT ADDRESSED AND CAN ONLY BE READ IN THE OVERFLOW READ MODE. THIS ALLOWS LOADING OF THE 12 FUNCTION BITS IN EACH CORRELATOR MODULE.

Fig. 3. Control and status register format



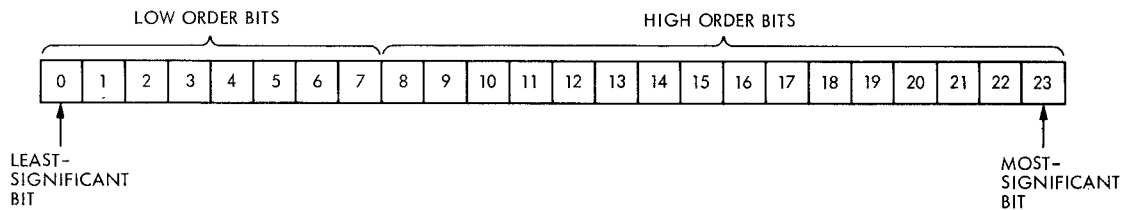
NOTE: A "1" IN ANY BIT POSITION INDICATES THAT THE UNIT IS GOOD.

Fig. 4. Good units register format



BIT ASSIGNMENTS		OCTAL CODE	FUNCTION
BITS 0-1	SIGNAL SOURCE	0	FROM STAGE N-1
		1	FROM STAGE N-2
		2	FROM DATA SAMPLER
		3	FROM AUX UNIT
BITS 2-3	DATA TYPE	0	REAL TWO LEVEL
		1	REAL THREE LEVEL
		2	COMPLEX TWO LEVEL
		3	COMPLEX THREE LEVEL
BITS 4-5	MULTIPLIER TYPE	0	TWO LEVEL
		1	THREE LEVEL
		2	FOUR LEVEL
		3	FIVE LEVEL
BIT 6	OVERSAMPLE	0	NYQUIST RATE
		1	2 x NYQUIST RATE
BIT 7	OVERFLOW READ	0	DATA READ MODE
		1	OVERFLOW READ MODE
BIT 8	LOW-SPEED CLOCK	0	HIGH-SPEED CLOCK
		1	LOW-SPEED CLOCK
BITS 9-11	TEST MODES	0	OPERATE
		1	DISABLE DATA BUS
		2	BUS TRANSMIT 1's
		3	BUS TRANSMIT 0's
		4	MEMORY LOAD 1's
		5	MEMORY LOAD 0's
		6	SPARE
		7	FAULT INDICATOR ON

Fig. 5. Correlator bus function word format



- NOTES:
1. IN THE READ MODE THE 24-BIT REGISTER REPRESENTS ONE ACCUMULATOR CHANNEL. 64 READ REQUESTS ARE REQUIRED TO UNLOAD EACH MODULE.
 2. IN THE OVERFLOW READ MODE THE 16 HIGH ORDER BITS REPRESENT THE MOST-SIGNIFICANT BIT OF EACH ACCUMULATOR CHANNEL. FOUR OVERFLOW READ REQUESTS ARE REQUIRED TO UNLOAD EACH MODULE. THE 8 LOW ORDER BITS REPRESENT D₀ THROUGH D₇ OF THE CORRELATOR BUS FUNCTION REGISTER FOR THE MODULE BEING UNLOADED.

Fig. 6. Correlator data word format