SDC-93-412

SDC
SOLENOIDAL DETECTOR NOTES

SDC PATTERN GENERATION MODULE (PGM)

PERFORMANCE SPECIFICATION

February 15, 1993

T. Gorski, J. Lackey, S. Dasu and W. H. Smith
University of Wisconsin

Contents

1 Introduction **1**

2 General Overview **1**

2.1 Block Diagram of PGM 1

2.2 TEM Interconnection 4

3 Trigger Control Bits **5**

4 PGM Frontpanel **6**

5 Functional Descriptions **8**

5.1 VME Slave Interface 8

5.1.1 Address Modifier Support 9

5.1.2 Slot Address Decoding 9

5.1.3 On-Board Busses 9

5.2 VME Bus Interrupter/Status Logic 9

5.2.1 Status Register Bit Definitions 10

5.2.2 VME IRQ Generation 11

5.2.3 IRQ Level Support and Interrupt Source ID 12

5.3 Reset/Configuration Logic 12

5.3.1 Configuration Register Bit Definitions 12

5.3.2 Board Resets 14

5.4 Pattern Generation State Machine 14

5.4.1 Descriptor Word Format 15

5.4.2 Defined Machine States 16

5.4.3 State Machine Architecture 18

5.4.4 Interrupt Branch Accept Rate 21

5.4.5 VME Memory Access 22

5.5 Interrupt Branch Input Logic 22

5.5.1 VME Write Port 23
5.5.2 Trigger A and B Interrupt Inputs 23
5.5.3 Vectored Interrupt Input 24
5.5.4 Interrupt Block Overrides 24
5.6 Reject Counter/Interval Timer Logic 25
5.6.1 Reject Counter Operation 25
5.6.2 Interval Timer Operation 26
5.6.3 Counter/Timer Readout 26

6 Programming Examples **27**
6.1 Example 1: Repeating L1A/L2A Sequence 27
6.2 Example 2: L2 Accepts/Rejects via Trigger Interrupts 29
6.3 Example 3: L1 Accept/Reset Sequences via Vectored Interrupts 31

A Backplane Signal Definitions **33**

B VME I/O Address Map **36**
B.1 Configuration Register 36
B.2 VME Interrupt ID Register 36
B.3 Status Bit Clear Mask 36
B.4 "Power-Up" Reset Strobe 39
B.5 Counter Clear Strobe 39
B.6 Pattern Generator VME Interrupt Write Port 39
B.7 Interval Timer Preload 39
B.8 BC0 Counter Preload Register 40
B.9 Status Register 40
B.10 Trigger A Interrupt Reject Counter 40
B.11 Trigger B Interrupt Reject Counter 40
B.12 Interval Timer 41

C Cable Interconnections **41**

C.1 Cable Termination 41

C.2 PGM Input and Output Connector Pinouts 41

C.3 Interconnection with TEM 43

D Specification Summary **44**

1 Introduction

This document outlines the specifications of the SDC Pattern generation Module (PGM). The PGM is designed to work with a second module, the Trigger Emulation Module (TEM)[1] to emulate much of the clock and control functionality anticipated for the final SDC Global Trigger System. The PGM uses programmable memory lookup tables and state machines to generate long and potentially complicated sequences of Clock and Control input data to drive the TEM. These sequences run at the full 60 MHz clock speed and provide detailed tests of the interface between trigger, DAQ and front end electronics. The purpose is to aid in the test bench debugging of subsystem front end electronics, DAQ and trigger circuits, and to serve the basic functions of a trigger module for test beam experiments. The TEM and PGM are designed to operate as part of a test of front end electronics, where they provide patterns of Level 1 and 2 Trigger and clock signals. These patterns can be started or changed via external signals from pulses on cables or a DAQ processor.

The PGM is programmed to provide a pattern of Level 1 Accept/Reject, Level 2 Accept, Level 2 Reject, Level 1 Synch, Level 1 Reset, and Bunch Crossing 0 signals. It also carries a bit that is used to stop the clocks. The bits that indicate the state of these signals are composed by sequences written into memory that are selected by descriptor words written in other memories. Sequences can be repeated and, when concluded, jump to additional sequences. Sequences can be interrrupted and changed by external signals or the DAQ processor. The PGM sequences are programmed by the DAQ processor and played back at 16 nsec speed for a practically limitless duration. The clock can be taken from the TEM or from another external source. If desired, PGM sequences can be set to be uninterruptable by external signals and unserviced interrupts are then counted.

2 General Overview

The PGM is a 6U high, 160mm long, .084" thick (as opposed to the standard .062" thickness) VME card. The card contains a standard A24:D16 VME slave interface, with the ability to generate VME bus interrupts.

Like the TEM, the card uses a mixture of ECL and TTL logic. For power, the card uses +5V, -5V, and -2V, which it receives from the crate via the backplane connectors P1 and P2. The -5V and -2V ECL voltages are applied to the card on P2 from a power adaptor card which plugs onto the other side of the backplane. This card is discussed in the TEM specification[1].

2.1 Block Diagram of PGM

Figure 1 shows an overall block diagram of the PGM. The diagram shows the six major sections of the card: the VME Slave Interface, the VME Bus Interrupter/Status Logic, the

Reset/Configuration Logic, the Pattern Generation State Machine, the Interrupt Branch Input Logic, and finally the Reject Counter/Interval Timer Logic.

The VME Slave Interface provides the link between the other sections of the PGM and the DAQ Processor. It does this through the decoding of VME read/write transfers into individual read/write strobes, and through the distribution of those strobes and local address and data busses throughout the card. Like the TEM[1], the slave is implemented for standard A24:D16 VME bus operation.

The VME Bus Interrupter/Status Logic section contains the status register for the card and the logic to generate a VME bus interrupt. VME interrupts can be generated as a result of a number of events on the board reported in the status register, including the overflow of an on-board Interval Timer.

The Reset/Configuration Logic section controls the operation of the PGM through a 16-bit configuration register. This section also distributes resets to the other sections of the module. The PGM recognizes two types of resets: an all-encompassing "power-on" reset, which resets the board, including most VME-programmable registers, but excluding the on-board RAMs, to a known state, and a "pushbutton" reset, which resets the board as above, except for any of the VME-programmed registers.

The Pattern Generation State Machine (PGSM) comprises the functional heart of the PGM. It is in this section that the Trigger Control word pattern stream is generated and then transmitted to the TEM. Pattern generation is controlled by the programming of two separate VME-accessable memories: a 512 $\times$ 32-bit *Descriptor Word Memory*, and a 4096 $\times$ 8-bit *Pattern Memory*. The Descriptor Word Memory (DWM) contains up to 512 Descriptor Words, each of which references a range of addresses in the Pattern Memory (PM), in which a pattern of Trigger Control Words resides, such as a Level 1 Accept trigger sequence. Descriptor Words (DW's) can be linked together in DWM to generate long, continuous streams of Trigger Control word patterns. After the DAQ Processor has initialized the two memories, the Pattern Generation State Machine "executes" the programmed pattern, as specified in the DWM and PM memories, delivering the output pattern stream to the TEM through a frontpanel connector.

The state machine also allows for the delivery of the current pattern to be interrupted, and for the delivery of a new pattern segment to start. This capability makes it possible for the PGM to respond to an external stimulus with an immediate, programmable, and predetermined Trigger Control word pattern response. The interrupt may come from any of several sources, including two edge-sensitive TTL frontpanel inputs, called Trigger Interrupts A and B, a separate, "Vectored Interrupt" input also on the frontpanel, and finally, the DAQ Processor via the VME interface. Each of these interrupts can result in a different pattern being started. The two Trigger Interrupts each point to a fixed address in Descriptor Word Memory. The "Vectored Interrupt" can point to any of 16 addresses in Descriptor Word Memory, depending on the value of the 4-bit vector code which this input receives. This code input to the PGM from an external subsystem allows that subsystem to force a different pattern sequence from the PGM, in response to an error or initialization request, for example. Through the VME Interface, the DAQ Processor may force an interrupt branch to any of the 512 locations in Descriptor Word Memory.

PATTERN GENERATOR MODULE
OVERALL BLOCK DIAGRAM

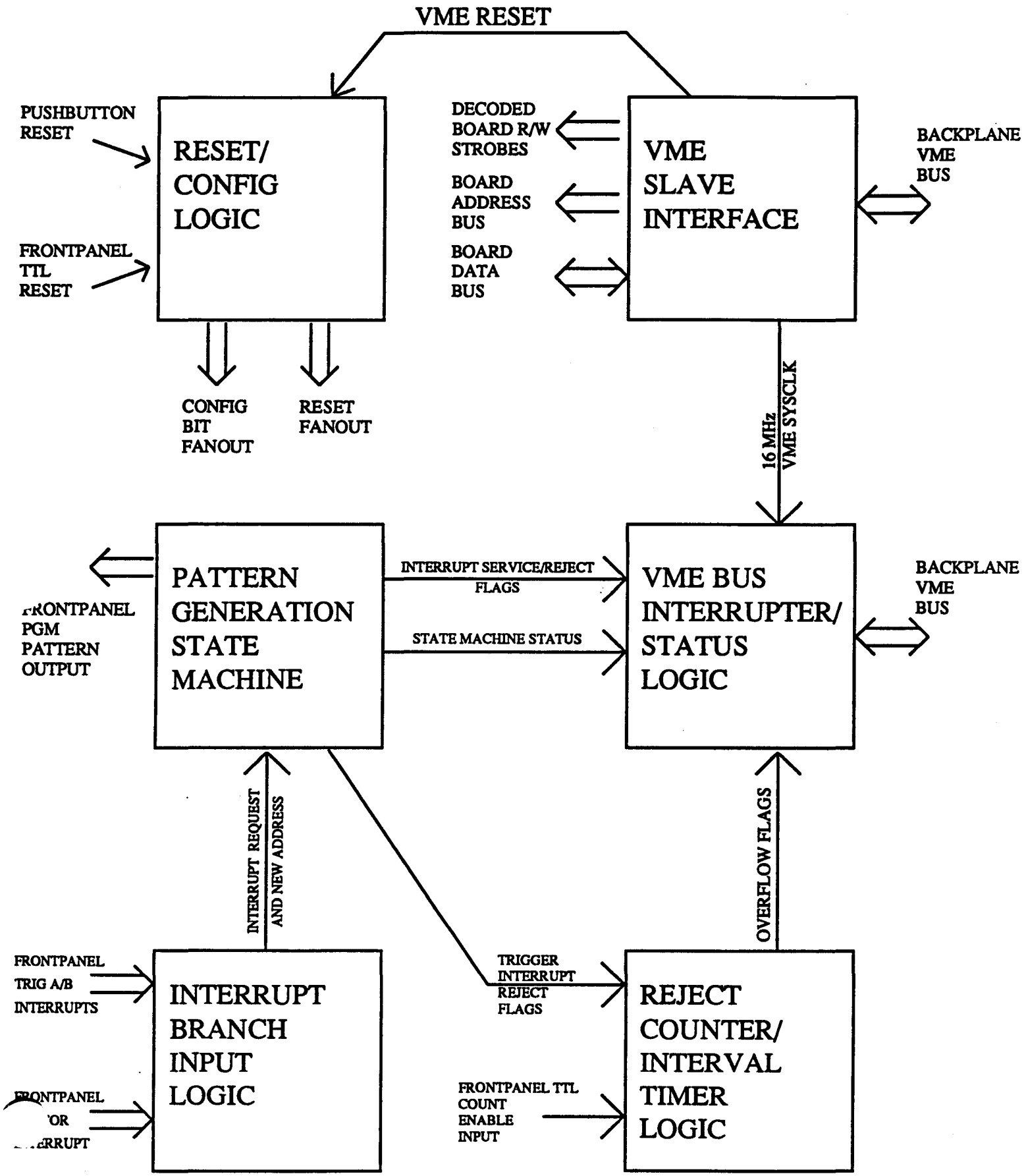


FIGURE 1

PGM Overall Bloack Diagram

The interrupt sources are individually enabled under control of the DAQ Processor via the configuration register. Additionally, each DW can be defined so as to enable or disable interruption of the pattern associated with it. Together, these capabilities can be used to safeguard the integrity of patterns which should not be interrupted.

Finally, the PGSM contains a special counter for driving the Trigger Control bit 6, Bunch Crossing 0. In addition to being driven like all other bits from pattern memory, this bit may be driven by the Bunch Crossing 0 counter. This is a 16-bit free-running counter with a user-programmable wraparound time which counts clock cycles. When enabled, this counter drives the Bunch Crossing 0 bit for one cycle each time it wraps around, with the wraparound cycle time being programmable from every 2 clocks to every 65536 clock cycles.

The Interrupt Branch Input Logic receives the frontpanel and VME interrupt requests, and redirects them to the Pattern Generation State Machine for action. Each interrupt branch source has associated with it a different Descriptor Word address or range of addresses. Each of the Trigger Interrupt sources has a fixed address. The DAQ Processor via VME can force a branch to any of the 512 Descriptor Word addresses, while the frontpanel Vector Interrupt source can force a branch to any of 16 different locations, depending on the value of the 4-bit "Vector" which comprises the input from this source.

The Reject Counter/Interval Timer Logic section contains two 16-bit event counters and a timer. The purpose of this logic is to provide a correction to the "dead time" calculations made by the Event Counters on the TEM[1], in the special case where a PGM Trigger Interrupt is blocked by the state machine. The two independent event counters tally the number of Trigger A and Trigger B Interrupts which are rejected by the TEM. The interval timer, meanwhile, can be used by the DAQ Processor to measure the time envelope over which the reject tally occurred.

For example, if Level 1 Accept sequences are *not* generated because the Trigger Interrupt that would have initiated it is frequently blocked, then the associated Reject Counter can be used to correct the total number of received Level 1 Accepts counted by the TEM.

2.2 TEM Interconnection

The principal connection between the TEM and the PGM occurs between the *Clock and Control Input connector* on the TEM and the *PGM Output connector* on the PGM, as shown in Figure 2. Across this link the TEM receives the Trigger Control Word pattern stream from the TEM and delivers a copy of the high-speed clock to the PGM. The PGM uses this high-speed clock as its sole timebase source. All connections on this link use differential 10K ECL logic levels.

When the TEM is used standalone (with no PGM) and with an external clock, that clock is applied directly to the appropriate pins on the TEM Clock and Control Input connector[1]. When the TEM is used with a PGM and also with an external clock, that clock is applied to the PGM Input connector, passed through the PGM without being referenced, and sent to the TEM on the link described above. This scheme prevents the

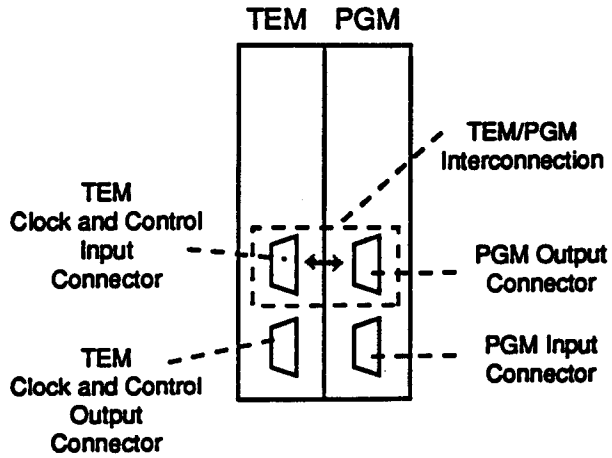


Figure 2: TEM/PGM Interconnection

Bit Number	Definition
0	Level 1 Accept
1	Level 2 Accept
2	Level 2 Reject
3	Level 1 Sync
4	Level 1 Reset
5	Bunch Crossing 0/Spare
6	Empty Bunch/Test Trigger
7	Clock Stop

Table 1: Trigger Control Word Bit Definitions

user from having to tap into the cable link from the PGM to the TEM in order to provide the external timebase. Note that even if the TEM receives the external timebase through the PGM, the PGM *always* receives the timebase that it uses *from* the TEM.

3 Trigger Control Bits

The function of the PGM is to provide a programmed stream of Trigger Control bit patterns to the TEM, for subsequent distribution to the front end electronics. As such, the PGM adheres to the same Trigger Control bit definitions as specified for the TEM[1]. These defintions are shown in Table 1. However, unlike the TEM, with the exception of the Bunch Crossing 0 bit, on the PGM the Trigger Control bits have no special individual functionality associated with them. Rather, the PGM treats them as generic data bits in its output pattern stream, with all bits being considered identical to each other. Any

and all special functions in hardware associated with these bits, such as gating Level 1 Accepts, or stopping the clock for example, occur downstream of the PGM, on the TEM.

The Bunch Crossing 0 bit is the exception because of the Bunch Crossing 0 counter on the PGM. This counter ORs a 1-cycle pulse onto the Bunch Crossing 0 bit at the PGM output, every time the counter wraps around, permitting this bit to be used by front end electronics as a sync source.

4 PGM Frontpanel

Figure 3 shows the frontpanel of the PGM card. The frontpanel has numerous LEDs to indicate the card status. It also has a pushbutton reset, and TTL inputs for two Trigger Interrupts, a remote reset, and a remote Counter enable. Finally, the PGM has a 44-pin "D" connector for driving the Trigger Control output pattern as differential ECL to the TEM, and a 25-pin "D" connector for receiving a 4-bit vectored interrupt, also as differential ECL. Each frontpanel item in Figure 3 is described below.

Pushbutton Reset. This reset switch provides a convenient means of resetting the PGM during use. This reset halts the pattern generation state machine, clears the status register, any pending VME interrupt, the Trigger Interrupt Reject counters, and the Interval Timer. This reset does not, however, affect the state of the board's configuration or interrupt vector registers.

RESET LED. This red LED illuminates while the PGM is disabled via the configuration register, or while the pushbutton or frontpanel-connector reset is active.

VME I/O LED. This green LED flashes momentarily each time the PGM services a VME bus cycle initiated by the DAQ Processor.

HALT LED. This red LED is lit when the pattern generation state machine is halted. While the state machine is halted, the Trigger Control word pattern output at the PGM output connector is all zeroes.

RUN LED. This green LED is lit when the pattern generation state machine is running.

INT LED. This amber LED flashes momentarily each time the pattern generation state machine takes a forced branch as a result of seeing an interrupt.

Trigger Interrupt Inputs. These are TTL logic levels, Lemo 00-type inputs. A rising edge on either of these inputs will cause the pattern generation state machine to immediately branch to start generating a new user-programmed Trigger Control output pattern. The inputs operate independently of each other, and are individually enabled from the configuration register which is controlled by the DAQ processor. A separate pattern can be programmed for each input.

An example configuration could use the TRIG A interrupt to initiate sequences containing a Level 1 Accept followed by a Level 2 Accept, and the TRIG B interrupt to initiate sequences containing a Level 1 Accept followed by a Level 2 Reject.

PGM FRONTPANEL

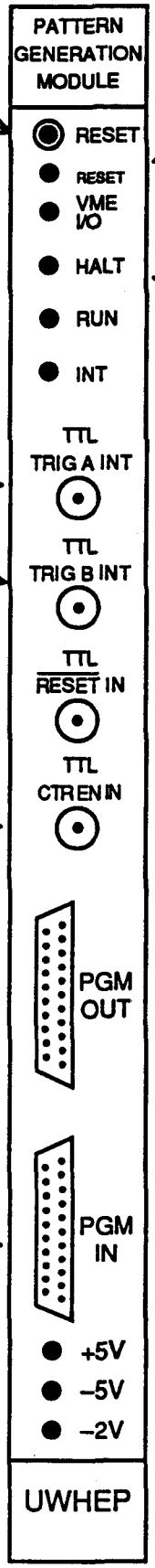
Pushbutton Reset.
Halts pattern generation, clears status register, interrupts reject counters and interval timer.

Run LED.
Lit to indicate pattern generation state machine is running.

Trigger Interrupt Inputs.
Independent inputs to force interrupt branch to new pattern sequence. Uses TTL inputs and Lemo 00 Connectors.

Counter Enable Input.
Enables Trig Interrupt Reject Counters and Interval Timer. Uses TTL logic levels and Lemo 00 Connector.

PGM Input Connector.
Receives 4-bit vector and Load strobe for vectored interrupt branches to new pattern sequences. Also provides alternate input point for external clock to TEM. Uses differential ECL logic levels.



RESET LED.
On during pushbutton or external reset or when PGM disabled by DAQ Processor.

VME I/O LED.
Flashes for each VME bus cycle serviced by card.

HALT LED.
Lit to indicate pattern generation state machine is halted.

INT LED.
Flashes to indicate state machine was interrupted to start new pattern.

Reset Input.
Same as pushbutton reset, except from remote or electronic source. Uses TTL logic levels and Lemo 00 Connector.

PGM Output Connector.
Connets to TEM Clock and Control Input. Delivers Trigger Control word pattern to TEM, and receives RF clock from TEM. Uses differential ECL logic levels.

Voltage Source LEDs.
Indicate blown fuse on card for respective voltage when not illuminated.

FIGURE 3

Reset Input. This is a TTL logic, Lemo 00-type input. A logic-low level on this input causes the same reset operation on the card as pressing the pushbutton reset frontpanel switch.

Counter Enable Input. This is a TTL logic, Lemo 00-type input. A logic-high level on this input enables the Trigger Interrupt Reject Counters and the Interval Timer. This signal may be tied to the same spill gate output or similiar source that is used to drive the Counter Enable Input on the TEM, in order to ensure that the same counting interval is used for both modules.

PGM Output Connector. The PGM Output Connector provides all links to the TEM Clock and Control Input connector on the same 44-pin, high-density, subminiature-D header as is used by the TEM, using differential ECL signal pairs. On this connector the PGM delivers the 8-bit Trigger Control output stream from its Pattern Memory to the TEM. Also on this connector, from the TEM, the PGM receives a copy of the TEM high-speed clock[1] which it uses as its own timebase. The PGM may also deliver the external timebase source to the TEM through this connector. The PGM Output Connector is mounted at the same position on the PGM as is the Clock and Control Input Connector on the PGM.

PGM Input Connector. The PGM Input Connector is a 25-pin, D-type header, which receives a 4-bit Vector and Load Strobe for Vectored interrupts of the PGM, and the TEM External Clock as explained previously in the document, as ECL differential pairs.

The Vectored Interrupt feature of the PGM allows an external module to request an interrupt branch to any of 16 different user-programmed pattern sequences, depending on the value of the input 4-bit vector code. The interrupt branch request is triggered by a rising edge on the load strobe. The Vectored Interrupt input is described in greater detail in the Interrupt Branch Input Logic section.

Voltage Source LEDs. These are green LEDs which when lit indicate the presence of the specified voltages on the card. Each of these voltages is fused on the PGM. Should a fault occur which causes any fuses to open, or if any of the power supplies fail, then the LEDs for the affected voltages will go out.

5 Functional Descriptions

In this section, the operation of the six main functional blocks shown in Figure 1 are further explained.

5.1 VME Slave Interface

The PGM is equipped with a VME slave interface which is compatible with standard IEEE 1014-1987.

The card acts as an A24:D16 slave. As such, it supports both D16, D08(Even), and D08(Odd) transfers. The appropriate transfer width for a bus cycle depends on the location being accessed within the card address space. The card supports Address-Only cycles[3], but *does not* support block transfers. It does support read-modify-write cycles.

5.1.1 Address Modifier Support

All I/O addressable space on the card is accessable in Standard Address Space, for Data Access. Thus, the card supports bus data transfer cycles only using Address Modifier Codes 39 hex (Standard Non-Priviliged Data Access), and 3D hex (Standard Supervisory Data Access). Address Modifier Codes are driven on lines AM0–AM5 of the VME Bus.

The card ignores bus requests associated with any and all other Address Modifier codes.

5.1.2 Slot Address Decoding

The upper 10 bits of the 24-bit address space are interpreted by the card as the *slot address*. The PGM is configured by the user for operation at a particular slot address through on-board Card Address jumpers. The card will service a data bus transfer request only when the card detects both a match in the slot address, and a recognized Address Modifier Code on the bus. Appendix B shows the map for the address space owned by the card.

5.1.3 On-Board Busses

The card has an on-board 16-bit data bus to support VME accesses. This bus is implemented in TTL, and is only active during VME bus cycles. Additionally, the card has an on-board address bus to support read/write access to the Descriptor Word and Pattern Memories in the Pattern Generation State Machine section. Again, this bus is implemented in TTL, with some trunk extensions into ECL. It is active only during VME bus cycles accessing the memories.

5.2 VME Bus Interrupter/Status Logic

This block contains the status register and the circuitry for generating VME bus interrupt requests. VME interrupts can be generated by the PGM as a result of a zero-to-one transition on any of 7 particular status bits, as described below.

Bit Number	Definitions
0–1	PGSM Status Code: 0 = Reset Halt (held in reset state) 1 = DW Halt (halted by DW Halt Bit) 2 = Running (generating patterns) 3 = Wait Int (waiting for start interrupt)
2	PGSM Vectored Interrupt Branch Serviced (Maskable VME IRQ Source)
3	PGSM Vectored Interrupt Branch Rejected (Maskable VME IRQ Source)
4	PGSM Trigger A Interrupt Branch Serviced (Maskable VME IRQ Source)
5	PGSM Trigger B Interrupt Branch Serviced (Maskable VME IRQ Source)
6	Interval Timer Overflow (Maskable VME IRQ Source)
7	Trigger A Reject Counter Overflow (Maskable VME IRQ Source)
8	Trigger B Reject Counter Overflow (Maskable VME IRQ Source)
9	Reject Counters/Interval Timer Enabled
10	Slow Clock Prescaler Jumper Installed (2^{13} Divide Rate Selected)
11	PGM Reset Since Last Status Register Read
12	PGSM VME Interrupt Branch Serviced
13	PGSM VME Interrupt Branch Rejected
14–15	Spare (Zero)

Table 2: PGM Status Register Bit Definitions

5.2.1 Status Register Bit Definitions

The PGM has a 16-bit status register which is readable by the DAQ Processor via the VME Interface. Table 2 contains the bit definitions for this register.

The Pattern Generation State Machine (PGSM) Status Code bits (bits 0–1) reflect the instantaneous state of the Pattern Generation State Machine, latched at the time of the status register read. Bits 2–5 and 12–13 show the outcome, either a service or reject, of attempts to interrupt the PGSM from their respective sources. All of these status bits are latched by the PGM hardware, meaning that once set, they stay set until explicitly cleared by the DAQ Processor via a write to the Status Bit Clear Mask. Operation of this mask is described in Appendix B.

Bits 6–8 reflect the overflow status of their associated counter or timer. The two reject

counters are automatically incremented for each occurence of a reject on their associated frontpanel Trigger Interrupt. They are 16-bit counters, and therefore will overflow after 65535 counts. Similiarly the Interval Timer uses a 16-bit counter, which also overflows after 65535 counts. The timer is driven by a "slow clock", which is a prescaled version of the same high-speed basic board clock used by the TEM and sent by the TEM to the PGM. As on the TEM, a jumper on the PGM allows the user to select between two different divide rates for the slow clock. Status bit 10 shows the position of the jumper. When the jumper is installed, a divisor of 2^{13} is used to make the slow clock. When the jumper is not installed, a divisor of 2^{17} is used. At an example high-speed clock rate of 60 MHz, the interval timer will overflow in approximately 8.94 seconds with a slow clock divisor of 2^{13}, and in approximately 143 seconds with a divisor of 2^{17}. The three overflow status bits are all latched, and once set, can only be cleared by a board reset, or Counter Clear strobe generated by the DAQ Processor.

Bit 9 shows the instantaneous status of the Count Enable signal for the Interval Timer and Reject Counters, latched at the time of the status register read. The Count Enable is made by logically ANDing the frontpanel Count Enable input with the Force Disable bit of the config register.

Bit 11 is set each time the card receives a reset, either through VME or the frontpanel, and is automatically cleared by a status register read.

5.2.2 VME IRQ Generation

As stated previously, VME interrupt requests can be generated by the PGM in response to a zero-to-one transition on any of 7 selected status register bits. In particular, bits 2–8 can be selected to cause a VME IRQ.

The mechanism for causing such a request operates as follows. Each status bit which can cause an interrupt has an associated mask enable bit in the config register. Each status bit and mask enable bits are logically AND'ed together, and the result is logically OR'ed with the AND results from the other status bits to produce a single request signal to the interrupt logic on the PGM. Those status bits selected to cause an interrupt have their associated mask enable bit set in the configuration register. When a status bit so enabled has a zero-to-one transition, it is seen by the interrupt logic and used to generate a VME IRQ.

Note that in this scheme, once an interrupt is generated, no subsequent VME interrupts can be generated, from any other status bit transitions until either:

1. The status bit causing the interrupt is cleared, or
2. the status bit is disabled as an interrupt source by clearing its mask enable bit.

This is due to the fact that the interrupt logic needs to see a zero-to-one transition coming out of the OR of the masking logic in order to generate a new interrupt. If the OR network

is still held high by the original interrupt source status bit, then any subsequent transitions by other bits will be impossible to detect.

The interrupt logic uses the Release on Acknowledge (ROAK) method to deassert the interrupt request line[3]. In this method, the active request signal on the backplane bus is automatically deasserted by the PGM during the acknowledge cycle for that interrupt, even though the on-board request line to the interrupt logic is still asserted.

5.2.3 IRQ Level Support and Interrupt Source ID

The PGM supports generating interrupt requests at any of the 7 VME bus priority levels[3]. The priority level used for interrupt request and acknowledge cycles is selected via jumpers on the card.

The card has an 8-bit Interrupt ID register which provides the bus master (in this case the DAQ Processor) with an interrupt source identifier during the interrupt acknowledge cycle. This register must be programmed with a valid user-defined ID number by the DAQ Processor prior to using VME interrupts.

5.3 Reset/Configuration Logic

This section contains the PGM configuration register and the logic for distributing fanout signals across the card.

5.3.1 Configuration Register Bit Definitions

The PGM has a 16-bit configuration register which is both writeable and readable by the DAQ Processor via the VME Interface. Table 3 contains the bit definitions for this register.

Bit 0 is the master enable bit for the PGM card. When this active-high bit is low, the card is held in a "pushbutton reset" state, which is described below. Bit 1 provides a forced-disable for the Reject Counters and Interval Timer. When this bit is zero, the counters and timer are disabled, regardless of the state of the frontpanel Count Enable input. When this bit is one, the counters and timer are enabled or disabled under the control of the frontpanel Count Enable input.

Bits 2–8 are the VME IRQ Mask Enable bits for the associated status register bits. These bits are set to one to enable a status bit as an VME interrupt source, and set to zero to block the status bit from generating a VME interrupt.

Bits 9–11 enable the frontpanel input sources for interrupt branches to the Pattern Generation State Machine. When these bits are at a logic zero, any signals on their associated inputs are ignored. When these bits are a logic one, a rising edge on either of the Trigger Interrupt inputs or a load strobe (also a rising edge) to the Vectored

Bit Number	Definitions
0	PGM Enable (0 = disabled, held in "pushbutton" reset state, 1 = enabled)
1	Reject Counter/Interval Timer Force Disable (0 = disabled, 1 = enabled/controlled by frontpanel input)
2	Status Bit 2 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
3	Status Bit 3 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
4	Status Bit 4 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
5	Status Bit 5 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
6	Status Bit 6 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
7	Status Bit 7 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
8	Status Bit 8 VME IRQ Mask Enable Bit (0 = no VME IRQ, 1= VME IRQ when status bit sets)
9	Trigger A Interrupt Branch Input Enable (0 = disable input, 1= enable input for pattern interrupts)
10	Trigger B Interrupt Branch Input Enable (0 = disable input, 1= enable input for pattern interrupts)
11	Vectored Interrupt Branch Input Enable (0 = disable input, 1= enable input for pattern interrupts)
12	Bunch Crossing 0 Counter Enable (0 = disabled, 1 = enabled)
13–15	Spare

Table 3: PGM Configuration Register Bit Definitions

Interrupt input will cause an attempted interrupt branch to a new pattern by the Pattern Generation State Machine.

Bit 12 is the enable for the Bunch Crossing 0 Counter. When this bit is a logic zero, the counter is disabled, and does not affect the Bunch Crossing 0 bit. When this bit is a logic one, the counter is enabled and asserts the Bunch Crossing 0 bit for one cycle every time it wraps around. Whether the counter is enabled or not, the Bunch Crossing 0 bit may always be driven from Pattern Memory.

The PGM powers up with the config register in the all-zero state. This means that the PGM is disabled, and must be initialized by the DAQ Processor prior to operation.

5.3.2 Board Resets

The PGM has 2 classes of resets, which are stacked in a hierarchy. The broadest in scope is the "power-on" reset, which resets the config register to zero, clears the status register, reject counters, and Interval Timer, and halts the Pattern Generation State Machine.

Of slightly lesser scope is the "pushbutton" reset, so-named as one of its sources is the reset button on the frontpanel. This reset covers the same circuitry as the "power-on" reset, with the exception that it does not affect the config register. The sources for these two resets are listed below.

Not covered by either reset are the Descriptor Word and Pattern Memories. These memories are undefined at power-up, and must be explicitly written by the DAQ Processor to be set to a known state.

Each reset type can originate from a number of different sources. The "power-on" reset can come either from the VME backplane SYSRST signal during crate powerup or reset, or from a decoded VME I/O strobe aptly called the "Power-Up" Reset Strobe. The DAQ Processor can assert this strobe at any time to induce this reset. See Appendix B for a further discussion of this strobe. The "pushbutton" reset has 3 possible sources: the frontpanel pushbutton, the frontpanel TTL Reset input, and the PGM Enable bit in the config register. The reset will be asserted for as long as the pushbutton is pressed, or for as long as a logic zero is present at either the frontpanel Reset input or PGM Enable bit.

5.4 Pattern Generation State Machine

As mentioned previously, the Pattern Generation State Machine (PGSM) comprises the functional heart of the PGM. It is in this section that the Trigger Control word pattern stream is generated and then transmitted to the TEM. This section contains two VME-accessable memories, a 4096 $\times$ 8-bit Pattern Memory (PM), which contains the pattern segments to be transmitted to the TEM, and a 512 $\times$ 32-bit Descriptor Word Memory (DWM), which contains the programmed instructions, called Descriptor Words, for the state machine.

VME Byte #	Byte 0			Byte 1	Byte 2	Byte 3
DW Bit #	3 1	3 0	2 2 2 2 2 2 9 8 7 6 5 4	2 2 2 2 1 1 1 1 3 2 1 0 9 8 7 6	1 1 1 1 1 1 0 0 5 4 3 2 1 0 9 8	0 0 0 0 0 0 0 0 7 6 5 4 3 2 1 0
Field Def.	H A L T	I B L K	Pattern Length (2..65)	Pattern Start Address (0x000..0xFF0)	Next DW Address (0x000..0x1FF)	Loop Count (1..128)

Figure 4: Descriptor Word Format

5.4.1 Descriptor Word Format

Figure 4 shows the field definitions for a Descriptor Word. Each Descriptor Word contains 6 fields, which control all aspects of state machine operation. The state machine executes one Descriptor Word at a time, until either completing transmission of the pattern sequence specified by that word, including repetitions of the pattern, or until an interrupt branch is received, causing execution to start on a new word. The use of Descriptor Words by the state machine is similiar to treatment of instructions by a microprocessor. As a microprocessor executes instructions to produce computational results, the Pattern Generation State Machine executes Descriptor Words to produce Trigger Control word pattern sequences. However, unlike the microprocessor, which may have many different instructions in numerous formats, the state machine has only one format for its one "instruction"–specifically that of the Descriptor Word.

The 6 fields of a Descriptor Word are defined as follows:

Pattern Start Address. This field contains the starting address of the pattern segment in Pattern Memory which is associated with this Descriptor Word. These 8 bits constitute the upper 8 bits of the 12-bit PM address, with the bottom 4 bits of the address initialized to zero. Thus, this field permits a pattern segment to start at any of 256 locations spread uniformly across the 4K PM space, or 1 location out of every 16.

Pattern Length. This field specifies the length of the pattern segment in PM which is associated with this Descriptor Word. It is a 6-bit field, where a value of 0x3F corresponds to a pattern length of 2, the minimum allowable value, and a value of 0x00 corresponds to a maximum pattern length of 65. Together with the Pattern Start Address field, the Pattern Length field of the Descriptor Word completely defines the contiguous range of PM which is transmitted to the TEM when this Descriptor Word is executed by the state machine.

Loop Count. This field specifies the loop count for the number of times that the above-defined pattern segment is transmitted in the course of executing this Descriptor Word. It is a 7-bit field, where a value of 0x7F corresponds to a loop count of 1,

the minimum, and a value of 0x00 corresponds to a maximum loop count of 128. Under circumstances of no interrupts, the state machine will repeatedly transmit the pattern segment the specified number of times before beginning execution of the next Descriptor Word.

Next DW Address. This field points to the address in DWM of the next Descriptor Word to be executed after the current Descriptor Word completes. It is a 9-bit field which can point to any of the 512 Descriptor Words in DWM. Under circumstances of no interrupts, the state machine will branch to execute the Descriptor Word pointed to by this field, after it has transmitted the associated pattern segment the number of times specified by the Loop Count field.

HALT bit. When this bit is set, it causes the state machine to immediately halt execution of the Descriptor Word. When this happens, the Pattern Start Address, Pattern Length, Next DW Address, and Loop Count fields in the current Descriptor Word are ignored. While the state machine is halted, the pattern output to the TEM is automatically forced to all zeroes.

Depending on the state of the IBLK bit in this same Descriptor Word, the state machine may or may not be interruptable while running or halted to start execution on another Descriptor Word. After halting, the status of the state machine will be observable at bits 0–1 of the board's status register, with a PGM State Code value of 1 (DW Halt) or 3 (Wait Int), again depending on whether or not the IBLK bit was also set. The machine states are described below.

IBLK bit. When this bit is set, interrupt branches are disabled for the duration of execution on the current Descriptor Word. This feature makes it possible to protect individual pattern sequences from interrupt branches that would abort their transmission prior to completion.

If the HALT bit is also set, then the value of the IBLK bit determines whether or not the state machine can be interrupted to start execution on a new Descriptor Word. If the IBLK bit is set, then the state machine remains in the "DW Halt" state, and cannot be interrupted. In this case, in order to resume processing of Descriptor Words, the state machine must first receive a reset. If the IBLK bit is not set, then the state machine enters the "Wait Int" state, in which it waits for an interrupt branch to resume Descriptor Word processing.

5.4.2 Defined Machine States

The Pattern Generation State Machine may occupy any of four different machine states, as viewed by the user through status register bits 0–1, as shown in Table 2. These states are all-encompassing as well as mutually-exclusive. the state machine will definitely occupy one of the four states at all times, but only one of the four at any given time. The definitions for each of these states are as follows:

0. Reset Halt:
   The state machine enters this state when any reset is applied to the PGM, and

remains in that state for the duration of the reset. It also resides in this state when bit 0 of the config register, the PGM Enable bit, is at a logic zero. When in this state, the state machine does not execute any Descriptor Words, and cannot be interrupted to start execution of a word. In addition, the Output Register for the pattern stream is turned off, forcing a null (all-zero) stream to be sent to the TEM. The state machine leaves the Reset Halt state and advances to the Wait Int state when the reset is released and the PGM Enable bit in the config register is set.

The red frontpanel HALT LED is lit while this state is occupied.

1. DW Halt:
   The state machine enters this state as a result of both the HALT and IBLK bits being set in a Descriptor Word. This state provides a means for the user to deliberately program a Halt into the DWM for circumstances such as unrecoverable errors. As with the Reset Halt state, interrupts are not serviced while in this state, and the Output Register is turned off. Once the state machine enters the DW Halt state, it must be reset before it can resume execution of Descriptor Words.

   The red frontpanel HALT LED is lit while this state is occupied.

2. Running:
   The state machine occupies this state when it is actively executing descriptor words. In this state, interrupt branches are enabled or not, depending on the level of the IBLK bit in the current descriptor word under execution. Once the state machine enters the Running state, it will remain in that state until it is reset, disabled through the config register, or encounters an active HALT bit in a Descriptor Word.

   This state is indicated by the green frontpanel RUN LED being lit.

3. Wait Int:
   This is a transition state between the two reset states and the Running state. The Wait Int state is like a Halt state, in that no Descriptor Word processing takes place and the Output Register is turned off. However, in the Wait Int state, interrupt branches are enabled, and once an interrupt is received, the state machine will begin execution on that Descriptor Word, and advance to the Running state.

   The state machine enters the Wait Int state from the Halt Reset state after the reset signal has cleared and after the PGM Enable bit is set in the config register. It also can enter the Wait Int state directly from the Running state when the HALT bit is set in a Descriptor Word while the IBLK bit is reset.

   This state is noted on the frontpanel by both the red HALT and green RUN LEDs being lit.

Figure 5 shows the state diagram for the Pattern Generation State Machine. The diagram shows the different transition paths between states and the conditions under which they occur.

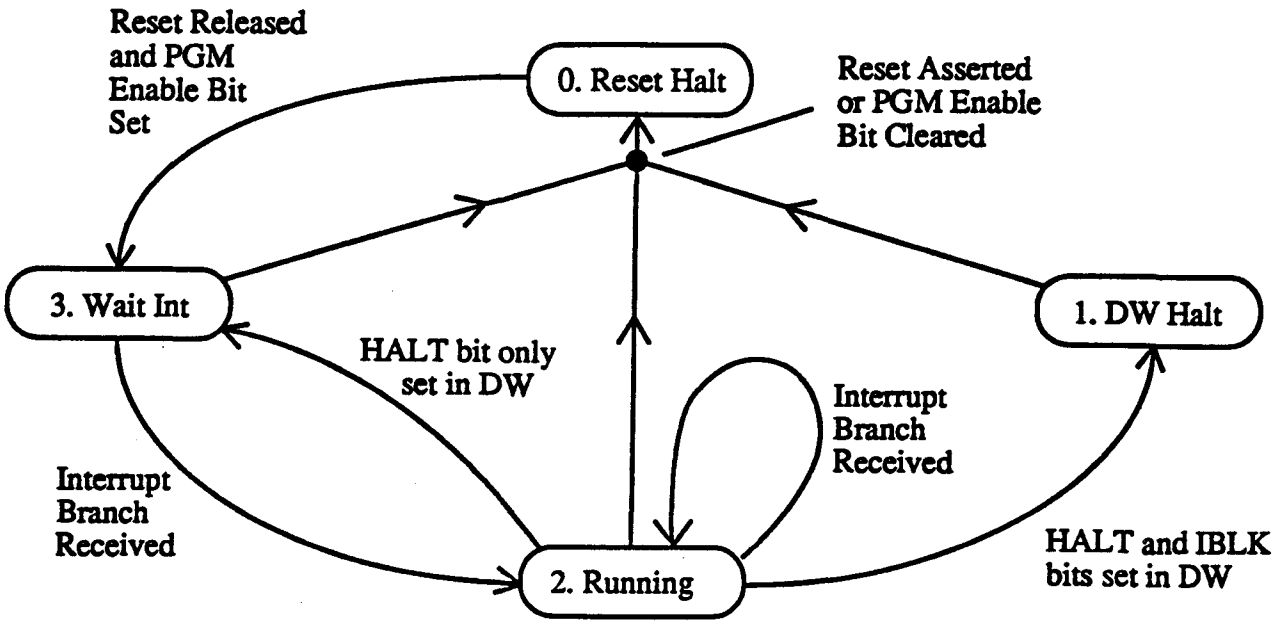


Figure 5: State Diagram for Pattern Generation State Machine

5.4.3 State Machine Architecture

Figure 6 shows a block diagram of the Pattern Generation State Machine architecture. The diagram shows the principal memory, register, counter and other building blocks which comprise the state machine. These blocks are described in the following paragraphs.

Descriptor Word Memory. The Descriptor Word Memory, or DWM, provides storage for the up to 512 Descriptor Words used by the state machine. It is a static RAM (SRAM) structure, which requires no dynamic refreshing of its contents. To the state machine, the DWM is a *read-only* structure, which receives its read address from the Next Address Register, and delivers its output data to various registers, counters, and other objects in the architecture, which are described below.

Next Address Register. The Next Address Register contains the 9-bit DWM Address of the next Descriptor Word to be executed. This register holds the address to the DWM while the Descriptor Word is being read out. The Next Address Register receives this address from the Address Select Mux, described next.

Address Select Mux. The Address Select Mux provides the switching path for the next Descriptor Word address from the various possible sources to the Next Address Register. Potential sources for the next address include the Next DW Address field of the current Descriptor Word, or any of the interrupt branch sources, which can have different addresses which are sent to the mux from the Interrupt Branch Input Logic. Source selection for the mux is governed by the State Machine Control Logic.

PATTERN GENERATION STATE MACHINE BLOCK DIAGRAM

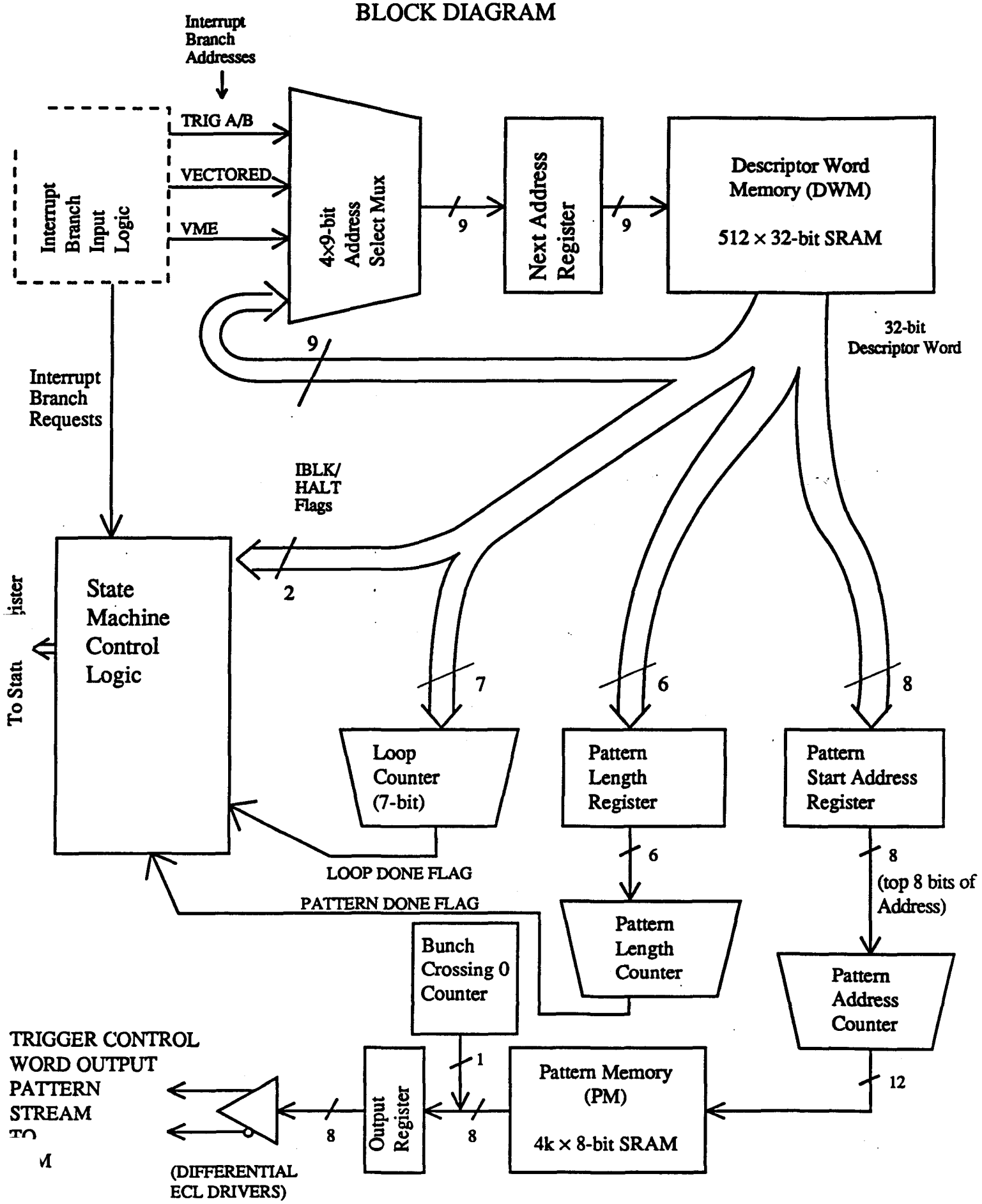


FIGURE 6

PGSM BLOCK DIAGRAM

Pattern Start Address Register. The Pattern Start Address Register is an 8-bit register which receives and holds the contents of the Pattern Start Address field when a new Descriptor Word is fetched from DWM. This register holds the field for subsequent loading into the Pattern Address Counter for the one or more iterations of the pattern, as specified by the Loop Count field of the Descriptor Word.

Pattern Length Register. The Pattern Length Register is a 6-bit holding register, which receives the contents of the Pattern Length field when a new Descriptor Word is fetched from DWM. This register holds the field for subsequent loading into the Pattern Length Counter for the one or more iterations of the pattern, as specified by the Loop Count field of the Descriptor Word.

Pattern Address Counter. The Pattern Address Counter is a 12-bit up counter which serves as the address generator to the Pattern Memory. At the start of each iteration of a pattern specified by the current Descriptor Word, it loads a starting address from the Pattern Start Address Register into its top 8 bits, and loads the bottom 4 bits of the counter to zero. The counter delivers the 12-bit address to the Pattern Memory, incrementing it once for each high-speed clock cycle, until the pattern is finished or interrupted. At which time, the counter again loads a start address.

Pattern Length Counter. The Pattern Length Counter is a 6-bit up counter which monitors the length of the current pattern. At the start of each iteration of a pattern specified by the current Descriptor Word, it loads a pattern length from the Pattern Length Register. On each high-speed clock thereafter, it increments by one count. When it reaches a terminal count of 0x3F, it signals the State Machine Control Logic that the current pattern iteration has completed via the Pattern Done Flag. The Pattern Length Counter is always reloaded at the same time as the Pattern Address Counter.

Pattern Memory. The Pattern Memory, or PM, provides storage for the up to 4096 8-bit Trigger Control word patterns used by the state machine. It is a static RAM (SRAM) structure, which requires no dynamic refreshing of its contents. To the state machine, the PM is a *read-only* structure, which receives its read address from the Pattern Address Counter, and delivers its output data to the Output Register for transmission to the TEM.

Output Register. The Output Register receives the Trigger Control word stream from the Pattern Memory and delivers to the differential ECL drivers for frontpanel transmission to the TEM card. The Output Register is enabled and disabled under the governance of the State Machine Control Logic, and is enabled only while the machine is in the Running state. In all other states it is disabled, transmitting a logic value of 0x00.

Loop Counter. The Loop Counter is a 7-bit up counter which is preloaded from the Loop Count field when a new Descriptor Word is fetched from DWM. This counter incre-

ments by one count each time the current pattern reaches the end of an iteration. When it reaches its terminal count of 0x7F, it signals the State Machine Control Logic via the Loop Done Flag that the current Descriptor Word has completed execution, and that the next word should be fetched from DWM.

Bunch Crossing 0 Counter. The Bunch Crossing 0 Counter is a 16-bit up counter which counts clock cycles. Every time it wraps around, it drives the Bunch Crossing 0 bit active for one cycle at the input to the Output Register.

To operate, the counter must be enabled at bit 12 of the config register. Additionally, the state machine must be in the Running or Wait Int state. The counter will not operate while the state machine is in the Reset Halt or DW Halt states. The wraparound modulus for the counter is held in the BC0 Counter Preload register, which is written by the DAQ Processor. The repeat rate at which the counter pulses the Bunch Crossing 0 bit may be programmed in the range from every 2 clock cycles to every 65536 clock cycles.

The BC0 Counter Preload register is not set to a predetermined value by a "power-up" reset, and must be initialized by the DAQ Processor prior to use of the counter.

State Machine Control Logic. This logic block contains the control logic which governs the operation of the objects described above. It contains a simple state machine which receives the various flags and interrupt requests and uses them to make the decisions inherent to the operation of the Pattern Generation State Machine, such as:

- whether or not to accept an interrupt branch and from which source,
- when to fetch a new Descriptor Word from DWM,
- when to start another pattern iteration in PM,
- and when to halt.

This section has an extensive connection to the Status Logic, through which it reports the machine status code, as well as the accept or reject outcome of interrupt branch attempts from the various sources.

5.4.4 Interrupt Branch Accept Rate

When not blocked, the Pattern Generation State Machine can accept a new interrupt branch on every clock cycle. Were this to occur over a number of consecutive clock cycles, then the first word only of each pattern sequence would be seen at the output, with each clock cycle bringing the first word of the pattern sequence associated with the next Descriptor Word. Simultaneous interrupt branch requests are prioritized by the Interrupt Branch Input Logic, as described below.

5.4.5 VME Memory Access

In order for the Pattern Generation State Machine to produce any useful output, its Descriptor Word and Pattern Memories must be initialized and verified by the DAQ Processor through the VME Interface. The following paragraphs describe the manner in which these memories are viewed from the VME bus, and the circumstances under which they may be accessed from the VME bus.

As seen from the VME interface, and further described in Appendix B, the 512 × 32-bit DWM is mapped into VME address space as 1K × 16-bits, while the 4K × 8-bit PM is mapped to every other byte in the VME address space, thereby spanning 8K worth of byte addresses.

In DWM, each 32-bit Descriptor Word occupies 2 adjacent 16-bit words as seen by the VME bus. Bytes 0 and 1 occupy the first word, and bytes 2 and 3 occupy the second. In PM, each 8-bit Trigger Control word is mapped to the odd VME byte (byte 1 of the data bus). Thus, bytes in PM are at odd VME addresses, and adjacent bytes in PM are one word address, or 2 byte addresses apart, as viewed from VME.

In order to avoid contention for the memories between the state machine and the VME interface, and to avoid the necessity of decidedly complex arbitration logic to resolve any such contention, VME access to the PM and DWM is restricted by the following simple rule:

> VME access to the state machine memories is permitted only when the state machine is in a non-interruptable halt condition. In particular, only when the state machine is either a Reset Halt or DW Halt state.

Should an access to either memory be attempted while the state machine is in either the Running or Wait Int states, in which the state machine either is or may start processing a Descriptor Word at any time, then the VME access will be rejected and a VME bus error will be sent to the DAQ Processor.

Note that access to the PGM on-board registers is not restricted in the above manner, and is permitted all times, regardless of the status of the state machine.

5.5 Interrupt Branch Input Logic

As discussed in the previous sections, "interrupt branch" is the term used to describe an external signal which causes the Pattern Generation State Machine to abort execution of the current Descriptor Word and immediately branch to begin execution on a new word. The PGM has four different potential sources for interrupt branches, each with its own associated Descriptor Word address or range of addresses. The Interrupt Branch Input Logic receives input signals from these sources and forms them into interrupt branch requests, which it passes onto the Pattern Generation State Machine, where the decision is made whether or not to accept or reject the request.

The operation of each of the potential interrupt branch sources is summarized in the following subsections.

5.5.1 VME Write Port

The VME Write Port source allows the DAQ Processor to send an interrupt branch request to the Pattern Generation State Machine through a normal VME bus cycle.

To request an interrupt branch, the DAQ Processor writes a 9-bit Descriptor Word address to this port. When the input logic detects the write operation, it sends the request on to the Pattern Generation State Machine with the address. Through this port, the DAQ Processor can request an interrupt branch to any Descriptor Word address from 0x000 to 0x1FF, even those at fixed locations associated with other interrupt sources, as described below.

The theoretical maximum rate at which the input logic can accept interrupt requests from the VME Write Port is equal to one-half of the PGM high-speed board clock rate. Practically speaking, however, this is much higher a rate than is possible through the VME Interface, unless the high-speed clock timebase sent to the TEM is slowed down significantly from a nominal speed of 60 MHz to something approaching 1 MHz.

Of the possible interrupt branch request sources, the VME Write Port is given highest priority by the input logic. That means that if a simultaneous requests occur on the VME Write Port and one of the other sources described below, the request from the write port will have precedence.

5.5.2 Trigger A and B Interrupt Inputs

These are asynchronous interrupt branch sources to the card, which are activated by a rising edge at their inputs . As described in the PGM Frontpanel section, they use Lemo 00-type inputs and TTL logic levels. Each input is individually enabled or disabled from bits 9 and 10 in the config register.

If enabled, a rising TTL edge to either of these inputs will cause the Interrupt Branch Input Logic to send an interrupt branch request and Descriptor Word address to the Pattern Generation State Machine. The logic always sends a Descriptor Word address of 0x1EE for the Trigger A input, and an address of 0x1EF for Trigger B. In terms of VME addresses, these addresses map to 0x03DC and 0x03DE respectively, relative to the base VME address of DWM. Note that the fixed entry point addresses for these inputs does not restrict the type of pattern response which may be programmed into the Descriptor Words at those addresses.

The maximum rate at which the PGM can accept recurring rising edges from these sources is equal to one-half of the PGM high-speed clock board rate. In the event of the occurance of "simulataneous" rising edges on both inputs (to the resolving ability of the high-speed clock on the PGM), Trigger A will automatically have priority over Trigger

B. Both inputs are subservient in priority to the VME Write Port, in the event of a contention with that port.

5.5.3 Vectored Interrupt Input

This interrupt branch input source allows the interrupter, a front end subsystem for example, to select from one of up to 16 different programmed PGM pattern responses to its interrupt request. This selection is made by using the value on a 4-bit vector code, which the subsystem delivers to the PGM via the PGM Input Connector. Through this input, the front end subsystem can direct the PGM to deliver a particular pattern, such as an initialization or reset sequence, with the vector code reflecting the "status" that initiates the pattern.

To load the 4-bit vector into the PGM, this interrupt branch input source uses a rising-edge load strobe. This strobe signals to the PGM that a valid Vectored Interrupt Branch request is at the input connector. As described in the PGM Frontpanel section, a 25-pin "D" connector is used for the PGM Input Connector, all of the Vectored Interrupt Input signals use differential ECL logic levels. This input is enabled or disabled from bit 11 in the config register.

If enabled, a rising edge on the load strobe causes the input logic to capture the state of the 4-bit vector, and also cause the Interrupt Branch Input Logic to send an interrupt branch request and Descriptor Word address to the Pattern Generation State Machine. The logic will send a Descriptor Word address of 0x1FY, where "Y" is the 4-bit vector. In terms of VME addresses, this range maps to a VME address offset of 0x03E0 for Descriptor Word address 0x1F0, and a VME address offset of 0x03FE for Descriptor Word address 0x1FF. These offsets are applied relative to the base VME address of DWM.

The maximum rate at which the PGM can accept recurring load strobes from this source is equal to one-half of the PGM high-speed clock board rate. This source has the lowest priority of any of the interrupt branch sources, below both the VME Write Port and Trigger Interrupts, in the event of contention with either of those sources.

5.5.4 Interrupt Block Overrides

Under most circumstances, the Pattern Generation State Machine may not be interrupted from any of the above sources when the IBLK bit is set in the Descriptor Word being currently executed. One facility exists, however, through which an interrupt request may override an active IBLK bit while the state machine is in the Running state.

Specifically, interrupt branch requests through the VME Write Port may override an IBLK bit, if the leftmost bit (bit 0) called the Override bit is set in the data word containing the Descriptor Word address. When this data word is written to the VME Write Port, the state machine will accept the interrupt branch request, even if the current Descriptor Word had an active IBLK bit.

This is the *only* method (other than a board reset) by which an IBLK bit may be overridden, and it applies only while the Pattern Generation State Machine is in the Running state. It does not apply in the DW Halt and Reset Halt states, because the state machine is halted. In the Wait Int state, interrupt branch requests are not blocked, so interrupt requests are handled normally, without the need for an override.

5.6 Reject Counter/Interval Timer Logic

As stated in the general description section, the Reject Counter/Interval Timer Logic section contains two 16-bit event counters and a interval timer. This logic measures the total number of Trigger A and Trigger B interrupt branches rejected by the Pattern Generation State Machine and the duration of time over which those rejections occurred, for the purpose of providing a correction to the dead time calculations made by the TEM under certain circumstances.

For example, the correction is required in the circumstance where one of the Trigger Interrupt inputs is used to generate a Level 1 Accept pattern sequence, and the possibility exists that the interrupt may be rejected by the state machine. This rejection could occur either as a result of the simultaneous occurance of a higher-priority interrupt request, or because the state machine was periodically executing Descriptor Words that were flagged as uninterruptable. When a rejection does occur, the result is that a Level 1 Accept is *not* sent to the TEM, and so cannot be counted on the module in the total number of Level 1 triggers received. The purpose of the reject counters is to tally the number of such triggers not delivered to the TEM.

5.6.1 Reject Counter Operation

The two event counters are designated as the Trigger A and Trigger B Interrupt Reject Counters. As their name implies, each counter tallies the number of times an interrupt request on their associated input is rejected by the Pattern Generation State Machine.

The counters are enabled through a combination of config register bit 1 and the front-panel Count Enable input. If bit 1 of the config register is zero, then the counters are automatically disabled. If bit one is set, then the counters are enabled or disabled from counting by the Count Enable input. If that input is high (its default unconnected state), then the counters are enabled, otherwise they are disabled.

The counters are readable by the DAQ Processor through the VME Interface. Appendix B details the read addresses. No writes or preloads of any kind are permitted to the counters. However, both counters will be reset to zero by a board reset or by a VME access to a Counter Clear strobe, as described in Appendix B.

Status register bits 7 and 8 are latched overflow bits, and possible VME bus interrupt request sources, for the Trigger A and Trigger B counters, respectively. If either of these counters increment past 65535 events, their associated status register overflow bit sets. The overflowing counter will wrap-around and can continue to count, but the its overflow

status bit remains set until a board reset of Counter Clear strobe occurs. This method of handling a event counter overflows is consistent with that for the Level 1 Accept counters on the TEM.

5.6.2 Interval Timer Operation

The Interval Timer consists of a prescaler to generate a "slow clock" timebase for a 16-bit up counter. The prescaler is a free-running counter which divides down the board high-speed clock by a factor of either 2^{13} or 2^{17}, selectable by an on-board jumper. The up counter is used to measure the accumulated elapsed time during which the reject counters are enabled. Specifically, the interval timer counts while the PGM Enable bit in the config register is set, while status bit 9 is active, and while no resets are applied to the card. This timer permits the DAQ Processor to determine the total duration of time over which the number of reject events occurred.

Like the Reject Counters, the Interval Timer is readable by the DAQ Processor through the VME Interface, as detailed in Appendix B. Also like the Reject Counters, it will be reset to zero by a board reset or by a VME access to a Counter Clear strobe. In addition, like the Busy Timer on the TEM, the Interval Timer is automatically reset to zero after a VME read by the DAQ Processor. Additionally, however, the Interval Timer may be preloaded with a non-zero value, which will shorten the length of the interval until an overflow occurs.

At a nominal timebase of 60 MHz, the interval timer overflows in approximately 8.94 seconds with a prescaler setting of 2^{13}, and in approximately 143 seconds with a prescaler setting of 2^{17}. When an overflow occurs, it is latched on status register bit 6. Once this happens, the Interval Timer stops counting, and automatically disables the Reject Counters. Additionally, if enabled, a VME bus interrupt request will be generated. The status register bit stays set and the timer/counter logic remains in this state until a board reset or Counter Clear strobe occurs. This method of handling a timer overflow is consistent with that for the Busy Timer on the TEM.

5.6.3 Counter/Timer Readout

The Reject Counters and Interval Timer are designed to operate over the same time window. They share a common count enable signal, which can be read at bit 9 of the status register. Prior to reading out the counters and timer, it is recommended that the common count enable be cleared. This is done by resetting bit 1 of the config register to zero.

Disabling the Reject Counters and Interval Timer in this manner prior to readout guarantees that the following two conditions, which are necessary to ensure the reliability of the read data, will apply:

1. the values in the two Reject Counters and Interval Timer all apply to the same

window of time, and

2. no counter or timer will change state during its readout via the VME interface.

6 Programming Examples

This section contains three programming examples which demonstrate the pattern generation capabilities of the PGM. The first pattern is a simple infinitely-repeating canned sequence, while the latter two make use of frontpanel interrupts. All of the examples assume the use of the TEM 60 MHz on-board timebase as the clock source, and therefore a clock period of 16.67 nsec.

6.1 Example 1: Repeating L1A/L2A Sequence

In this first example, the PGM is programmed to generate an infinitely repeating pattern, in which a 4-cycle duration Level 1 Accept is followed 50 μsec later by a Level 2 Accept, which is followed 50 μsec later by another Level 1 Accept, and so on. The pattern is programmed into the PGM memories while the PGM Enable bit (bit 0) in the config register is reset. When the memories have been loaded, the PGM Enable bit is set, and the sequence is kicked off by a DAQ Processor Interrupt via the Pattern Generator VME Interrupt Write Port. Once running, the pattern will continue to repeat until the DAQ Processor disables the PGM or until a reset is applied to the card.

Figure 7 shows the Descriptor Word programming map for the sequence. The actual values programmed into memory are shown in hexadecimal, while the decimal equivalents for the counts are shown in parenthesis. Figure 8 shows the Pattern Memory programming map. Each row contains 16 8-bit Trigger Control words, with the row address in the leftmost column being equivalent to a value in the Descriptor Word Patter Start Address Field. This example is programmed using a total of 7 Descriptor Words which reference 3 segments in Pattern Memory.

Before building the program, it is first necessary to reinterpret the timing parameters in the example in terms of PGM clock cycles. For instance, at a clock rate of 60 MHz, the 50 μsec interval between the start of the L1 Accept and the L2 Accept is equivalent to 3000 clock cycles, as is the 50 μsec interval between the L2 Accept and the next L1 Accept. Since the example calls for a 4-cycle wide L1 Accept, this means that there are 2996 cycles of *null* (zero) data between the last cycle of the L1 Accept and the L2 Accept. Finally, there are 2999 null cycles between the L2 Accept and the start of the next L1 Accept.

With these counts in hand, the example can now be explained, on a Descriptor Word basis:

DW #0: This is starting word for the pattern. Each iteration of the pattern loop begins execution here. To start, the DAQ Processor puts the state machine in a Wait

Descriptor Word Number	HALT Bit	IBLK Bit	Pattern Length	Pattern Start Address	Next DW Address	Loop Count
0	0	0	0x35 (12)	0x00	0x001	0x7F (1)
1	0	0	0x35 (12)	0x01	0x002	0x03 (125)
2	0	0	0x35 (12)	0x01	0x003	0x04 (124)
3	0	0	0x3F (2)	0x02	0x004	0x7F (1)
4	0	0	0x35 (12)	0x01	0x005	0x03 (125)
5	0	0	0x35 (12)	0x01	0x006	0x04 (124)
6	0	0	0x37 (10)	0x01	0x000	0x7F (1)

Figure 7: Example #1 Descriptor Word Programming Map

DW Pattern Start Address Field	Byte Number															
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0x00	01	01	01	01	00	00	00	00	00	00	00	00	XX			
0x01	00	00	00	00	00	00	00	00	00	00	00	00	XX			
0x02	02	00	XX													

Note: "XX" denotes "don't care" locations which are not used as part of the pattern.

Figure 8: Example #1 Pattern Memory Programming Map

Int state by applying and then clearing an equivalent "pushbutton" reset. From this state, the DAQ Processor kicks off pattern execution by writing the Descriptor Word address 0x0000 to the VME Interrupt Write Port. This starts execution of the pattern.

This Descriptor Word references PM locations 0x000–0x00B for one interation, to generate the 4-cycle L1 Accept, and the first 8 cycles out of 2996 of null pattern between the L1 Accept and the L2 Accept. It then branches to DW #1.

DW #1: This DW generates the next 1500 out of 2988 remaining null pattern cycles between the L1 Accept and the L2 Accept. It does this by referencing 12 cycles of null patterns, at PM addresses 0x010–0x01B, with a loop count of 125. Upon completion, it branches to DW #2.

DW #2: This DW generates the remaining 1488 cycles of null pattern between the L1 Accept and L2 Accept. It does this by referencing the same range of PM as DW #1, except with a loop count of 124. Upon completion, it branches to DW #3.

DW #3: This DW generates the L2 Accept and the first null cycle out of 2999 between the L2 Accept and the next L1 Accept. It references PM address range 0x020–0x021, with a length of 2, the minimum allowable value. It uses a loop count of 1, and branches to DW #4.

DW #4: This DW generates the next 1500 out of 2998 remaining null cycles between the L2 Accept and the next L1 Accept. This DW is programmed to use the exact same range of PM and loop count as DW #1, but branches instead to DW #5.

DW #5: This DW generates the next 1488 out of 1498 remaining null cycles between the L2 Accept and the next L1 Accept. This DW is programmed to use the exact same range of PM and loop count as DW #2, but branches instead to DW #6.

DW #6: This is the last DW in the loop. It generates the remaining 10 null cycles until the next L1 Accept. It references the same null range of PM as is used by words 1–2 and 4–5, except with a pattern length of 10 cycles, and loop count of 1. Upon completion, it branches back to DW #0, and the entire sequence begins again.

6.2 Example 2: L2 Accepts/Rejects via Trigger Interrupts

In this second example, the PGM is programmed to respond to interrupt requests on its frontpanel Trigger A and Trigger B inputs. While waiting for interrupts to occur, the state machine will sit in an infinite loop, generating a null output pattern. When an interrupt occurs on Trigger A, the PGM will generate a 1-cycle Level 1 Accept which is followed 250 nsec later by a Level 2 Accept. When an interrupt occurs on Trigger B, the PGM will generate a 1-cycle Level 1 Accept which is followed 250 nsec later by a Level 2 Reject. Both of these patterns will use the IBLK flag to prevent them from being interrupted in mid-execution by another interrupt. Additionally, upon completion both interrupt patterns branch back to the null loop.

Descriptor Word Number	HALT Bit	IBLK Bit	Pattern Length	Pattern Start Address	Next DW Address	Loop Count
0	0	0	0x3F (2)	0x00	0x000	0x7F (1)
1EE	0	1	0x31 (16)	0x01	0x000	0x7F (1)
1EF	0	1	0x31 (16)	0x02	0x000	0x7F (1)

Figure 9: Example #2 Descriptor Word Programming Map

DW Pattern Start Address Field	Byte Number															
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0x00	00	00	XX													
0x01	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	02
0x02	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	04

Figure 10: Example #2 Pattern Memory Programming Map

Figure 9 shows the Descriptor Word programming map for the sequence. Figure 10 shows the Pattern Memory programming map. This example is programmed using a total of 3 Descriptor Words which reference 3 segments in Pattern Memory.

In terms of conversions from absolute times to clock cycle counts, the only one required for this example is to convert the 250 nsec between a L1 Accept and a L2 Accept/Reject into 15 clock cycles at 60 MHz. This means that for the two interrupt patterns, 14 null cycles exist between the L1 Accept and the L2 Accept/Reject.

The 3 Descriptor Words in this example are now described:

DW #0: This is the starting point for the pattern. As in example 1, the DAQ Processor starts execution at this word by writing the Descriptor Word address 0x0000 to the VME Interrupt Write Port.

This word comprises the idle loop for the example. The the specified pattern is 2 cycles of nulls at PM address 0x000–0x001, which are repeated only once per word, but the word branches back to itself at address 0x000, thereby producing an infinite loop which outputs null Trigger Control words unless an interrupt occurs to force execution to a different Descriptor Word.

DW #1EE: This word is the predefined entry point for a Trigger A interrupt branch. When an interrupt request is received from the Trigger A input while the state machine is in the DW #0 idle loop, a branch is immediately taken to this Descriptor Word.

The pattern segment associated with this word resides at PM addresses 0x010–0x01F, and is repeated just once. The first cycle of the segment contains a L1 Accept, the next 14 cycles are nulls, and the last cycle contains a L2 Accept, coming a total of 250 nsec after the L1 Accept.

The IBLK bit is set, which prevents execution of this word from being interrupted by a subsequent interrupt request. One execution of the word has completed, the state machine branches back to DW #0, to wait for another interrupt.

DW #1EF: This word is the predefined entry point for a Trigger B interrupt branch. When an interrupt request is received from the Trigger B input while the state machine is in the DW #0 idle loop, a branch is immediately taken to this Descriptor Word.

The pattern segment associated with this word resides at PM addresses 0x020–0x02F, and is repeated just once. The first cycle of the segment contains a L1 Accept, the next 14 cycles are nulls, and the last cycle contains a L2 Reject, coming a total of 250 nsec after the L1 Accept.

The IBLK bit is set, which prevents execution of this word from being interrupted by a subsequent interrupt request. One execution of the word has completed, the state machine branches back to DW #0, to wait for another interrupt.

6.3 Example 3: L1 Accept/Reset Sequences via Vectored Interrupts

This third example demonstrates the use of Vectored Interrupts to generate Level 1 Accepts a Level 1 Reset sequence. As in the previous example, the state machine is booted into an infinite loop in which it generates at null input pattern. As vectored interrupts with specific codes are received, the state machine branches to and executes the appropriate Descriptor Words.

This example uses 4 out of 16 possible interrupt codes. Code 0 causes the state machine to generate a 1-cycle L1 Accept and then return to the idle loop. Code 1 also generates a L1 Accept, but follows it 4 cycles later with an active Clock Stop bit, and then returns to the idle loop. When the Clock Stop bit propagates to the output of the TEM, it will stop the TEM/PGM clock. Code 2 starts a L1 Reset pattern sequence. This sequence consists of a 2-cycle of L1 Reset which branches back to itself, thereby continuously driving the L1 Reset bit until another interrupt occurs. Code 3 drops the L1 Reset, follows it with a Bunch Crossing 0 pulse a few cycles later, and then branches back to the idle loop.

In this example, the source of the interrupts, presumably a front end electronic subsystem, needs to properly manage the order in which some interrupts are sent. In particular, to properly perform a Level 1 Reset sequence, a code 2 interrupt, which starts the sequence, must be followed by a code 3 interrupt, which stops the sequence and provides a Bunch Crossing 0 start sync. To follow the code 2 interrupt with any other code would violate the L1 Reset protocol used in this example.

Descriptor Word Number	HALT Bit	IBLK Bit	Pattern Length	Pattern Start Address	Next DW Address	Loop Count
0	0	0	0x3F (2)	0x00	0x000	0x7F (1)
1F0	0	0	0x3F (2)	0x01	0x000	0x7F (1)
1F1	0	0	0x3C (5)	0x01	0x000	0x7F (1)
1F2	0	0	0x3F (2)	0x02	0x1F2	0x7F (1)
1F3	0	0	0x3D (4)	0x03	0x000	0x7F (1)

Figure 11: Example #3 Descriptor Word Programming Map

DW Pattern Start Address Field	Byte Number															
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0x00	00	00	XX													
0x01	01	00	00	00	80	XX										
0x02	10	10	XX													
0x03	00	00	00	20	XX											

Figure 12: Example #3 Pattern Memory Programming Map

Figure 11 shows the Descriptor Word programming map for the sequence. Figure 12 shows the Pattern Memory programming map. This example is programmed using a total of 5 Descriptor Words which reference 4 segments in Pattern Memory.

The Descriptor Words used in this example are now described:

DW #0: Exactly the same as in Example 2, this is the starting point for the pattern. The DAQ Processor starts execution at this word by writing the Descriptor Word address 0x0000 to the VME Interrupt Write Port, and each of the interrupt sequences branches back to this word when completed.

This word comprises the idle loop for the example. The the specified pattern is 2 cycles of nulls at PM address 0x000–0x001, which are repeated only once per word, but the word branches back to itself at address 0x000, thereby producing an infinite loop which outputs null Trigger Control words unless an interrupt occurs to force execution to a different Descriptor Word.

DW #1F0: This word is the predefined entry point for a Vectored Interrupt with code 0. This word references PM range 0x010-0x011. The first cycle of this range contains a L1 Accept, and the second cycle a null. This segment is repeated just once, and the state machine branches back to DW #0 upon completion.

DW #1F1: This word is the predefined entry point for a Vectored Interrupt with code 1. In the example, this pattern is a variation on that of code 0, with a Clock Stop signal following the L1 Accept by 4 cycles. This word references PM range 0x010–0x014. The first cycle of this range contains a L1 Accept, the next three cycles nulls, and the last cycle a Clock Stop bit. Note that the Clock Stop signal will actually have its effect after it propagates through the FIFO on the TEM to reach the output of that module. This pattern segment is repeated just once, and the state machine branches back to DW #0 upon completion.

DW #1F2: This word is the predefined entry point for a Vectored Interrupt with code 2. This word references PM range 0x020–0x021. This range contains 2 cycles of L1 Reset. Note that this Descriptor Word branches back to itself, so once started, L1 Resets will continuously appear at the pattern output until another interrupt is received.

DW #1F3: This word is the predefined entry point for a Vectored Interrupt with code 3. As an interrupt with this code completes the L1 Reset sequence started by a code 2 interrupt, this code should logically follow a code 2 interrupt. This word references PM range 0x030–0x033. The first 3 cycles of this range are nulls, and the forth contains a Bunch Crossing 0 pulse. This segment thus clears the L1 Reset, and follows it up with a Bunch Crossing 0 as a start sync. Upon completion of this segment, the state machine branches back to the idle loop at DW #0.

References

[1] W. H. Smith, T. Gorski, S. Dasu, J. Lackey, *SDC Prototype Trigger Emulation Module*, U. Wisconsin SDC Note SDC-92-00348, 1992.

[2] W. H. Smith, T. Gorski, J. Lackey, *SDC Global Level 1 Processor: Clock & Control*, U. Wisconsin SDC Note SDC-91-00090, 1991.

[3] *The VMEbus Specification*, Revision C.1, 1985.

A Backplane Signal Definitions

Tables 4 and 5 show the backplane DIN 41612 connector pin assignments for TEM connectors P1 and P2. The pin assignments on P1 are per the standard VME definition. Only those signals with an actual electrical connection on the TEM are shown.

P2 is used by the TEM only for power and ground connections. The pin assignments on this connector follow the VXI standard for connector P2, with additional –5V, –2V, and ground assignments on rows A and C to support the ECL power requirements of the TEM, Fanout, or Pattern Generation Modules

Table 4: PGM P1 Pin Assignments

Pin Number	Row a Signal Mnemonic	Row b Signal Mnemonic	Row c Signal Mnemonic
1	D00		D08
2	D01		D09
3	D02		D10
4	D03	BG0IN*	D11
5	D04	BG0OUT*	D12
6	D05	BG1IN*	D13
7	D06	BG1OUT*	D14
8	D07	BG2IN*	D15
9	GND	BG2OUT*	GND
10	SYSCLK	BG3IN*	
11	GND	BG3OUT*	BERR*
12	DS1*		SYSRESET*
13	DS0*		LWORD*
14	WRITE*		AM5
15	GND		A23
16	DTACK*	AM0	A22
17	GND	AM1	A21
18	AS*	AM2	A20
19	GND	AM3	A19
20		GND	A18
21	IACKIN*		A17
22	IACKOUT*		A16
23	AM4	GND	A15
24	A07	IRQ7*	A14
25	A06	IRQ6*	A13
26	A05	IRQ5*	A12
27	A04	IRQ4*	A11
28	A03	IRQ3*	A10
29	A02	IRQ2*	A09
30	A01	IRQ1*	A08
31			
32	+5V	+5V	+5V

Note: "*" denotes active-low signal.

Table 5: PGM P2 Pin Assignments

Pin Number	Row a Signal Mnemonic	Row b Signal Mnemonic	Row c Signal Mnemonic
1	–2V	+5V	
2	–2V	GND	
3	–2V		GND
4	GND		–5V
5			–5V
6			–5V
7	–5V		GND
8			
9			
10	GND		GND
11			–2V
12		GND	–2V
13	–5V	+5V	–2V
14			–2V
15			–2V
16	GND		GND
17			
18			
19	–5V		–5V
20			–5V
21			–5V
22	GND	GND	GND
23			
24			
25			GND
26			
27			
28	GND		GND
29			
30			GND
31	GND	GND	
32		+5V	

B VME I/O Address Map

This appendix describes the VME addressable range of the PGM. It lists the addressable objects defined within that range, and describes their function. Table 6 shows the VME I/O Address Map. In it are 3 addressable regions, one for PGM registers, one for the Descriptor Word Memory, and one for the Pattern Memory. Together, these regions exist within the 16Kb region of the 24-bit VME address space allocated to the PGM. The on-board Card Address jumpers, which set a match value for the topmost 10 bits of the 24-bit address bus, determine the location of this range.

Table 7 shows the mapping on-board register space. Each item in this table includes a board-relative byte address, an access type, and a general item description. Defined access types for the card include 8-bit or 16-bit reads and writes, plus "address only" (AO) accesses, which are performed simply as 16-bit write operations, where the write data is not used, and is considered a "don't care." AO accesses are simply used to generate control strobes on the board.

B.1 Configuration Register

Shown in Table 3, the 16-bit configuration register contains the various static signals required to control the PGM. It resets to an all-zero state when the crate is powered-up or reset, or when the PGM receives a "power-up" reset strobe issued by the DAQ Processor. This register supports both write and read access, with write access at board-relative address byte 0x00, and read address at address 0x10.

B.2 VME Interrupt ID Register

This 8-bit register contains the user-defined interrupt ID for the PGM, which is used to identify the PGM as the interrupt source during a VME Bus interrupt acknowledge cycle. The ID number to be written to this register is system dependent, and must be determined by the user.

The register supports both write and read access, with write access at board-relative byte address 0x03, and read access at 0x1B. This register is not initialized to a known state by a "power-up" reset, and must be written by the user prior to use of VME Bus interrupts by the PGM.

B.3 Status Bit Clear Mask

The Status Bit Clear Mask is a 16-bit location at board-relative byte address 0x04 which is written to clear selected latched bits in the status register. Referring to the status register definition in Table 2, bits 2–5 and 12–13 are latched bits, which once set, can only be cleared by a "pushbutton" reset or by a write to the Status Clear Bit Mask.

Table 6: PGM VME I/O Address Map

Address Range	Description
XXY[1]:0000 – XXY:001F	On-Board Read/Write Register Access
XXY:0020 – XXY:07FF	Unused
XXY:0800 – XXY:0FFF	Descriptor Word Memory VME Read/Write Access[2]
XXY:1000 – XXY:1FFF	Unused
XXY:2000 – XXY:3FFF	Pattern Memory VME Read/Write Access[3]

Notes:

1. "XXY" denotes the upper 10 address bits in the 24-bit VME Address, which are determined by the Card Address Jumpers on the PGM.
2. Since the PGM uses a D16 data bus for VME I/O, each 32-bit Descriptor Word is mapped into adjacent words in memory. For example, bytes 0 and 1 of the first DW are mapped to address XXY:0800, and bytes 2 and 3 are mapped to address XXY:0802, while bytes 0 and 1 of the *second* DW are mapped to address XXY:0804, and so on. A Descriptor Word address of 0x020 to the Pattern Generation State Machine thus corresponds to a VME address of XXY:0880.
3. Pattern Memory allows byte access only, with all bytes mapped to the *odd* byte on the VME data bus. Thus, the first byte of Pattern Memory is mapped to address XXY:2001, and the second to XXY:2003, and so on.

Relative Address	Access	Description
0000	W16	Configuration Register (Write Access)
0003	W8	VME Interrupt ID Register
0004	W16	Status Bit Clear Mask
0006	AO	"Power-Up" Reset Strobe
0008	AO	Counter Clear Strobe
000A	W16	Pattern Generator VME Interrupt Write Port
000C	W16	Interval Timer Preload
000E	W16	BC0 Counter Preload Register
0010	R16	Configuration Register (Read Access)
0012	R16	Status Register
0014	R16	Trigger A Interrupt Reject Counter
0016	R16	Trigger B Interrupt Reject Counter
0018	R16	Interval Timer
001B	R8	VME Interrupt ID Register (Read Access)
001C		(Spare Read Address)
001E		(Spare Read Address)

Table 7: PGM Register Address Map

A given status bit is cleared when its corresponding bit in the Status Clear Bit Mask is set and written to the PGM by the DAQ Processor. Thus, only bits 2–5 and 12–13 in the mask are used; all other bits are ignored.

B.4 "Power-Up" Reset Strobe

This strobe is generated through an AO access (16-bit write data transfer on the VME bus, where the data is a "don't care") to byte address 0x06 on the card. This strobe produces a "power-up" reset signal on the PGM for the duration of time that it is active, and is equivalent in function to the VME SYSRST (VME system reset) signal, which goes active at crate power-up or reset.

B.5 Counter Clear Strobe

This strobe is generated through an AO access to byte address 0x08 on the card. The Counter Clear signal asynchronously resets both Trigger Interrupt Reject counters and the Interval Timer to zero. It also clears their associated overflow bits, bits 6–8, in the status register. It does not affect any other components on the card.

B.6 Pattern Generator VME Interrupt Write Port

This 16-bit port resides at board address 0x0A hex on the card. Through this port the DAQ Processor sends interrupt branch requests to the Pattern Generation State Machine.

When the DAQ Processor writes to the port, the rightmost 9 bits, bits 7–15, contain the Descriptor Word number to which to branch. The leftmost bit, bit 0, is the Override Flag bit. If this bit is set, it means that this request will override any interrupt blocking programmed into the currently executed Descriptor Word. The remaining bits 1–6 are ignored by the PGM.

After writing an interrupt branch request to the port, the DAQ Processor can read the status register to determine if the request was accepted or rejected. The PGSM will set bit 12 of the status register if the request was serviced, and bit 13 if the request was rejected. Note that once set, these bits remain so until explicitly cleared by a "pushbutton" reset or write to the Status Bit Clear Mask. Thus, the DAQ Processor should routinely clear the service/reject outcome of the previous interrupt branch request on bits 12 and 13 before attempting another.

B.7 Interval Timer Preload

The Interval Timer Preload is a 16-bit write location at board address 0x0C. At this location the DAQ Processor may preload the interval timer to a starting count value

other than the default of 0x0000. A non-zero preload causes a reduction in the interval of time until the timer overflows (and possibly sends a VME interrupt request).

The preload should be written to the Interval Timer while counting is disabled via bit 1 of the config register, and after the Interval Timer/Reject Counters have received a "pushbutton" or "power-up" reset or Counter Clear strobe.

B.8 BC0 Counter Preload Register

The BC0 Counter Preload Register is a 16-bit write-only register at board address 0x0E. It holds the repeat rate value for the Bunch Crossing 0 counter in the Pattern Generation State Machine.

A value of 0x0000 in the register corresponds to a Bunch Crossing 0 pulse being generated every 65536 clock cycles. A value of 0xFFFE corresponds to a pulse being generated every 2nd clock cycle. Value 0xFFFF is not valid to the counter, and should not be used.

This register is *not* initialized by a "power-up" reset to a known state. It must be written by the DAQ Processor prior to enabling the counter.

B.9 Status Register

Shown in Table 2, the status register is a 16-bit read-only register. It shows the current operational status of the PGM, and is accessed at board byte address 0x12. Its individual bit definitions are described in Section 5.2.

B.10 Trigger A Interrupt Reject Counter

The 16-bit Trigger A Interrupt Reject Counter resides for read access only at board address 0x14. The counter tallies the number of times which a frontpanel Trigger A Interrupt Branch request is rejected by the Pattern Generation State Machine. The counter is enabled/disabled by the same circuitry as the Interval Timer, specifically bit 1 of the configuration register and the frontpanel Count Enable input.

Prior to reading the counter, the DAQ Processor should disable counting via bit 1 of the config register. This will guarantee that a counter transition does not occur in the middle of the read operation.

B.11 Trigger B Interrupt Reject Counter

The 16-bit Trigger B Interrupt Reject Counter resides for read access only at board address 0x16. This counter tallies the number which a frontpanel Trigger B Interrupt

Branch request is rejected by the Pattern Generation State Machine. In this and in other respects it operates identically to the Trigger A Reject Counter.

B.12 Interval Timer

Board address 0x18 is the read address for the Interval Timer. As described in Section 5.6.2, the Interval Timer measures the duration of time over which the two Reject Counters were enabled.

As is the case with the Reject Counters, counting should be disabled prior to reading the timer.

C Cable Interconnections

This section describes various electrical and mechanical aspects of making cable interconnections to the PGM in the Test Beam or bench environment, including termination schemes, connector pinouts, and the link to the TEM.

C.1 Cable Termination

The PGM uses the same cable termination schemes as are specified for the TEM. TTL inputs to the card should be terminated at the source with a series resistor equivalent in value to the impedance of the connecting cable (50Ω if RG-58 or RG-174 coaxial cable is used). Differential ECL signals are terminated at the receiver with a 50Ω resistor to -2 Volts. This scheme provides a end-to-end impedance of 100Ω seen at the load. Therefore, it is recommended that 100Ω twisted-pair cable be used to carry the ECL signals[1].

C.2 PGM Input and Output Connector Pinouts

Figure 13 shows the pinout on the high-density, subminiature-D connector used for the PGM Output. This connector is a female receptacle, with shell dimensions indentical to those of the standard-density, 25-pin subminiature-D connector. This high-density, 44-pin version is made by at least two manufacturers, specifically AMP and Amphenol. On the PGM, we use an AMP version of the connector, part #1-748482-5. A similiar part from Amphenol has part #17BH-044-S-AM-2-10. This part is identical to the Clock and Control Input and output connecters used on the TEM[1], and is used for the PGM Output connector for compatability with the TEM.

Table 8 gives the pin assignments for the PGM Output connector. Note that all inputs are differential ECL pairs. Those pins listed as "reserved" and "spare" in the TEM Clock and Control Input pins defintions are also listed as such here.

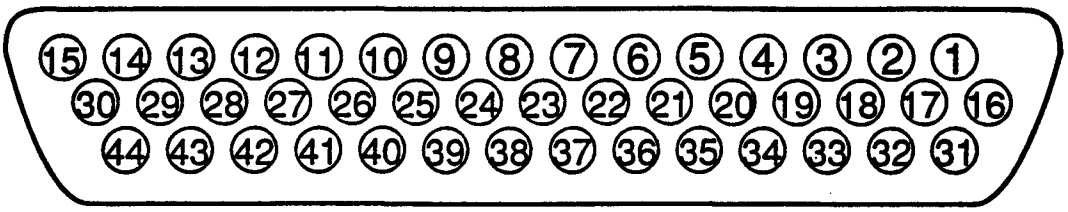


Figure 13: 44-pin High-Density Connector Pinout

Table 8: PGM Output Coneector Pin Assignments

Pin #	Signal Name	Pin #	Signal Name	Pin #	Signal Name
1	L1A OUT+	16	Spare	31	Spare
2	L1A OUT-	17	Spare	32	L2A OUT+
3	L2R OUT+	18	Spare	33	L2A OUT-
4	L2R OUT-	19	Spare	34	L1SYNC OUT+
5	L1RST OUT+	20	Spare	35	L1SYNC OUT-
6	L1RST OUT-	21	Spare	36	BC0 OUT+
7	EBTT OUT+	22	Spare	37	BC0 OUT-
8	EBTT OUT-	23	Spare	38	CSTOP OUT+
9	Spare	24	Spare	39	CSTOP OUT-
10	Spare	25	Spare	40	Spare
11	PGM CLK IN+	26	Spare	41	Spare
12	PGM CLK IN-	27	Spare	42	Reserved
13	EXT CLK OUT+	28	Reserved	43	Reserved
14	EXT CLK OUT-	29	Spare	44	Spare
15	Spare	30	Spare		

Table 9: PGM Input Connecter Pin Assignments

Pin #	Signal Name	Pin #	Signal Name
1	VLOAD IN+	14	VLOAD IN drain
2	VLOAD IN-	15	VBIT0 IN+
3	VBIT0 IN drain	16	VBIT0 IN-
4	VBIT1 IN+	17	VBIT1 IN drain
5	VBIT1 IN-	18	VBIT2 IN+
6	VBIT2 IN drain	19	VBIT2 IN-
7	VBIT3 IN+	20	VBIT3 IN drain
8	VBIT3 IN-	21	spare
9	spare	22	spare
10	spare	23	spare
11	spare	24	EXT CLK IN+
12	EXT CLK IN drain	25	EXT CLK IN-
13	spare		

Table 9 gives the pin assignments for the PGM Input connector. As with the PGM Output connector, all signals are differential pairs, but with an additional a drain wire pin assigned to each pair. A standard-density, 25-pin subminiature-D receptacle is used for this connector.

C.3 Interconnection with TEM

As indicated throughout this document, the PGM is designed to operate only when paired with a TEM card, with all essential interrconnections between the two modules taking place on the link between the PGM Output connector, shown in Table 8, and TEM Clock and Control Input connector.

To facilitate this interconnection, one of two options will be implemented. Either a short link cable or small printed circuit board will be provided with each PGM. This cable or board will support all signals carried between the TEM and PGM, using the same type 44-pin D connectors as are used on both of the modules.

D Specification Summary

This section provides a summary of some of the principal specifications for the Pattern Generation Module.

- **General Mechanical Specifications:**
  - Board Height: 9.187" (6 Eurocard Units).
  - Board Width: 6.30" (160 mm).
  - Board Thickness: 0.084" (2.13 mm), nominal.
  - Backplane Connectors: DIN 41612-C J1 and J2 connectors, and VME Bus standard 3U and 6U locations.
  - Frontpanel Width: 0.8" (Single-width in standard VME crate).
- **General Electrical Specifications:**
  - Backplane Bus:
    * Specification: VME Bus, IEEE 1014-1987.
    * Mode of Operation: Slave-only, with Bus Interrupter capability.
    * Address/Data Path Support: A24/D16 (Standard I/O Space).
    * Address Modifier Code Support: 39, 3D hex only (Standard I/O, user & supervisor data access).
    * Address Decoding: A14–A23 decoded as module address, A01–A13 decoded as module-owned space.
  - Power Requirements:
    * +5V: To be determined, est. < 5A. Supplied from VME Backplane.
    * -5V: To be determined, est. < 5A. Supplied by rearpanel custom Power Adapter Card via J2 connector.
    * -2V: To be determined, est. < 3A. Supplied by rearpanel custom Power Adapter Card via J2 connector.
    * +12V: Not used.
    * -12V: Not used.
- **Performance Specifications:**
  - Timebase:
    * Source: TEM via frontpanel connector.
    * Maximum Operating Frequency: 60 MHz minimum.
  - Reset Sources:
    VME, Frontpanel Pushbutton, frontpanel Lemo connector.
  - VME Bus Interrupter Capability:

* VME Interrupt ID Support: 8-bit on-board, User-Programmable register. R/W access supported.
* Bus IRQ Level Support: IRQ1–IRQ7 via on-board jumpers.
* Interrupt Sources: 0→1 transitions on selected status register bits.
* Source Mask Control: All sources independently maskable.

– Pattern Generation:

* Output Rate: 8 bits (one Trigger Control word) per TEM clock.
* Pattern Content: Pattern Memory user-programmable via VME.
* Pattern Length: 2 words minimum, 4096 words maximum (achieved by chaining together shorter pattern segments).
* Maximum Number of Non-overlapping Patterns: 256, each up to 16 words in length.
* Available Descriptor Words: 512.

– Bunch Crossing 0 Counter:

* Repetition Rate: User-programmable via VME, minimum rate 2 cycles, maximum rate 65536 cycles.
* Bunch Crossing 0 Pulse Width: 1 TEM clock cycle.

– Memory Specifications:

* Pattern Memory Size: 4096×8 bits.
* Descriptor Word Memory Size: 512×32 bits (viewed as 1024×16 bits from VME).
* VME I/O Access: VME R/W Access fully supported while state machine halted. VME memory I/O locked out while state machine running or waiting for interrupt.

– Pattern Interrupts:

* Sources: Frontpanel Trigger A & Trigger B connectors, Frontpanel Vector Input, VME.
* Maximum Repeat Rate: one-half TEM clock frequency per each input.
* Interrupt-to-Pattern Output Latency: 6 clock cycles.
* Minimum Separation Between Interrupts from Separate Sources: one TEM clock cycle.
* Source Enabling: Per interrupt source, via VME.
* Interrupt Block Capability: Programmable into Descriptor Words.
* Blocked Interrupt Override Capability: From VME–sourced interrupts only.

– Interval Timer:

* Timebase: Input clock from TEM, prescaled by factor of 2^{13} or 2^{17}, selectable by on-board jumper.
* Maximum full-scale range: $2^{16} - 1$ counts, equivalent to approx. 143 sec. @ TEM Clock rate of 60MHz.

    * Interval Enable Control: Via VME or frontpanel connector.
    * Preload Capability: Via VME.
    * Reset Capability: Via board reset or VME.
    * Read Capability: Via VME.
- Reject Event Counters:
    * Event Sources: Rejected frontpanel Trigger A & B interrupt requests.
    * Counter Size: 2^{16}, with wrapaounds flagged.
    * Interval Enable Control: Via VME or frontpanel connector.
    * Preload Capability: none.
    * Reset Capability: Via board reset or VME.
    * Read Capability: Via VME.