SWEA/STE Interface FPGA (SIF)
Description/Operation

Revision 0.2
9 April 2002

D. Gordon
ELF Electronics

References.

1. SWEA/STE Interface FPGA (SIF) Specification, Version B, 2002-Mar-15
2. SWEA UCB to CESR ICD, Version E, 2002-Mar-29
3. STEREO IMPACT Intra-Instrument Serial Interface, Version E, 2002-Mar-7

Revision History

Revision Number	Date	Change Summary
0.1	29 March 2002	Initial Draft
0.2	9 April 2002	Design Evolution & Corrections

Table of Contents

1.0 Introduction

The SIF (SWEA/STE Interface FPGA), logic contained in an Actel 54SX32S, handles the overall control and communication requirements for two types of instruments: the SWEA/STE-D and the STE-U. It incorporates components required by both instruments; however certain of the modules can be disabled via command, enabling the same design to be used for both systems.

An external reset signal (active low) is controlled by an RC with a time constant of approximately TBD seconds.

Figure 1 on page 4 shows an overview of the main subsystems comprising the SIF.

NOTE: Figure 1 is the actual FPGA schematic. At this stage, all the blocks are empty, and the schematic shows the general subsystem breakdown. I have left the internal connects unhooked until the subsystems have been designed. However, the external I/O are shown as in/out arrows.

The current I/O count is 162 pins (the limit is 173 pins for the 54SX32S in the 208 pin package).

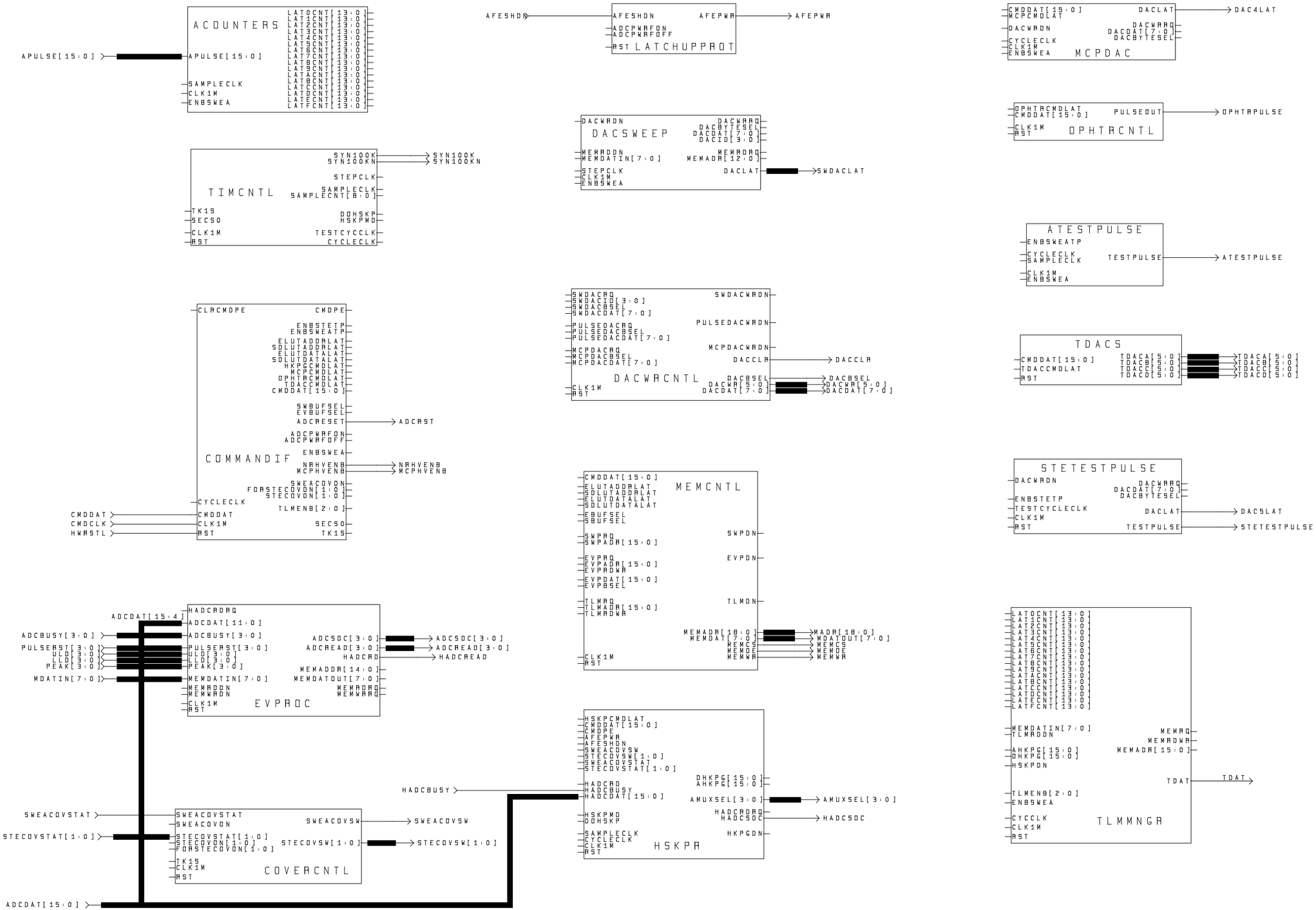

Figure 1: SIF FPGA – Overall Block Diagram

2.0 Subsystem Descriptions

2.1 Timing Control

The timing (TIMCNTL) module receives the 1MHz Spacecraft clock (CLK1M), the 1 Second Tick (TK1S) and the seconds count bit 0. It creates the following signals/strobes and fowards them to the various SIF subsystems:

CYCLECLK: Occurs every 2 seconds, as parsed from the "F0" command data (every "even" instance of TK1S). Only SECS[0] is used by the TIMCNTL module.

STEPCLK: STEPCLK divides the 2 second cycle clock into 1345 intervals as follows: at the beginning of a CYCLECLK period 1344 1.45 ms intervals are generated (totally 1948.8ms). The last interval is 51.2 ms.

SAMPLECLK and SAMPLECNT[8:0]: SAMPLECLK occurs every 4th STEPCLK during the 1344 1.45ms interval period, restarting at the next CYCLECLK. The SAMPLECNT, reset by CYCLECLK is a 9-bit counter, incremented every SAMPLECLK.

TESTCYCLECLK: Occurs every ten seconds. TBR: The TESTCYCLECLK hardware does not use the F0 data directly to determine timing, but counts CYCLECLKs, "ticking" every 5th CYCLECLK (different than as described in Reference 1).

HSKPMD: Toggles between 0 and 1 every CYCLECLK.

DOHSKP: Occurs 8 times per second, or 16 times per CYCLECLK.

SYN100K and SYN100KN: 100KHz Reference Supply (and an inverse version) which is forwarded to the SWEA-HVPS. (TBR: Is a specific enable needed for these signals, and/or should it be gated with any other signals (for example ENBSWEA))?

2.2 Command Interface

The command interface module performs serial to parallel conversion, receiving commands in the 24-bit format describes in Reference 3. If a parity error is detected, the command is ignored and an error indicator set (CMDPE). CMDPE is sent down as part of the digital housekeeping word (see Section 2.14, "Housekeeping Control," on page 12) and is reset following transmission.

The command interface module directly drives the simple control signals, either to internal SIF subsystems or off-chip to the various SWEA/STE-D or STE-U subsystems. For these outputs, any required staging is performed in the command interface module itself Other commands are forwarded to the SIF subsystems using the latched command data bus and subsystem select strobes.

A preliminary list of commands is shown below:

The Command ID start at E0.

The commands are:

COMMAND NAME	ID (Hex)	DESCRIPTION	Staged - When Effective
SRAM Buffer Select	E0	Sets Buffer Use D[1]: SWEEP Table Buffer Select - the value of this bit determines which buffer is the read buffer. D[0]: Energy Table Buffer Select - the value of this bit determines which buffer is the read buffer.	Next CYCLECLK
MCP DAC Load	E1	Loads MCP DAC (8 bit value) D[7:0]: DACValue[15:8] (DACValue[7:0] automatically loaded with zeros)	Next CYCLECLK
SWEA/STE Controls/ Enables	E2	Sets/Clears Various Controls D[15:13]: Telemetry Subsystem Enbs D15: Enable ACOUNTER Messages D14: Enable HSKPG Messages D13: Enable STE-PHA Messages D[12]: ADCs Active D[11]: AFEPWR Force On D[10]: AFEPWR Force Off D[9]: STE Test Pulser Enable D[8]: SWEA Test Pulser Enable D[7:4]: Shaper Chain Enable[3:0] D[3]: Spare D[2]: Enable SWEA (enables the following SWEA/STE Subsystems: DAC SWEEP and SWEA Counter Readout) D[1]: Enable NR HV D[0]: Enable MCP HV All bits default to 0 (disabled) at reset	Most are immediate. Test Pulser, Telemetry Subsystem and Shaper Chain Enables take effect at the next CYCLKCLK.
Actuator Enables	E3	Sets/Clears Cover Actuators D[4]: SWEA Cover Actuator ON D[3:2]: FORCE STE Cover Actuator ON D3: Force Cover In (Close) D2: Force Cover Out (Open) D[1:0]: STE Cover Actuator ON D1: Cover In (Close) D0: Cover Out (Open) NOTE: Only one bit of D[3:0] should be set; if more than one are set, the command is ignored. An “ARM” must be set in order for the FORCE commands to take effect All bits default to 0 (disabled) at reset	Immediate

COMMAND NAME	ID (Hex)	DESCRIPTION	Staged - When Effective
Operational Heater Control	E4	Sets PWM Duty Cycle to OpHeater D[3:0]: Pulse Width Value 0-> 0% Duty Cycle (Pulse held at logic zero) 1-> 10% Duty Cycle (on 1µs, off 9µs) ... 9-> 90% Duty Cycle (on 9µs, off 1µs) 10-> 100% Duty Cycle (Pulse held at logic one)	Immediate
Threshold DAC Load	E5	Sets One of Four Threshold DACs D[7:6]: DAC Select D[5:0]: DAC Data DACs are initialized to all zeros at reset.	Next CYCLECLK (TBR: depends on how full FPGA becomes)
ARM Force STE Cover Actuators	E6	ARMs Force Reclosable Cover Command D1: ARM Force Cover In (Close) D0: ARM Force Cover Out (Open) NOTE: Only D0 or D1 should be set; if more than one are set, the command is ignored.	Immediate (Enables future command (ID: E2): FORCE STE Cover Actuator - expires in 2 seconds)
SWEEP Housekeeping Channel Select	E7	Selects Active Housekeeping Channel for Sweep Housekeeping Mode D[3:0]: SWEEP Housekeeping Select	Next CYCLECLK
Sweep LUT Address Pointer Write	E8	D[12:1] - Word Address[12:1] Address pointer for IDPU writes to the Sweep Look-up Table. The address is incremented by one following each Sweep LUT Data Write	Immediate
Sweep LUT Data Word Write	E9	D[15:0] - Data to be written to current LUT location, determined by both the Sweep Table Buffer Select and the Sweep LUT Address Pointer	Immediate
Energy LUT Address Pointer Write	EA	D[14:1] - Word Address[14:1] Address pointer for IDPU writes to the Energy Look-up Table. The address is incremented by one following each Energy LUT Data Write	Immediate
Energy LUT Data Word Write	EB	D[15:0] - Data to be written to current LUT location, determined by both the Energy Table Buffer Select and the Energy LUT Address Pointer	Immediate

2.3 Latchup Protect

The Latchup Protect (LATCHUPPROT) module manages AFEPWR based on the states of an external overcurrent detect: AFESHDN and two commanded controls. The two commanded parameters, which both default to zero, are AFEPWR Force Off (AFEPWROFF) and AFEPWR Force On (AFEPWRON).

The following truth-table describes the AFEPWR control:

AFESHDN	AFEPWROFF	AFEPWRON	AFEPWR
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

AFEPWR is active high (TBR: Polarity of AFESHDN): a logic 1 indicates that the supplies are turned on, and the system is active. AFEPWR is jammed low by reset, but is allowed to assert following reset if AFESHDN is deasserted.

TBD: Should AFESHDN be based only on the external input, or should it be latched and kept set by the SIF once detected? In the latched case, only a reset or an explicit "CLRAFESHDN" command via the IDPU interface could allow the power to be restored.

AFEPWR, which controls the external power switches for the analog supplies, is also fed to SIF subsystems so that the analog circuitry inputs can be suppressed when the power is off. All DACs are cleared, and the five ADCs are reset. The following modules are held in reset when AFEPWR is deasserted: ACOUNTERS, DACSWEEP, DACWRCNTL, EVPROC, HSKPR, TDACS, STETESTPULSE, ATESTPULSE and MCPDAC.

Both AFEPWR and AFESHDN are available as status via the digital housekeeping register.

2.4 DAC Write Control

The DAC Write Control (DACWRCNTL) module receives write request (8-bit data format) to the six DACs that share the common DAC bus. It handles the requests on a fixed (TBD) priority basis, driving the appropriate address, byte-select, data and write strobe onto the external DAC control bus. The client DAC control subsystems are responsible for generating their own DAC latch pulses and sequencing 16-bit writes if necessary.

All the DACs share a command CLEAR signal, which is asserted during power-on reset, and when AFEPWR is off. (TBR: Does it matter what state the Sweep and MCP DACs are held when the ENBSWEA line is deasserted? If so, perhaps we should break the DACCLEAR into two signals -- one for all the DACs that should be held at zero by ENBSWEA=0, and the other for the STETESTPULSE DAC.)

2.5 SWEA/STE DAC Sweep Control

The Sweep Control subsystem (DACSWEEP) is responsible for loading four DACs at the STEPCLK frequency, and cycling through a preloaded pattern every CYCLECLK. The IDPU loads the SWEEP LUT portion of memory and initiates LUT buffer swapping via the command interface.

For every 8-bit value loaded into the DACs, a memory read must be performed. DACSWEEP tracks the current address and requests SRAM cycles via the memory arbiter. When the read has been granted, DACSWEEP requests the DAC bus via the DAC controller/arbiter, continuing on until all the values for the next step have been written. The next values remain in the DACs' holding registers until DACSWEEP issues the SWDACLAT strobe, coincident with the next STEPCLK.

TBR: Most likely a "flat" structure will be implemented for the LUT.

2.6 MCP DAC Control

The MCP DAC Control (MCPDAC) module receives a signal from the Command I/F following the reception of an MCP DAC Load Command. It then requests the DAC bus and writes the 8-bit value into the upper byte of the MCP DAC. The value is latched into the DAC at the occurrence of the next CYCLECLK.

The lower byte, set to zero by the DACCLEAR signal (asserted at RESET), is never changed.

2.7 Operational Heater Control

The Operational Heater Control (OPHTRCNTL) module receives a signal from the Command I/F following the reception of an Operational Header Control Load Command. It then latches the control data as input to the Pulse Width Modulation Circuitry. The control data (PWM_VALUE) occupies the lower nibble of the 16-bit command word, and is defined as follows: the output pulse is high for a period of PWM_VALUE x 1μs. The period of the PWM is 10 μs.

The PWM_VALUE, which is cleared at reset, varies from 0-10 in order to cover all possibilities, all off to all on. If PWM_VALUE is set to an illegal number, OPHTRCNTL substitutes a value of zero.

2.8 SWEA/STE (Anode) Test Pulser

The Anode Test Pulser (ATESTPULSE) module generates a variable frequency test pulse, used to stimulate the SWEA anodes during testing. It is held in a reset state (at which time the output is a logic low) when the SWEA is disabled or when the SWEA Test Pulser is disabled. The test pulse is activated by command: Controls/Enables (command ID E2).

When enabled, the ATESTPULSE frequency is stepped (up--?) each SAMPLECLK and returns to the start frequency at each CYCLECLK.

The Anode Test pulser generates a square wave based on CLK1M divided by SAMPLECNT[8:0]. (SAMPLECNT[8:0] is incremented each SAMPLECLK and set back to the starting number each CYCLECLK). The top frequency is 1MHz (TBR: perhaps 0.5MHz if that is simpler).

TBC (to be clarified): The current description has the Anode Test Pulser starting at the highest frequency and stopping at the lowest frequency. If it truly should step up in frequency each SAMPLECLK interval, SAMPLECNT[8:0] should be a downcounter which is loaded with MAXCNT[8:0]. CLK1M divided by MAXCNT would then produce the minimum frequency, at the beginning of each CYCLE.

2.9 Threshold DAC Registers

The Threshold DAC module contains four 6-bit registers which directly drive four on-board DACs. They are programmable via the threshold DAC Load Command, with bits[7:6] selecting the DAC to be loaded, and bits[5:0] containing the DAC data to be loaded.

The DAC data lines are staged; any change registers at the next CYCLECLK (TBR).

TDAC registers all default to zero at reset.

2.10 STE Test Pulser

The STE Test Pulser (STETESTPULSE) module both programs an external DAC and generates a test pulse. It is enabled via command: (Controls/Enables command ID E2), STE Test Pulser Enable.

When enabled, the DAC ramps up from zero to its maximum count (65536) incrementing by one every 100μs. The Ramp begins upon detection of a TESTCYCLECLK. The pulse, generated before each DAC increment, is active low and lasts 10μs. When the DAC count/value reaches its maximum, the DACs are cleared and the pulses are stopped. The DACs remain at zero until the next TESTCYCLECLK, at which time the DAC ramping and test pulses are restarted.

All DAC writes must be arbitrated via the DAC controller/arbiter, as for the Sweep and MCP DACs.

2.11 Cover Actuator Control

The control logic in this module (COVERCNTL) provides switch control for the cover actuators.

2.11.1 SWEA Cover

There is one 1-time opening cover on the SWEA. This switch shall be activated via the IDPU command I/F (Actuator Enables - Command E3).

TBD: Should the Activator status signal be used to deassert the SWEACOVACT switch control?
TBS: Should this be a protected command as the STE switch controls are?

2.11.2 STE Cover

STE has a reclosable cover. There are two switch controls to the STE and two status signals from the STE. The controls are activated via the command I/F (Actuator Enables): one commands the cover open and the other commands the cover shut. When one of these commands is received, power is supplied to the appropriate actuator, until the status signal feedback indicates that the cover has reached its desired position. If at any time, the OPEN or CLOSE commands are

received with the data bit set to zero, the actuator control will be deasserted immediately, regardless of the state of the status feedback. If both an OPEN and CLOSE command are asserted simultaneously, neither command shall register.

There is also a “force” command available for each STE actuator. This allows power to be supplied to the appropriate actuator regardless of the state of the feedback switch. This command is protected via an “arm-force” command (one arm per actuator). The appropriate arm-force must be issued prior to the force command. They two commands must occur within the same 2 second period.

TBD: Should we allow any other intervening commands in between the arm and the force commands (except for the F0 command)?

TBD: Should the actuator (when asserted via the FORCE command) automatically turn off after a few seconds if the FORCE command has not explicitly deasserted?

2.12 Anode Counters

The Anode Counter (ACOUNTERS) module contains 16 counters, each 14-bits, which register pulses generated by the SWEA anodes.

Anode Pulses are 250-300ns, positive-going pulses; rising edges are counted. As the fastest clock used by the SIF is 1MHz, each anode pulses must be fed directly to the clock of its counter. (So if the input is noisy, the counter will increment. Also, runt pulses may be registered as counts.

Each SAMPLECLK, the count values are transferred to holding registers, and the counters are cleared. The Telemetry Manager subsystem (see Section 2.16 on page 13) is responsible for reading out the holding registers (and the sweep housekeeping value) following each SAMPLECLK. There may be a few microseconds of downtime during the count latching/clearing every two seconds.

This module is held in a reset state when ENBSWEA is deasserted.

2.13 Event Processing

The event processing subsystem handles the outputs from the STE shaper circuits. The four analog chains each drive a separate set of control/pulse signals, but share a common 12-bit ADC data bus. (This ADC data bus is also shared with the 16-bit analog housekeeping ADC data bus. Arbitration, between the four analog chains as well as the HSKPR module, is performed by the Event Processing subsystem.)

A combination of control signals (as described in Reference 1) determine whether an event should be processed. If the criteria are met, the SIF issues an ADCSOC pulse to the active chain. ADCSOC is initiated via combinational logic only, in order to insure a small time delay. The SIF then waits for the busy line to transition high (25ns MAX) and deasserts ADCSOC. When the busy line returns to low, it is queued up for readout and processing.

After conversion, a request is made to the Event Arbiter/Processor subsystem. Each chain is allowed to convert when another event from a different chain is being processed. However, any further events in the same chain that arrive while an event is still pending are dropped.

The arbiter is non-preferential. If all four detector chains are continuously receiving events, each will receive one-fourth of the event processing subsystem bandwidth. However, if only one detector is active, this chain will receive the total available bandwidth. An enable is provided for each analog chain, so that a noisy channel can be disabled.

Following readout, the EVPROC module uses the ADC Data along with the Detector ID as an index to a lookup table, located in external RAM. The 8-bit value returned is used as an index to a 16-bit counter, also located in external RAM. EVPROC reads the counter value, increments it, and writes it back into memory.

2.14 Housekeeping Control

The Housekeeper (HSKPR) module receive a status signal (HSKPMD) and a timing signal from the TIMCNTL subsystem. HSKPMD, which toggles every CYCLECLOCK, selects either CYCLING or SWEEP Mode. The timing signal (DOHSKP) is a tick which occurs 8 times/second, or 16 times every CYCLECLK.

When in CYCLING mode: at CYCLECLK the AMUX channel is set to zero. At the first DOHSKP, HSKPR performs an ADC conversion and latches the result. (NOTE: All ADC Reads must be negotiated through the Event Processing subsystem ADC bus arbiter.) When the readout has completed, HSKPR informs TLMMNGR, which constructs a message containing the 16-bit ADC result followed by the 16-bit digital housekeeping register. After latching the current result, HSKPR increments the AMUX channel in preparation for the next conversion, which occurs at the next DOHSKP tick. This process continues until the 16-values have been scanned through in one CYCLECLK period.

When in SWEEP mode: The Housekeeper maintains a 4-bit register, loaded via command, which sets the analog channel for ADC conversions. At the next CYCLECLK when HSKPMD=SWEEP, the AMUX channel is set to the commanded value. Conversions are then performed every SAMPLECLK. TLMMNGR still reads the result and constructs messages (the last ADC conversion followed by the digital housekeeping register) eight times per second; however the AMUX channel is not incremented. TLMMNGR also appends the ADC result onto the SWEEP messages.

The digital housekeeping register provides status from the various digital and external subsystems. The register is defined as follows:

Bit Position(s)	Signal Name	Definition
15:12	HSKPCH	AMUX channel for last conversion
11	HSKPMD	Current Housekeeping Mode
10:8		TBS
7:6	STECOVSW[1:0]	STE Cover Switch Control Outputs
5:4	STECOVSTAT[1:0]	STE Cover Status
3	SWEACOVSTAT	SWEA Cover Status
2	AFESHDN	AFE Shutdown
1	AFEPWR	AFE Power
0	CMDPE	Command Parity Error in previous 1/8 second interval

2.15 Memory Cycle Control

An external 512Kx8 SRAM holds the various data buffers used by the SIF: the Energy LUT, the Accumulator RAM, and the Sweep LUT. All of these SRAM Areas are double buffered. The LUT buffer select is determined by IDPU command; the Accumulator RAM buffer select is swapped regularly as messages are read-out.

TBD: We could use bit 18 of the SRAM as a general RAM buffer select, which might be useful if a memory cell develops a problem during the life of the mission. The general select be set as the LUT Buffers are, via the IDPU Command I/F. This feature would be most valuable if memory dump command is also offered, so that the different memory sections could be tested/verified via the command and telemetry interfaces. Addition of these features will be considered if FPGA space appears to be available.

The Memory Controller (MEMCNTL) manages the cycles to the 8-bit SRAM, selecting between the following memory clients: DACSWEEP, EVPROC, TLMMNGR, and COMMANDIF.

The DACSWEEP module must read out of SRAM in order to update the Sweep DACs. At most, this would require eight byte reads every STEPCLK.

EVPROC requires one byte read, one word read and one word write for every event processed.

TLMMNGR requires 256 word reads followed by 256 word clears every CYCLECLK, for the accumulator messages.

COMMANDIF requires one word write each time a LUT DATA WRITE command is received. Both LUT address registers and write control subsystems are included in the MEMCNTL module.

MEMCNTL coordinates all the SRAM buffer controls, based on input from the Command I/F and the various timing signals. It selects the client for the current cycle (delineated by CLK1M), drives the selected address, data and control strobes onto the memory bus, and informs the client that the cycle has been completed. The memory bus arbitration scheme must assure that each client is able to obtain the necessary bandwidth. For example, when TLMMNGR is reading out the accumulator RAM, two byte reads must occur at least every 16μs in order to keep up with the message generation.

2.16 Telemetry Manager

The telemetry manager (TLMMNGR) module constructs, formats and serializes messages in accordance with the requirements outlined in Reference 3. The following message types are constructed by TLMMNGR:

Message Type	ID	Length	Frequency	Description
Anode Counters without HSKPG	C0	17	336 msg/sec	The 16 latched counter values accumulated by the SWEA Anodes every SAMPLECLK
Anode Counters with HSKPG	C1	18	336 msg/sec	The 16 latched counter values accumulated by the SWEA Anodes every SAMPLECLK with the SWEEP Housekeeping Value appended to the end of the message

Message Type	ID	Length	Frequency	Description
Energy Bin Counters TESTCYCLKCLK =0	C2	257	one msg/ CYCLKCLK	The 256 event counters used for STE PHA.
Energy Bin Counters TESTCYCLKCLK =1	C3			
Housekeeping	C4	3	8 msg/sec	The most recently converted HSKPG-ADC value, followed by one Digital Housekeeping value

The length is the number of telemetry message words, including the first (header) word. (The header word contains the message ID in the upper 6 bits and length in the lower 10 bits, as specified in Reference 3.) TLMMNGR handles the timing for Anode Counter readout, starting the message construction following every SAMPLECLK, and extracting the necessary data from the ACOUNTER (and HSKPR if in SWEEP Housekeeping mode) modules.

TLMMNGR constructs and transmits the housekeeping message upon receipt of the signal HKPGDN from HSKPR, generated following each ADC conversion, 8 times/second.

The Energy Bin Counters: While TLMMNGR is constructing Energy Bin Counter messages, it requests RAM via MEMCNTL for two byte reads per counter readout. The Accumulator Buffer is cleared as a block action after the message is constructed, but preceding the Accumulator Buffer Swap.

Each telemetry message type has its own enable, which is set via the command I/F.