HSA8000 (IXC2) Data Sheet
Digital Power Processor

Configurable Analog Front End™

EnSilica 32-bit RISC processor

Contents

Contents 2
List of Figures 4
General Description 6
HSA8000 applications 7
Ordering Information 8
Features 8
Absolute maximum electrical specifications 9
Architectural block diagram 10
HSA8000 pinout and pin functions 11
HSA8000 48 - QFN Pinout 11
Pin descriptions 12
Functional Description 14
Analog Interface 14
Analog inputs ANx_CMPx (x = 0 to 3) 15
Analog inputs CMPx (x = 4, 5 and 7) 17
Analog inputs AN4, AN5, AN6, and AN7_DACVR 18
Analog inputs AN10P/N and AN11P/N 19
Analog inputs AN12_ACP and AN13_ACN 22
Analog to digital converters ADC0, ADC1 and ADC2 23
Analog DACs 26
Temperature sensor 27
Electrical characteristics 27
Supply voltages and power consumption 28
Analog and digital regulators specifications 28
Power on Reset 28
Digital Interface 29
CPU 29
ROM and RAM 30
Memory map 30
Switching Engine 31
I/O Block 38
Capture (Timing Measurements) block 40
Compare (Timing Generator) 42
AC PLL 42
Serial Peripheral Interface 46
UART interface 47
I^2C serial interface 48
Watchdog 48
Math Accelerators Block 49

Timer 49
Digital DAC 49
DEBUG 50
Applications, Implementation and Layout 53
Application 53
Implementations 53
PFC Architecture based on HSA8000 54
LLC Architecture based on HSA8000 58
MI-P700A PV inverter HSA8000 based 59
1000 W AC battery inverter 60
HSA8000 schematic checklist 61
Power supply connections 61
Regulators connections 62
0.9 V mid-rail voltage reference connections 62
Reset circuit connections 63
Crystal oscillator connections 63
Unused pins 64
Layout Example 64
Documentation Support 66
HSA8000 Packaging 65

List of Figures

FIGURE 1- HSA800 SIMPLE BLOCK DIAGRAM 7
FIGURE 2 FUNCTIONAL BLOCK DIAGRAM 10
FIGURE 3 - HSA8000 48-LQFP PINOUT 11
FIGURE 4 - ANALOG INTERFACE SIMPLIFIED BLOCK DIAGRAM. 15
FIGURE 5 - BLOCKS ASSOCIATED WITH ANALOG INPUTS ANX_CMPX (X = 0 …3) 16
FIGURE 6- BLOCKS ASSOCIATED WITH ANALOG INPUTS CMPX (X = 4, 5 AND 7) 18
FIGURE 7 - BLOCKS ASSOCIATED WITH ANALOG INPUTS AN4, AN5, AN6, AND AN7_DACVR, INPUTS. 18
FIGURE 8 - BLOCKS ASSOCIATED WITH ANALOG INPUTS AN10P/N AND AN11P/N INPUTS 20
FIGURE 9 - BLOCKS ASSOCIATED WITH ANALOG INPUTS AN12_ACP AND AN13_ACN. 22
FIGURE 10 – ADC0 AND ADC1 OUTPUT STATUS CIRCUIT 24
FIGURE 11 -ADC2 OUTPUT STATUS CIRCUIT 24
FIGURE 12 - ADC0 BLOCK DIAGRAM. 25
FIGURE 13 - ADC1 BLOCK DIAGRAM. 25
FIGURE 14 - ADC2 BLOCK DIAGRAM. 26
FIGURE 15 - HIGH LEVEL BLOCK DIAGRAM OF HSA8000 SWITCHING ENGINE 31
FIGURE 16 - AN EVEN PROCESSOR BLOCK DIAGRAM 32
FIGURE 17 - A WINDOW CHANNEL OF THE HSA8000 WINDOW CONTROL BLOCK 33
FIGURE 18 – AN EVENT TIMER BLOCK DIAGRAM 34
FIGURE 19 - EVENT DRIVEN TIMER BLOCK DIAGRAM 35
FIGURE 20 - A FAULT CHANNEL [X] BLOCK DIAGRAM 36
FIGURE 21 - THE BLOCK DIAGRAM OF THE GATE CONTROL BLOCK 37
FIGURE 22 - THE BLOCK DIAGRAM OF THE PAIR PROCESSOR BLOCK 38
FIGURE 23 - THE BLOCK DIAGRAM OF GPIOX BLOCK (X = 0…7) 39
FIGURE 24 - THE BLOCK DIAGRAM OF GPIOX BLOCK (X = 8 … 31) 40
FIGURE 25 – CAPTURE (TIMING MEASUREMENTS) BLOCK DIAGRAM 41
FIGURE 26 - A CHANNEL OF THE CAPTURE (TIMING MEASUREMENTS) BLOCK 41
FIGURE 27 - BLOCKS ASSOCIATED WITH ANALOG INPUTS AN12_ACP AND AN13_ACN. 42
FIGURE 28 - AC PLL BLOCK DIAGRAM 43
FIGURE 29 - GRID FREQUENCY MEASUREMENT BLOCK DIAGRAM 44
FIGURE 30 - GRID REGENERATION BLOCK DIAGRAM 45
FIGURE 31 – PERIOD DIVIDER BLOCK DIAGRAM 46
FIGURE 32 - SPI BLOCK DIAGRAM 47
FIGURE 33 - UART INTERFACE BLOCK DIAGRAM 47
FIGURE 34- HSA8000 WATCHDOG SYSTEM BLOCK DIAGRAM. 49
FIGURE 35 HSA8000 DEBUG BLOCK 50
FIGURE 36 - SIMPLIFIED HSA8000-BASED TOTEM-POLE PFC CIRCUIT. (POWER TRAIN IN BLACK, DRIVER CIRCUITS IN GREEN, CURRENT SENSING CIRCUITS IN BLUE, AND VOLTAGE MEASUREMENT CIRCUITS IN RED.) 54
FIGURE 37 - SIMPLIFIED HSA8000-BASED TOTEM-POLE PFC CONTROL CIRCUIT 55
FIGURE 38 HSA8000 SWITCHING ENGINE CONFIGURATION FOR HUADA TOTEM-POLE PFC. 56
FIGURE 39 -TIMING DIAGRAM OF THE HUADA TOTEM-POLE PFC BOOST 1. 57
FIGURE 40 - SIMPLIFIED HSA8000-BASED LLC CONVERTER CIRCUIT. 58
FIGURE 41 - SIMPLIFIED HSA8000-BASED LLC CONTROLLER CONTROL CIRCUIT. 59
FIGURE 42- MI-P700 4Q HIGH LEVEL VIEW OF STAGES 60
FIGURE 43 - ACBI HIGH LEVEL VIEW BLOCK DIAGRAM 61
FIGURE 44 - POWER SUPPLY CONNECTIONS 62
FIGURE 45 - REGULATORS CONNECTIONS 62
FIGURE 46 - 0.9 V MID-RAIL VOLTAGE REFERENCE CONNECTIONS. 63

FIGURE 47 - RESET CIRCUIT CONNECTIONS 63
FIGURE 48 - CRYSTAL OSCILLATOR CONNECTIONS 64
FIGURE 49 - LAYOUT EXAMPLE. 64
FIGURE 69 – HSA4021 IXC2 QFN48L PACKAGE 65

General Description

The HSA8000 is a digital power processor. One IC offers high-speed analog peripherals, digital accelerators, event control, and digital processing. Its flexibility and performance enable designers to meet demanding compliance standards. Industrial, automotive, and renewable energy applications can benefit significantly from the enhanced performance and reduced component count it offers. The solution addresses many power conversion applications. It easily fits into advanced topologies for AC-DC inverters, battery chargers, and isolated DC-DC converters.
The HSA8000 has an advanced mixed-signal architecture. The core is a 32-bit RISC 50 MHz micro-processor. A rich set of high-performance digital power peripherals supports the core. Communications, data memory, and general-purpose inputs and outputs (GPIO) are also provided.
The HSA8000 is a fully software-programmable platform. Programming enables control, monitoring and optimization. Those features allow a design solution to meet aggressive requirements. It also allows for easily differentiated products for competitive markets. Solantro bundles a software development environment with application-specific evaluation hardware to enable customers to achieve faster time-to-market.

The HSA8000 arose from extensive experience in power systems design. Those systems frequently require optimization of multiple voltages through control loops with adaptive dead-time control of multiple power trains and phases. For example, Totem-Pole PFC, LLC, and Interleaved PFC need such complex control schemes. Here, the core architecture optimizes processor usage by using high speed analog peripherals, digital accelerators and a high performance PLL. The peripherals control high speed loop functions leaving the digital core processor free to maintain low-speed functions. Those functions include slower control loops, protection, optimization and housekeeping. The HSA8000's peripheral set is the industry's most complete single-chip offering. It includes configurable high-speed voltage/current sensing, and high-speed comparators (10nS), a variety of high-performance ADCs, DACs, programmable filtering, a multi-event interrupt-based timing engine, a high performance digital PLL, and PWM control for up to 8 power devices are integrated into a single digital power processor.

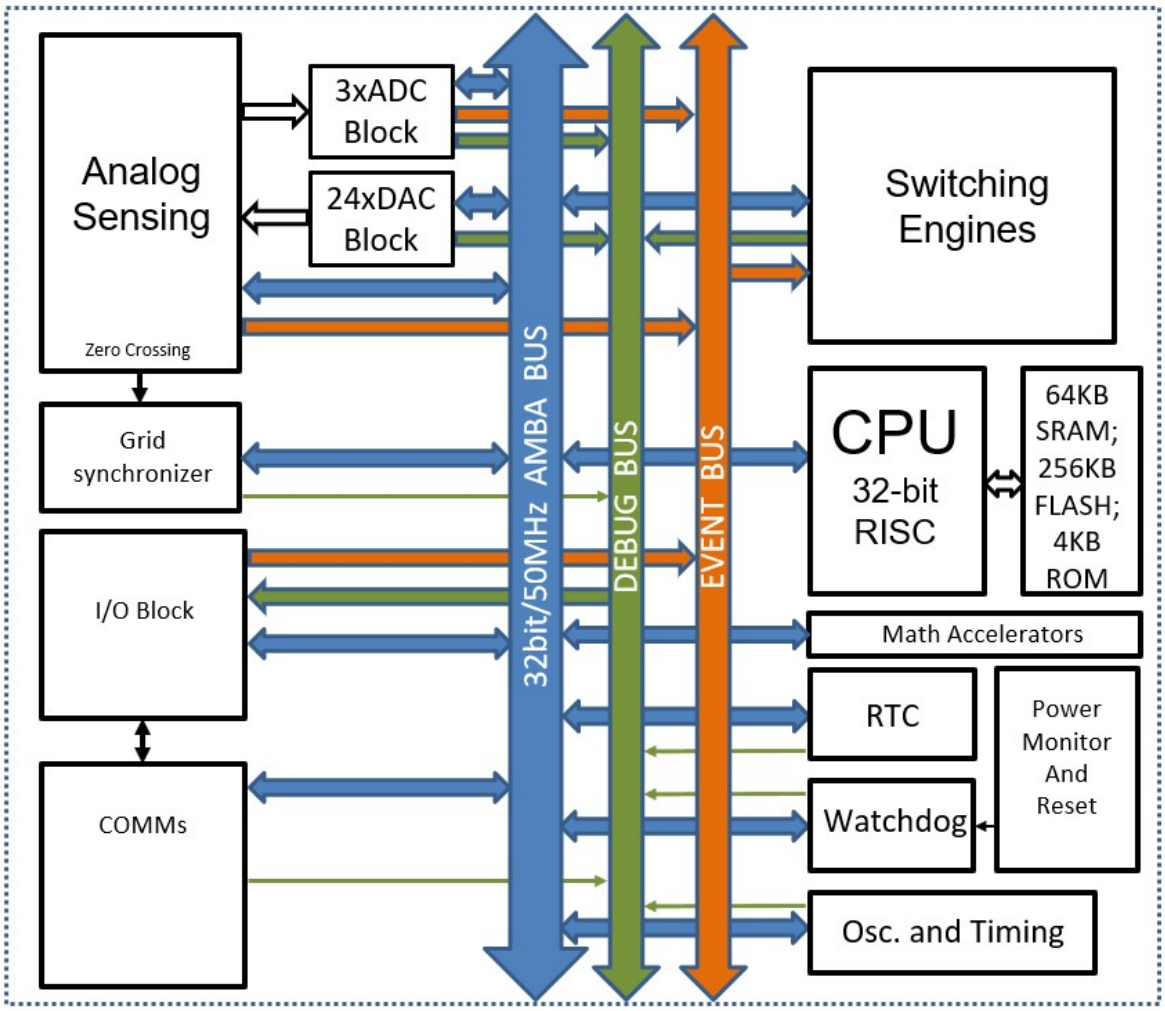

Figure 1- HSA800 simple block diagram

HSA8000 applications

- AC/DC - Power-Factor-Corrected- Bridgeless & Interleaved
- DC/DC - LLC, Half-Bridge, Phase-Shifted Full-Bridge, etc.
- Charging – On-board (EV), Charge Stations, Off-grid
- Inverters – Bi-Directional, Automotive, UPS and Storage
- Heavy Industrial – Electrified Equipment, HVAC, Welding
- Single-panel and dual-panel micro-inverters
- Battery and fuel cell inverters (uni-directional and bi-directional)
- VAR compensator
- Interleaved multi-phase battery charger for high-power applications

Ordering Information

Part Number	Temperature range	Package	Pins	Packing
HSA8000	-40 to +105 °C	48-QFN	48+Exposed Pad	Tray

Features

Rich power control-centric analog Peripherals:

- Sixteen-channel 10-bit 1.4 MS/s ADC
- 2 four-channel 10-bit, 1.4 MS/s ADC
- 15 10 ns fast comparators
- 13 10-bit analog DACs for internal comparator threshold settings
- Two differential high-speed current sensing amplifier interfaces
- Programmable anti-alias low-pass filters
- Temperature sensing

Digital Power Engine and Peripherals:

- 32-bit RISC CPU with 64 KiB RAM
- 256 KiB internal SPI flash memory
- Switching engine for up to 8 drivers with gate control for cross-conduction protection
- Four event-driven timing engines with sixteen event processing channels for hysteresis control.
  - Four PWM timers (10 ns resolution, 625 ps fractional)
  - Up to four interleaved timers with real-time programmable phase shift (any phase shift) with 10 ns resolution
- Dedicated high-performance digital AC PLL with a dedicated sensing comparator for grid synchronization
- Simultaneous adjustable real-time update of frequency and duty cycle.

Junction temperature - 40 to 125° C.

Absolute maximum electrical specifications

Table 1 lists the absolute maximum electrical specifications for the HSA8000.

Warning! Operating beyond the limits specified in the following table may cause permanent damage to the device. Operating at the limits specified for extended periods may affect device reliability and lifetime.

Table 1 - HSA8000 Absolute Maximum Electrical Specifications

Rating	Symbol/Pin	Value	Units
Supply voltage	VDDA and VDD	-0.3 to +3.6	V
Low-voltage analog pin voltage	Vin_LV_analog	-0.3 to VDDA	V
Digital pin voltage	V_digital	-0.3 to VDD	V
Storage temperature range	T_storage	- 65 to +150	°C
ESD immunity (Human body model)	V_ESD	4000	V
Operating temperature	Ta	-40 to +105	°C
Junction temperature	Tj	-40 to +125	°C
Note: Unless otherwise specified, all voltages are with respect to the voltage at the GND. VD) return pins. Operating ratings are conditions under which operation of the device is specified. For specific operating limits and associated test conditions, see the electrical characteristics.			

Architectural block diagram

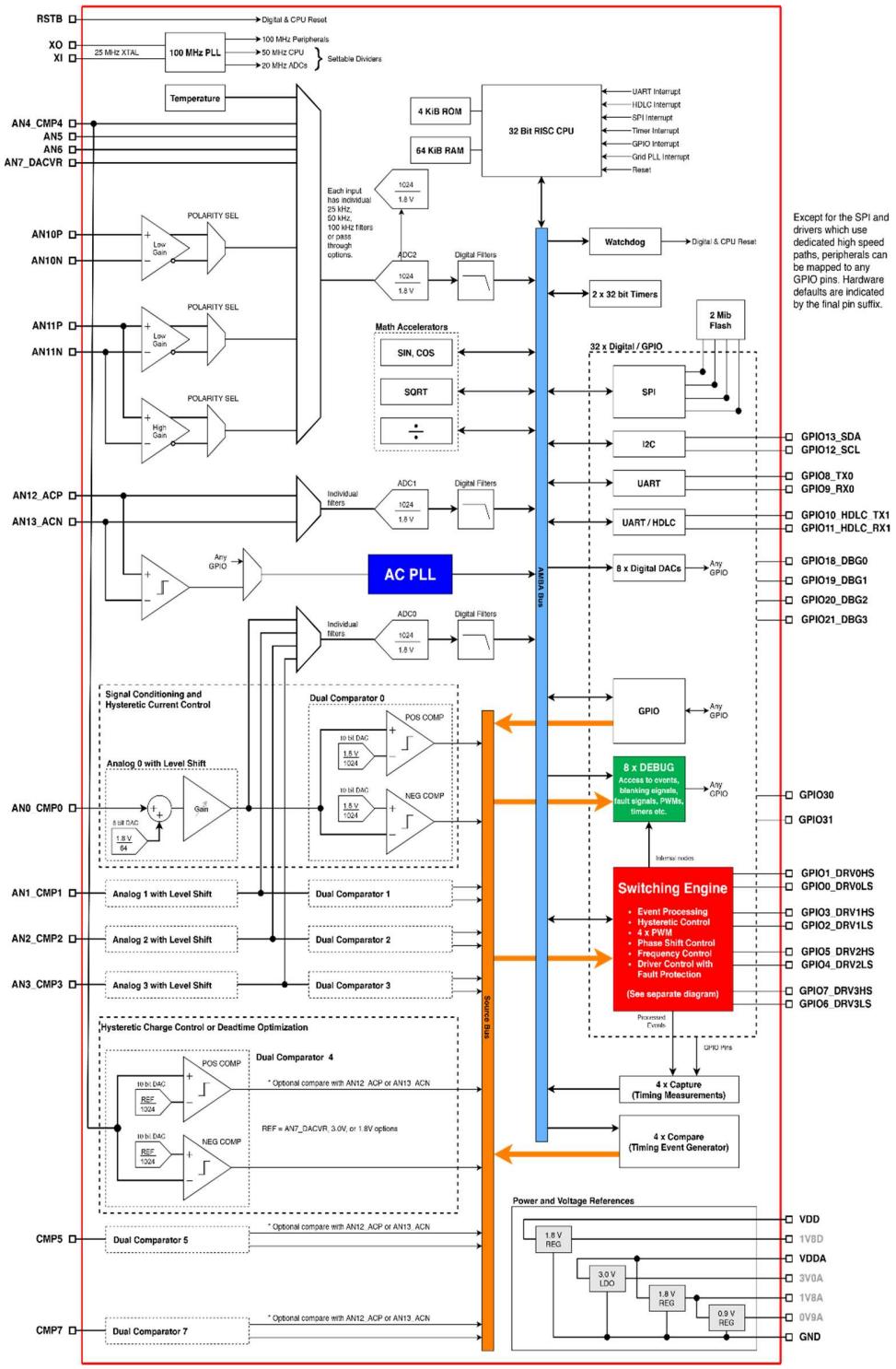

Figure 2 Functional block diagram

HSA8000 pinout and pin functions

HSA8000 48 - QFN Pinout

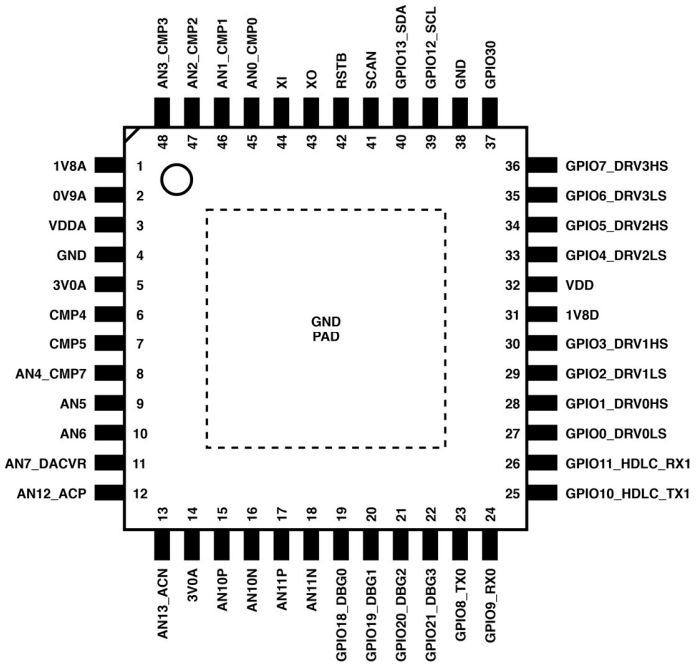

Figure 3 - HSA8000 48-LQFP Pinout

Pin descriptions

Table 2 - HSA8000 Pin functionality

Name	HSA8000 (48 pin)	Description
GND		Analog ground.
1V8A	1	1.8 V internally regulated analog supply. External decoupling required.
0V9A	2	0.9 V mid-rail voltage reference. External decoupling required.
VDDA	3	Analog power supply. External decoupling required.
GND	4	Analog ground.
3V0A	5	3.0 V internal analog supply. Connect externally to other 3V0A pins and decouple.
CMP4	6	Analog dual comparators.
CMP5	7	Analog dual comparators.
CMP7	8	Analog dual comparators.
AN4	8	Analog input on ADC2.
AN5	9	Analog input on ADC2.
AN6	10	Analog input on ADC2.
AN7_DACVR	11	Analog input on ADC2. Selected DAC voltage references.
AN12_ACP	12	Analog input on ADC1. AC PLL input.
AN13_ACN	13	Analog input on ADC1. AC PLL input.
3V0A	14	3.0 V Power for ADCs. Connect externally to other 3V0A pins and decouple.
AN10P	15	Differential analog positive input on ADC2.
AN10N	16	Differential analog negative input on ADC2.
AN11P	17	Differential analog positive input on ADC2.
AN11N	18	Differential analog negative input on ADC2.
GPIO18_DBG0	19	GPIO and alternate functions. Debug Channel 0 default.
GPIO19_DBG1	20	GPIO and alternate functions. Debug Channel 1 default.
GPIO20_DBG2_TCK	21	GPIO alternate functions and JTAG. JTAG TCK default. Debug Channel 2 default with JTAG disabled.
GPIO21_DBG3_TMS	22	GPIO alternate functions and JTAG. JTAG TMS default. Debug Channel 3 default with JTAG disabled.
GPIO8_TX0	23	GPIO and alternate functions. TX0 Default.
GPIO9_RX0	24	GPIO and alternate functions. RX0 Default.
GPIO10_HDLC_TX1	25	GPIO and alternate functions. TX1 Default. HDLC off by default.
GPIO11_HDLC_RX1	26	GPIO and alternate functions. RX1 Default. HDLC off by default.
GPIO0_DRV0LS	27	GPIO, alternate functions and driver control. Driver low output default.

Name	HSA8000 (48 pin)	Description
GPIO1_DRV0HS	28	GPIO, alternate functions and driver control. Driver low output default.
GPIO2_DRV1LS	29	GPIO, alternate functions and driver control. Driver low output default.
GPIO3_DRV1HS	30	GPIO, alternate functions and driver control. Driver low output default.
1V8D	31	1.8 V internally regulated digital supply. External decoupling required.
VDD	32	Power supply for the digital interface. External decoupling required.
GPIO4_DRV2LS	33	GPIO, alternate functions and driver control. Driver low output default.
GPIO5_DRV2HS	34	GPIO, alternate functions and driver control. Driver low output default.
GPIO6_DRV3LS	35	GPIO, alternate functions and driver control. Driver low output default.
GPIO7_DRV3HS	36	GPIO, alternate functions and driver control. Driver low output default.
GPIO30	37	GPIO and alternate functions. GPIO input default.
GPIO31		GPIO and alternate functions. GPIO input default.
GND	38	Digital ground
GPIO12_SCL	39	GPIO and alternate functions. I2C SCL Default.
GPIO13_SDA	40	GPIO and alternate functions. I2C SDA Default.
SCAN	41	Scan test input
RSTB	42	Bidirectional active low reset. External pull-up resistor to VPWRD.
XO	43	Output for external 25 MHz crystal with external loading capacitor.
XI	44	Input for external 25 MHz crystal with external loading capacitor.
AN0_CMP0	45	Analog input on ADC0 and dual comparators.
AN1_CMP1	46	Analog input on ADC0 and dual comparators.
AN2_CMP2	47	Analog input on ADC0 and dual comparators.
AN3_CMP3	48	Analog input on ADC0 and dual comparators.

Functional Description

The HSA8000 is a mixed-signal integrated circuit optimized for power conversion using digital control methods. The analog inputs can create timing events using high speed comparators while monitoring voltages and currents using three independent ADCs. The digital peripherals of which the Switching Engine is the heart can drive the gate signals of complex power applications. The functional diagram, in Figure 2, gives an overview of the HSA8000. The 32-bit RISC core oversees the configuration and monitoring of the peripherals while implementing control algorithms and state machines not possible in purely analog circuits.

Analog Interface

The HSA8000 is a highly integrated IC with rich power control-centric analog features:

- 18 analog input pins
- 1 sixteen-channel, 10-bit, 1.4 MS/s ADC with digital filters
- 2 four-channel, 10-bit, 1.4 MS/s ADCs with digital filters
- 17 10 ns fast comparators for event generation and fault detection
- 16 10-bit analog DACs for comparators internal references
- 1 10-bit analog DAC for debagging
- 4 6-bit analog DACs for level shifting
- 2 differential high-speed current sensing amplifier interfaces
- 1 differential high gain amplifier
- 14 programmable 4^{th} order anti-alias low-pass filters
- 1 25 kHz low pass filter
- Internal temperature sensor

All comparator outputs are connected to the Event Bus, whereas all ADCs and DACs are connected to the AMBA bus. HSA8000 has 18 analog inputs that can be used for sensing the controls signal for different power applications.
A simplified block diagram of analog interface is shown in Figure 4.

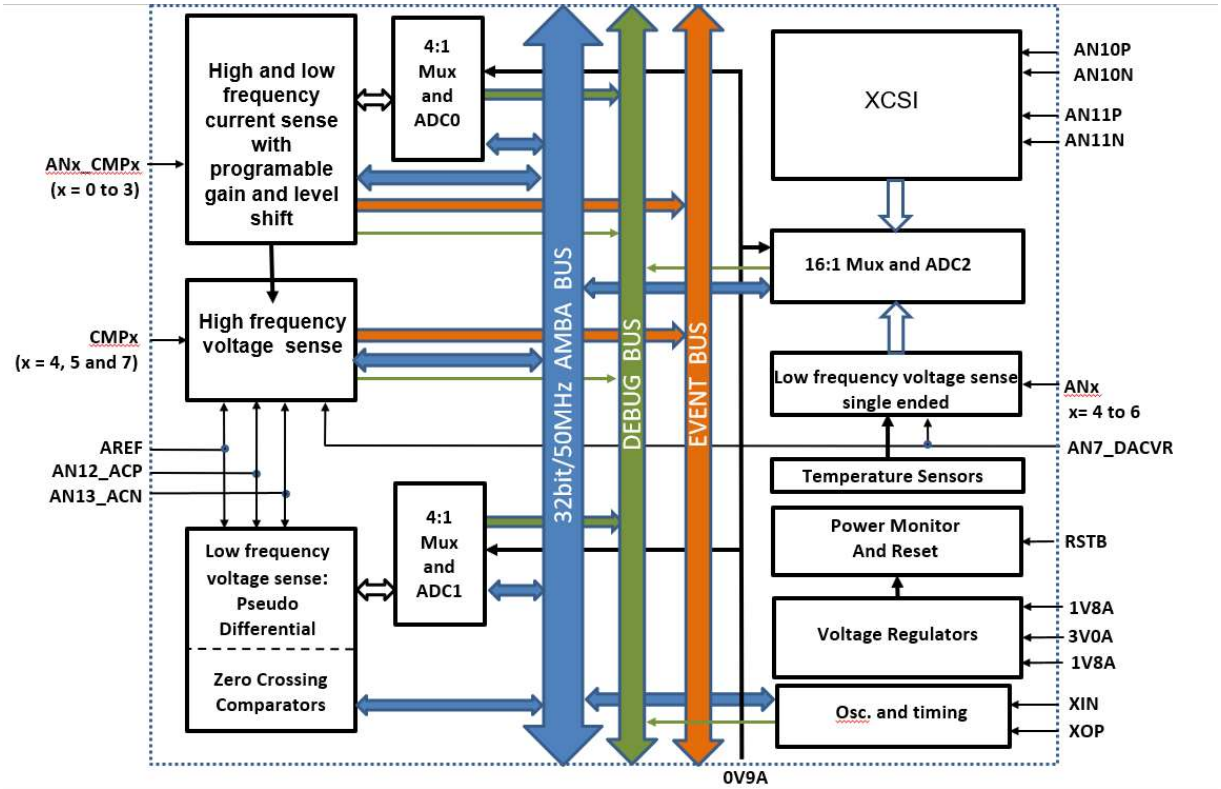

Figure 4 - Analog interface simplified block diagram.

Analog inputs ANx_CMPx (x = 0 to 3)

These are dual-function analog inputs feeding ADC0 and two high speed comparators as shown in Figure 5. The level shift and gain stage allow small signals to be scaled to the 0 - 1.8V input range of both the ADC0 and comparators. The shifting is done by the level shift DAC. The amplifier has four gains: 1, 2, 4 or 8. To reduce the impact of the switching ripple common for the power supply signals, the ADC0 input can be optionally filtered by a 4th order low pass filter 25 kHz, 50 kHz, 100 kHz or by passed.

For performing hysteretic current or voltage control or fault detection, the amplified signal is also sent to the dual comparator block. The block has two comparators. Each comparator has its own DAC reference which can be set or controlled through software to implement a dynamic behavior and advanced control methods. The comparators' outputs are sent to the Source Bus and to SWE_DEBUG.

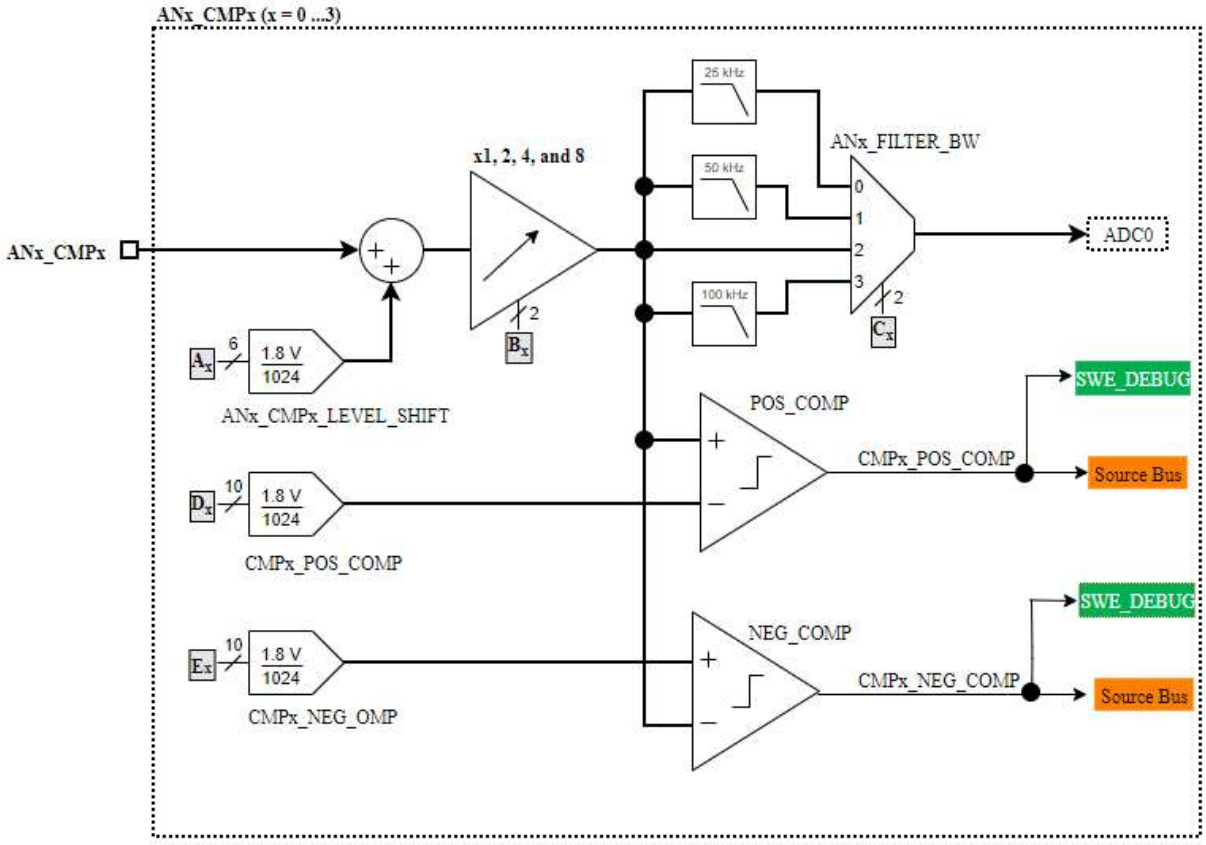

Figure 5 - Blocks associated with analog inputs ANx_CMPx (x = 0 …3)

ANx_CMPx (x = 0 to 3) specifications

Parameter	Minimum	Typical	Maximum	Unit	Conditions
Input signal range at ANx_CMPx pins	0		1.8	V	
DAC Full scale output range (absolute)			1.8	V	Absolute
DAC full scale absolute accuracy			0.25	%	
DAC resolution	10			bits	
Channel matching (POS and NEG levels)			1	%	Between separate channels
Channel matching (over/under levels)			10	%	Between separate channels
Full scale (ADC) signal window	0		1.8	V	
Full scale (ADC) signal window accuracy			0.2	%	Meter specification driven (ANSI spec of 0.2% and 0.5%)
ADC resolution	10			bits	
ADC effective sampling rate	0.06		1	MHz	

ANx_CMPx (x = 0 t0 3) Variable Gain Amplifier (VGA) Gain 1, 2, 4, and 8

Parameter	Min	Typ	Max	Units	Conditions
VGA_GAIN=1	0.96	1	1.01	V/V	25 °C
VGA_DRIFT_GAIN=1	-44	-20	4	ppm/°C	-40 - 125 °C
OFFSET	-12	4	22	mV	25 °C
OFFSET_DRIFT	-16.5	20.2	56.9	uV/°C	-40 - 125 °C
VGA_GAIN=2	1.92	2	2.03	V/V	25 °C
VGA_DRIFT_GAIN=2	-133	-59	16	ppm/°C	-40 - 125 °C
OFFSET	-10	3	19	mV	25 °C
OFFSET_DRIFT	-0.2	0.1	0.4	uV/°C	-40 - 125 °C
VGA_GAIN=4	3.86	4	4.06	V/V	25 °C
VGA_DRIFT_GAIN=4	-295	-140	15	ppm/°C	-40 - 125 °C
OFFSET	-11	3	18	mV	25 °C
OFFSET_DRIFT	-33.2	15.8	64.7	uV/°C	-40 - 125 °C
VGA_GAIN=8	7.7	8	8.12	V/V	25 °C
VGA_DRIFT_GAIN=8	-622	-312	-2	ppm/°C	-40 - 125 °C
OFFSET	-11	3	18	mV	25 °C
OFFSET_DRIFT	-33.8	15.7	65.2	uV/°C	-40 - 125 °C

ANx_CMPx (x = 0 t0 3) 4th order low pass filter (25 kHz, 50 kHz, or 100 kHz) specifications

Parameter	Min	Typ	Max	Units	Conditions
LPF_100kHz_BW	75	100	125	kHz	25 °C
LPF_100kHz_BW-Drift	75	111	148	Hz/°C	-40 to 125 °C
LPF_50kHz_BW	37.5	50	60.5	kHz	25 °C
LPF_50kHz_BW-Drift	39	57	75	Hz/°C	-40 to 125 °C
LPF_25kHz_BW	18.75	25	31.25	kHz	25 °C
LPF_25Hz_BW-Drift	19	29	40	Hz/°C	-40 to 125 °C

Analog inputs CMPx (x = 4, 5 and 7)

These analog inputs are connected to three high speed comparators as shown in Figure 6. The comparators can create events for controlling the event driven timers that are switching the powertrain switches.

For performing hysteretic charge control or dead time optimization, the input signal is sent to two comparators POS_COMP and NEG_COMP. The references to the comparators are set by their corresponding DACs. To implement the dynamic behavior and perform advanced control methods, the comparators' references can be set or controlled through the software. The DACs range can be chosen to be 1.8 V, 3V or AN7_DACVR_BUF. The CMPx block has also a zero-crossing comparator, ZC_COMP that compares the input signal with signals applied to AREF pin, AN12_BUF, AN13_ BUF or 0 V.

The comparator output CMPx _NEG_COMP is sent to the Source Bus while only one output of the comparators POS_COMP and ZC_COMP is selected to the Source Bus and SWE_DEBUG. By default, the comparator POS_COMP is selected to the Source Bus and SWE DEBUG.

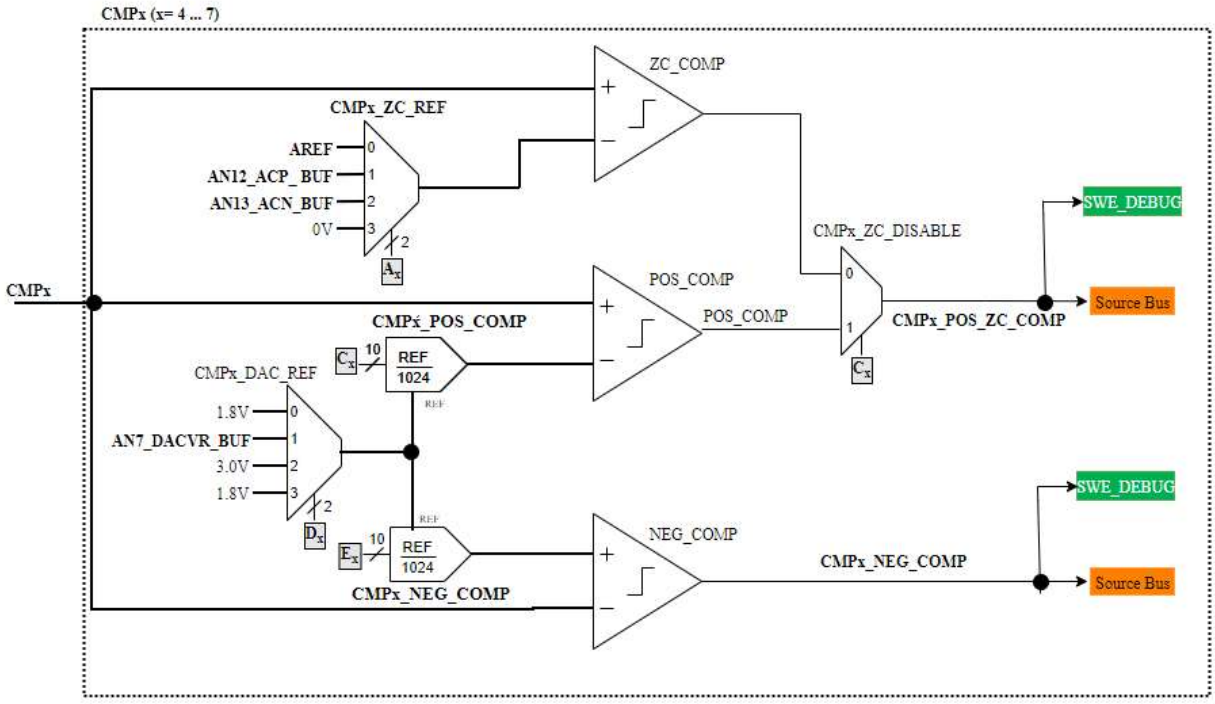

Figure 6- Blocks associated with analog inputs CMPx (x = 4, 5 and 7)

Analog inputs AN4, AN5, AN6, and AN7_DACVR

The inputs are intended for sensing of low frequency signals by ADC2 which data is controlling the powertrains. The blocks associated with them are shown in Figure 7. The signals applied to the blocks can be optionally subtracted by a signal applied to AREF input. To reduce the impact of the switching ripple common for the power supply signals, the ADC2 input can be optionally filtered by a 4th order low pass filter 25 kHz, 50 kHz, or 100 kHz.

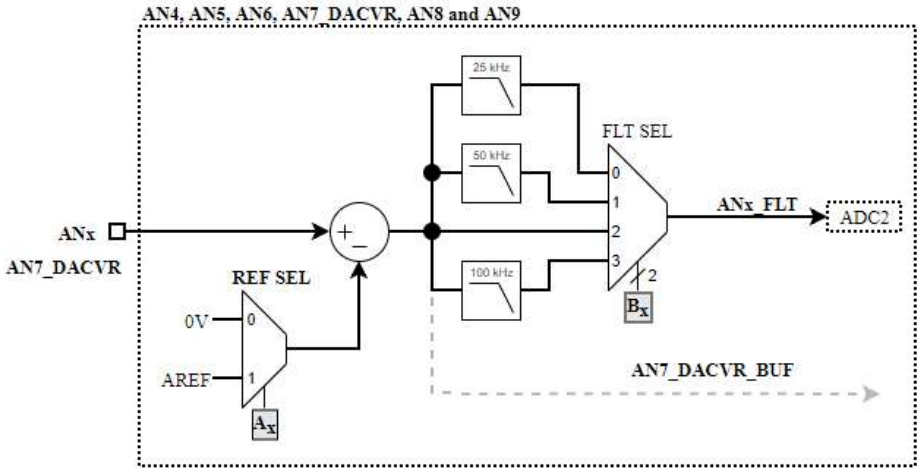

Figure 7 - Blocks associated with analog inputs AN4, AN5, AN6, and AN7_DACVR, inputs.

AN4, AN5, AN6, and AN7_DACVR specifications

Parameter	Minimu	Typical	Maximu	Unit	Conditions
Input signal range at the pin	0		1.8	V	
Full scale (ADC) signal window	0		1.8	V	
Full scale (ADC) signal window accuracy			0.2	%	Meter specification driven (ANSI spec of 0.2% and 0.5%))
ADC resolution	10			bits	
ADC effective sampling rate	0.06		1	MHz	
Internal OSR noise reduction	12			bits	In 5 kHz bandwidth
	16			bits	In 60 Hz bandwidth (used for DC offset cancellation)

Subtractor (SUB) for AN4, AN5, AN6, AN7_DACVR, AN12_ACP and AN13_ACN specifications

Parameter	Min	Typ	Max	Units	Conditions
SUBx_GAIN	0.85	1	1.13	V/V	25 °C
SUBx_GAIN_DRIFT	-0.51	-0.005	0.50	%/°C	-40 to 125 °C
OFFSET	-13	2	17	mV	25 °C
OFFSET_DRIFT	-5.97	-0.31	5.36	mV/°C	-40 - 125 °C

Subtractor (SUB) and low pass filter (LPF) for AN4, AN5, AN6, AN7_DACVR, AN12_ACP and AN13_ACN specifications

Parameter	Min	Typ	Max	Units	Conditions
SUBx+LPFx_GAIN	0.988	0.993	0.996	V/V	25 °C
SUBx+LPFx_GAIN_DRIFT	-0.4	-0.09	0.02	%/°C	-40 to 125 °C
OFFSET	-16.4	6.5	31.5	mV	25 °C
OFFSET_DRIFT	-4.05	-0.91	2.2	mV/°C	-40 to 125 °C

4th order low pass filter (25 kHz, 50 kHz, or 100 kHz) for AN4, AN5, AN6, AN7_DACVR, AN12_ACP and AN13_ACN specifications

Parameter	Min	Typ	Max	Units	Conditions
LPF_100kHz_BW	75	100	125	kHz	25 °C
LPF_100kHz_BW-Drift	75	111	148	Hz/°C	-40 to 125 °C
LPF_50kHz_BW	37.5	50	60.5	kHz	25 °C
LPF_50kHz_BW-Drift	39	57	75	Hz/°C	-40 to 125 °C
LPF_25kHz_BW	18.75	25	31.25	kHz	25 °C
LPF_25Hz_BW-Drift	19	29	40	Hz/°C	-40 to 125 °C

Analog inputs AN10P/N and AN11P/N

Figure 8 shows three blocks, two external current sense interfaces, XCSI0 and XCSI1, and high gain amplifier, HGA, associated with analog inputs AN10P/N and AN11P/N. The XCSI0 and XCSI1 blocks are designed to work with fully-differential isolation amplifier AMC1100 (used for current measurements). The blocks contain a differential to single ended programmable low gain

amplifier. To reduce the impact of the switching ripple common for the power supply signals, the ADC2 input can be optionally filtered by a 4th order low pass filter 25 kHz, 50 kHz, or 100 kHz. The AN11P/N inputs are also connected to the HGA block. The block has a programmable high gain amplifier and a 25 kHz low pass filter.

The low gain amplifier settings are 0.45, 0.4, or 0.3. The input range voltage of XCSI0 and XCSI1 blocks is -2 V to +2 V differentially applied with a common mode of 1.2 V. The amplifier output voltage is somewhere between -0.9 V and +0.9 V. Since, the ADC input range is 0 V to 1.8 V, 0.9 V is added to shift the amplifier output voltage to be in ADC2 range (see Figure 8).

The HGA block gain settings are 30, 60, or 120. The HGA input range depends on the gain settings. The HGA allows small signals to be amplified for the 0 V - 1.8 V ADC2 range.

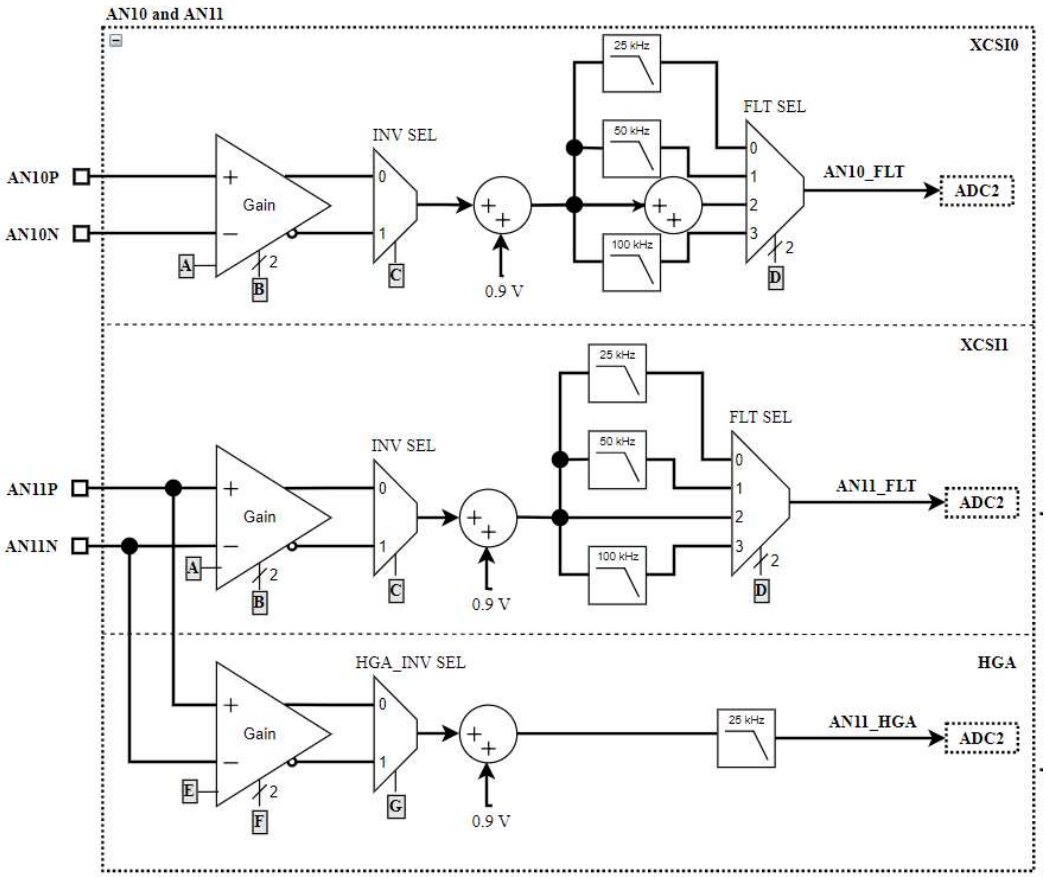

Figure 8 - Blocks associated with analog inputs AN10P/N and AN11P/N inputs

AN10P/N and AN11P/N (XCSIx x = 0 and 1) specifications

Parameter	Min	Typ	Max	Unit	Conditions
Input signal range at AN10P/N and AN11P/N	-0.04	1.2	3	V	Absolute range
Channel matching					Maintained by 0.2% absolute

AN10P/N and AN11P/N (XCSIx x = 0 and 1) offset and gain specifications

Parameter	Min	Typ	Max	Units	Conditions

XCSIx Gain=0.45_CM	0.898	0.912	0.927	V	25 °C
XCSIx Gain=0.45_CM_Drift	-51	15	82	uV/°C	-40 to 125 °C
XCSIx Gain=0.45_Gain	0.416	0.45	0.46	V/V	25 °C
XCSIx Gain=0.45_Gain_Drift	-64.4	-1	62.5	ppm/°C	-40 to 125 °C
XCSIx Gain=0.40_CM	0.873	0.905	0.938	V	25 °C
XCSIx Gain=0.40_CM_Drift	-54	15	85	uV/°C	-40 to 125 °C
XCSIx Gain=0.40_Gain	0.38	0.40	0.42	V/V	25 °C
XCSIx Gain=0.40_Gain_Drift	-108.8	-50	8.82	ppm/°C	-40 to 125 °C
XCSIx Gain=0.30_CM	0.887	0.911	0.935	V	25 °C
XCSIx Gain=0.30_CM_Drift	-55	19	93	uV/°C	-40 to 125 °C
XCSIx Gain=0.30_Gain	0.29	0.3	0.33	V/V	25 °C
XCSIx Gain=0.30_Gain_Drift	-65.9	-51	-35.86	ppm/°C	-40 to 125 °C

4th order low pass filter (25 kHz, 50 kHz, or 100 kHz) AN10P/N, AN11P/N, specifications

Parameter	Min	Typ	Max	Units	Conditions
LPF_100kHz_BW	75	100	125	kHz	25 °C
LPF_100kHz_BW-Drift	75	111	148	Hz/°C	-40 to 125 °C
LPF_50kHz_BW	37.5	50	60.5	kHz	25 °C
LPF_50kHz_BW-Drift	39	57	75	Hz/°C	-40 to 125 °C
LPF_25kHz_BW	18.75	25	31.25	kHz	25 °C
LPF_25Hz_BW-Drift	19	29	40	Hz/°C	-40 to 125 °C

AN11P/N High Gain Amplifier (HGA) specifications

Parameter	Min	Typ	Max	Units	Conditions
HGA Gain =120_CM	0.88	0.93	0.98	V	25 °C
HGA Gain =120_CM_Drift	-0.31	0.09	0.49	uV/°C	-40 to 125 °C
HGA Gain = 120_Gain	56.49		60	V/V	25 °C
HGA Gain =120_Gain_Drift	-5.69	0.19	-5.32	ppm/°C	-40 to 125 °C
HGA Gain =60_CM	0.9	0.93	0.96	V	25 °C
HGA Gain =60_Offset_Drift	-0.53	-0.2	0.13	uV/°C	-40 to 125 °C
HGA Gain =60_Gain	27.5	30	30.1	V/V	25 °C
HGA Gain =60_Gain_Drift	8.11	8.5	8.89	ppm/°C	-40 to 125 °C
HGA Gain =30_CM	0.89	0.9	0.94	V	25 °C
HGA Gain =30_CM_Drift	5.35	5.6	5.57	mV/°C	-40 to 125 °C
HGA Gain= 30_Gain	14.48	15	15.18	V/V	25 °C
HGA Gain =30_Gain_Drift	-0.4	-0.09	0.22	%/°C	-40 to 125 °C

4th order low pass filter (25 kHz, 50 kHz, or 100 kHz) AN10P/N, AN11P/N, specifications

Parameter	Min	Typ	Max	Units	Conditions
HGA_LPF_25kHz_BW	18.75	25	31.25	kHz	25 °C
HGA_LPF_25Hz_BW-Drift	19	29	40	Hz/°C	-40 to 125 °C

Analog inputs AN12_ACP and AN13_ACN

The inputs are intended for sensing of low frequency signals specially AC grid voltages. The blocks associated with them are shown in Figure 9. The signals applied to the blocks can be optionally subtracted by a signal applied to AREF input. To reduce the impact of the switching ripple common for the power supply signals, the ADC1 input can be optionally filtered by a 4th order low pass filter 25 kHz, 50 kHz, or 100 kHz. The unfiltered signals are also connected to the zero-crossing comparator (PLL_COMP) which output signal is used by the GRID PLL block. They can be as well selected as input of ZC comparators of the CMPx (x = 4 to 7) blocks.

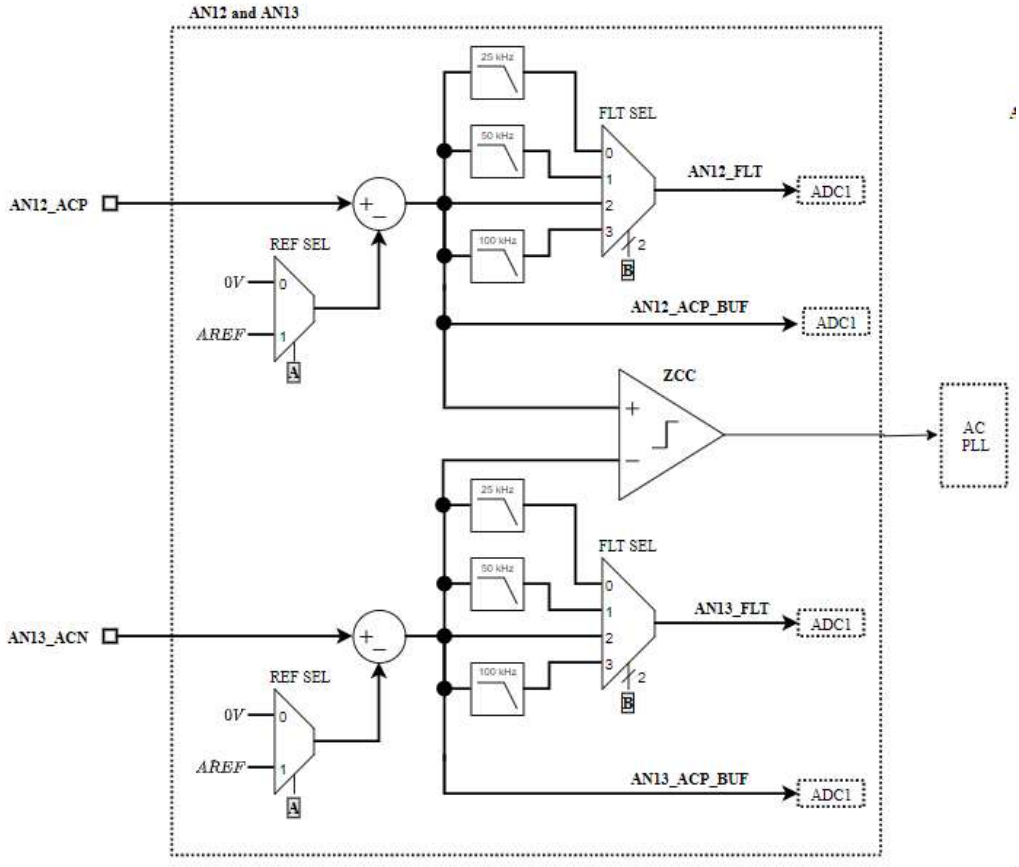

Figure 9 - Blocks associated with analog inputs AN12_ACP and AN13_ACN.

Subtractor (SUB) for AN12_ACP and AN13_ACN specifications

Parameter	Min	Typ	Max	Units	Conditions
SUBx_GAIN	0.85	1	1.13	V/V	25 °C
SUBx_GAIN_DRIFT	-0.51	-0.005	0.50	%/°C	-40 to 125 °C
OFFSET	-13	2	17	mV	25 °C
OFFSET_DRIFT	-5.97	-0.31	5.36	mV/°C	-40 - 125 °C

Subtractor (SUB) and low pass filter (LPF) for AN12_ACP and AN13_ACN specifications

Parameter	Min	Typ	Max	Units	Conditions
SUBx+LPFx_GAIN	0.988	0.993	0.996	V/V	25 °C
SUBx+LPFx_GAIN_DRIFT	-0.4	-0.09	0.02	%/°C	-40 to 125 °C
OFFSET	-16.4	6.5	31.5	mV	25 °C
OFFSET_DRIFT	-4.05	-0.91	2.2	mV/°C	-40 to 125 °C

4th order low pass filter (25 kHz, 50 kHz, or 100 kHz) for AN12_ACP and AN13_ACN specifications

Parameter	Min	Typ	Max	Units	Conditions
LPF_100kHz_BW	75	100	125	kHz	25 °C
LPF_100kHz_BW-Drift	75	111	148	Hz/°C	-40 to 125 °C
LPF_50kHz_BW	37.5	50	60.5	kHz	25 °C
LPF_50kHz_BW-Drift	39	57	75	Hz/°C	-40 to 125 °C
LPF_25kHz_BW	18.75	25	31.25	kHz	25 °C
LPF_25Hz_BW-Drift	19	29	40	Hz/°C	-40 to 125 °C

Analog to digital converters ADC0, ADC1 and ADC2

HSA8000 has three 10-bit analog to digital converters that convert the analog signals into digital. The ADC0 and ADC1 have four input and four output channels, while ADC2 has sixteen input and eight output channels. By default, the ADC clock is 20 MHz, but can be reduced to save power. It takes 14 clock cycles to convert, which results in a maximum conversion rate f_{s_max} = 1.429 MHz. When n channels are sampled, then the sampling frequency is reduced by the same factor:

$$f_s = \frac{f_{s\_max}}{n}$$

Each ADC channel output includes a digital filter. The raw data from the ADC occupies bits 15 … 6, and the bits 5 … 0 are 0. If the digital filtering is enabled, then all the bits are used. The filter adds more resolution hence the bits 5 … 0 will be populated, but the MSB stays bit 15. The filter transfer function is similar to a first order RC low pass filter.

The 3-dB bandwidth, B, is calculated by the following equation:

$$B = -\frac{f_s}{2\pi}\ln\left(1 - \frac{\alpha}{4096}\right)$$

where $0 < \alpha < 255$ value set in the corresponding ADC register. When $\alpha = 255$ the filter is bypassed, while when $\alpha = 0$ the filter will latch the current value indefinitely. For a given bandwidth and sampling frequency, the value of α that should be set in the register is calculated by the equation:

$$\alpha = \left(1 - e^{-\frac{2\pi B}{f_s}}\right) 4096$$

For example, let assume that the desired filter 3-dB bandwidth is 10 kHz and the sampling frequency is f_s = 1.429 MHz, then substituting in the equation above, the value that should be written in the filter register is 176.

The unfiltered output of every ADC channel is connected to two digital comparators (See Figure 10 and Figure 11). The signal is compared between the maximum and minimum values. When the signal is larger than the maximum or smaller than the minimum values, a bit of the corresponding

channel is set to 1 and exported to the Source Bus. This bit is sticky and is cleared manually. The status for the minimum and maximum values of the ADC0 and ADC1 comparators are OR and shown in field C_x (Figure 10). For ADC2, the comparators' status for the minimum and maximum values are and shown in C_x and D_x fields (Figure 11).

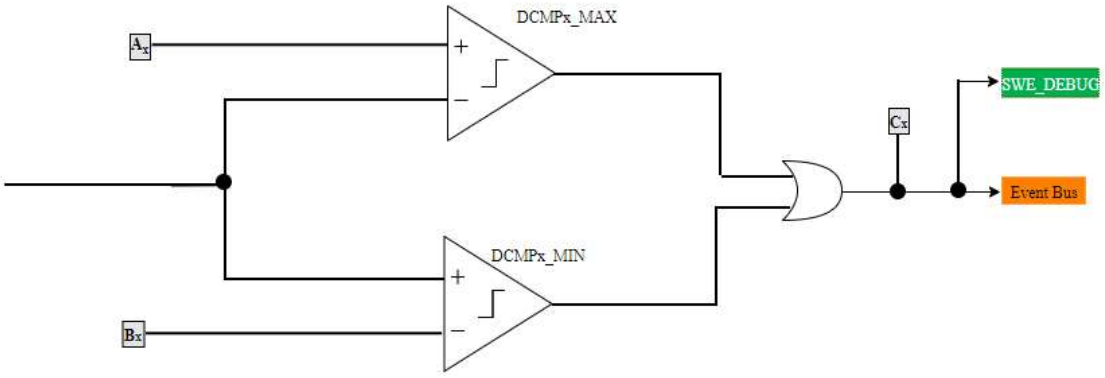

Figure 10 – ADC0 and ADC1 Output Status circuit

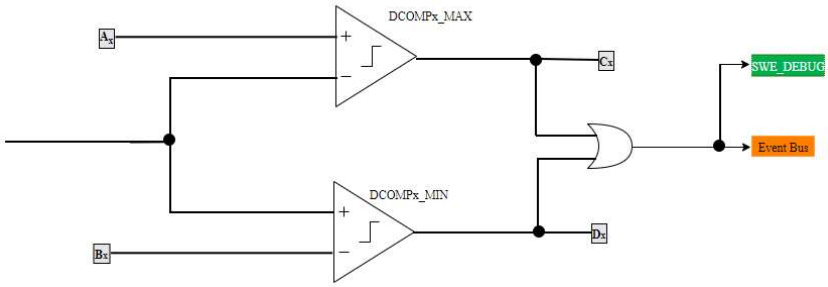

Figure 11 -ADC2 Output status circuit

ADC0

The ADC0 digitizes signals coming from the ANx_CMPx (x = 0 to 3) inputs. Figure 12 shows the ADC0 block diagram. ADC0 has four sequences. For initializing the ADC0, first the number of time slots should be specified:

00b: all four of the time slots are read (0, 1, 2, 3, 0, 1, 2, 3, 0, ...)
01b: only first of the time slots is read (0, 0, 0, 0, 0, 0, ...)
10b: only the first two time slots are read (0, 1, 0, 1, 0, 1, ...)
11b: only the first three time slots are read (0, 1, 2, 0, 1, 2, …)

The second step is to select the channels for the time slots:

00: AN0_FLT
01: AN1_FLT
10: AN2_FLT
11: AN3_FLT

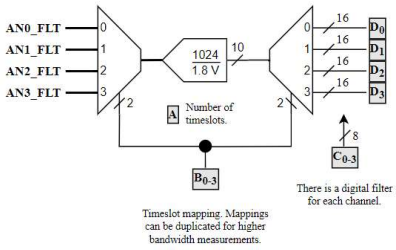

Figure 12 - ADC0 block diagram.

ADC1

The ADC1 digitizes signals coming on the AN12_ACP and AN13_ACN inputs. Figure 13 shows the ADC1 block diagram. ADC1 has four time slots for measuring the input filtered and unfiltered signals. For initializing of ADC1, first the number of time slot should be specified:

00b: all four of the time slots are read (0, 1, 2, 3, 0, 1, 2, 3, 0, ...)
01b: only first of the time slots is read (0, 0, 0, 0, 0, 0, ...)
10b: only the first two time slots are read (0, 1, 0, 1, 0, 1, ...)
11b: only the first three time slots are read (0, 1, 2, 0, 1, 2, …)

The second step is to select the channel for the time slot:
00b: AN12_FLT
01b: AN13_FLT
10b: AN12_ACP_BUF
11b: AN13_ACN_BUF

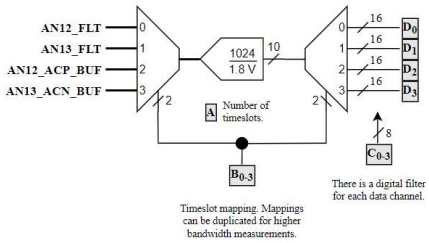

Figure 13 - ADC1 block diagram.

ADC2

Figure 14 shows the ADC2 block diagram. ADC2 has eight time slots. The initialization of ADC2 is done in three steps. First the number of time slots is selected:

000b - eight of the time slots are read (0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7, 0, ...)
001b - only the first of the time slots is read (0, 0, 0, 0, 0, 0, ...)
010b - only the first two time slots are read (0, 1, 0, 1, 0, 1, ...)
011b - only the first three time slots are read (0, 1, 2, 0, 1, 2, 0, ...)
100b - only the first four time slots are read (0, 1, 2, 3, 0, 1, 2, 3, 0, ...)
101b - only the first five time slots are read (0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, ...)
110b - only the first six time slots will be read (0, 1, 2, 3, 4, 5, 0, 1, 2, 3, 4, 5, 0, ...)
111b - only the first seven time slots are read (0, 1, 2, 3, 4, 5, 6, 0, 1, 2, 3, 4, 5, 6, 0, ...)

Then the channels associated for each time slot are selected.
000b – CH0
001b – CH1
010b – CH2
011b – CH3
100b – CH4

101b – CH5
110b – CH6
111b – CH7

The third step is to select the input for the corresponding channel:

0000b - Internal Temperature
0001b - AN4_FLT
0010b - AN5_FLT
0011b - AN6_FLT
0100b - AN7_FLT
0101b - High Impedance (no connect)
0110b - 1V8A
0111b – AN10_FLT
1000b – AN11_FLT
1001b – Reserved
1010b – Reserved
1011b - GND
1100b - AN11_HGA
1101b - CAL_DAC
1110b - Reserved
1111b – Reserved

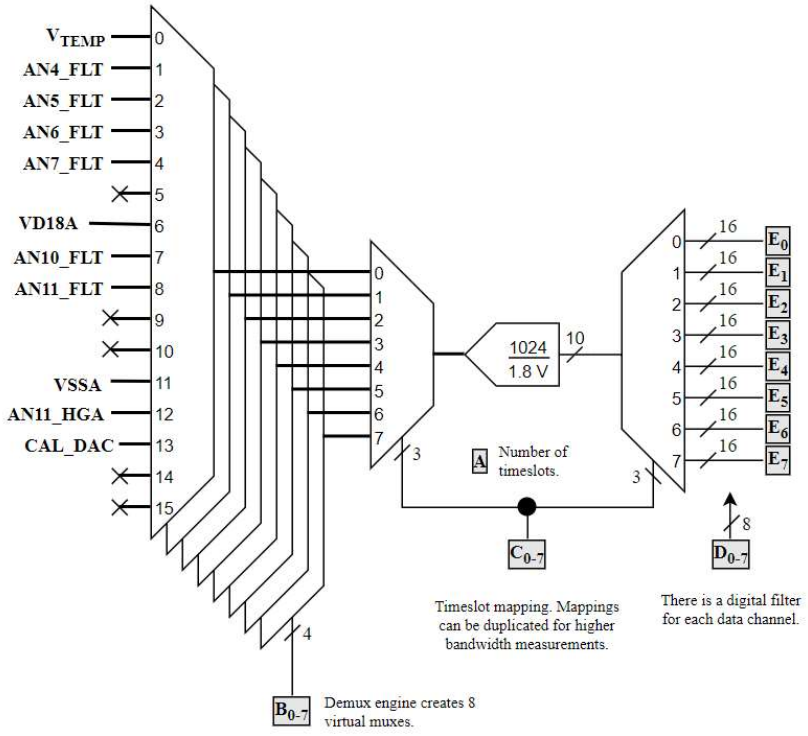

Figure 14 - ADC2 block diagram.

Analog DACs

The HSA8000 has the following analog DACs:

- 16 10-bit DACs for comparators internal references
- 1 10-bit DAC for debugging

- 4 6-bit DACs for level shifting

The DACs values for the comparator references and debugging, DAC is calculated by:

$$DAC = \frac{1024*V_{ref}}{1.8}$$

where V_{ref} is the desire voltage reference.

The DAC value for level is calculated by:

$$DAC = \frac{64*V_{shift}}{1.8}$$

where V_{shif} is the level shifting voltage.

DAC Specifications

Parameter	Min	Typ	Max	Unit	Conditions
Input signal range	0		1.8	V	Referenced to local ground or to separate AREF
DAC (reference level setting) range	0		1.8	V	
DAC full scale accuracy (absolute)			1.0	%	
DAC (reference level setting) resolution	10			bits	
DAC (reference) update rate	0		1	MHz	
dv/dt full scale detection range	1		1000	mV/µs	
dv/dt range accuracy	-20		20	%	
dv/dt range resolution			1024	steps	
dv/dt threshold update rate	0		1	MHz	

Temperature sensor

The HSA8000 temperature is measured by a temperature sensor. The sensor is designed to change its output voltage based upon HSA8000 temperature. The output voltage is measured by ADC2. The relationship between the temperature and the voltage is linear. The slope is approximately a constant while the offset is process dependent. Therefore, the temperature sensor requires one-point calibration.

Temperature sensor specification

Parameter	Min	Typ	Max	Unit
Range	-40		125	°C
Resolution	-5		5	°C

Parameter	Min	Typ	Max	Units	Conditions
Temperature_Sensor_output_voltage	0.937	1.156	1.748	V	25 °C, VDDA=VDD =3.3 V
Temperature_Sensor_Slop	3.605	3.624	3.643	mV	-40 to 125 °C, VDDA=VDD =3.3V

Electrical characteristics

The following tables list the characterization parameters for the HSA8000.

Supply voltages and power consumption

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Supply voltage	VDDA	3	3.3	3.5	V	
Supply voltage	VDD	2.7	3.3	3.5	V	
Internally regulated voltage	VD3A		3.0		V	VDDA > 3.2V
Internally regulated voltage	V1P8A, V1P8D		1.8		V	VDDA, VDD > 2.7V
Power consumption (operating mode)	Pw_opmode			300	mW	VDDA, VDD = 3.3V
Power consumption (low power mode)	Pw_lpmode			10	mW	VDDA, VDD = 3.3 V

Analog and digital regulators specifications

Parameter	Min	Typ	Max	Units	Conditions
Digital_1.8V_Linear_Regulator_output_voltage	1.789	1.809	1.829	V	25 ºC, VDDA=VDD =3.3 V
Digital_1.8V_Linear_Regulator_output_voltage set to 1.7V	1.693	1.711	1.730	V	25 ºC, VDDA=VDD =3.3V
Analog_1.8V_Linear_Regulator_output_voltage	1.796	1.821	1.845	V	25 ºC, VDDA=VDD =3.3 V
Analog_3V_Linear_Regulator_output_voltage	2.970	3.013	3.057	V	25 ºC, VDDA=VDD =3.3V
ADCCM_output_voltage	0.896	0.909	0.9220	V	25 ºC, VDDA=VDD =3.3V
Analog_Core_Current_Consumption	35.3	35.2	37.2	mA	25 ºC, VDDA=VDD =3.3V
Digital_Core_Current_Consumption	60.2	63.9	65.6	mA	25 ºC, VDDA=VDD =3.3 V
Total_Supply_Current_Consumption	95.5	99.1	102.8	mA	25 ºC, VDDA=VDD =3.3 V

Power on Reset

Parameter	Min	Typ	Max	Unit	Conditions
Threshold voltage, Vth0			1.4	V	Threshold below which the power on reset circuit is not active
Threshold voltage,Vth1	2.62		2.67	V	Threshold at which V_{CC} is declared “valid”
Threshold voltage, Vth2	2.54		2.61	V	Threshold at which V_{CC} is declared “lost”
Reset time, t_Reset			200	ms	

Digital Interface

The digital interface processes the analog signals sensed by the analog interface and sends signals to control the power train switches. The digital interface includes the following blocks:

- CPU
- ROM and RAM
- Switch Engine
- I/O Block
- Grid PLL
- COMMs
- Serial Structure interface
- Watchdog system
- Math Accelerators (sin, cos, sqrt, divider).
- RTC
- ADCs
- DACs.
- DEBUG infrastructure

As it is shown in Figure 2, there are two buses for communication between the different HSA8000 blocks:

- AMBA Bus (32 bit/ 50 MHz)
- EVENT SOURCE Bus.

The AMBA BUS connects all HSA8000 blocks to the CPU. The EVENT SOURCE BUS connects the analog interface and the I/O block to the switching engine. The EVENT SOURCE BUS is carrying data from the comparators, GPIOs, and other digital AUX sources to the timing engine and is synchronized with 100 MHz clock. In the following section the functionality of the digital interface blocks is presented.

CPU

The 32-bit RISC micro-controller operates at 50 MHz with 64 kiB internal RAM, 4 kiB ROM (boot loader) and provides the execution of code for customizable control algorithms.
The CPU core is an EnSilica 32-bit RISC processor. For more information, refer to the EnSilica web site at ***http://www.ensilica.com***.

The specific configuration used is outlined in the following table. The interface to the various functional blocks is through the AMBA bus.

Configuration type	**Configuration options used**
Architectural	BITS:32, REGISTERS:16, von Neumann
Clock speed	50 MHz maximum
Memory	ENDIAN (Little), 32-bit byte-addressable

ROM and RAM

The memory-model is 32 bits wide and is byte-addressable using appropriate byte-access load/store instructions. There are 24 addressing bits, but for 32-bit word accesses the 2 least significant address bits are ignored.

Memory map

The HSA8000 memory map is listed in the following table below:

Address	Description
0x000000 - 0x000FFF	ROM (4 kB)
0x020000 - 0x027FFF	RAM0 (32 kB)
0x028000 - 0x02FFFF	RAM1 (32 kB)
0x800000 - 0xFFFFFF	AMBA

Internal ROM

The HSA8000 internal ROM is 4 kB of on-chip ROM, organized as 1k 32-bit words. The internal ROM starts at address 0x000000. When the processor resets, it starts executing instructions from this ROM. The ROM-based routines include:

- initialization - stack pointer, exception vectors, watchdogs
- self-checks - ROM checksum, RAM test, flash code checksum
- flash-boot-load instructions into the RAM from external flash memory connected to the SPI, check the checksum, and if OK jump to the first instruction.

Internal SRAM

The HSA8000 internal SRAM is organized into two adjacent 32 kB blocks (64 kB total), in von-Neumann mode.

AMBA Structure access

Memory addresses from 0x800000 to 0xFFFFFF map to the embedded AMBA Structure bus. HSA8000 complies with the AMBA 4 standard. Accesses to AMBA-space must be 32-bit word aligned (the two least-significant-bits of the address must be zero). An exception is generated when unaligned accesses are attempted.
Details for each of the set of registers within each Structure is provided in the ***IXC_EP2_Register Map*** document.

Flash memory

HSA8000 contains 256 kB flash memory. The ***Solantro Test and Control Tool User's Guide*** **and the** ***Solantro Test and Control Tool Programmer's Guide*** details the compilation and debug environment for building loads that are downloaded into the HSA8000 flash memory for auto-booting. The flash memory is accessed serially through the SPI interface, which is connected to the CPU through the AMBA bus, like any other peripheral. At reset, the bootloader (located in ROM) will copy the executable image of the code from the flash into the RAM and will launch

it in execution.

Switching Engine

The Switching Engine is one of the most important HSA8000 blocks. It is used to control the power switches of different applications such as DC/DC convertors, DC-AC invertors, battery chargers, and others. The high-level block diagram of the HSA8000 Switching Engine block is shown in Figure 15. The Switching Engine block has four main blocks: Event Control, Fault Processor, Timing Engine, and Driver Control.

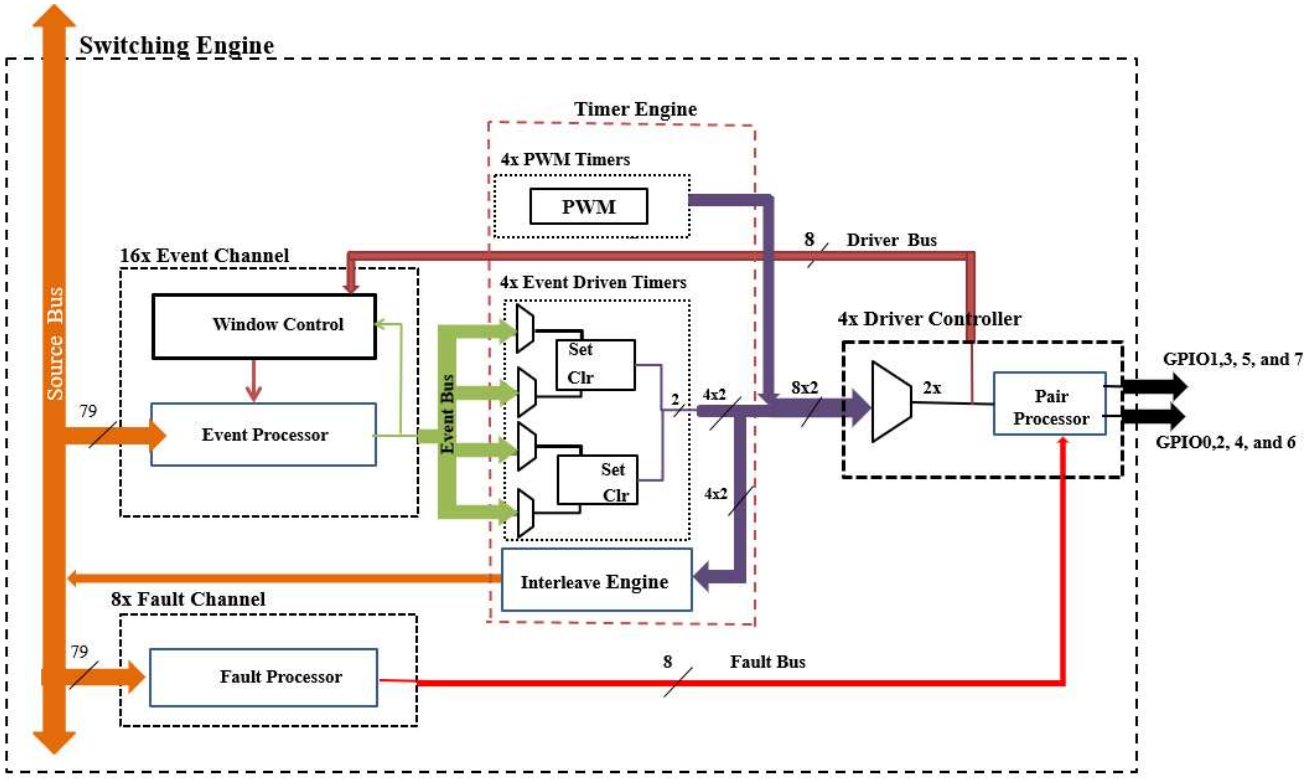

Figure 15 - High level block diagram of HSA8000 Switching Engine

Event Control block

The Event Control block consists of 16 Event Channels. Every channel has an Event Processor and a Window Control block. The function of Event Control block is to select up to 16 signals from the Source Bus, process them and output up to 16 events. The Source Bus includes signals from Analog Sensing blocks, Digital blocks (ADC, timing engine) and any GPIO.

Event Processor

The Event Processor[x] block diagram (**x** is from 0 to 15 - the channel number) is shown in Figure 16.

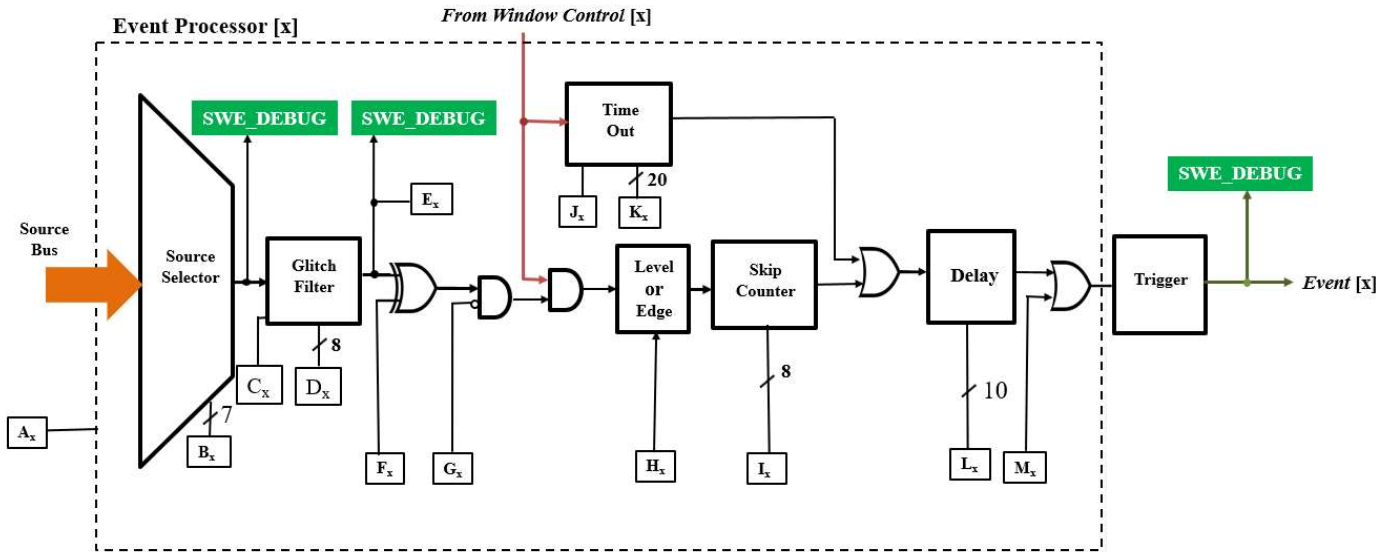

Figure 16 - An Even Processor block diagram

Window Control block

Each of the 16 Event Control channels have its associated Window Channel[x], where x is the channel number from 0 to 15. An Event processor only looks for the selected event source when its associated window is turned ON. Basically, a window is turned ON based on a rising or falling edge of a driver output or a combination of those. The **Window Triger Bus** coming from the eight driver output signals feeds all 16 window control channels (see Figure 15). The window is always turned OFF when the associated event occurs (either based on the hardware source or on timeout). The output of the windows is connected also to SWE_DEBUG. A Window Channel is shown in Figure 17.

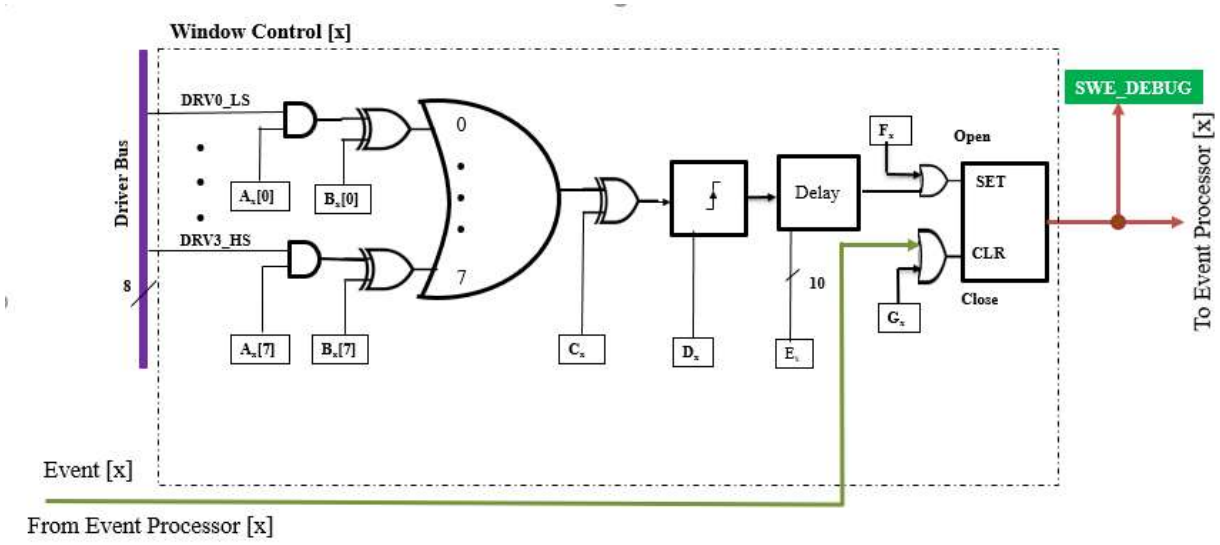

Figure 17 - A Window Channel of the HSA8000 Window Control block

Timing Engine block

The time engine block consists of Event Driven Timer and Pulse Width Modulation (PWM) blocks.

Event Driven Timer

The events from the Event Processor block are sent to the Event Driven Timer block. These events are used by the Event Driven Timer block to create signals for the Driver Controller block (Figure 15). The Event Driven Timer block has 4 timers (Event Driven Timer [**x**], where x is equal from 0 to 3). Every Event Driven Timer has 16 inputs and one output pair. The signals of the four pairs are sent to the Driver Controller block and also to the Interleave Engine (Figure 15).

The Event Driven Timer[x] block contains two identical blocks EDT**x**_LS and EDT**x**_HS (Figure 18). The blocks have two selectors and a Flip Flop. The selectors select the turn ON and OFF events for the driver. If necessary, the driver can be forced ON and OFF by the controller. The timer can also be disabled after the first event.

The EDT**x**_LS and EDT**x**_HS are also connected to the SWE_DEBUG.

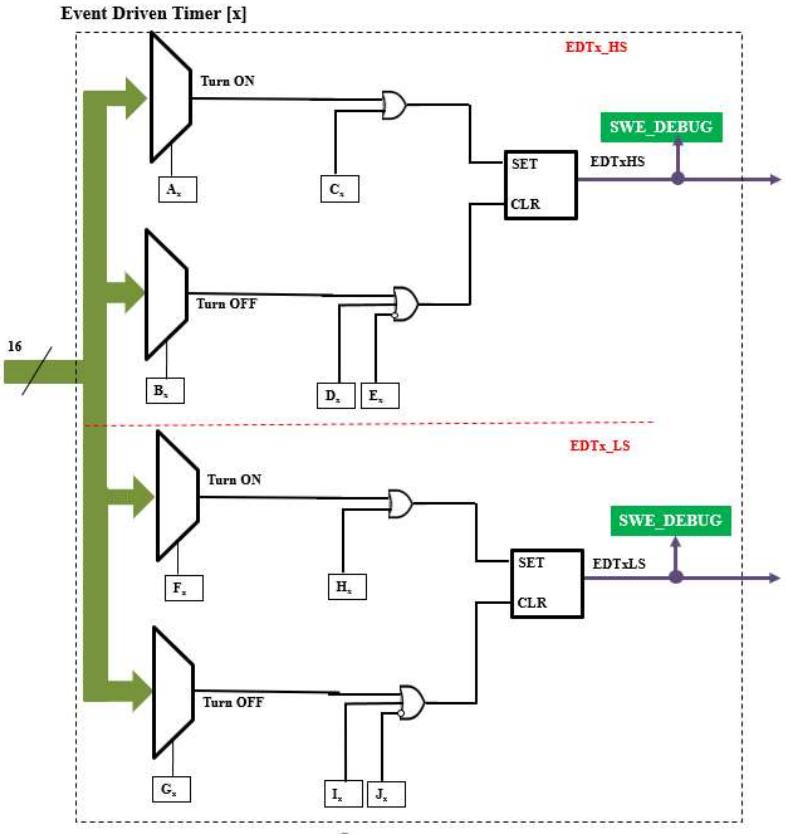

Figure 18 – An Event Timer block diagram

Interleave Engine

The Interleave Engine is part of the Event Driven Timer block (Figure 19). It has a selector selecting a master phase from the inputs of the four pairs coming from the Event Driven Timers.

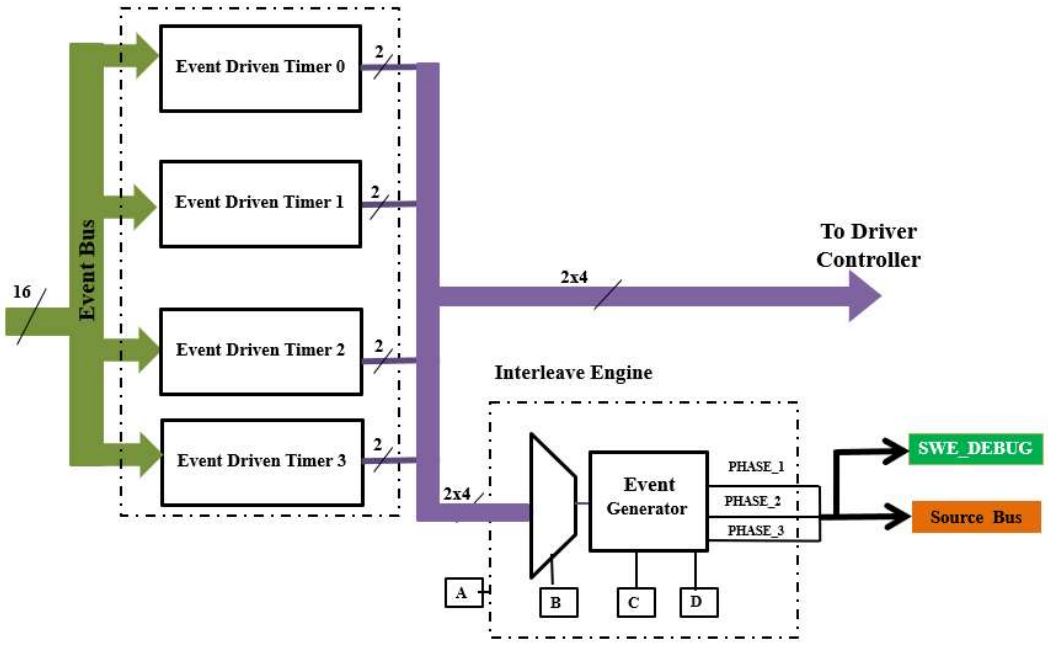

Figure 19 - Event Driven Timer block diagram

PWM Timer

The PWM timer block is a part of the HSA8000 Timing Engine (Figure 15). The PWM timing is only controlled by the software. It has four channels (CH. x for x equal from 0 to 3); every channel has direct and complementary outputs. The PWM timer outputs are connected to the Driver Controller block of the HSA8000 Switching Engine.

Fault Processing block

Fault Processing block selects and processes up to 8 faults from the Source Bus (sources from the analog interface and I/O block). It has 8 channels. Every channel sends fault signals to the 4 Pair Processors from (Figure 15).

Figure 20 shows a Fault Channel[x] diagram (x is equal from 0 to 7). Every channel has a fault selector selecting one source signal from the Source Bus. A channel is enabled with (A_x) register.

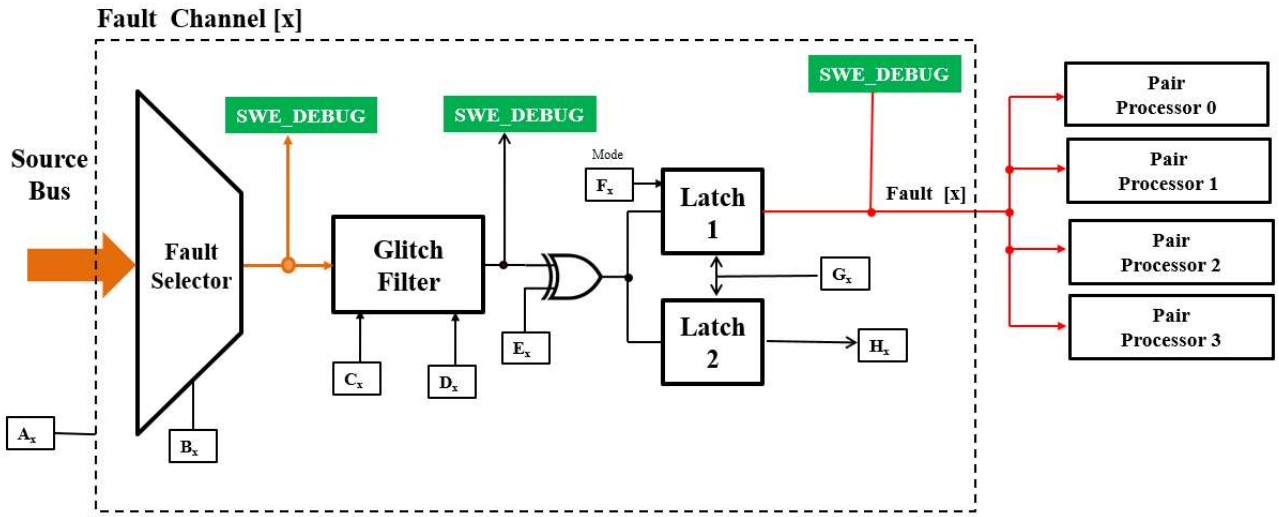

Figure 20 - A Fault Channel [x] block diagram

Driver Controller block

As shown in Figure 15, 8 pairs - 4 from Event Driven Timer block and 4 from PWM block are inputs to the Driver Controller. The block diagram of the Driver Controller block is shown in Figure 21.

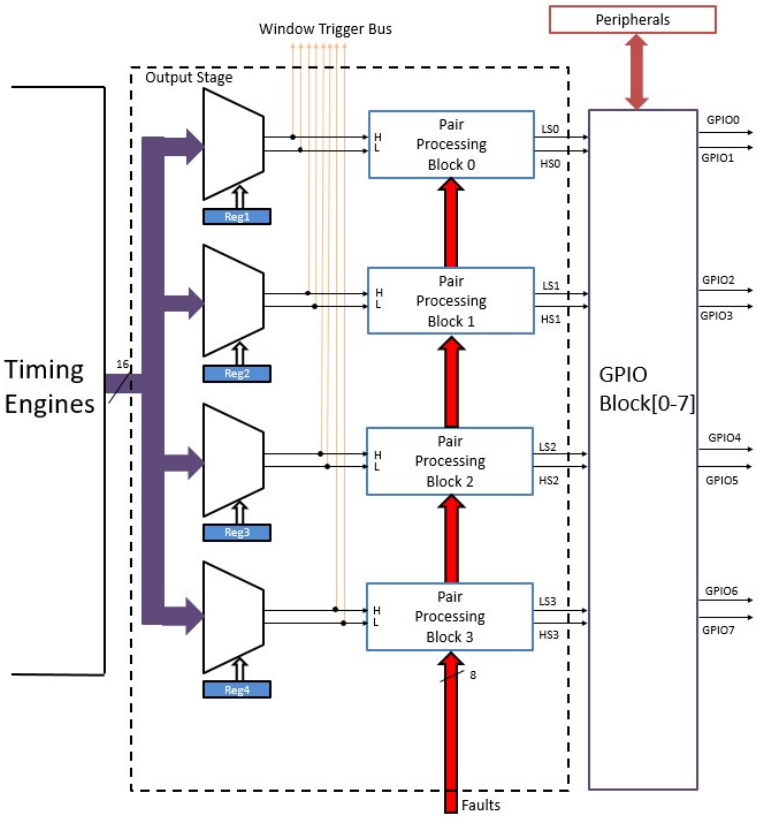

Figure 21 - The block diagram of the Gate Control block

The Driver Controller has 4 selectors that select 4 pairs to be exported to the driver pins. The selected pair is sent to a Pair Processor[x] block whose diagram is shown in Figure 22.

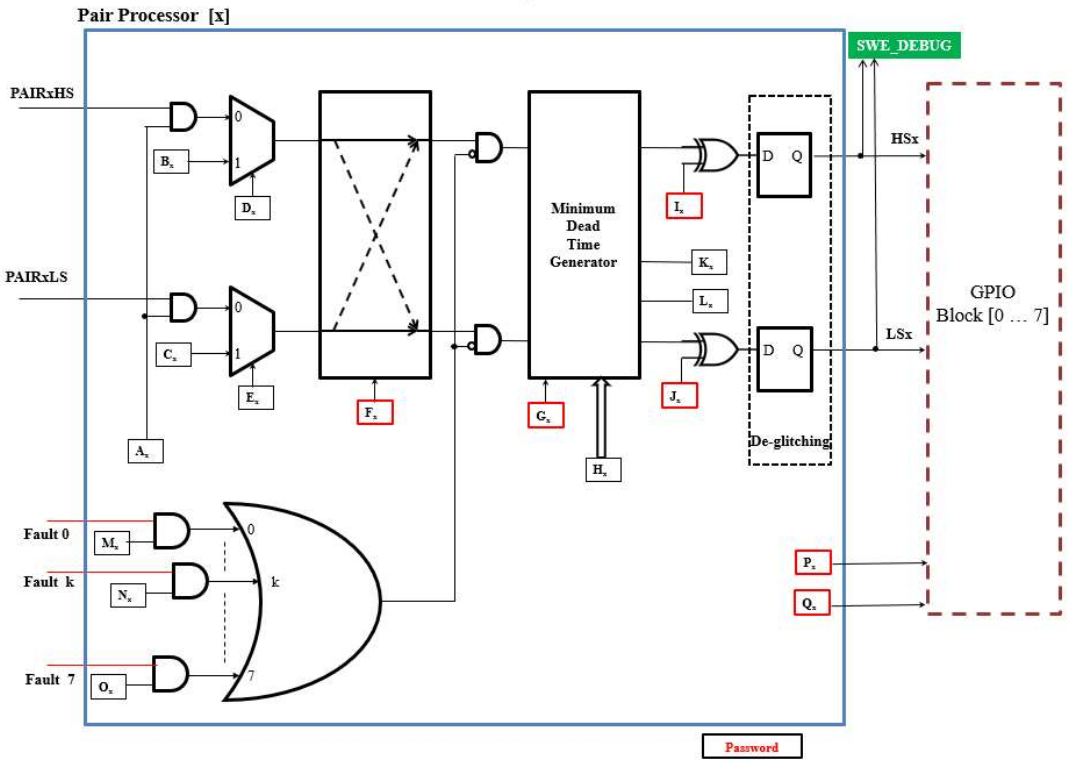

Figure 22 - The block diagram of the Pair Processor block

I/O Block

The I/O block controls the functionality of the digital input/output pins: The GPIO pins can be configured as:

- Push-pull outputs
- Open-drain outputs
- Inputs with optional pull-up or pull-down.

The HSA8000 has 32 GPIOs. In addition to the basic IO functions, to every GPIO can be assigned an alternate function, i.e. to be connected to a hardware peripheral (either an input or an output). GPIO0 to GPIO7 are slightly different than GPIO8 to GPIO31, since they have faster dedicated paths to export the driver signals.

GPIOx Block (x = 0 to 7) is shown in

Figure 23. GPIO0 to GPIO7 can export the driver signals coming from the Driver Controller block when P_x and Q_x fields of the Pair Processor block () are set to 1 (Figure 22) . By default, the drivers are enabled (P_x and Q_x fields are set 1). The fast driver paths are illustrated in Figure 23.

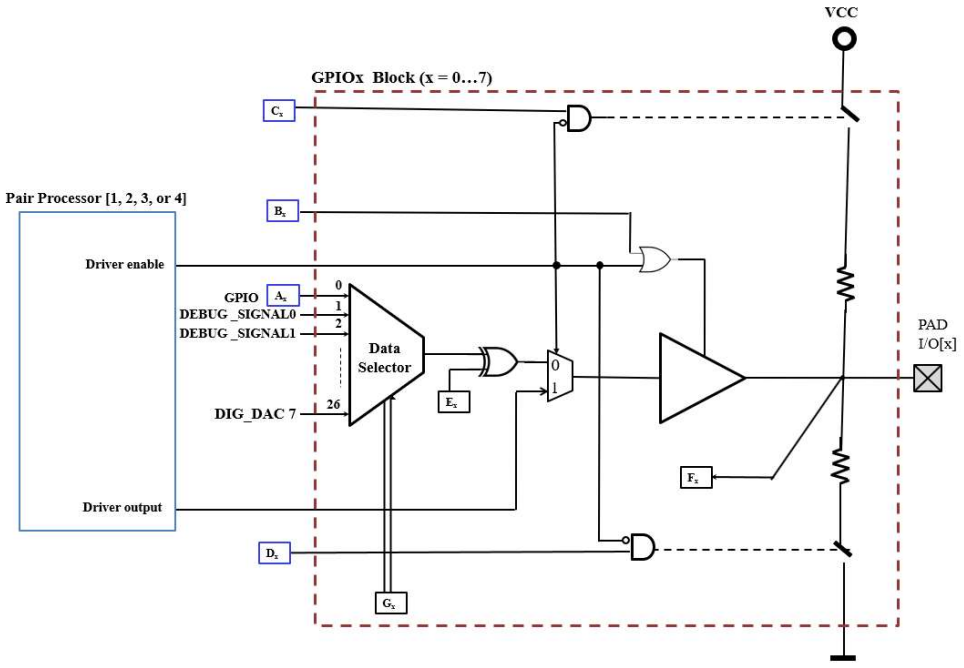

Figure 23 - The block diagram of GPIOx Block (x = 0…7)

GPIOx Block (x = 8 to 31) is shown in

Figure 24. The blocks do not have special driver paths and therefore, the gate drivers cannot be exported on these GPIOs.

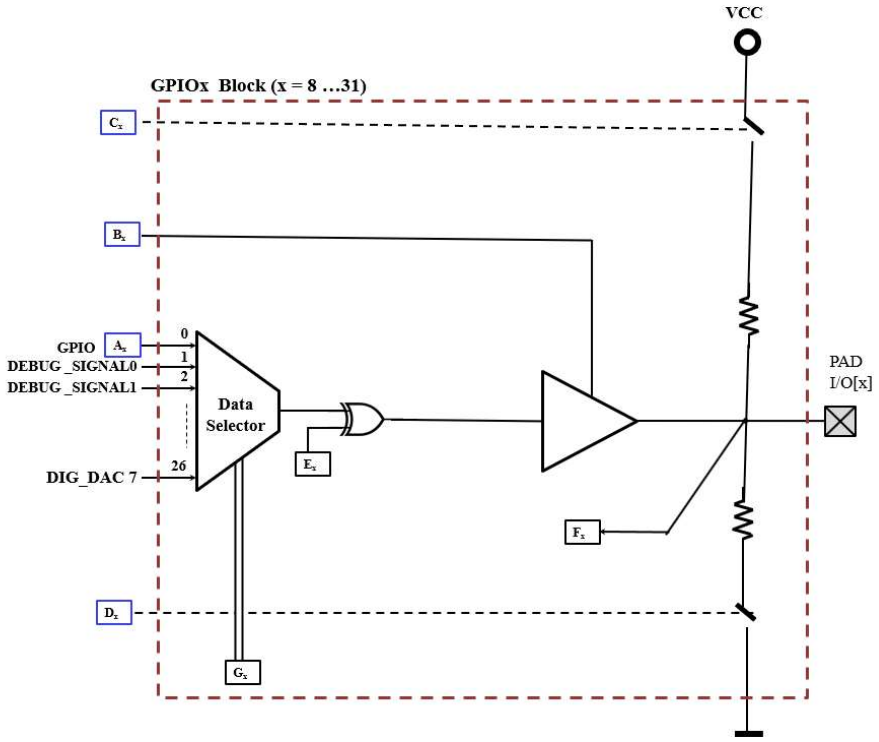

Figure 24 - The block diagram of GPIOx Block (x = 8 … 31)

Every GPIO block has an Alternate Function selector. The selector has 27 inputs. The signals from different hardware blocks are connected to the selectors' inputs. The following table shows the alternate function names.

Selector Number	Alternative Function Name	Selector Number	Alternative Function Name	Selector Number	Alternative Function Name
0	GPIO	9	UART TXD	18	UART HDLC RX1
1	DEBUG_SIGNAL0	10	UART RXD	19	DIG DAC 0
2	DEBUG_SIGNAL1	11	SPI SEN	20	DIG DAC 1
3	DEBUG_SIGNAL2	12	SPI SCK	21	DIG DAC 2
4	DEBUG_SIGNAL3	13	SPI SDO	22	DIG DAC 3
5	DEBUG_SIFNAL4	14	SPI SDI	23	DIG DAC 4
6	DEBUG_SIGNAL5	15	I2C SDA	24	DIG DAC 5
7	DEBUG_SIGNAL6	16	I2C SCK	25	DIG DAC 6
8	DEBUG_SIGNAL7	17	UART HDLC TX1	26	DIG DAC 7

Capture (Timing Measurements) block

The Capture block measures the actual time interval between two processed events, GPIO signals, or any combination of those. The block has four channels: Measure Channel x (x = 0 to 3). Each of the channels contains three independent registers – current measurement and min/max values – which are continuously refreshed in the background. When a read request is issued for a particular channel, the three data registers from that channel will be transferred into the AMBA read

registers (Figure 25). The AMBA read registers are common for all four channels, hence only one channel can be read at the time. The time intervals (measurements) are expressed in clock ticks (10 ns).

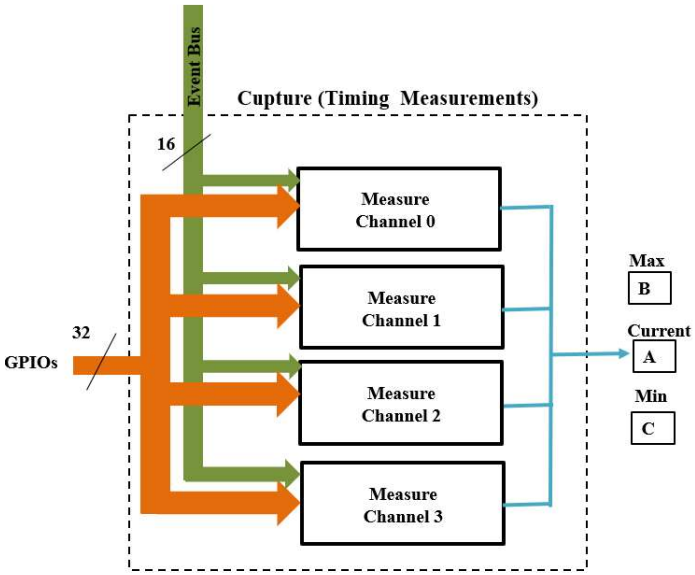

Figure 25 – Capture (Timing Measurements) block diagram

A channel of the Capture block is shown in Figure 26. The channel has two selectors for selecting a start and a stop signal for the measurement.

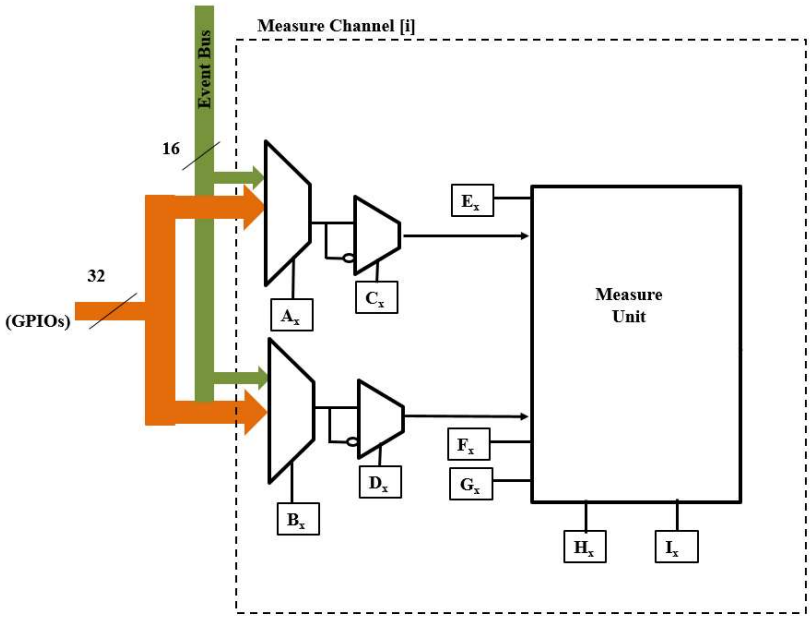

Figure 26 - A channel of the Capture (Timing Measurements) block

If the minimum and/or maximum functions are enabled, the minimum and/or maximum values of the currently measured data will be recorded in the dedicated min/max registers.
The maximum and minimum values can be reinitialized by disabling and then re-enabling the maximum and minimum functions respectively.

Compare (Timing Generator)

The Timing Generator can create four signals with the same period but different phases. The period of the pulses in ns, is given by the PERIOD register. A tick is equal to 10 ns. The start of every pulse depends of the value written in the TIMEx register (**x** is the output signal number from 0 to 3). The output signals are sent to Source Bus and SWE_DEBUG.

AC PLL

AC PLL block synchronizes the HSA8000 controller with the grid frequency. The basic functions of AC PLL block are:

- lock on the grid frequency (50/60Hz) based on zero-crossings;
- track the drift of the grid frequency up to 0.1 Hz/s;
- generate an integer multiple of the grid frequency used for processing in the CPU program.

AC PLL can be used for DC to AC inversion and ensures efficient grid current injection operation or in PFC applications to generate grid current reference. The grid voltages are scaled down by voltage dividers and then applied to the subtractors inputs (pins AN12_ACP and AN13_ACN).

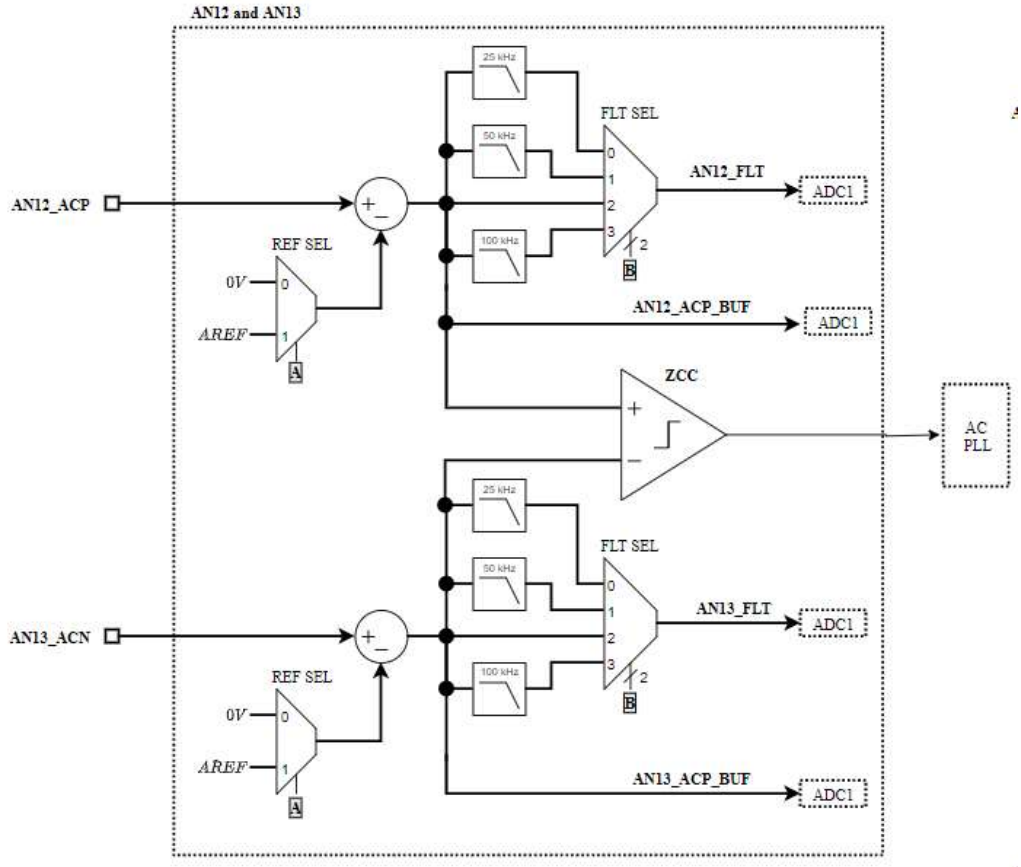

Figure 27 - Blocks associated with analog inputs AN12_ACP and AN13_ACN.

The subtractors outputs are inputs to the zero-crossing comparator, ZCC, as shown in Figure 27. The block diagram in Figure 28 illustrates the AC PLL main blocks:

- Glitch filter;
- Grid frequency measurements;
- Grid reconstruction;
- Period divider:
- Interrupt request (IRQ) generation;
- Configuration and report registers.

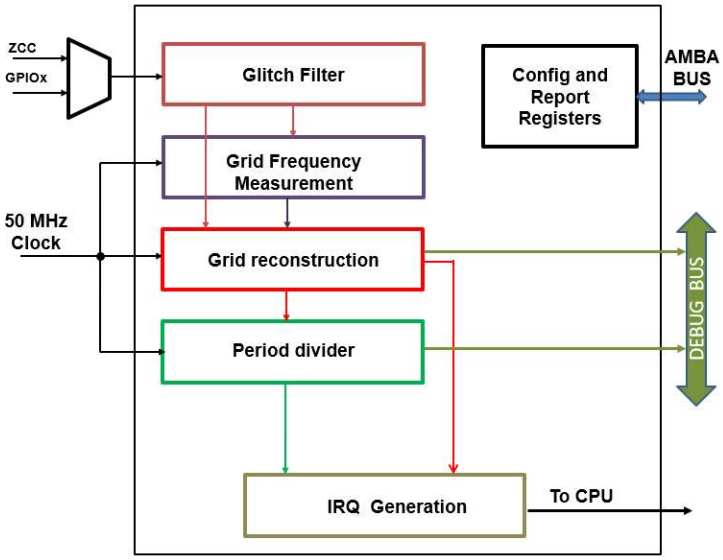

Figure 28 - AC PLL block diagram

To estimate the grid frequency, AC PLL should be set to acquisition mode. During acquisition mode AC PLL is not locked. The acquisition mode is performed during HSA8000 start. It also can be forced by the HSA8000 CPU.

Glitch filter

For the glitch filter input, ZCC output or any of the GPIOs can be selected. When ZCC is selected, the comparator should be enabled. The signal coming from ZCC is filtered by a glitch filter. The filter can be set to consecutive or integration mode.

Grid frequency measurements

AC PLL can be synchronized with the grid frequency in different programed limits. The period limits are provided for 50 Hz and 60 Hz through programmable registers. For each frequency, there are four limits.

Figure 29 shows the Grid frequency measurement block diagram. The block measures the grid period and locks the PLL if the frequency is in the range. For this purpose, the signal from the glitch filer is applied to the “Capture grid rising and falling edges” block. The outputs of the block are the negative and positive grid durations. From the two durations, period and asymmetry are calculated

in the “ADD/Subtract” block and their values are stored in the report registers. When the period is in the wide limits for time, t_{lock}, PLL enables smaller limits locking mode. The wide period limits and the lock time are set by the used in the config registers.
The period limits for 50 Hz and 60 Hz can be programed by a user.
When the “smaller limits locking mode” is enabled, the calculated period and asymmetry are filtered by 1st order low pass filter and then checked with the smaller range limits given in the config registers.

The output of the Grid frequency measurement block is the filtered frequency.

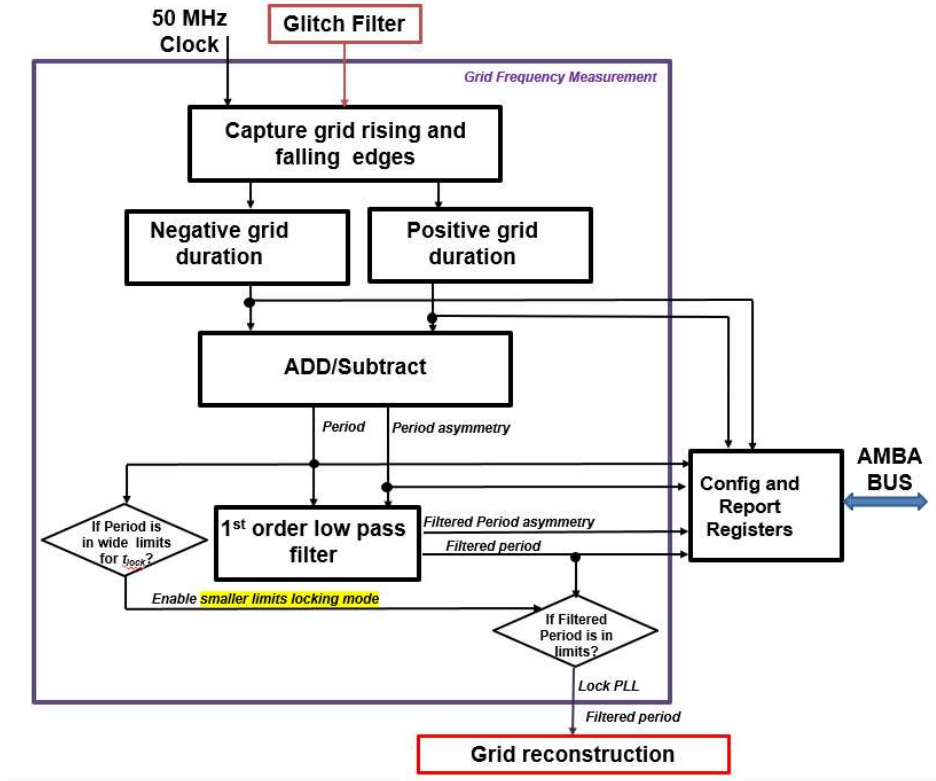

Figure 29 - Grid frequency measurement block diagram

Grid regenerations

The grid regeneration block diagram is shown in Figure 30.

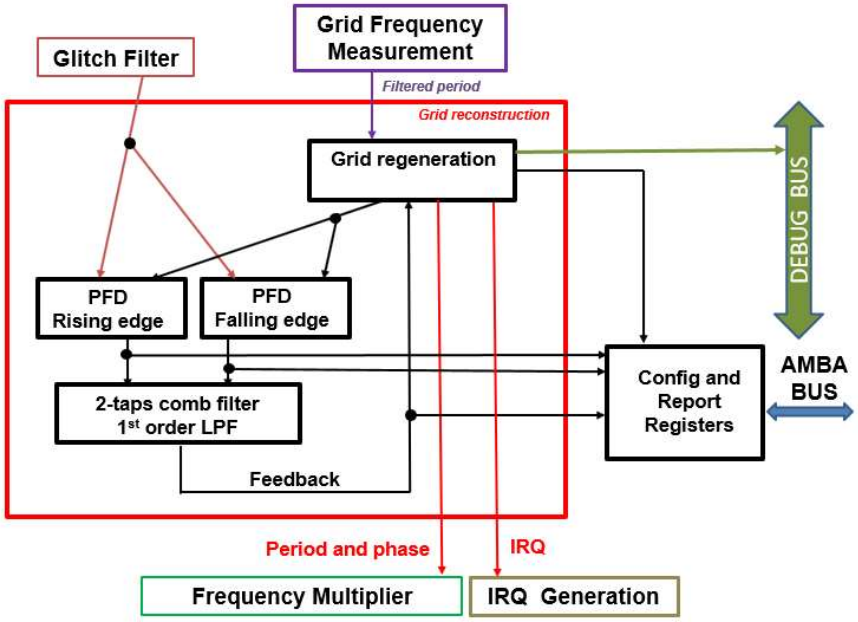

Figure 30 - Grid regeneration block diagram

The filtered period from the "Grid Frequency Measurement" block is sent to the "Grid regeneration" block for grid regeneration. The grid and the regenerated grid are dual phase/frequency detectors (PFD) for compensating any asymmetries due to imperfect rising edge/falling edge timing matching of ZCC and not exactly 0 threshold.

The detected phase errors by PFDs are filtered by a 2-taps comb filter to completely cancel the asymmetries due to the grid comp offset and due to the differences between rising and falling edges and a 1st order low-pass filter after the comb which guarantees stability.

Grid regeneration counter is:

- real-time frequency and phase programmability;
- when measured frequency and phase are steady for a certain period of time, the counter is preloaded with the best estimate of the frequency, initial phase, and the PFDs are enabled;
- it generates interrupts at the rising edge of the grid signal;
- the reconstructed grid can be observed through DEBUG BUS.

Once the PLL is locked:

- it tracks the frequency as long as the Grid frequency measurement block is in "smaller limits locking mode";
- short term grid glitches are reported to the software immediately, but are not fed to the filters nor PFDs, therefore the grid regenerator keeps running with the previous frequency and phase;
- software decides what short-term means, and forces a new acquisition or not.

Period divider

Period divider block diagram is shown in Figure 31. The regenerated grid period is used as reference for the period divider.

- the regenerated grid has 50% duty (or very close) of filtered period;
- the period division is recalculated at every half-grid cycle;
- rising edge of the divided signal is detected and sent to create an interrupt;
- the divider output can be seen through DEBUG BUS.

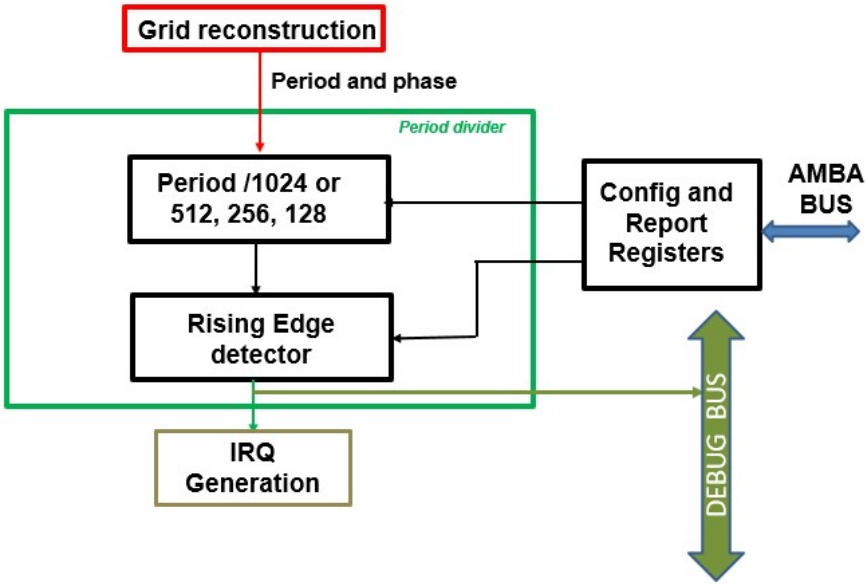

Figure 31 – Period divider block diagram

The period is divided by a number, N, set by SAMPLES_PER_GRID_PERIOD. The number can be 32, 64, 128, 256, 512, and 1024. The divider output is a sequence of N pulses (samples). By default, the samples per grid period is 256. Every sample sets IRQ to the CPU. During operation, the sample number can be seen by SAMPLE_NUMBER register.

For debugging, DEBUG_SAMPLE_NUMBER_OUTPUT register is used. The register sets the sample number that will be output on DEBUG.

Serial Peripheral Interface

The HSA8000 Serial Peripheral Interface (SPI) implements full-duplex, synchronous, serial communications, and has the following capabilities:

- master operation
- programmable word size (8 or 16-bits), bit ordering (MSB first / LSB first), clock polarity and phase, and bit rate

Figure 32 shows a block diagram of the HSA8000 SPI.

Figure 32 - SPI block diagram

BIT_WIDTH shows how many ns is each UART bit. The bit duration is reciprocal of the baudrate. Since the baudrate is 115200, the value of the BIT_WIDTH register in ns is:

$$\text{BIT\_WIDTH register} = \frac{1}{115200\frac{\text{bit}}{\text{s}}} = 8680\ ns$$

The value of the BIT_WIDTH register, depends on the CPU clock frequency, CPU_{Clk}, and the baudrate, BR. Since, the CPU clock frequency is 50 MHz and the BIT_WIDTH register in clock cycle is:

$$\text{BIT\_WIDTH register} = 8680 * 10^{-9}\ s * 50\ 000\ 000\ \frac{\text{clock}}{\text{s}} = = 434 \text{ clocks /bit} = 1B2(\text{HEX})$$

DEVICE_SELECT is a single bit used to control the GPIO14_SEN output (for bootloader portability), otherwise the user can use any available GPIO to generate SEN function.

UART interface

There are two UARTs in the HSA8000. UART0, connected by default as alternate function to pins GPIO8_TX0 and GPIO9_RX0, is a general purpose one. UART1 is connected by default as alternate function to pins GPIO10_HDLC_TX1 and GPIO11_HDLC_RX1. It can be used as a general purpose one, or it can be configured to work with the hardware HDLC interface.

The UARTs have the following capabilities:

- 7 or 8 data bits
- 1 or 2 stop bits
- parity bit (None / Even / Odd / Mark / Space)
- programmable bit rate

Figure 33 shows a block diagram of the HSA8000 UARTs.

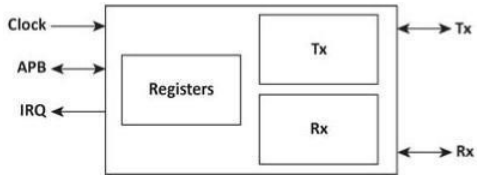

Figure 33 - UART interface block diagram

UART0

Data to be transmitted over the serial interface should be written to the lower 8 bits of the TX_DATA register. The register should not be written to while the TX_FULL bit in the STATUS register is set, otherwise data loss may occur.

Data that is received over the serial interface can be read in the lower 8 bits of the RX_DATA register.

The STATUS register contains a selection of flags that indicate the current status of the UART. To clear a bit in the register, write a 1 to it. Writing 0 will leave it unchanged.

BIT_WIDTH is a 16-bit register that specifies how many cycles of the CPU clock, each bit is transmitted for. Use of a 16-bit register provides support for a wide range of clock frequencies and baud rates.

UART1 HDLC

TX_CONTROL is the HDLC control field to be transmitted.
TX_ADDRESS shows the HDLC address field to be transmitted.
TX_DATA shows the HDLC data field to be transmitted. Writing to this field will trigger a transmission of an HDLC packet that consists the current contents of the control (TX_CONTROL), address (TX_ADDRESS) and data (TX_DATA) fields.
RX_CONTROL shows the HDLC control field of the last received HDLC packet.
RX_ADDRESS is the HDLC address of the last received HDLC packet.
RX_DATA - HDLC data field of the last received HDLC packet. Reading this field will send an acknowledgement to the RX framer thereby enabling the receipt of the next packet. Until this field is read, the RX framer will hold off all new incoming packets. Pending packets will be stored in the RX FIFO.
RX_CRC - HDLC cyclic redundancy check of the last received HDLC packet.
The fields of the CONTROL register control UART1 HDLC.

- acknowledging to the RX framer that data was received.

I²C serial interface

The HSA8000 has an I²C-compatible master interface which implements a subset of the I²C protocol with the following restrictions:

- Master only (no slave supported)
- 7-bit address space only
- No clock stretching
- Only 1 or 2 bytes of data (programmable through a register, default 2)

Read/write messages are sent in the following format:

- 1 byte device address
- 1 byte register address
- 1 or 2 bytes of data (programmable,).

Data to be transmitted over the I²C interface should be written to the WRITE register.

Watchdog

The watchdog's main purpose is to watch over the user program and make sure that it is executing properly. It will produce a signal which resets all digital circuitry including the CPU if it is not fed within the set timeout period. Feeding the watchdog is achieved by writing specific values to the "bone" registers. Typically, the user program writes one of the bones in the main routine and the other bone in the interrupt routine. If either routine is not running due to a stalled loop or the lack of CPU cycles, then a reset will be initiated which may allow the system to recover. Figure 34 shows HSA8000 watchdog system a block diagram.

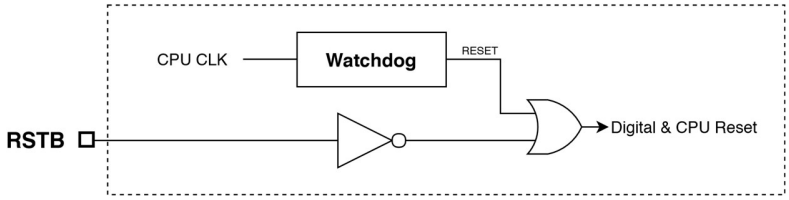

Figure 34- HSA8000 Watchdog system block diagram.

Math Accelerators Block

The Math Accelerator Block includes three blocks that perform mathematical function necessary for calculating. The blocks functionality is explained below:

MATH_SQRT

ARGUMENT register

The number to be square rooted is written to the ARGUMENT register. By writing to it, a BUSY flag is set, and the SQRT state machine is started automatically. The operation takes up to 16 clock ticks at the Peripheral clock rate 100 MHz to complete the calculation. The BUSY flag is cleared automatically after the calculation is done. The operation duration depends on the operand effective size:

- up to 16 bits for 8 clock ticks
- 17 to 24 bits for 12 clock ticks
- 25 to 32 bits for 16 clock ticks.

The machine automatically detects the effective size of the operand and decides the cycles number necessary to be completed.

MATH_SIN_COS

ARGUMENT register

The register represents the angle of the desired sin and cos, which is quantized on 10 bits. A value of “0” represents 0^o, a value of “512” represents 180^o, and the max value, “1023”, represents 359.6484375^o. A value of 1024 would represent 360^o, but in this implementation, it will wrap down to 0. Writing to the ARGUMENT register triggers the state machine and the result is available in the very next AMBA cycle so a busy flag is not necessary.

Timer

Two 32-bit timers (x = 0 and 1) are located inside the timer peripheral for the main purpose of generating interrupts. They are clocked at the same speed as the CPU which is typically 50 MHz. Each counter goes from 0 up to COMP3_PERIOD - 1 before repeating. Three other COMP registers provide counts for comparison, creating additional interrupt signals which can be individually enabled and masked giving significant flexibility in dividing up the count sequence in time.

Digital DAC

HSA8000 has 8 digital DACs. The digital DACs can function in either PWM or Sigma-Delta modes. A register controls the DACs clock frequency. When the DAC is set in PWM mode, a register can set the duty cycle. The maximum value is when the register is set to 0x400 which responds to

100%. Setting the register to 0x200, the duty cycle is 50% and setting to 0x00 the duty cycle is 0%. Every digital DAC can be exported on a GPIOx pin (x = 0-31) by setting its ALT_FUNCx.

DEBUG

HSA8000 is designed to export signals in real-time. Figure 35 shows three selectors used for signals debugging.

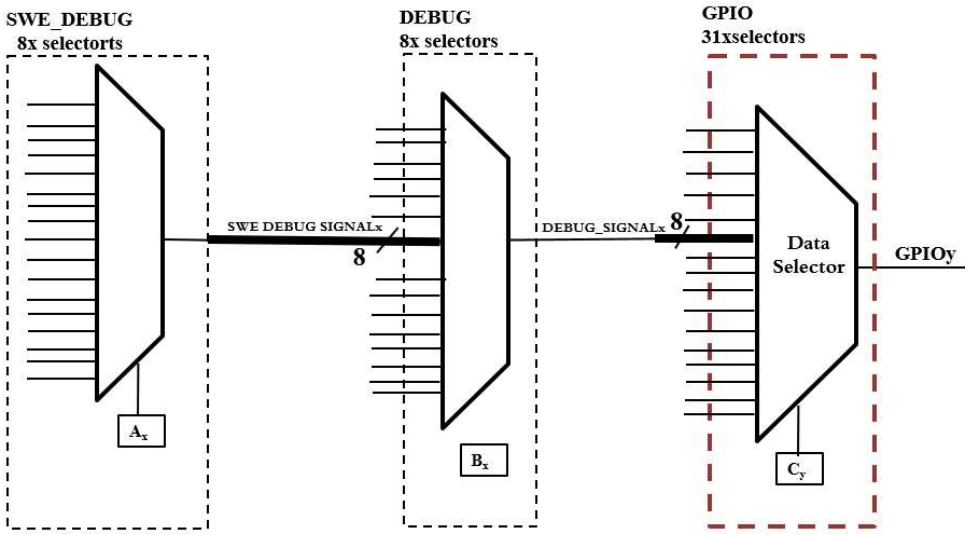

Figure 35 HSA8000 Debug block

The SWE_DEBUG block has eight selectors A_x that select signals from the Switching Engine Block.

0 = CMP0_POS_COMP
1 = CMP0_NEG_COMP
2 = CMP1_POS_COMP
3 = CMP1_NEG_COMP
4 = CMP2_POS_COMP
5 = CMP2_NEG_COMP
6 = CMP3_POS_COMP
7 = CMP3_NEG_COMP
16 = CMP4_POS_COMP
17 = CMP4_NEG_COMP
18 = CMP5_POS_COMP
19 = CMP5_NEG_COMP
20 =
21 =
22 = CMP7_POS_COMP
23 = CMP7_NEG_COMP
24 = ADC0_AN0_COMP_STATUS
25 = ADC0_AN1_COMP_STATUS
26 = ADC0_AN2_DCOMP_STATUS
27 = ADC0_AN3_COMP_STATUS
28 = ADC1_AN12_FLT_COMP_STATUS
29 = ADC1_AN13_FLT_COMP_STATUS
30 = ADC1_AN12_BUF_COMP_STATUS
31 = ADC1_AN13_BUF_COMP_STATUS
32 = ADC2_CHANNEL0_COMP_STATUS
33 = ADC2_CHANNEL1_COMP_STATUS
34 = ADC2_CHANNEL2_COMP_STATUS

35 = ADC2_CHANNEL3_COMP_STATUS
36 = ADC2_CHANNEL4_COMP_STATUS
37 = ADC2_CHANNEL5_COMP_STATUS
38 = ADC2_CHANNEL6_COMP_STATUS
39 = ADC2_CHANNEL7_COMP_STATUS
40 = COMPARE_TIME0_EQUAL
41 = COMPARE_TIME1_EQUAL
42 = COMPARE_TIME2_EQUAL
43 = COMPARE_TIME3_EQUAL
44 = SWE_EDT_INTERLEAVE_PHASE1
45 = SWE_EDT_INTERLEAVE_PHASE2
46 = SWE_EDT_INTERLEAVE_PHASE3
64 = SWE_DRIVER0_LS
65 = SWE_DRIVER0_HS
66 = SWE_DRIVER1_LS
67 = SWE_DRIVER1_HS
68 = SWE_DRIVER2_LS
69 = SWE_DRIVER2_HS
70 = SWE_DRIVER3_LS
71 = SWE_DRIVER3_HS
72 = SWE_EDT0_HS
73 = SWE_EDT0_LS
74 = SWE_EDT1_HS
75 = SWE_EDT1_LS
76 = SWE_EDT2_HS
77 = SWE_EDT2_LS
78 = SWE_EDT3_HS
79 = SWE_EDT3_LS
80 = SWE_PWM0_LS
81 = SWE_PWM0_HS
82 = SWE_PWM1_LS
83 = SWE_PWM1_HS
84 = SWE_PWM2_LS
85 = SWE_PWM2_HS
86 = SWE_PWM3_LS
87 = SWE_PWM3_HS
88 = SWE_PWM0_REFERENCE
89 = SWE_PWM1_REFERENCE
90 = SWE_PWM2_REFERENCE
91 = SWE_PWM0_REFERENCE
92 = SWE_FAULT0
93 = SWE_FAULT1
94 = SWE_FAULT2
95 = SWE_FAULT3
96 = SWE_FAULT4
97 = SWE_FAULT5
98 = SWE_FAULT6
99 = SWE_FAULT7
104 = SWE_FAULT0_SELECTED_SOURCE_RAW
105 = SWE_FAULT1_SELECTED_SOURCE_RAW
106 = SWE_FAULT2_SELECTED_SOURCE_RAW
107 = SWE_FAULT3_SELECTED_SOURCE_RAW
108 = SWE_FAULT4_SELECTED_SOURCE_RAW
109 = SWE_FAULT5_SELECTED_SOURCE_RAW
110 = SWE_FAULT6_SELECTED_SOURCE_RAW
111 = SWE_FAULT7_SELECTED_SOURCE_RAW
112 = SWE_FAULT0_SELECTED_SOURCE_FLT
113 = SWE_FAULT1_SELECTED_SOURCE_FLT
114 = SWE_FAULT2_SELECTED_SOURCE_FLT
115 = SWE_FAULT3_SELECTED_SOURCE_FLT
116 = SWE_FAULT4_SELECTED_SOURCE_FLT
117 = SWE_FAULT5_SELECTED_SOURCE_FLT
118 = SWE_FAULT6_SELECTED_SOURCE_FLT
119 = SWE_FAULT7_SELECTED_SOURCE_FLT
128 = SWE_WINDOW0

129 = SWE_WINDOW1
130 = SWE_WINDOW2
131 = SWE_WINDOW3
132 = SWE_WINDOW4
133 = SWE_WINDOW5
134 = SWE_WINDOW6
135 = SWE_WINDOW7
136 = SWE_WINDOW8
137 = SWE_WINDOW9
138 = SWE_WINDOW10
139 = SWE_WINDOW11
140 = SWE_WINDOW12
141 = SWE_WINDOW13
142 = SWE_WINDOW14
143 = SWE_WINDOW15
144 = SWE_EVENT0
145 = SWE_EVENT1
146 = SWE_EVENT2
147 = SWE_EVENT3
148 = SWE_EVENT4
149 = SWE_EVENT5
150 = SWE_EVENT6
151 = SWE_EVENT7
152 = SWE_EVENT8
153 = SWE_EVENT9
154 = SWE_EVENT10
155 = SWE_EVENT11
156 = SWE_EVENT12
157 = SWE_EVENT13
158 = SWE_EVENT14
159 = SWE_EVENT15
160 = SWE_EVENT0_INPUT
161 = SWE_EVENT1_INPUT
162 = SWE_EVENT2_INPUT
163 = SWE_EVENT3_INPUT
164 = SWE_EVENT4_INPUT
165 = SWE_EVENT5_INPUT
166 = SWE_EVENT6_INPUT
167 = SWE_EVENT7_INPUT
168 = SWE_EVENT8_INPUT
169 = SWE_EVENT9_INPUT
170 = SWE_EVENT10_INPUT
171 = SWE_EVENT11_INPUT
172 = SWE_EVENT12_INPUT
173 = SWE_EVENT13_INPUT
174 = SWE_EVENT14_INPUT
175 = SWE_EVENT15_INPUT
176 = SWE_EVENT0_FILTERED_INPUT
177 = SWE_EVENT1_FILTERED_INPUT
178 = SWE_EVENT2_FILTERED_INPUT
179 = SWE_EVENT3_FILTERED_INPUT
180 = SWE_EVENT4_FILTERED_INPUT
181 = SWE_EVENT5_FILTERED_INPUT
182 = SWE_EVENT6_FILTERED_INPUT
183 = SWE_EVENT7_FILTERED_INPUT
184 = SWE_EVENT8_FILTERED_INPUT
185 = SWE_EVENT9_FILTERED_INPUT

The DEBUG selector can select the following signals:

0 = LOW
1 = HIGH
2 = 50 MHz
3 = RX0
4 = 100 MHz
5 = ADC0_CLK

6 = ADC1_CLK
7 = ADC2_CLK
8 = ADC0_BUSY
9 = ADC1_BUSY
10 = ADC2_BUSY
11 = ADC2_SUMPLE_HOLD
12 = TIMER_COUNTER0_0_1_EQUAL
13 = TIMER_COUNTER0_2_3_EQUAL
14 = TIMER_COUNTER1_0_1_EQUAL
15 = TIMER_COUNTER1_2_3_EQUAL
16 = SWE_DBG_DEBUG_SIGNAL0
17 = SWE_DBG_DEBUG_SIGNAL1
18 = SWE_DBG_DEBUG_SIGNAL2
19 = SWE_DBG_DEBUG_SIGNAL3
20 = SWE_DBG_DEBUG_SIGNAL4
21 = SWE_DBG_DEBUG_SIGNAL5
22 = SWE_DBG_DEBUG_SIGNAL6
23 = SWE_DBG_DEBUG_SIGNAL7
25 = RX1
26 = TX1
42 = AC_PLL_sync_grid_comp
43 = AC_PLL_comp_filtered
44 = AC_PLL_out_of_range
45 = AC_PLL_compare_0_pulse
46 = AC_PLL_compare_1_pulse
47 = AC_PLL_phdet_re_up_pulse
48 = AC_PLL_phdet_re_dn_pulse
49 = AC_PLL_phdet_fe_up_pulse
50 = AC_PLL_phdet_fe_dn_pulse
51 = AC_PLL_reconstruct_not_locked
52 = AC_PLL_reconstruct_positive
53 = AC_PLL_reconstruct_be
54 = AC_PLL_fmul_pulse
55 = AC_PLL_sample_pulse
56 = AC_PLL_sample_N

Applications, Implementation and Layout

Application

HSA8000 is a highly integrated mixed-signal IC with the industry's most complete set of analog and digital power peripherals for high-performance power system designs. It can be used for the following applications:

- AC/DC - Power-Factor-Corrected- Bridgeless & Interleaved
- DC/DC - LLC, Half-Bridge, Phase-Shifted Full-Bridge, etc.
- Charging – On-board (EV), Charge Stations, Off-grid
- Inverters – Bi-Directional, Automotive, UPS and Storage
- Heavy Industrial – Electrified Equipment, HVAC, Welding
- Single-panel and dual-panel micro-inverters
- Battery and fuel cell inverters (uni-directional and bi-directional)
- VAR compensator
- Interleaved multi-phase battery charger for high-power applications

Implementations

Solantro has developed different HSA8000 based platforms. Some of them are listed below.

PFC Architecture based on HSA8000

Figure 36 shows a simplified HSA8000-based totem-pole PFC circuit. PFC is achieved by sensing the inductor current, the grid voltage, the PFC output voltage, and the boost switching node "*B*" voltage.

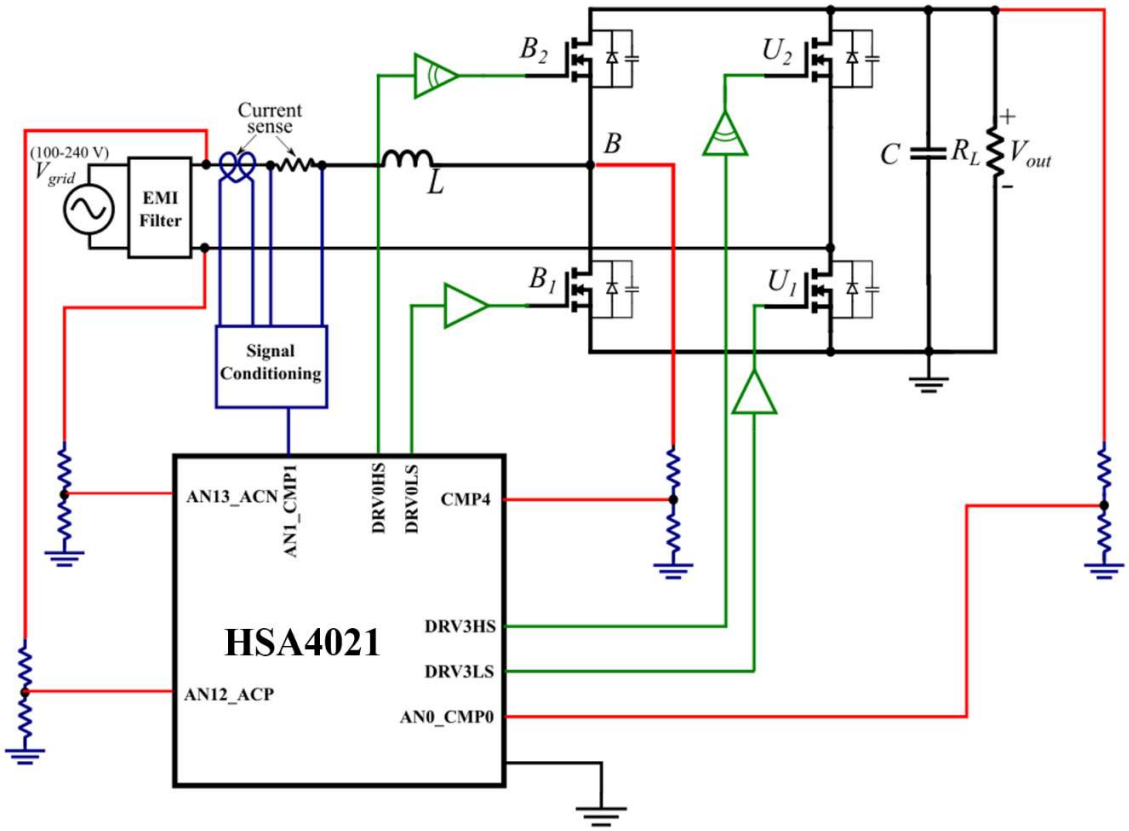

Figure 36 - Simplified HSA8000-based totem-pole PFC circuit. (Power train in black, driver circuits in green, current sensing circuits in blue, and voltage measurement circuits in red.)

Figure 37 shows the internal components of the HSA8000 that are essential for implementing control of Huada's totem-Pole PFC.

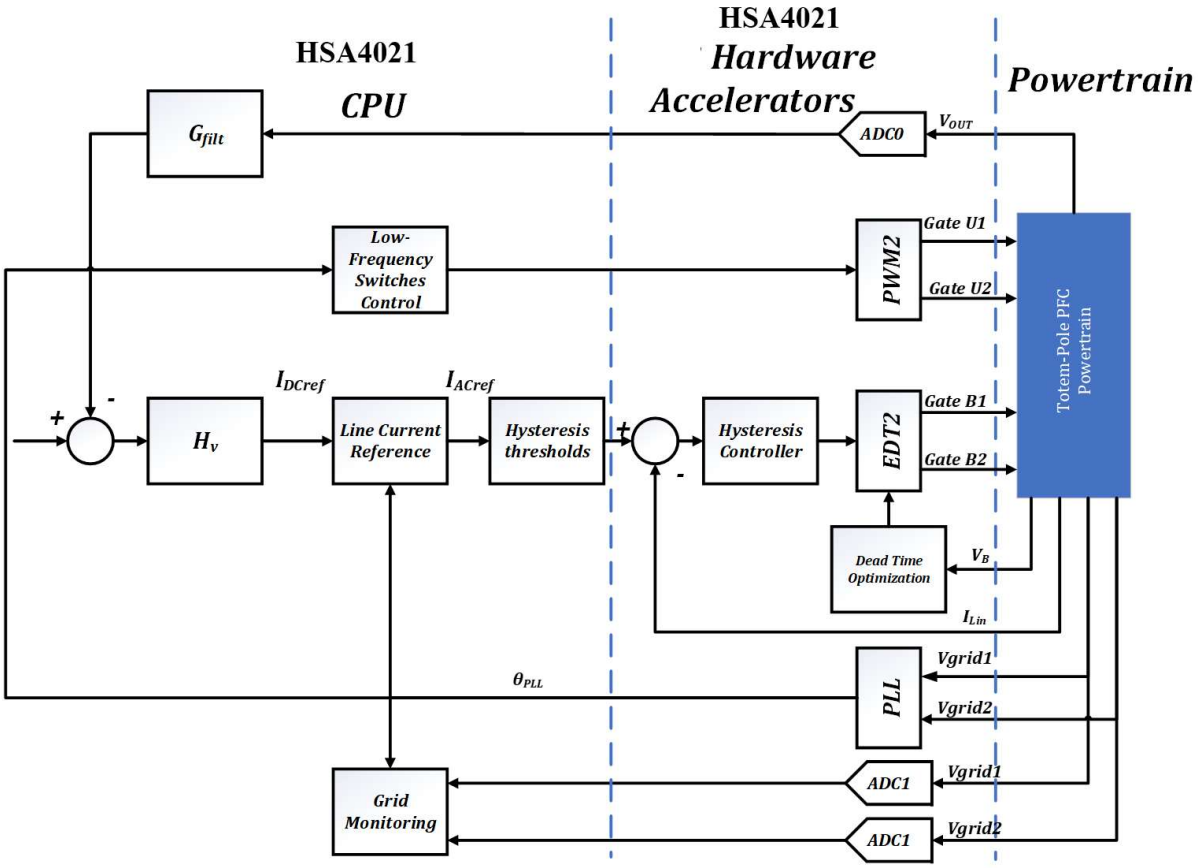

Figure 37 - Simplified HSA8000-based totem-pole PFC control circuit

The HSA8000 also provides four logical signals to control the drivers for the four transistors used in the totem-pole architecture. The grid monitoring block is a grid-tied, high-performance digital PLL. Accurate grid voltage monitoring is necessary for grid compliance. Instantiating this function as a separate hardware peripheral achieves grid compliance with a minimum of software/processor burden. This leaves many more CPU resources to implement housekeeping, telemetry, and communications in comparison with other controller solutions.

Depending upon the application, a designer must select one of two methods for deriving the waveform. The HSA8000's rich hardware peripheral set supports either method. In the first method, the input grid current follows an ideal sinusoidal waveform independent of the grid voltage waveform. In the second method, the input current waveform is directly tracked to the grid voltage waveform. Both methods can be implemented through firmware utilizing the integrated digital PLL and supporting hardware.

The HSA8000 controls the operation of two synchronous boost converters. Boost 1 during the positive half of the grid cycle and Boost 2 during the negative half of the grid cycle. It does this through hysteresis control using two event driven timers. To maintain high efficiency and low electromagnetic interference (EMI), the boosts should work either in Transition Mode (TM) or Continuous Conduction Modes (CCM). The proper mode depends on the instantaneous input current value. At low instantaneous currents, TM switching is desirable to ensure soft switching. At high instantaneous currents, CCM and hard switching are desirable. That keeps the ripple within reasonable limits that minimize conduction losses. Overall, the switching frequency should be kept within limits that minimize the switching losses while still being outside of the bandwidth of the output low-pass filter. The constraints mentioned above can be easily addressed by the hardware peripherals implemented in the HSA8000 controller.
To perform hysteresis control, the input inductor current (grid current) creates events for turning *OFF* switches B_1 and B_2 while the switching node *B* voltage creates events for turning *ON* switches B_1 and B_2. Figure 38shows HSA8000 Switching Engine configuration for Huada Totem-Pole PFC.

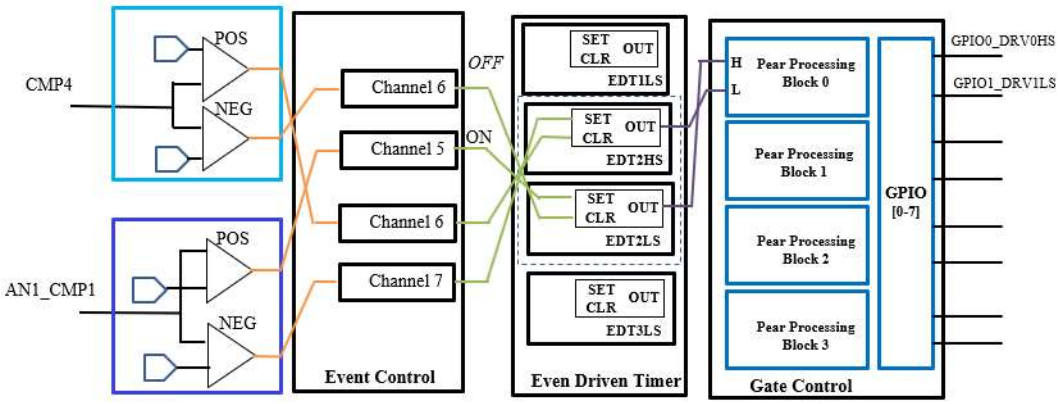

Figure 38 HSA8000 switching engine configuration for Huada totem-pole PFC.

The timing diagram of the boost converter, Boost 1, is shown in Figure 39. The diagram includes the waveforms of the sensing signals, the inductor current i_{Lin}, the switching node *B* voltage V_B, events 4 to 7, their corresponding windows, and the drivers' signals DRV0HS and DRV0LS.

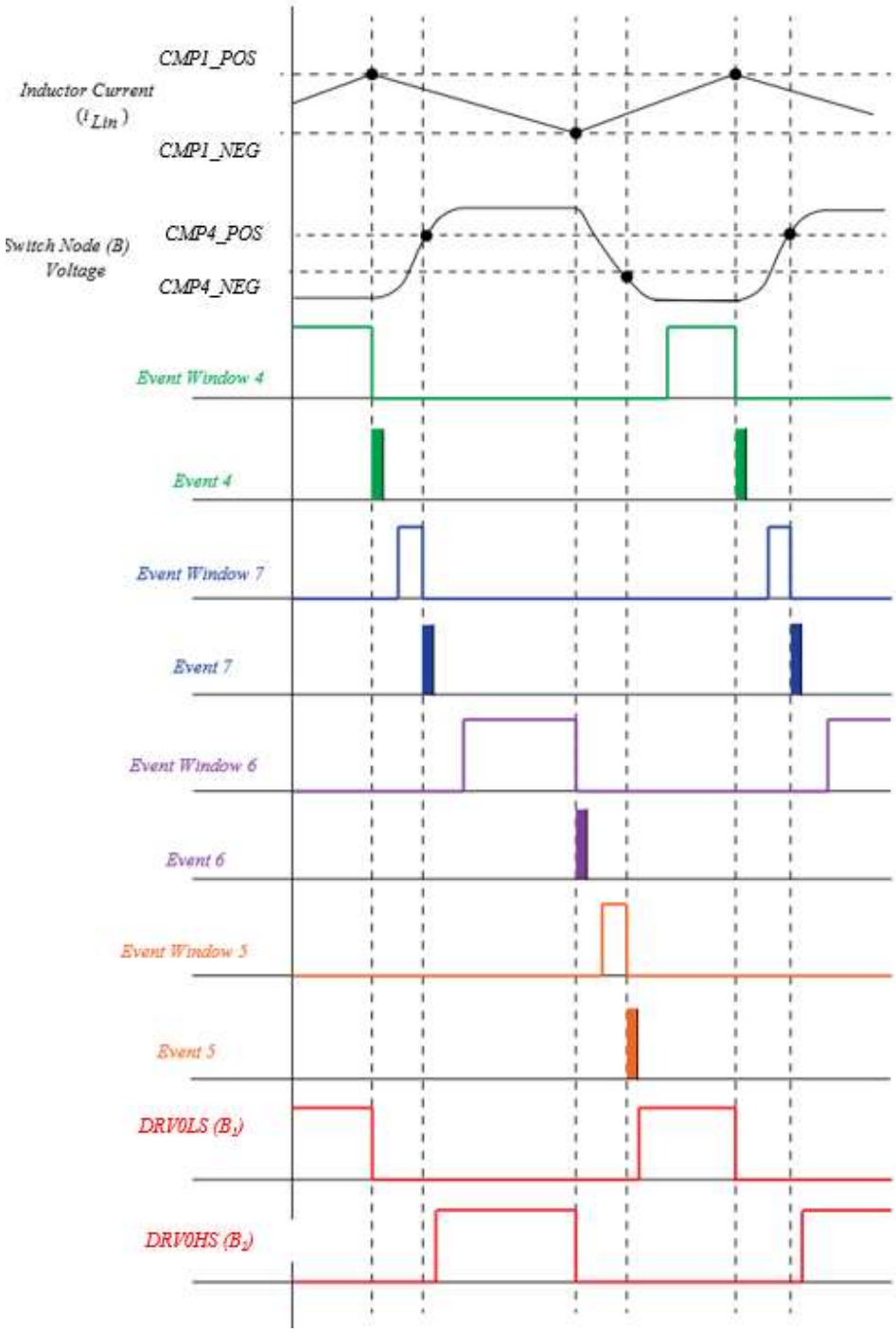

Figure 39 -Timing diagram of the Huada totem-pole PFC Boost 1.

For more information on 1 kW Huada's PFC platform see the following document.

DPD1153-1_HSA8000_Totem-Pole_Bridgeless_PFC_Platform_AppNote.

LLC Architecture based on HSA8000

Figure 40 shows a simplified HSA8000-based LLC converter circuit. The LLC dynamic response is controlled by the charge control method. By controlling the input charge cycle-by-cycle, the resonant power stage can be reduced to a 1st order system. The input charge is determined by sensing the series resonant capacitor voltage at the switch's turn off time.

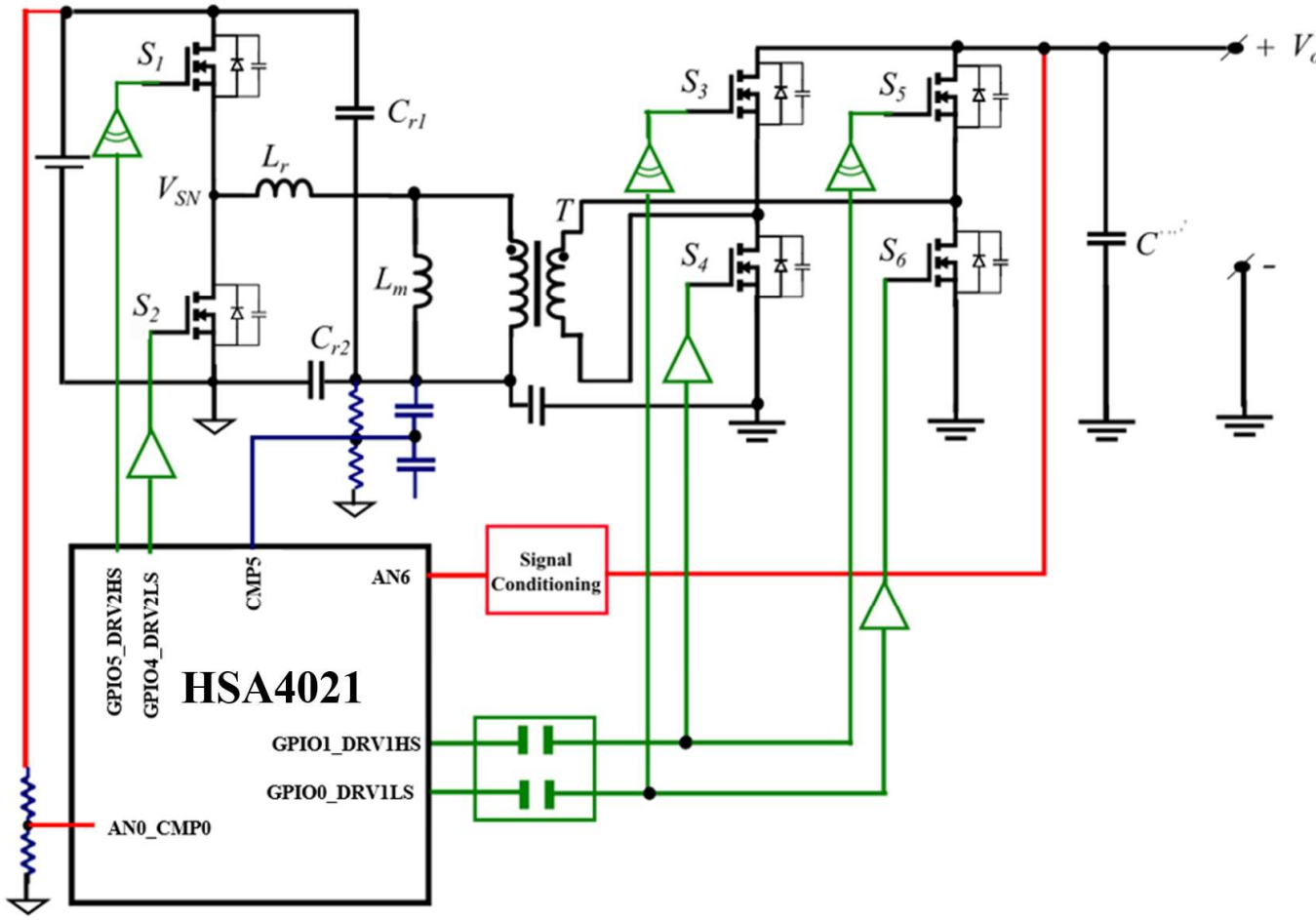

Figure 40 - Simplified HSA8000-based LLC converter circuit.

As shown in Figure 40, the voltage of the resonant capacitor is scaled down and used for the charge control. The input and output voltage are also measured by the HSA8000. The HSA8000 also provides logical signals to control the drivers for the LLC controller. Figure 41 shows a simplified HSA8000-based LLC converter control circuit.

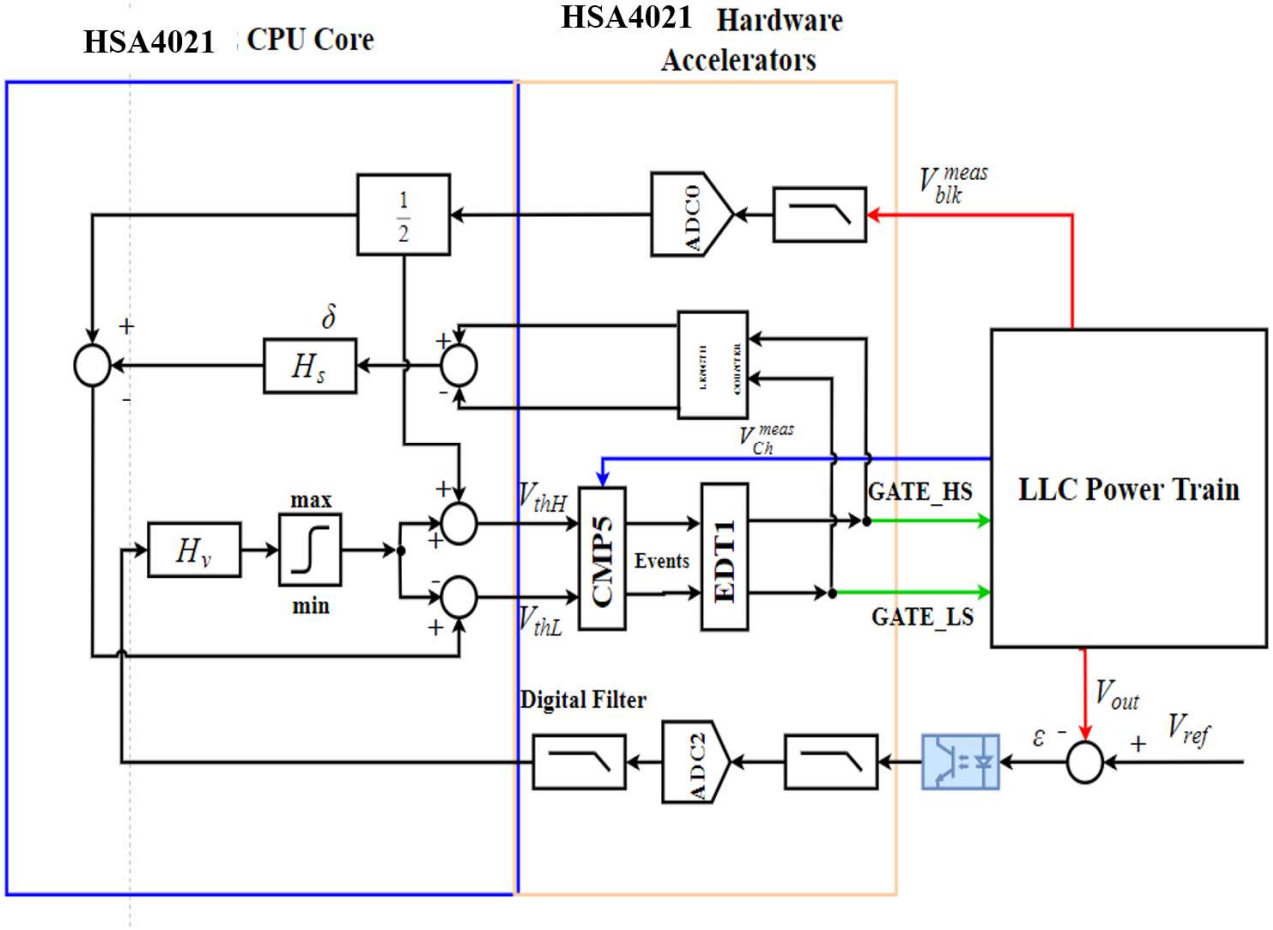

Figure 41 - Simplified HSA8000-based LLC controller control circuit.

For more information on 1 kW Huada's LLC platform see the following document.

DPD1154-1_HSA8000_ LLC_SeriesResonantConverter_AppNote.

MI-P700A PV inverter HSA8000 based

The MI-P700A is a PV inverter supplied from two PV panels. The inverter's five stages include; two parallel synchronous buck converters; a solid-state step-up transformer (SST) consisting of a series resonant tank; an unfolding bridge; a reactive shunt, and an output filter and surge protection block. A block diagram of the MI-P700 4Q is shown in Figure 42. The buck converter stages maintain the PV panels at their respective maximum power points and act as current sources to supply the SST.

The SST stage has a transformer with a turns ratio of 1:15. The output voltage of the transformer is set by the grid and therefore the input voltage (output of the bucks) is 1/15th of the grid voltage. The transformer galvanically isolates the PV panels from the grid. Since the output of the bucks and therefore the output of the SST is a rectified sine wave, an unfolding bridge is used to create a proper sinusoidal voltage. The output of the

unfolding bridge is connected to the reactive shunt. The reactive shunt can adjust the power factor of the inverter output from 0.85 leading to 0.85 lagging. The output filter and surge protection stage provide high frequency filtering of the switching noise and protects against excessive transient voltages. It also limits electromagnetic interference and radio frequency interference.

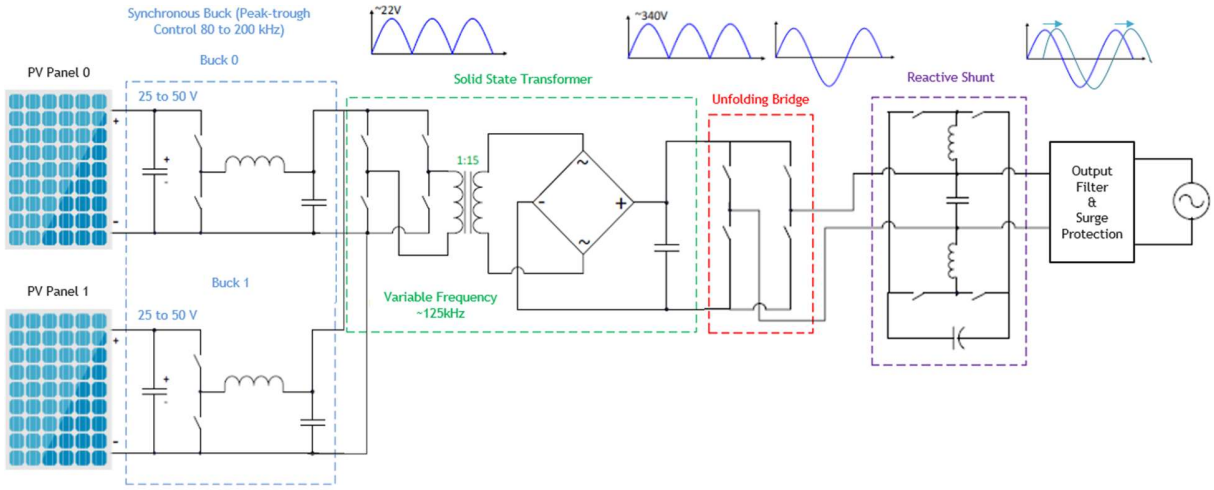

Figure 42- MI-P700 4Q high level view of stages

The entire control of the PV inverter MI-P700A is done by HSA8000. For more information on platform operation see the following document: DPD1140 MI-P700_4Q_Development_Platform_Application_Note

1000 W AC battery inverter

The AC battery inverter (ACBI) is a 1000VA, four-quadrant, bi-directional battery inverter controlled by Solantro's microcontroller HSA8000. It can charge a battery from the grid or supply power from the battery to the grid. The ACBI is designed with a "blade" form factor, suitable for mounting in an industry standard 19-inch electrical rack. Figure 43 shows a simplified block diagram of ACBI Development Platform. The ACBI consists of five blocks: DC filter, Solid state transformer (SST), Buck-boost DC- DC converter, H-bridge inverter, and AC Filter and Surge Protection. The ACBI is a bi-directional inverter operating at two modes: charging and discharging the battery. The energy flow and the functions of the blocks depend on the operation mode.

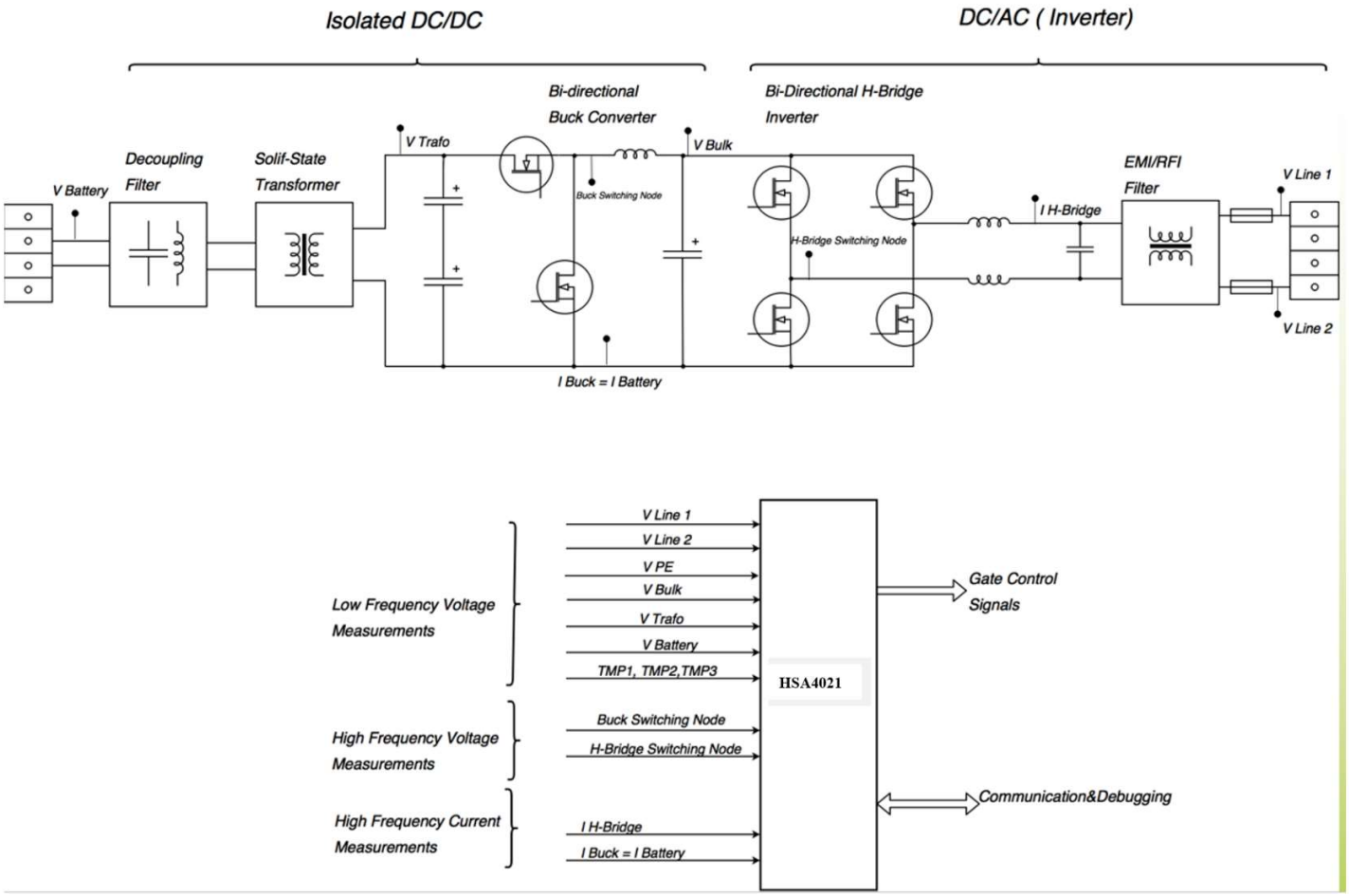

Figure 43 - ACBI high level view block diagram

HSA8000 schematic checklist

The HSA8000 is intended to work in a noisy environment, therefore, to ensure reliable operation it must be protected from the noise. The designer should follow the best practice for electromagnetic compatibility and the recommendations listed in this section must be followed. Specifically, decoupling capacitors should be placed very close to the power pins, the RSTB pin should be connected to pull-up 10 kΩ resistor. It is also relevant to eliminate or attenuate noise in order to avoid that the noise reaches supply, I/O and crystal pins.

Power supply connections

The HSA8000 supports a single power supply from 3 to 3.6 V. It has four power pins: a VDDA (pin 5) and two VDD pins (pins 49, 61 and 68). Figure 44 shows power supply connections. Two capacitors are recommended: 100 nF and 4.7 µF. The decoupling capacitor 100 nF should be placed close to the device for each supply pin pair. Also, for better decoupling low ESR capacitors

should be used.

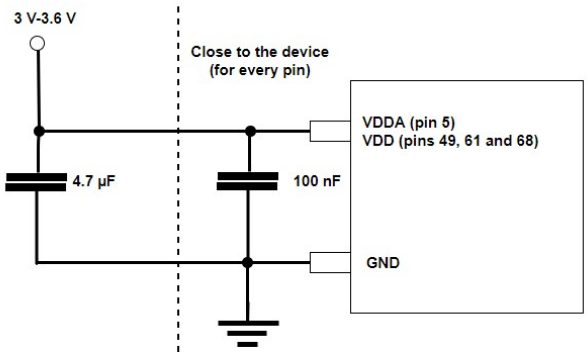

Figure 44 - Power supply connections

Regulators connections

The HSA8000 has two 1.8 V regulators, 1V8A (pin 2) and 1V8D (pin 21) and one 3 V regulator 3V0A (pins 7 and 23). Figure 45 shows regulators connections. Two capacitors are recommended:100 nF and 4.7 µF. The decoupling capacitor 100 nF should be placed close to the device for each regulator pin pair. Also, for better decoupling, low ESR capacitors should be used.

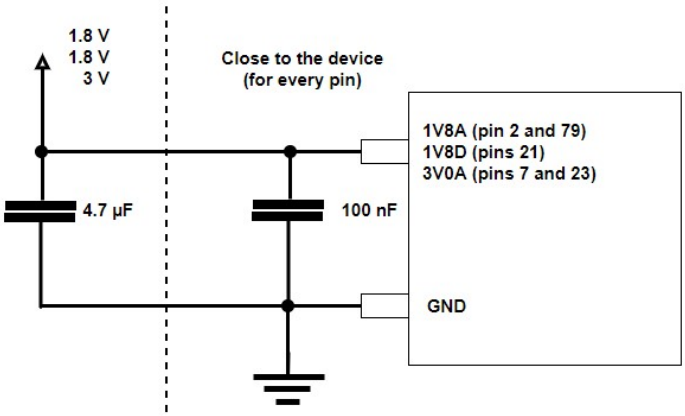

Figure 45 - Regulators connections

0.9 V mid-rail voltage reference connections

The HSA8000 has a 0.9 V mid-rail voltage reference that should also be decoupled as shown in Figure 46. A capacitor of 10 nF placed close to the device should be used for decoupling. For better decoupling low ESR capacitors should be used.

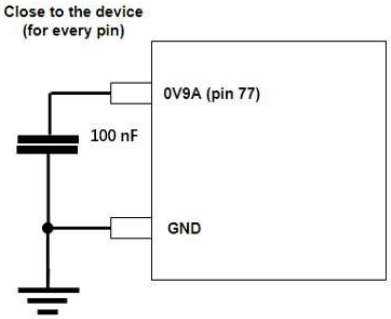

Figure 46 - 0.9 V mid-rail voltage reference connections.

Reset circuit connections

An external reset circuit consisting of a resistor and capacitor is connected to the RSTB (pin 70) see Figure 47. The 10 kΩ pull-up resistor makes sure that the reset does not go low unintended causing a device reset. The 100 nF (0402 package) capacitor filters the noise coming to the pins.

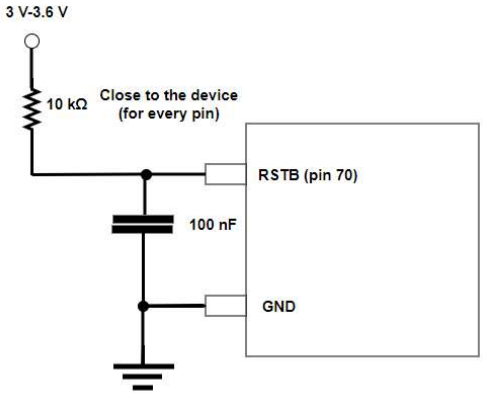

Figure 47 - Reset circuit connections

Crystal oscillator connections

An external crystal oscillator applies an input signal of 25 MHz to XI and XO (pins 7 2 and 71) shown in Figure 48. The crystal should be located as close to the device as possible. Long signal lines may cause too high load to operate the crystal, and cause crosstalk to other parts of the system. To prevent noise coming to the XI and XO pins, two 15 pF decoupling capacitor should be placed close to the device.

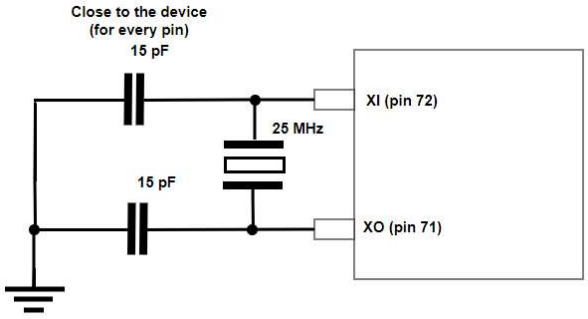

Figure 48 - Crystal oscillator connections

Unused pins

All unused analog pins should be grounded.

Layout Example

Figure 49 shows a layout example with HSA8000.

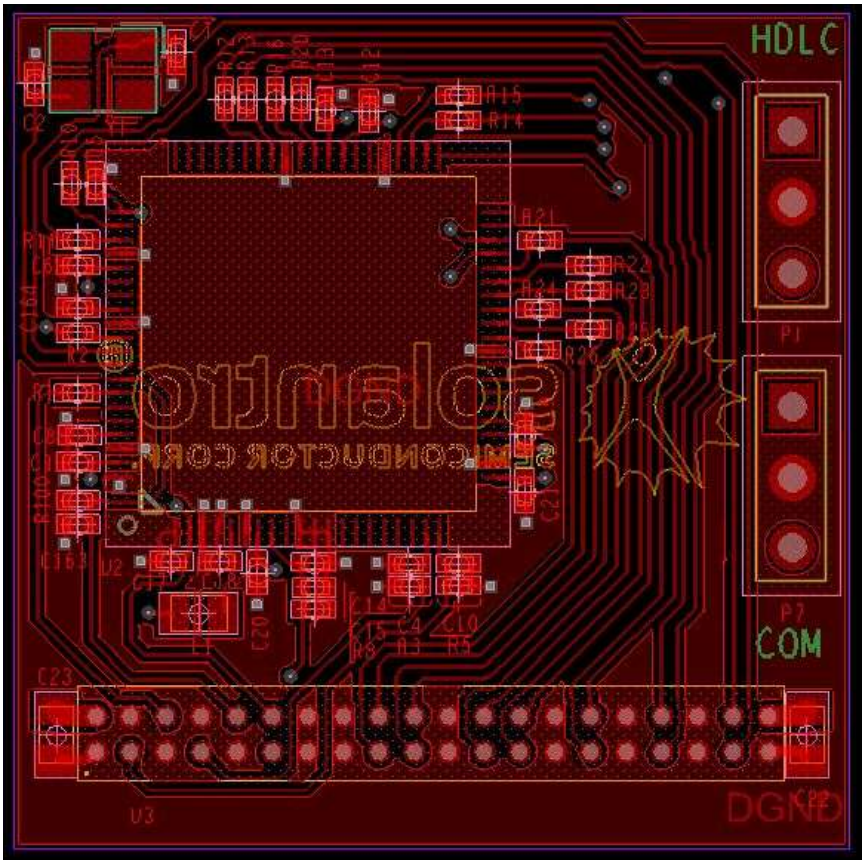

Figure 49 - Layout example.

HSA8000 Packaging

The HSA8000 is packaged in an QFN48L package, as seen in Figure 50.

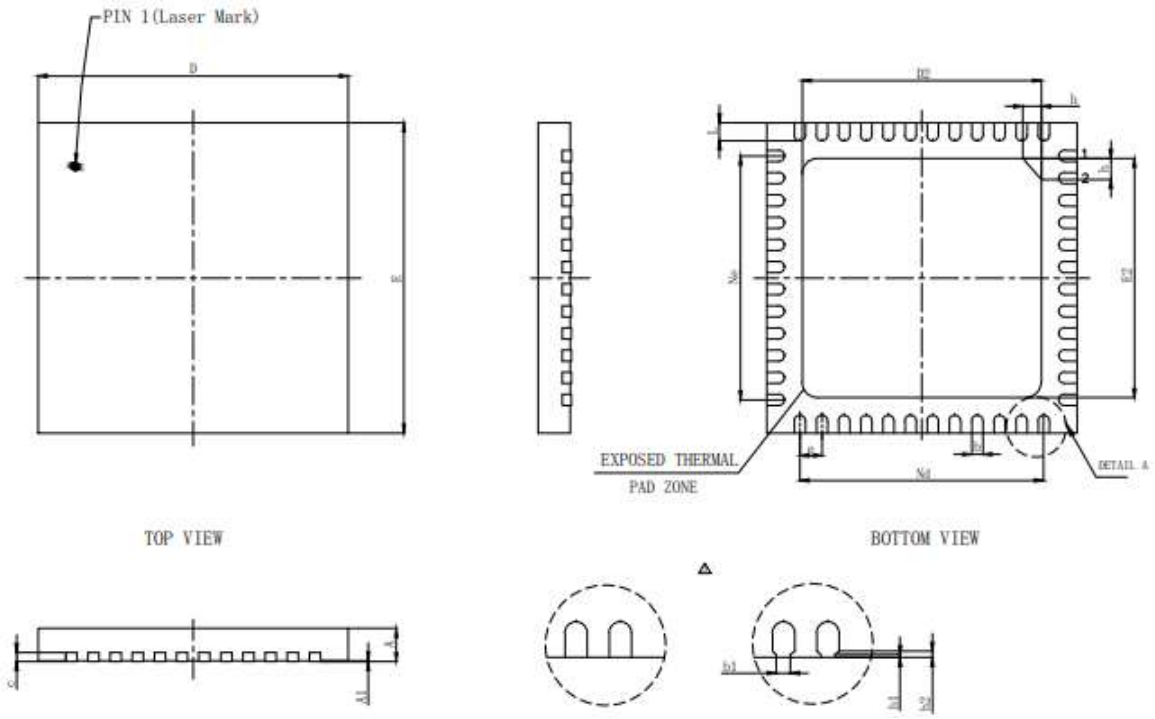

Figure 50 – HSA4021 IXC2 QFN48L Package

Table 3 lists the packaging dimensions for the HSA8000.

Table 3 - HSA8000 LQFP Package Dimensions

Symbol	Millimeters		
	Min.	Nom.	Max.
A	0.85	0.90	0.95
A1	0	0.02	0.05
b	0.18	0.25	0.30
B1	0.11	0.16	0.21
c	0.18	0.20	0.23
D	6.90	7.00	7.10
D2	5.30	5.40	5.50
e	0.50BSC		
Ne	5.50BSC		
Nd	5.50BSC		
E	6.90	7.00	7.10
E2	5.30	5.40	5.50
L	0.35	0.40	0.45
h	0.30	0.35	0.40
h1	0.03REF		
h2	0.10REF		

L/F	Symbol	Millimeters
232*232	D2	5.60 ±0.10
	E2	5.60 ±0.10

Documentation Support

DPD1126-4_HSA8000_IXC2_HW_and_Register_Users_Guide

DPD1144-2_HSA8000_IXC2_Register_Map

DPD1153-1_HSA8000_Totem-Pole_Bridgeless_PFC_Platform_AppNote

DPD1154-1_HSA8000_ LLC_SeriesResonantConverter_AppNote

DPD1140-1 MI-P700_4Q_Development_Platform_Application_Note