

# 47 Harmonic Analysis Engine (HAE)

The harmonic analysis engine (HAE) analyzes harmonic frequencies present on the voltage and current input samples. The HAE receives input samples from two source channels whose frequencies are 45–65 Hz. The HAE then processes the input samples and produces output results. The output results consist of power quality measurements of the fundamental and up to 12 more harmonics.

## HAE Features

The HAE features include:

- Processing of two 24-bit signed input channels, consisting of one voltage and one current. The input full scale is limited to  $\sim \pm 6,000,000$
- Processing of fundamental frequencies between 45-65 Hz
- Processing of input samples at a nominal 8 kHz rate
- Processing of the fundamental plus 12 harmonic frequencies
- Active, reactive, apparent,  $I_{RMS}$ ,  $V_{RMS}$ , and power factor on the fundamental frequency
- Active, reactive, apparent,  $I_{RMS}$ ,  $V_{RMS}$ , power factor,  $I_{HD+n}$ , and  $V_{HD+n}$  on the 12 harmonic frequencies
- The accuracy of the measurements, relative to a full scale of  $\pm 6,000,000$ :
  - Fundamental active or reactive powers 0.1% down to 1/1000 of full scale
  - Fundamental apparent power 0.2% down to 1/1000 of full scale
  - Fundamental  $I_{RMS}/V_{RMS}$  0.1% down to 1/1000 of full scale
  - Fundamental power factor 0.3% based on active and apparent accuracy
  - Harmonic active or reactive powers 1% down to 1/1000 of full scale
  - Harmonic apparent power 2% down to 1/1000 of full scale
  - Harmonic  $I_{RMS}/V_{RMS}$  1% down to 1/1000 of full scale
  - Harmonic power factor 3% based on active and apparent accuracy
  - Harmonic  $I_{HD+n}$  2% down to 1/1000 FS based on the fundamental and harmonic  $I_{RMS}$

- Harmonic  $V_{HD+n}$  2% down to 1/1000 FS based on the fundamental and harmonic  $V_{RMS}$
- Twelve 6-bit fields for selecting harmonics to analyze, limited by bandwidth of the input signal. The fundamental component always is provided.

## HAE Functional Description

The following sections provide a functional description of the HAE.

### Harmonic engine

The hardware block of the harmonic engine works with other HAE blocks to co-process full and partial results

### Harmonic Analyzer

The harmonic analyzer block works with the harmonic engine to co-process full and partial results.

### Data Transfer Module

The data transfer module transfers three channels of data to and from the HAE module.

## ADSP-SC58x HAE Register List

The Harmonic Analysis Engine (HAE) analyzes harmonics present on voltage and current input samples. The HAE receives input samples from two source channels, processes the samples, and produces output results. The output results consist of power quality measurements of the fundamental and up to twelve additional harmonic components. A set of registers governs HAE operations. For more information on HAE functionality, see the HAE register descriptions.

**Table 1:** ADSP-SC58x HAE Register List

Name	Description
HAE_CFG0	Configuration 0 Register
HAE_CFG1	Configuration 1 Register
HAE_CFG2	Configuration 2 Register
HAE_CFG3	Configuration 3 Register
HAE_CFG4	Configuration 4 Register
HAE_DIDT_COEF	DIDT Coefficient Register
HAE_DIDT_GAIN	DIDT Gain Register

**Table 1:** ADSP-SC58x HAE Register List (Continued)

Name	Description
HAE_H[nn]_INDX	Harmonic n Index Register
HAE_ISAMPLE	I (Current) Sample Register
HAE_IWAVEFORM	I (Current) Waveform Register
HAE_RESULTS_RAM	Results RAM Register
HAE_RUN	Run Register
HAE_STAT	Status Register
HAE_VLEVEL	Voltage Level Register
HAE_VSAMPLE	V (Voltage) Sample Register
HAE_VWAVEFORM	V (Voltage) Waveform Register

## ADSP-SC58x HAE Interrupt List

**Table 2:** ADSP-SC58x HAE Interrupt List

Interrupt ID	Name	Description	Sensitivity	DMA Channel
158	HAE0_RXDMA_CH0	HAE0RX DMA Channel 0	Level	32
159	HAE0_RXDMA_CH1	HAE0RX DMA Channel 1	Level	33
160	HAE0_TXDMA	HAE0TX DMA Channel 0	Level	31
161	HAE0_STAT	HAE0Status	Level	
212	HAE0_RXDMA_CH0_ERR	HAE0RX DMA Channel 0 Error	Level	
213	HAE0_RXDMA_CH1_ERR	HAE0RX DMA Channel 1 Error	Level	
214	HAE0_TXDMA_ERR	HAE0TX DMA Channel Error	Level	

## ADSP-SC58x HAE Trigger List

**Table 3:** ADSP-SC58x HAE Trigger List Masters

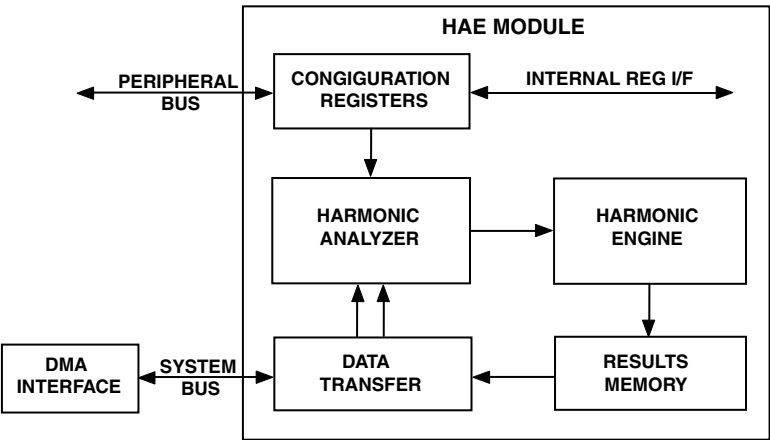
Trigger ID	Name	Description	Sensitivity
46	HAE0_RXDMA_CH0	HAE0RX DMA Channel 0	Level
47	HAE0_RXDMA_CH1	HAE0RX DMA Channel 1	Level
48	HAE0_TXDMA	HAE0TX DMA Channel 0	Level

**Table 4:** ADSP-SC58x HAE Trigger List Slaves

Trigger ID	Name	Description	Sensitivity
39	HAE0_RXDMA_CH0	HAE0RX DMA Channel 0	Level
40	HAE0_RXDMA_CH1	HAE0RX DMA Channel 1	Level
41	HAE0_TXDMA	HAE0TX DMA Channel 0	Level

HAE Block Diagram

The **HAE Block Diagram** shows the functional blocks within the HAE.



**Figure 1:** HAE Block Diagram

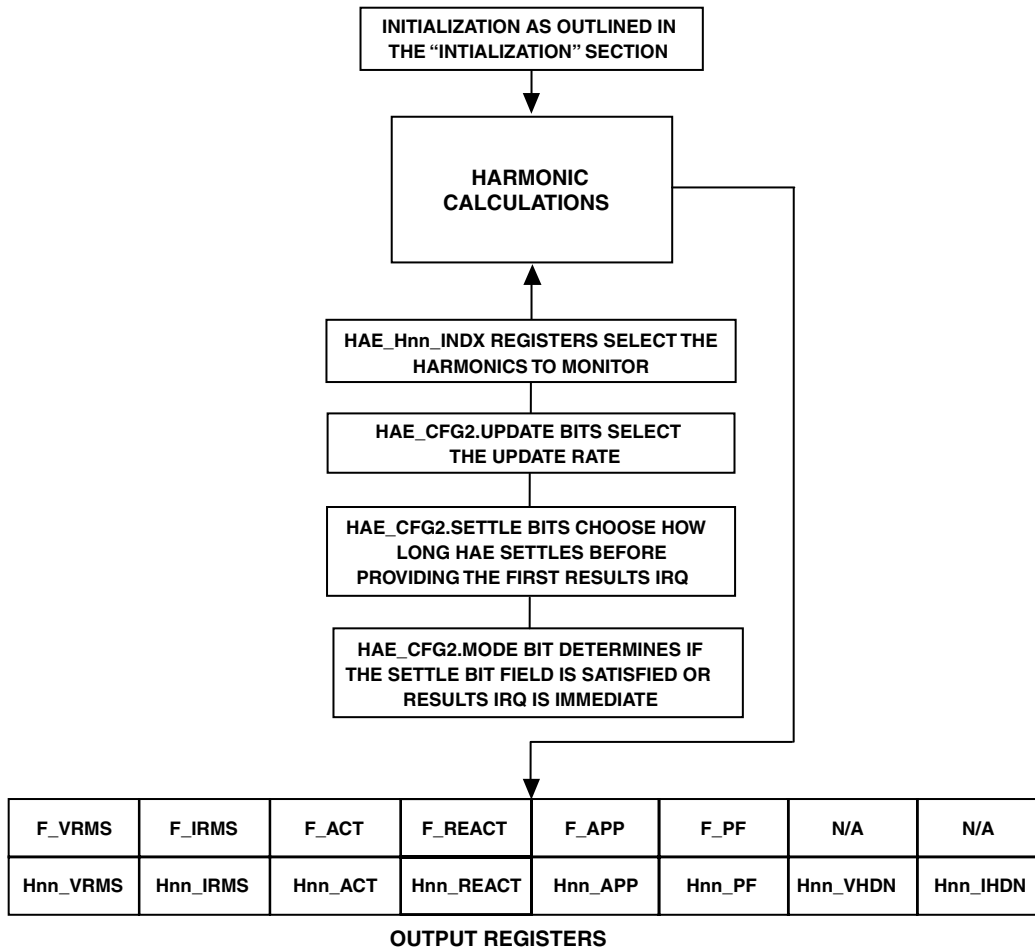
HAE Architectural Concepts

Using the HAE features and event control to their greatest potential requires an understanding of these architectural concepts.

- [Harmonic Engine](#)
- [Harmonic Analyzer](#)
- [Data Transfer Module](#)
- [Results Memory](#)

## Harmonic Engine

The **HAE Engine Block Diagram** presents a synthesized diagram of the harmonic engine, its settings, and its output registers.



**Figure 2:** HAE Engine Block Diagram

The harmonic engine hardware block works with other HAE blocks to co-process full and partial results (see [Harmonic Calculations](#)). At the start of a new sampling period (described in [Initialization](#)), the harmonic engine cycles through predefined locations in data RAM, which contain the analyzer processing results. See [Harmonic Analyzer](#) for more information.

As the harmonic engine produces results in their final formats (described in [HAE Result Ranges and Formats](#)), the results are stored in the results memory (see [Results Memory](#)).

The HAE engine computes harmonic information for line frequencies 45–66 Hz. Neutral current can also be analyzed simultaneously with the sum of phase currents. See [Theory of Operation](#) for more information.

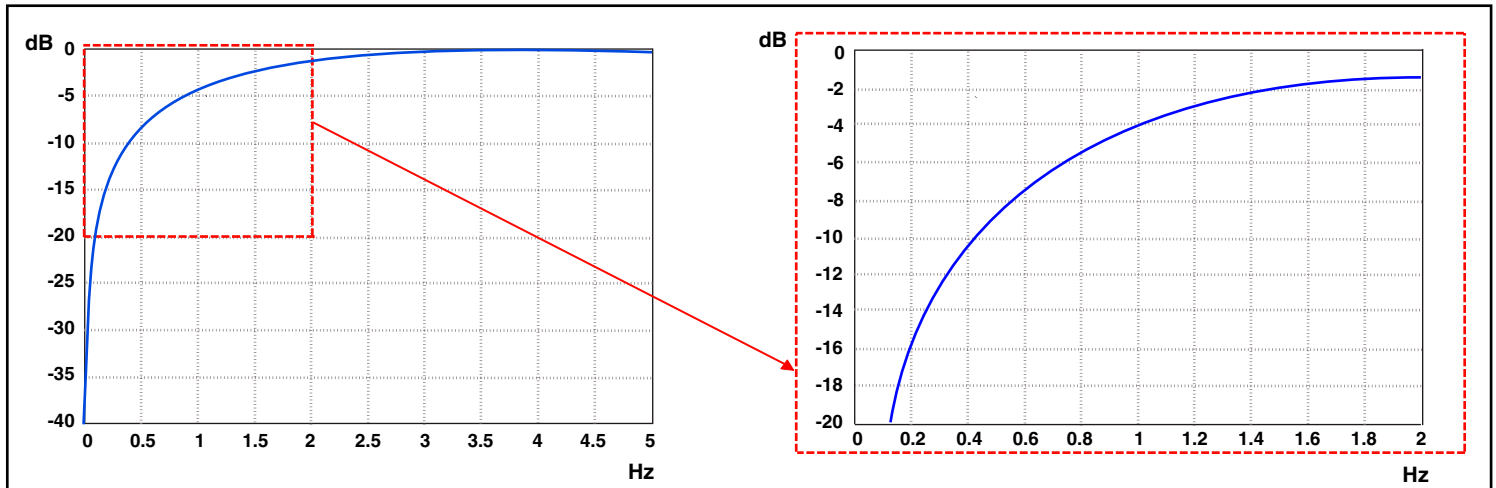
## Harmonic Analyzer

The harmonic analyzer block works with the [Harmonic Engine](#) to co-process full and partial results.

To perform harmonic analysis, the HAE contains a pair of voltage and current inputs and a set of 12 indexes to indicate which harmonic components to extract. See [Theory of Operation](#) and [Harmonic Calculations](#) for more information.

### High-Pass Filters (HPFs)

The voltage and current have the option of being high-pass filtered to remove DC offsets. The **Frequency Response for High-Pass Filters** figure shows the frequency response of the high-pass filters (the -3 dB point is around 1.1 Hz). The HPFs are enabled by default.

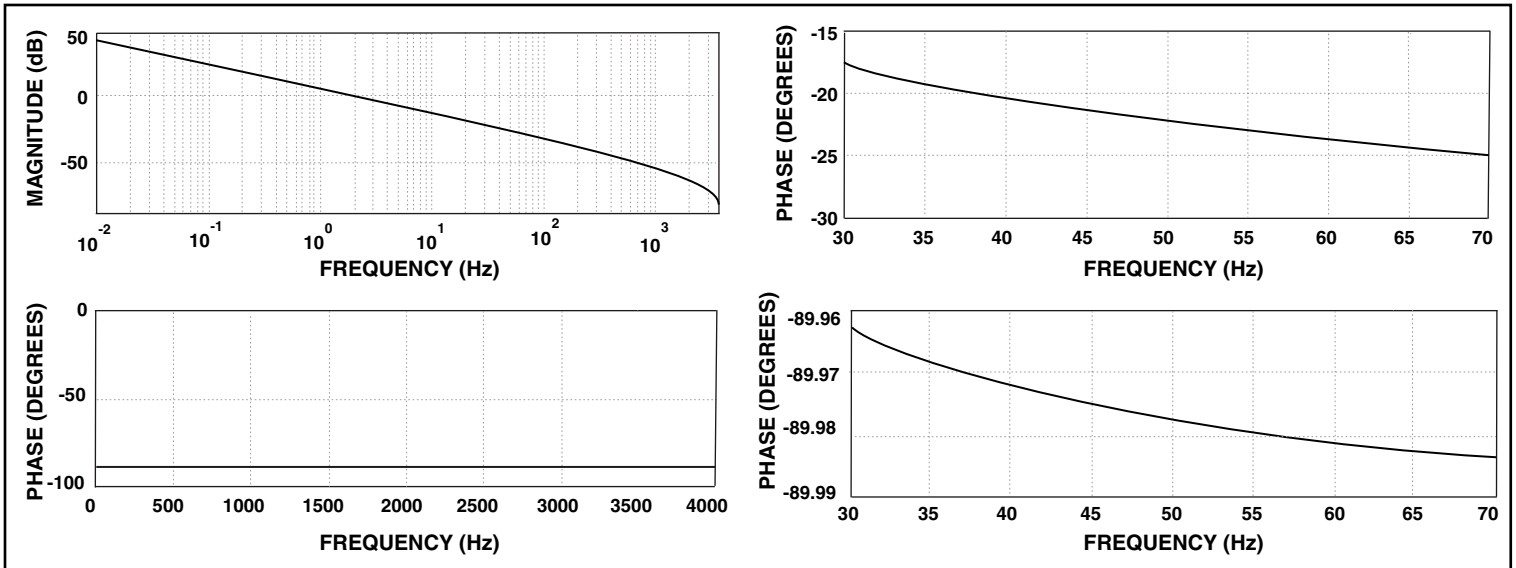


**Figure 3:** Frequency Response for High-Pass Filters

### Digital Integrators

For cases when the current is sensed with a  $di/dt$  sensor, such as a Rogowski coil, the HAE offers an internal digital integrator for compensation. Ideally, it causes a perfect 90 degrees of phase shift at all the frequencies; at 50 Hz it is close to that value (see the **Digital Integrator** figure). The integrator is necessary to restore the signal to its original form before using the signal in HAE calculations. The digital integrator is disabled by default.

With digital integrator enabled for current channel, the amplitude of current will change with the frequency according to a slope of 20 dB per decade. For frequency of 50 Hz, the amplitude of current remains same after passing through the integrator. However, for other frequencies in the range 45-65 Hz, the amplitude of current changes with a slope of 20 dB per decade after passing through integrator.



**Figure 4:** Digital Integrator

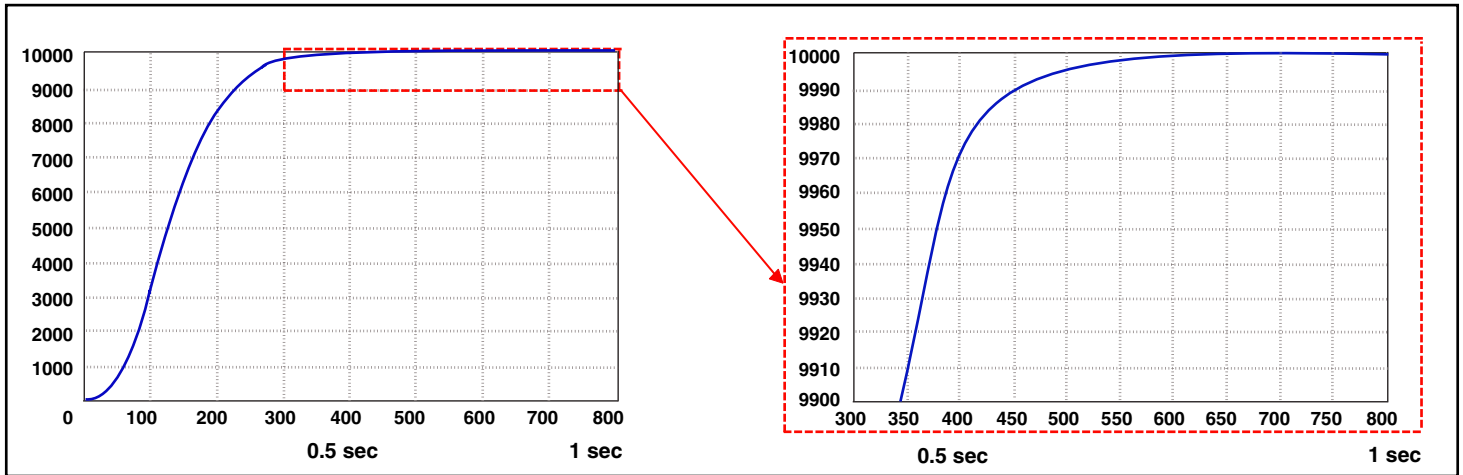
### Phase-Locked Loop and Clock Control (PLL)

The fundamental frequency of the system is extracted from the voltage signal using the phase-locked loop and clock control (PLL) techniques. PLL techniques are optimized for signals with frequencies used in standard power grids around the world (50 Hz or 60 Hz). The techniques consider possible deviations of up to 5 Hz, so the final guaranteed operating range is from 45 Hz to 65 Hz.

The initial detection time of the frequency depends on its value and can take up to several seconds. This activity only happens at the start-up of the HAE block. Once the value of the fundamental frequency is detected, the PLL tracks it continuously.

### Settling Times

The block that extracts all of the harmonic RMS and power values is replicated 12 times in parallel for all of the harmonic indexes plus once for the fundamental. Once an index for a particular harmonic changes, there is a settling time of about 700 mSec ( $\sim 700 / 0.125 = 5600$  8K samples). This settling time achieves less than 0.1% error needed for all the internal RMS and powers computations (see the **Fundamental and Harmonic Values - Settling Times** plots).



**Figure 5:** Fundamental and Harmonic Values - Settling Times

## Data Transfer Module

The data transfer module transfers three channels of data to and from the HAE module.

### RX I (Receive Current Sample) Channel

The RX I channel transfers HAE current (I) input samples from system memory through direct memory access (DMA). The RX I request occurs at a nominal 8 kHz rate, depending on the `HAE_CFG1.STARTDIV` bit field. The RX I samples typically come from a SINC filter through a memory buffer. For accurate harmonic results, derive the input samples at the same rate as the HAE sample loop. Therefore, program the SINC and HAE modules to be timing-matched. The `SCLK/HAE_CFG1.STARTDIV` determines the HAE sample loop, which must be programmed to be nominally 8 kHz in frequency. There is one DWORD transfer each HAE sample loop.

There is also an RX I memory-mapped location (`HAE_ISAMPLE` register) in peripheral space (MMR), which enables the MCU to provide the I channel samples, timed with the `HAE_STAT.RXIRQ` bit, if desired.

### RX V (Receive Voltage Sample) Channel

The RX V channel transfers HAE voltage (V) input samples from system memory through DMA. The RX V request occurs at a nominal 8 kHz rate, depending on the `HAE_CFG1.STARTDIV` bit field. The RX V samples typically come from a SINC filter through a memory buffer. For accurate harmonic results, derive the input samples at the same rate as the HAE sample loop. Therefore, program the SINC and HAE modules to be timing-matched. The `SCLK/HAE_CFG1.STARTDIV`, determines the HAE sample loop which must be programmed to be nominally 8 kHz in frequency. There is one DWORD transfer each HAE sample loop.

There is also an RX V memory-mapped location (`HAE_VSAMPLE` register) in peripheral space (MMR), which enables the MCU to provide the V channel samples, timed with the `HAE_STAT.RXIRQ` bit, if desired.



Example:

- The `HAE_STAT.RXIRQ` bit toggles prior to RX transfers:

The RX interrupt is used internally in hardware to request I and V samples. The MCU can also use the interrupt when it supplies the input samples, rather than the DMA interface.

- RX transfers indicate that the RX channel is ready:

The HAE RX channels request one RX sample at an 8 kHz rate, depending on the `HAE_CFG1.STARTDIV` bit field. The holding register stores the RX data, replacing the previous data. The holding register contents are moved up to an output register and driven to the [Harmonic Analyzer](#) module for processing. This operation amounts to a two-deep FIFO, allowing a full sample time of latency on the input sample arrival, nominally 125 uSec (8 kHz).

If the MCU supplies the input samples, rather than the DMA RX interfaces, there are two memory-mapped locations (`HAE_ISAMPLE` and `HAE_VSAMPLE`), where the next samples are written. As previously mentioned, the HAE can use the `HAE_STAT.RXIRQ` bit to time the sample delivery.

### TX (Transmit Results) Channel

The TX channel transfers HAE results from results memory to system memory through DMA. The results for the fundamental and 12 harmonics are stored in 13 8-location fields in the results memory. Therefore, the maximum number of DWORDs to transfer is  $13 \times 8 = 104$ . Use the `HAE_CFG3.CHANEN` bits to select the channels to transfer.

The HAE can request a transfer of results for each 8 kHz sample period, meaning that up to 104 DWORDs can be transferred each 125 uSec. Use the `HAE_CFG2.UPDATE` bit field to set the request rate to longer intervals. The results also are memory-mapped to peripheral address space, which enables the MCU to read the results, timed with the `HAE_STAT.TXIRQ` bit, if desired.

Example:

- The `HAE_CFG3.CHANEN` bit field is set to 0x009 to transfer the fundamental and harmonic contained in the `HAE_H3_INDX` register (programmed to 5<sup>th</sup>).
- The `HAE_STAT.TXIRQ` bit toggles prior to TX transfers:

The HAE uses the TX interrupt internally in hardware to start the transfer. The MCU can also use the interrupt when the MCU reads the HAE results, rather than the DMA interface transferring it.

- TX transfers indicate that the HAE results data is ready:

The HAE TX channel can transfer one DWORD every other clock. The internal hardware parses through the `HAE_CFG3.CHANEN` bit field, eventually traversing all 13 bits. If consecutive channels are enabled, two more IDLE clocks are inserted between the adjacent channels for a total of three IDLE clocks. When channels are skipped, one more IDLE clock is inserted for each unselected channel. Therefore, there are 3 (inter-channel IDLE) + 2 (skipped channels) = 5 IDLE clocks between the fundamental and `HAE_H3_INDX` transfers.

## Results Memory

The results memory is organized by the fundamental and 12 harmonic indexes. Each of the 13 potentially analyzed frequencies has eight locations dedicated to store various power quality measurements. Each frequency is offset from the next by eight locations.

The **HAE Results Memory** figure shows the results memory contents.

	INDEX	0	1	2	3	4	5	6	7
GROUP	OFFSET								
FUNDAMENTAL	0x00	F_IRMS	F_VRMS	F_ACT	F_REACT	F_APP	F_PF	N/A	N/A
H1_INDEX	0x08	H1_IRMS	H1_VRMS	H1_ACT	H1_REACT	H1_APP	H1_PF	H1_IHDN	H1_VHDN
H2_INDEX	0x10	H2_IRMS	H2_VRMS	H2_ACT	H2_REACT	H2_APP	H2_PF	H2_IHDN	H2_VHDN
H3_INDEX	0x18	H3_IRMS	H3_VRMS	H3_ACT	H3_REACT	H3_APP	H3_PF	H3_IHDN	H3_VHDN
H4_INDEX	0x20	H4_IRMS	H4_VRMS	H4_ACT	H4_REACT	H4_APP	H4_PF	H4_IHDN	H4_VHDN
H5_INDEX	0x28	H5_IRMS	H5_VRMS	H5_ACT	H5_REACT	H5_APP	H5_PF	H5_IHDN	H5_VHDN
H6_INDEX	0x30	H6_IRMS	H6_VRMS	H6_ACT	H6_REACT	H6_APP	H6_PF	H6_IHDN	H6_VHDN
H7_INDEX	0x38	H7_IRMS	H7_VRMS	H7_ACT	H7_REACT	H7_APP	H7_PF	H7_IHDN	H7_VHDN
H8_INDEX	0x40	H8_IRMS	H8_VRMS	H8_ACT	H8_REACT	H8_APP	H8_PF	H8_IHDN	H8_VHDN
H9_INDEX	0x48	H9_IRMS	H9_VRMS	H9_ACT	H9_REACT	H9_APP	H9_PF	H9_IHDN	H9_VHDN
H10_INDEX	0x50	H10_IRMS	H10_VRMS	H10_ACT	H10_REACT	H10_APP	H10_PF	H10_IHDN	H10_VHDN
H11_INDEX	0x58	H11_IRMS	H11_VRMS	H11_ACT	H11_REACT	H11_APP	H11_PF	H11_IHDN	H11_VHDN
H12_INDEX	0x60	H12_IRMS	H12_VRMS	H12_ACT	H12_REACT	H12_APP	H12_PF	H12_IHDN	H12_VHDN

**Figure 6:** HAE Results Memory

### HAE Results Upper Byte ID

The HAE results are stored in a 24-bit wide memory, as shown in [Results Memory](#).

When the system bus moves the results into memory, the HAE hardware appends a unique CHANNEL : INDEX identifier to the upper byte of each results location. The intent of the ID byte is to help in data parsing of HAE results.

The 32-bit appended result is as follows:

CHANNEL[3:0]	INDEX[3:0]	RESULTS[23:0]
--------------	------------	---------------

The CHANNEL in the upper byte corresponds to nn in Hnn\_INDXX of that particular results location. The INDXX in the upper byte corresponds to the INDXX within the Hnn\_INDXX grouping.

For example, location H7\_PF has 0x75 in the upper byte of the system bus transfer of that HAE results memory location.

## HAE Result Ranges and Formats

The HAE results, stored in the results memory, have formats appropriate for the given measurement. Measurements accuracy is relative to the full scale value of the input samples, and the assumed full scale is  $\pm \sim 6,000,000$ . If the full scale is above this value, overflow can occur and the results are undefined. A lower full scale limits the dynamic range accuracy of the measurements. For example, a full scale, which is 50% lower than the assumed 6,000,000 full scale, has the dynamic range reduced by 50%. Potentially, this result can be better or worse, as determined by the ADC noise floor supplying the input samples.

The HAE results are as linear as the input samples, within the accuracy ranges specified earlier. The results assume a full-scale input sample of  $\sim \pm 6,000,000$  and sufficiently low ADC noise floor. Evaluate the system using your specific ADC and full scale specifications to predict the HAE range of accuracy and expected results.

The RMS values of voltage and current calculated from HAE are exactly not same as the theoretical RMS values. The ratio between HAE calculated RMS value and the theoretical value is around 1.11. A gain factor should be added in the software so as to match the theoretical and HAE calculated RMS values. Calibration can be done by calculating the ratio between HAE calculated RMS value and the theoretical one for a particular amplitude of V and I signal and line frequency.

The following list gives the HAE result formats.

- $I_{RMS}$  and  $V_{RMS}$ : both fundamental and harmonic  $I_{RMS}$  and  $V_{RMS}$  are unsigned magnitudes with full scale of  $\sim 4,200,000$
- Active and reactive power: both active and reactive powers are signed numbers with full scale of  $\sim \pm 4,200,000$
- Apparent power: apparent power is an unsigned magnitude with full scale of  $\sim 4,200,000$
- Power factor: each LSB of the fundamental and harmonic power factors equates to a weight of  $2^{-23}$ . Hence, the maximum register value of 0x7FFFFFFF equates to a power factor value of 1. The minimum register value of 0x800000 corresponds to a power factor of -1. If the power factor is outside of the -1 to +1 range because of offset and gain calibrations, the result is set at -1 or +1. The result depends on the sign of the fundamental reactive power.
- $I_{HD+n}/V_{HD+n}$ : the harmonic distortion plus noise ratios are computed using the RMS of the fundamental and the RMS of the harmonic under analysis. In other words, the ratio only covers the particular harmonic under analysis versus the fundamental. The ratios are stored as 24-bit values in 3.21 signed

format. The ratios are limited to +3.9999, and all greater results are clamped to it. The  $H_{D+n}$  ratios cannot be negative.

## HAE Operating Modes

The HAE uses the DMA interface to transfer data to and from system memory. The HAE configuration registers enable the module and calibrate its frequency (clock) divide and other parameters, as described in [HAE Programming Model](#). The HAE triggers and status signals indicate system events and errors.

## HAE Data Transfer Modes

The HAE uses the RX DMA interface to transfer data from system memory. Samples for the current (I) and voltage (V) channels of the AFE or SINC filter compose the data: two WORDS are transferred each 8 kHz sample period into the HAE (I and V).

The HAE uses the TX DMA interface to transfer data to system memory. The fundamental and selected harmonic results compose the data: up to 13 (fundamental + 12 harmonics)  $\times 8 = 104$  DWORDs are transferred each 8K sample period.

See [Data Transfer Module](#) for more information.

## HAE Event Control

The HAE module uses DMA to transfer samples to and data from system memory. The HAE also can use TX and RX events to time the arrival of input samples and extract the results by the MCU:

- The MCU uses the RX event (`HAE_STAT.RXIRQ`) as an IRQ to time when to write the I and V samples into the HAE.
- The MCU uses the TX event (`HAE_STAT.TXIRQ`) as an IRQ to time when to read the HAE results.

## HAE Interrupt Signals

The interrupt signals to the HAE module include:

- The RX interrupt to time the delivery of waveform samples as inputs to the HAE.
- The TX interrupt to time when the HAE results memory is ready with new values for the current sample.

The HAE generates the RX interrupts at a rate specified by the `HAE_CFG1.STARTDIV` bit field and the TX interrupts at a rate specified by the `HAE_CFG2.UPDATE` bit field. Refer to [Data Transfer Module](#) for more information.

## HAE Status and Error Signals

The trigger and status signals to the HAE module include:

- `HAE_STAT.RDY` (HAE ready status). When the bit is set, the HAE is fully accessible.
- `HAE_STAT.RXIRQ` (RX IRQ status). The bit mirrors the RX interrupt.
- `HAE_STAT.TXIRQ` (TX IRQ status). The bit mirrors the TX interrupt.

The status bit requires a 1 to clear and reenables the corresponding interrupt for the next sample period. Refer to [Data Transfer Module](#) for more information.

## HAE Programming Model

The following sections provide basic procedures for configuring various HAE operations.

Current and Voltage data can be transferred from system memory to HAE and HAE results can be transferred to system memory using core and DMA accesses. DMA transfers can be set up to transfer a configurable number of I and V samples and HAE results between HAE and system memory automatically. Core-driven transfers use HAE interrupts to signal the processor core to provide I and V data to the sample registers and read the HAE results from the system memory.

The following sections provide recommended programming guidelines to be followed for core and DMA mode.

## Configuring the HAE for DMA Transfers

- Enable HAE operations by configuring the `HAE_CFG2.EN` bit.
- Poll for HAE ready status from the `HAE_STAT.RDY` bit configuration.
- Initialize the HAE configuration registers to configure update and settle rates, HPF, integrator for di/dt sensor and line frequency.
- Choose the harmonic channels to be monitored by configuring the `HAE_H[nn]_INDX` registers and enable channels by configuring the `HAE_CFG3.CHANEN` bits.
- Configure the RX DMA channel 0 and the RX DMA channel 1 to receive I and V samples. Configure the TX DMA channel 0 to transmit harmonic calculation results.
- Set the HAE clock divider using the `HAE_CFG1.STARTDIV` field to 8 kHz.
- Set the `HAE_RUN` register.

## Configuring the HAE for Core Transfers

- Enable HAE operations by configuring the `HAE_CFG2.EN` bit.
- Poll for HAE ready status using the `HAE_STAT.RDY` bit.
- Initialize the HAE configuration registers to configure update and settle rates, HPF, integrator for di/dt sensor and line frequency.
- Enable receive and transmit IRQ using the `HAE_CFG0` register.
- Configure the `HAE_VLEVEL` register depending on input voltage magnitude requirements.
- Configure the `HAE_H[nn]_INDX` registers to choose the harmonic channels to monitor. Enable the channels using the `HAE_CFG3.CHANEN` bits.
- Set the HAE clock divider using the `HAE_CFG1.STARTDIV` field to 8 kHz.
- Set the `HAE_RUN` register.
- At each receive IRQ (`HAE_STAT.RXIRQ = 1`), write I and V sample values to the `HAE_ISAMPLE` and the `HAE_VSAMPLE` registers respectively.
- At each transmit IRQ (`HAE_STAT.TXIRQ = 1`), read the HAE results from the HAE result RAM. The address is provided by the `HAE_RESULTS_RAM` register.

## HAE Programming Concepts

Using the HAE features to their greatest potential requires an understanding of some HAE-related concepts.

- [Theory of Operation](#)
- [Initialization](#)
- [Harmonic Calculations](#)
- [Configuring Harmonic Calculations Update Rate](#)

### Theory of Operation

The HAE theory of operation is as follows.

Consider a nonsinusoidal AC system supplied by a voltage,  $v(t)$ , that consumes the current,  $i(t)$ . Then,

$$v(t) = \sum_{k=1}^{\infty} V_k \sqrt{2} \sin(k\omega t + \varphi_k)$$

$$i(t) = \sum_{k=1}^{\infty} I_k \sqrt{2} \sin(k\omega t + \gamma_k)$$

where:

- $V_k, I_k$  are the RMS voltage and current, respectively, of each harmonic.
- $\Phi_k, \gamma_k$  are the phase delays of each harmonic.
- $\omega$  is the angular velocity at the fundamental (line) frequency  $f$ .

The HAE harmonic calculations are specified for line frequencies 45–65 Hz.

The maximum number of harmonics which the HAE can accept for a particular line frequency will depend upon the bandwidth of the input waveforms supplied.

When the HAE analyzes I and V samples, the following metering quantities are computed:

- Fundamental current RMS:  $I_1$
- Fundamental voltage RMS:  $V_1$
- RMS of up to 12 harmonics of the current channel:  $I_n, n = 2, 3, \dots, 12$
- RMS of up to 12 harmonics of the voltage channel:  $V_n, n = 2, 3, \dots, 12$
- Fundamental active power:  $P_1 = V_1 I_1 \cos(\varphi_1 - \gamma_1)$
- Fundamental reactive power:  $Q_1 = V_1 I_1 \sin(\varphi_1 - \gamma_1)$
- Fundamental apparent power:  $S_1 = V_1 I_1$
- Power factor of the fundamental:

$$pf_1 = \text{sgn}(Q_1) \times \frac{P_1}{S_1}$$

Active power of up to 12 harmonics:

$$P_n = V_n I_n \cos(\varphi_n - \gamma_n), n = 2, 3, \dots, 12$$

- Reactive power of up to 12 harmonics:

$$Q_n = V_n I_n \sin(\varphi_n - \gamma_n), n = 2, 3, \dots, 12$$

- Apparent power of up to 12 harmonics:

$$S_n = V_n I_n, n = 2, 3, \dots, 12$$

- Power factor of up to 12 harmonics:

$$pf_n = \text{sgn}(Q_n) \times \frac{P_n}{S_n}, n = 2, 3, \dots, 12$$

- Total harmonic distortion of the current channel:

$$(THD)_I = \frac{\sqrt{I^2 - I_1^2}}{I_1}$$

- Total harmonic distortion of the voltage channel:

$$(THD)_V = \frac{\sqrt{V^2 - V_1^2}}{V_1}$$

- Harmonic distortion of up to 12 harmonics on the current channel:

$$HD_{I_n} = \frac{I_n}{I_1}, n = 2, 3, \dots, 12$$

- Harmonic distortion of up to 12 harmonics on the voltage channel:

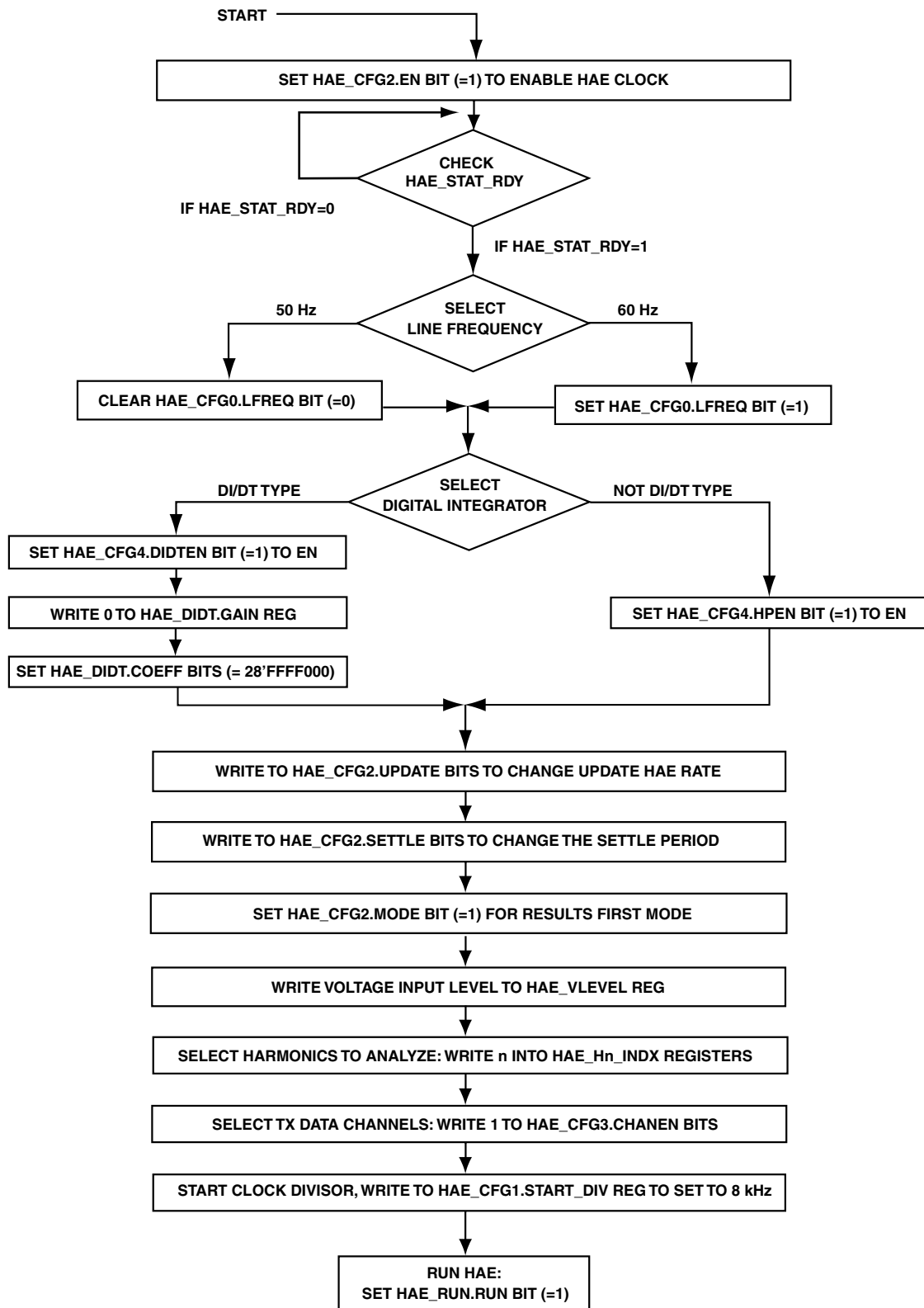
$$HD_{V_n} = \frac{V_n}{V_1}, n = 2, 3, \dots, 12$$

## Initialization

The HAE typically is initialized with parameters and settings for a given usage scenario, then is interrupt-driven when new results are available.

The HAE programming depends on whether the line frequency is 50 Hz or 60 Hz. If the external sensor is a di/dt type, there are also some differences. The **HAE Initialization** flowchart shows the initialization sequence.





**Figure 7:** HAE Initialization

## Harmonic Calculations

When the harmonic engine runs, it computes information about the fundamental and up to 12 harmonics. The HAE simultaneously monitors the indexes of the additional 12 harmonics, provided by the 8-bit registers HAE\_H[nn]\_INDX. Write the index of the harmonic into the register for the harmonic to be monitored. If the second harmonic is monitored, write 2. If harmonic 51 is desired, write 51. The HAE always monitors the fundamental component, independent of the values written into HAE\_H[nn]\_INDX. Therefore, if one of these registers is equal to 1, the HAE monitors the fundamental components multiple times. The maximum index allowed in the HAE\_H[nn]\_INDX registers is 63.

As a reference, the **Harmonic Engine Outputs and Registers where Values are Stored** table presents the harmonic engine outputs and registers that store the outputs.

**Table 5:** Harmonic Engine Outputs and Registers where Values are Stored

Quantity	Definition	HAE Registers
RMS of the Fundamental Component	$V_1, I_1$	F_VRMS, F_IRMS
RMS of a Harmonic Component	$V_n, I_n, n = 2, 3, \dots, 12$	Hnn_VRMS, Hnn_XIRMS
Active Power of the Fundamental Component	$P_1 = V_1 I_1 \cos(\varphi_1 - \gamma_1)$	F_ACT
Active Power of a Harmonic Component	$P_n = V_n I_n \cos(\varphi_n - \gamma_n), n = 2, 3, \dots, 12$	Fnn_ACT
Reactive Power of the Fundamental Component	$Q_1 = V_1 I_1 \sin(\varphi_1 - \gamma_1)$	F_REACT
Reactive Power of a Harmonic Component	$Q_n = V_n I_n \sin(\varphi_1 - \gamma_1), n = 2, 3, \dots, 12$	Hnn_REACT
Apparent Power of the Fundamental Component	$S_1 = V_1 I_1$	F_APP
Apparent Power of a Harmonic Component	$S_n = V_n I_n, n = 2, 3, \dots, 12$	Hnn_APP
Power Factor of the Fundamental Component	$pf_1 = \text{sgn}(Q_1) \times \frac{P_1}{S_1}$	F_PF
Power Factor of a Harmonic Component	$pf_n = \text{sgn}(Q_n) \times \frac{P_n}{S_n}, n = 2, 3, \dots, 12$	Hnn_PF
Harmonic Distortion of a Harmonic Component	$HD_{V_n} = \frac{V_n}{V_1}, HD_{I_n} = \frac{I_n}{I_1}, n = 2, 3, \dots, 12$	Hnn_VHDN, Hnn_IHDN

## Configuring Harmonic Calculations Update Rate

The harmonic engine functions at an 8 kHz rate. From the moment the HAE\_CFG2 register is initialized, and the harmonic indexes are set in the HAE\_H[nn]\_INDX index registers, the HAE calculations take typically 750 mSec to settle within the specification parameters.

The HAE module uses the HAE\_CFG2.UPDATE bits to manage the update rate of the output registers for the harmonic engine. It manages the update rate independent of the calculations rate of 8 kHz for the engine. The default value of 000 means that the registers are updated every 125 uSec (8 kHz rate). Other update periods are: 250 uSec (001), 1 mSec (010), 16 mSec (011), 128 mSec (100), 512 mSec (101), 1.024 mSec (110). If the HAE\_CFG2.UPDATE bits are 111, the harmonic calculations are disabled.

The HAE module provides two ways to manage the harmonic computations. It enables the first approach when bit HAE\_CFG2.MODE is cleared to its default value of 0. The state sets status bit HAE\_STAT.TXIRQ to 1 after a certain period and then every time the harmonic calculations are updated at HAE\_CFG2.UPDATE frequency. This functionality allows an external microcontroller to access the harmonic calculations only after they have settled. The HAE uses the state of bits HAE\_CFG2.SETTLE to determine the time period. The possible values of settling time of the harmonic calculations are 512 mSec (00), 768 mSec(01), 1024 mSec (10), and 1280 mSec (11).

The HAE module enables the second approach when bit HAE\_CFG2.MODE is set to 1. The state sets the status bit HAE\_STAT.TXIRQ to 1 every time the harmonic calculations are updated at the HAE\_CFG2.UPDATE frequency, without waiting for the harmonic calculations to settle. This functionality allows an external microcontroller to access the harmonic calculations immediately after starting. A write to the HAE\_STAT register clears the status bit. The corresponding bit (HAE\_STAT.TXIRQ) is set to 1.

## ADSP-SC58x HAE Register Descriptions

Harmonic Analysis Engine (HAE) contains the following registers.

**Table 6:** ADSP-SC58x HAE Register List

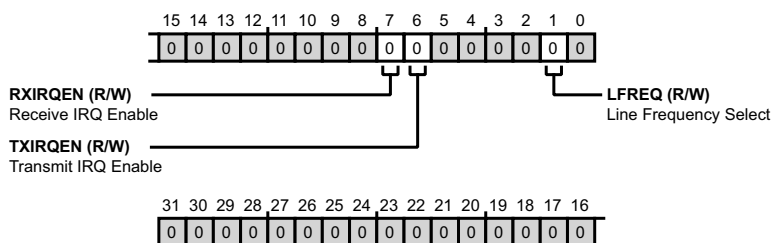
Name	Description
HAE_CFG0	Configuration 0 Register
HAE_CFG1	Configuration 1 Register
HAE_CFG2	Configuration 2 Register
HAE_CFG3	Configuration 3 Register
HAE_CFG4	Configuration 4 Register
HAE_DIDT_COEF	DIDT Coefficient Register
HAE_DIDT_GAIN	DIDT Gain Register

**Table 6:** ADSP-SC58x HAE Register List (Continued)

Name	Description
HAE_H[nn]_INDX	Harmonic n Index Register
HAE_ISAMPLE	I (Current) Sample Register
HAE_IWAVEFORM	I (Current) Waveform Register
HAE_RESULTS_RAM	Results RAM Register
HAE_RUN	Run Register
HAE_STAT	Status Register
HAE_VLEVEL	Voltage Level Register
HAE_VSAMPLE	V (Voltage) Sample Register
HAE_VWAVEFORM	V (Voltage) Waveform Register

## Configuration 0 Register

The HAE\_CFG0 register configures high-level interrupts and specifies the line frequency for HAE operations.



**Figure 8:** HAE\_CFG0 Register Diagram

**Table 7:** HAE\_CFG0 Register Fields

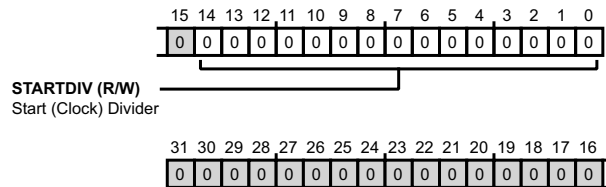
Bit No. (Access)	Bit Name	Description/Enumeration
7 (R/W)	RXIRQEN	Receive IRQ Enable. The HAE_CFG0.RXIRQEN bit enables an interrupt, which the HAE triggers on each request for a new input sample on both the I and V channels.
		0 Disable
		1 Enable

**Table 7:** HAE\_CFG0 Register Fields (Continued)

Bit No. (Access)	Bit Name	Description/Enumeration
6 (R/W)	TXIRQEN	Transmit IRQ Enable. The HAE_CFG0.TXIRQEN bit enables an interrupt, which the HAE triggers on each result as the result is calculated and ready to transmit.
		0 Disable
		1 Enable
1 (R/W)	LFREQ	Line Frequency Select. The HAE_CFG0.LFREQ bit specifies the line frequency for the HAE. Set the bit to match the line frequency being analyzed.
		0 50 Hz
		1 60 Hz

## Configuration 1 Register

The HAE\_CFG1 register configures the HAE frequency (clock) divider.



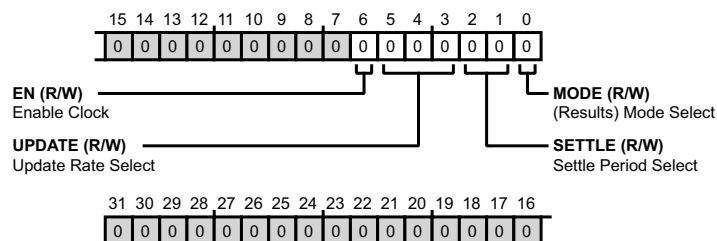
**Figure 9:** HAE\_CFG1 Register Diagram

**Table 8:** HAE\_CFG1 Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
14:0 (R/W)	STARTDIV	Start (Clock) Divider. The HAE_CFG1.STARTDIV bits provide the sample clock divider. Write the value to divide the main clock down to 8 kHz. The HAE sample loop is determined by SCLK / (HAE_CFG1.STARTDIV).

## Configuration 2 Register

The HAE\_CFG2 register enables and configures HAE operations as related to output results.



**Figure 10:** HAE\_CFG2 Register Diagram

**Table 9:** HAE\_CFG2 Register Fields

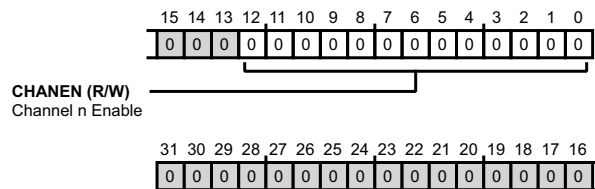
Bit No. (Access)	Bit Name	Description/Enumeration
6 (R/W)	EN	Enable Clock. The HAE_CFG2.EN bit enables the HAE the main clock, enabling HAE operation.
		0   Disable
		1   Enable
5:3 (R/W)	UPDATE	Update Rate Select. The HAE_CFG2.UPDATE bits determine the rate (in microseconds or milliseconds) at which the HAE updates the output results. The HAE_CFG2.UPDATE bits can also disable harmonic calculations.
		0   125 uSec
		1   250 uSec
		2   1 mSec
		3   16 mSec
		4   128 mSec
		5   512 mSec
		6   1024 mSec
		7   Disable
2:1 (R/W)	SETTLE	Settle Period Select. The HAE_CFG2.SETTLE bits determine the time (in milliseconds) that the HAE waits before producing results. The time is based on an 8K sample count.
		0   512 mSec
		1   768 mSec
		2   1024 mSec
		3   1280 mSec

**Table 9:** HAE\_CFG2 Register Fields (Continued)

Bit No. (Access)	Bit Name	Description/Enumeration
0 (R/W)	MODE	(Results) Mode Select. The HAE_CFG2.MODE bit determines whether the HAE produces results immediately or waits per the HAE_CFG2.SETTLE bit field.
		0 Results after Settle
		1 Results First

## Configuration 3 Register

The HAE\_CFG3 register configures HAE data transfer operations by selecting the fundamental and potential of twelve additional harmonic channels. The selected harmonics have their data (results) transferred to memory using DMA. First, the fundamental; followed by the selected channel n, in the order from the lowest to the highest numbered channel. Each selected channel has its eight result words transferred.



**Figure 11:** HAE\_CFG3 Register Diagram

**Table 10:** HAE\_CFG3 Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
12:0 (R/W)	CHANEN	Channel n Enable. Each HAE_CFG3.CHANEN bit enables the fundamental and potential of twelve additional harmonic data channels. The following enumerations apply to each bit. Bit 0 denotes the fundamental channel. Bits 1-12 denote the harmonic channels 1-12, accordingly.
		0 Disable channel n
		1 Enable channel n

## Configuration 4 Register

The HAE\_CFG4 register configures the internal digital integrator and high-pass filters (HPS) for HAE operations. The digital integrator is disabled and the filters are enabled by default.

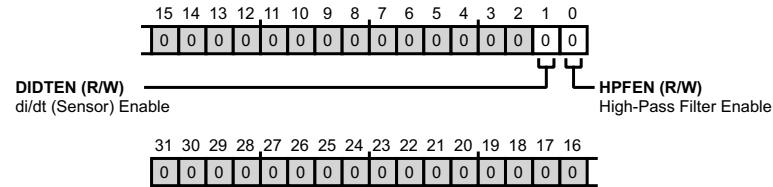


Figure 12: HAE\_CFG4 Register Diagram

Table 11: HAE\_CFG4 Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
1 (R/W)	DIDTEN	di/dt (Sensor) Enable. The HAE_CFG4.DIDTEN bit enables the internal digital integrator for a di/dt sensor. Set the bit (=1) if the sensor is a di/dt type sensor.
		0 Disable
		1 Enable
0 (R/W)	HPFEN	High-Pass Filter Enable. The HAE_CFG4.HPFEN bit enables the high-pass filters. Set the bit (=1) unless measuring DC levels.
		0 Disable
		1 Enable

DIDT Coefficient Register

The HAE\_DIDT\_COEF register sets the di/dt sensors coefficient.

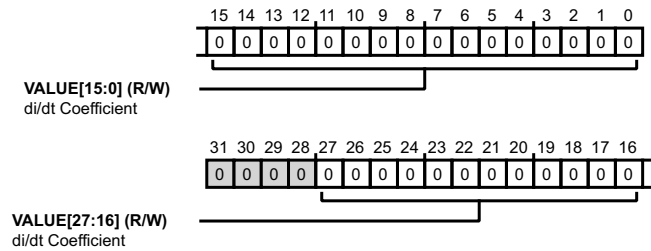


Figure 13: HAE\_DIDT\_COEF Register Diagram

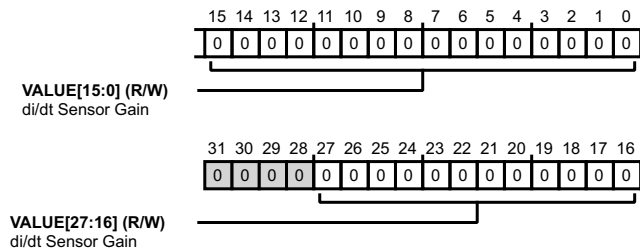


**Table 12:** HAE\_DIDT\_COEF Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
27:0 (R/W)	VALUE	di/dt Coefficient. The HAE_DIDT_COEF.VALUE bits provide the coefficient for a di/dt type sensor. If the HAE_CFG4.DIDTEN bit is set (=1), set this bit field to 28'FFFF000.

## DIDT Gain Register

The HAE\_DIDT\_GAIN register provides the di/dt sensors gain.



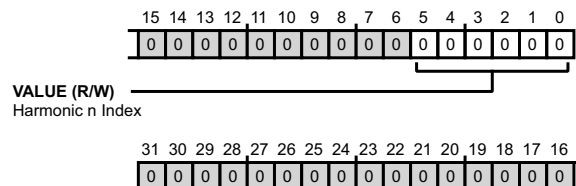
**Figure 14:** HAE\_DIDT\_GAIN Register Diagram

**Table 13:** HAE\_DIDT\_GAIN Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
27:0 (R/W)	VALUE	di/dt Sensor Gain. The HAE_DIDT_GAIN.VALUE bits provide the gain for a di/dt type sensor. If the HAE_CFG4.DIDTEN bit is set (=1), set this bit field to 0.

## Harmonic n Index Register

The HAE\_H[nn]\_INDX registers select harmonics for HAE operations. The fundamental always is provided. The harmonic results appear in the results RAM (HAE\_RESULTS\_RAM register) based on the index.



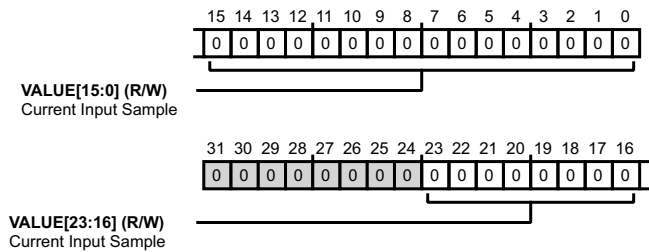
**Figure 15:** HAE\_H[nn]\_INDX Register Diagram

**Table 14:** HAE\_H[nn]\_INDX Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
5:0 (R/W)	VALUE	Harmonic n Index.  The HAE_H[nn]_INDX.VALUE bits select the harmonic channel n to analyze. Write the index of the harmonic into the HAE_H[nn]_INDX register for that harmonic to be selected.

I (Current) Sample Register

The HAE\_ISAMPLE register provides current (I) input samples for HAE calculations.



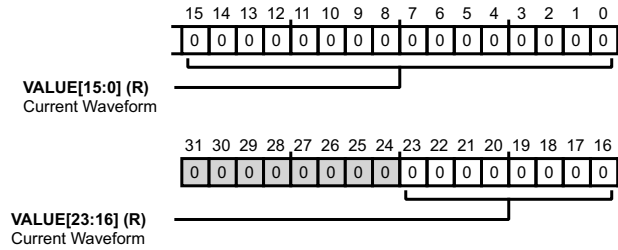
**Figure 16:** HAE\_ISAMPLE Register Diagram

**Table 15:** HAE\_ISAMPLE Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
23:0 (R/W)	VALUE	Current Input Sample.  The HAE_ISAMPLE.VALUE bits hold the current sample if provided by the MCU. The sample can be timed with the HAE_STAT.RXIRQ bit.

I (Current) Waveform Register

The HAE\_IWAVEFORM register holds processed current waveforms produced by the HAE. After some amplitude and phase delay, the waveform follows a sample input.



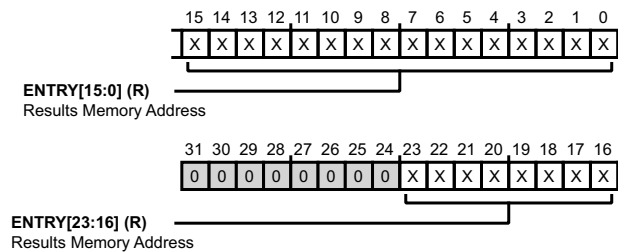
**Figure 17:** HAE\_IWAVEFORM Register Diagram

**Table 16:** HAE\_IWAVEFORM Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
23:0 (R/NW)	VALUE	Current Waveform. The HAE_IWAVEFORM.VALUE bits hold the processed current input sample.

## Results RAM Register

The HAE\_RESULTS\_RAM register points to a memory block holding HAE results.



**Figure 18:** HAE\_RESULTS\_RAM Register Diagram

**Table 17:** HAE\_RESULTS\_RAM Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
23:0 (R/NW)	ENTRY	Results Memory Address. The HAE_RESULTS_RAM.ENTRY bits hold the base pointer (address) for the HAE results RAM. The address range is x0c00 - 0x0dff. The size is 128 x 24 bits.

## Run Register

The HAE\_RUN register starts/idles HAE harmonic calculations.

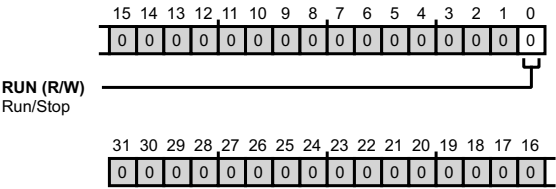


Figure 19: HAE\_RUN Register Diagram

Table 18: HAE\_RUN Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
0 (R/W)	RUN	Run/Stop. The HAE_RUN.RUN bit starts the HAE harmonic calculations.
		0 Stop
		1 Run

Status Register

The HAE\_STAT register indicates status for the HAE module and its interrupts.

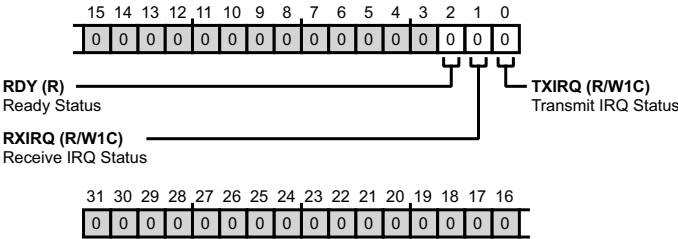


Figure 20: HAE\_STAT Register Diagram

Table 19: HAE\_STAT Register Fields

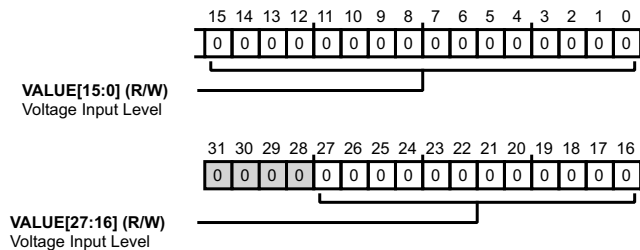
Bit No. (Access)	Bit Name	Description/Enumeration
2 (R/NW)	RDY	Ready Status. The HAE_STAT.RDY bit indicates status for the HAE. When the bit is set (=1), the HAE is fully accessible, following the setting of the HAE_CFG2.EN bit.
		0 No Status
		1 Ready

**Table 19:** HAE\_STAT Register Fields (Continued)

Bit No. (Access)	Bit Name	Description/Enumeration
1 (R/W1C)	RXIRQ	Receive IRQ Status. The HAE_STAT.RXIRQ bit indicates status for an HAE RX interrupt. The bit mirrors the HAE_CFG0.RXIRQEN bit status.
		0 No Status
		1 Interrupt Detected
0 (R/W1C)	TXIRQ	Transmit IRQ Status. The HAE_STAT.TXIRQ bit indicates status for an HAE TX interrupt. The bit mirrors the HAE_CFG0.TXIRQEN bit status.
		0 No Status
		1 Interrupt Detected

## Voltage Level Register

The HAE\_VLEVEL register is used to scale the fixed fundamental voltage level internally. This assists the HAE in locking onto the fundamental voltage channel.



**Figure 21:** HAE\_VLEVEL Register Diagram

**Table 20:** HAE\_VLEVEL Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
27:0 (R/W)	VALUE	Voltage Input Level. The HAE_VLEVEL.VALUE bits hold a value to scale the fixed fundamental voltage level. Use this formula: $VLEVEL = FS/V_{IN} * 5033168$ , where $V_{IN}$ is the fundamental voltage input level.

## V (Voltage) Sample Register

The HAE\_VSAMPLE register provides voltage (V) input samples for HAE calculations.

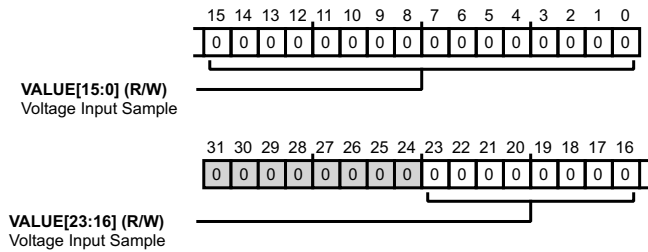


Figure 22: HAE\_VSAMPLE Register Diagram

Table 21: HAE\_VSAMPLE Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
23:0 (R/W)	VALUE	Voltage Input Sample. The HAE_VSAMPLE.VALUE bits hold the voltage sample if provided by the MCU. The sample can be timed with the HAE_STAT.RXIRQ bit.

V (Voltage) Waveform Register

The HAE\_VWAVEFORM register contains processed voltage waveforms produced by the HAE. After some amplitude and phase delay, the waveform follows a sample input.

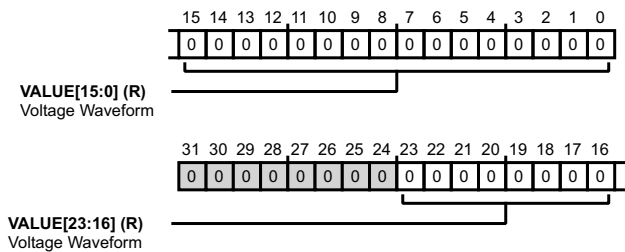


Figure 23: HAE\_VWAVEFORM Register Diagram

Table 22: HAE\_VWAVEFORM Register Fields

Bit No. (Access)	Bit Name	Description/Enumeration
23:0 (R/NW)	VALUE	Voltage Waveform. The HAE_VWAVEFORM.VALUE bits hold the processed voltage input sample.