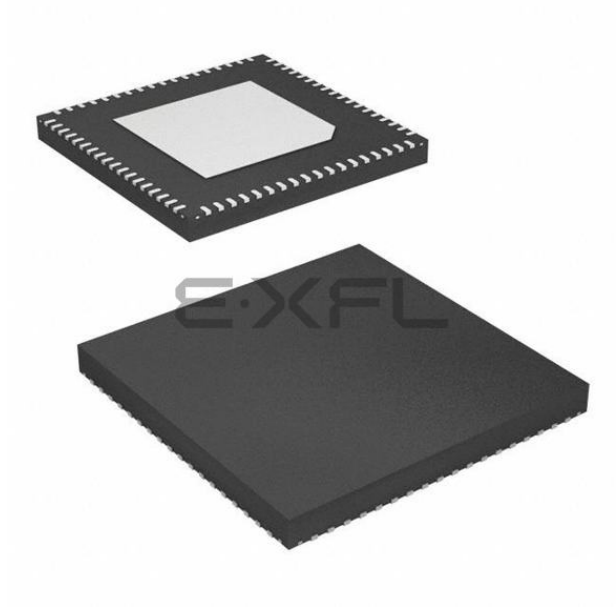

Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Sigma
Interface	I²C, SPI
Clock Rate	294.912MHz
Non-Volatile Memory	ROM (64kB)
On-Chip RAM	192kB
Voltage - I/O	3.30V
Voltage - Core	1.20V
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	72-VFQFN Exposed Pad, CSP
Supplier Device Package	72-LFCSP-VQ (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adau1462wbcpz300

Multipurpose Pins (MPx)

T_A = −40°C to +105°C, DVDD = 1.2 V ± 5%, IOVDD = 1.8 V − 5% to 3.3 V + 10%.

Table 9.

Parameter	Min	Max	Unit	Description
f_{MP}		24.576	MHz	MPx maximum switching rate when pin is configured as a general-purpose input or general-purpose output
t_{MPIL}	10 × t_{CORE}	6144 × t_{CORE}	sec	MPx pin input latency until high/low value is read by core; the duration in the Max column is equal to the period of one audio sample when the DSP is processing 6144 instructions per sample

S/PDIF Transmitter and Receiver

T_A = −40°C to +105°C, DVDD = 1.2 V ± 5%, IOVDD = 1.8 V − 10% to 3.3 V + 10%.

Table 10.

Parameter	Min	Max	Unit	Description
AUDIO SAMPLE RATE				
Transmitter	18	192	kHz	Audio sample rate of data output from S/PDIF transmitter
Receiver	18	192	kHz	Audio sample rate of data input to S/PDIF receiver

System Initialization Sequence

Before the IC can process the audio in the DSP, the following initialization sequence must be completed.

1. If possible, apply the required voltage to all four power supply domains (IOVDD, AVDD, PVDD, and DVDD) simultaneously. If simultaneous application is not possible, supply IOVDD first to prevent damage or reduced operating lifetime. If using the on-board regulator, AVDD and PVDD can be supplied in any order, and DVDD is then generated automatically. If not using the on-board regulator, AVDD, PVDD, and DVDD can be supplied in any order following IOVDD.
2. Start providing a master clock signal to the XTALIN/MCLK pin, or, if using the crystal oscillator, let the crystal oscillator start generating a master clock signal. The master clock signal must be valid when the DVDD supply stabilizes.
3. If the SELFBOOT pin is pulled high, a self boot sequence initiates on the master control port. Wait until the self boot operation is complete.
4. If SPI slave control mode is desired, toggle the SS/ADDR0 pin three times. Ensure that each toggle lasts at least the duration of one cycle of the master clock being input to the XTALIN/MCLK pin. When the SS/ADDR0 line rises for the third time, the slave control port is then in SPI mode.
5. Execute the register and memory write sequence that is required to configure the device in the proper operating mode.

Table 19 contains an example series of register writes used to configure the system at startup. The contents of the data column may vary depending on the system configuration. The configuration that is listed in Table 19 represents the default initialization sequence for project files generated in SigmaStudio.

Recommended Program/Parameter Loading Procedure

When writing large amounts of data to the program or parameter RAM in direct write mode (such as when downloading the initial contents of the RAMs from an external memory), use the hibernate register (Address 0xF400) to disable the processor core, thus preventing unpleasant noises from appearing at the audio output. When small amounts of data are transmitted during real-time operation of the DSP (such as when updating individual parameters), the software safeload mechanism can be used (see the Software Safeload section).

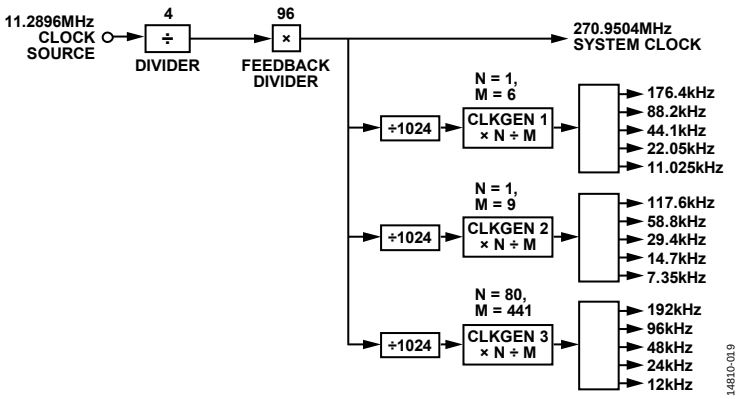

Figure 19. PLL and Audio Clock Generators with Default Settings and Resulting Clock Frequencies Labeled, XTALIN/MCLK = 11.2896 MHz

Figure 19 shows an example where the master clock input has a frequency of 11.2896 MHz, and the default settings are used for the PLL predivider, feedback divider, and Clock Generator 1 and Clock Generator 2. The resulting system clock is

$$11.2896 \text{ MHz} \div 4 \times 96 = 270.9504 \text{ MHz}$$

The base output of Clock Generator 1 is

$$270.9504 \text{ MHz} \div 1024 \times 1 \div 6 = 44.1 \text{ kHz}$$

The base output of Clock Generator 2 is

$$270.9504 \text{ MHz} \div 1024 \times 1 \div 9 = 29.4 \text{ kHz}$$

In this example, Clock Generator 3 is configured with N = 80 and M = 441; therefore, the resulting base output of Clock Generator 3 is

$$270.9504 \text{ MHz} \div 1024 \times 80 \div 441 = 48 \text{ kHz}$$

Master Clock Output

The master clock output pin (CLKOUT) is useful in cases where a master clock must be fed to other ICs in the system, such as audio codecs. The master clock output frequency is determined by the setting of the MCLK_OUT register (Address 0xF005). Four frequencies are possible: 1×, 2×, 4×, or 8× the frequency of the predivider output.

- The predivider output × 1 generates a 3.072 MHz output for a nominal system clock of 294.912 MHz.
- The predivider output × 2 generates a 6.144 MHz output for a nominal system clock of 294.912 MHz.
- The predivider output × 4 generates a 12.288 MHz output for a nominal system clock of 294.912 MHz.
- The predivider output × 8 generates a 24.576 MHz output for a nominal system clock of 294.912 MHz.

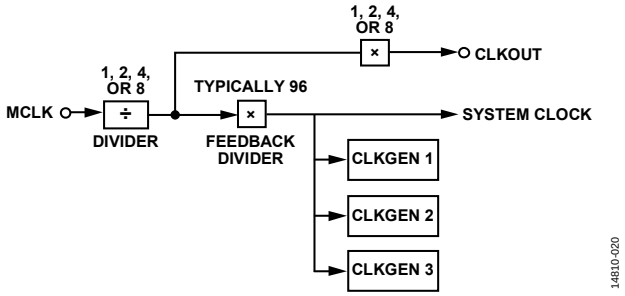

Figure 20. Clock Output Generator

The CLKOUT pin can drive more than one external slave IC if the drive strength is sufficient to drive the traces and external receiver circuitry. The ability to drive external ICs varies greatly, depending on the application and the characteristics of the PCB and the slave ICs. The drive strength and slew rate of the CLKOUT pin is configurable in the CLKOUT_PIN register (Address 0xF7A3); therefore, its performance can be tuned to match the specific application. The CLKOUT pin is not designed to drive long cables or other high impedance transmission lines. Use the CLKOUT pin only to drive signals to other integrated circuits on the same PCB. When changing the settings for the predivider, disable and then reenable the PLL using Register 0xF003 (PLL_ENABLE), allowing the frequency of the CLKOUT signal to update.

Dejitter Circuitry

To account for jitter between ICs in the system and to handle interfacing safely between internal and external clocks, dejitter circuits are included to guarantee that jitter related clocking errors are avoided. The dejitter circuitry is automated and does not require interaction or control from the user.

SLAVE CONTROL PORTS

A total of four control ports are available: two slave ports and two master ports. The slave I²C port and slave SPI port allow an external master device to modify the contents of the memory and registers. The master I²C port and master SPI port allow the device to self boot and to send control messages to slave devices on the same bus.

Slave Control Port Overview

To program the DSP and configure the control registers, a slave port is available that can communicate using either the I²C or SPI protocols. Any external device that controls the ADAU1462/ADAU1466, including a hardware interface used with SigmaStudio for development or a microcontroller in a large running system, uses the slave control port to communicate with the DSP. This port is unrelated to the master communications port that also uses the I²C or SPI protocols. The master port enables applications without an external controller and can read from an external EEPROM to self boot and control external ICs.

The slave communications port defaults to I²C mode; however, it can be put into SPI mode by toggling SS (SS/ADDR0), the slave select pin, from high to low three times. The slave select pin must be held low for at least one master clock period (that is, one period of the clock on the XTALIN_MCLK input pin). Only the PLL configuration registers (0xF000 to 0xF004) are accessible before the PLL locks. For this reason, always write to the PLL registers first after the chip powers up. After the PLL locks, the remaining registers and the RAM become accessible. See the System Initialization Sequence section for more information.

SLAVE CONTROL PORT ADDRESSING

Unlike earlier SigmaDSP processors, the ADAU1462/ADAU1466 slave control port 16-bit addressing cannot provide direct access to the total amount of memory available to the DSP core on its wider internal busses. Full read/write access to all memory and addressable registers is possible, but it must be accessed as two pages of memory in the slave control port address space. Page 0 is referred to as lower memory and Page 1 as upper memory. The single-bit register SECONDPAGE_ENABLE (0xF899) selects the active page.

Within a page, all addresses are accessible using both single address mode and burst mode. The first byte (Byte 0) of a control port write contains the 7-bit chip address plus the R/$\overline{W}$ bit. The next two bytes (Byte 1 and Byte 2) together form the subaddress of the register location within the memory maps of the ADAU1462/ADAU1466. This subaddress must be two bytes long because the memory locations within the devices are directly addressable, and their sizes exceed the range of single byte addressing. The third byte to the end of the sequence contain the data, such as control port data, program data, or parameter data. The number of bytes written per word depends on the type of data. For more information, see the Burst Mode Writing and Reading section. The ADAU1462/ADAU1466 must have a valid master clock to write to the slave control port, with the exception of the PLL configuration registers, 0xF000 to 0xF004.

If large blocks of data must be downloaded, halt the output of the DSP core (using Register 0xF400, HIBERNATE), load new data, and then restart the device (using Register 0xF402, START_CORE). This process is most common during the booting sequence at startup or when loading a new program into RAM because the ADAU1462/ADAU1466 has several mechanisms for updating signal processing parameters in real time without causing pops or clicks.

When updating a signal processing parameter while the DSP core is running, use the software safeload function. This function allows atomic writes to memory and prevents updates to parameters across the boundary of an audio frame, which can lead to an audio artifact such as a click or pop sound. For more information, see the Software Safeload section.

The slave control port supports either I²C or SPI, but not simultaneously. The function of each pin is described in Table 25 for the two modes.

Burst Mode Writing and Reading

Burst write and read modes are available for convenience when writing large amounts of data to contiguous registers. In these modes, the chip and memory addresses are written once, and then a large amount of data can follow uninterrupted. The subaddresses are automatically incremented at the word boundaries. This increment happens automatically after a single word write or read unless a stop condition is encountered (I²C mode) or the slave select is disabled and brought high (SPI mode). A burst write starts like a single word write, but, following the first data-word, the data-word for the next address can be written immediately without sending its 2-byte address. The control registers in the ADAU1462/ADAU1466 are two bytes wide, and the memories are four bytes wide. The auto-increment feature knows the word length at each subaddress; therefore, it is not necessary to manually specify the subaddress for each address in a burst write.

The subaddresses are automatically incremented by one address, following each read or write of a data-word, regardless of whether there is a valid register or RAM word at that address.

AUDIO SIGNAL ROUTING

A large number of audio inputs and outputs are available in the device, and control registers are available for configuring how the audio is routed between different functional blocks.

All input channels are accessible by both the DSP core and the ASRCs. Each ASRC can connect to a pair of audio channels from any of the input sources or from the DSP to ASRC channels of the DSP core. The serial outputs can obtain their data from a number of sources, including the DSP core, ASRCs, PDM microphones, S/PDIF receiver, or directly from the serial inputs.

See Figure 43 for an overview of the audio routing matrix with its available audio data connections.

To route audio to and from the DSP core, select the appropriate input and output cells in SigmaStudio. These cells can be found in the **IO** folder of the SigmaStudio algorithm toolbox.

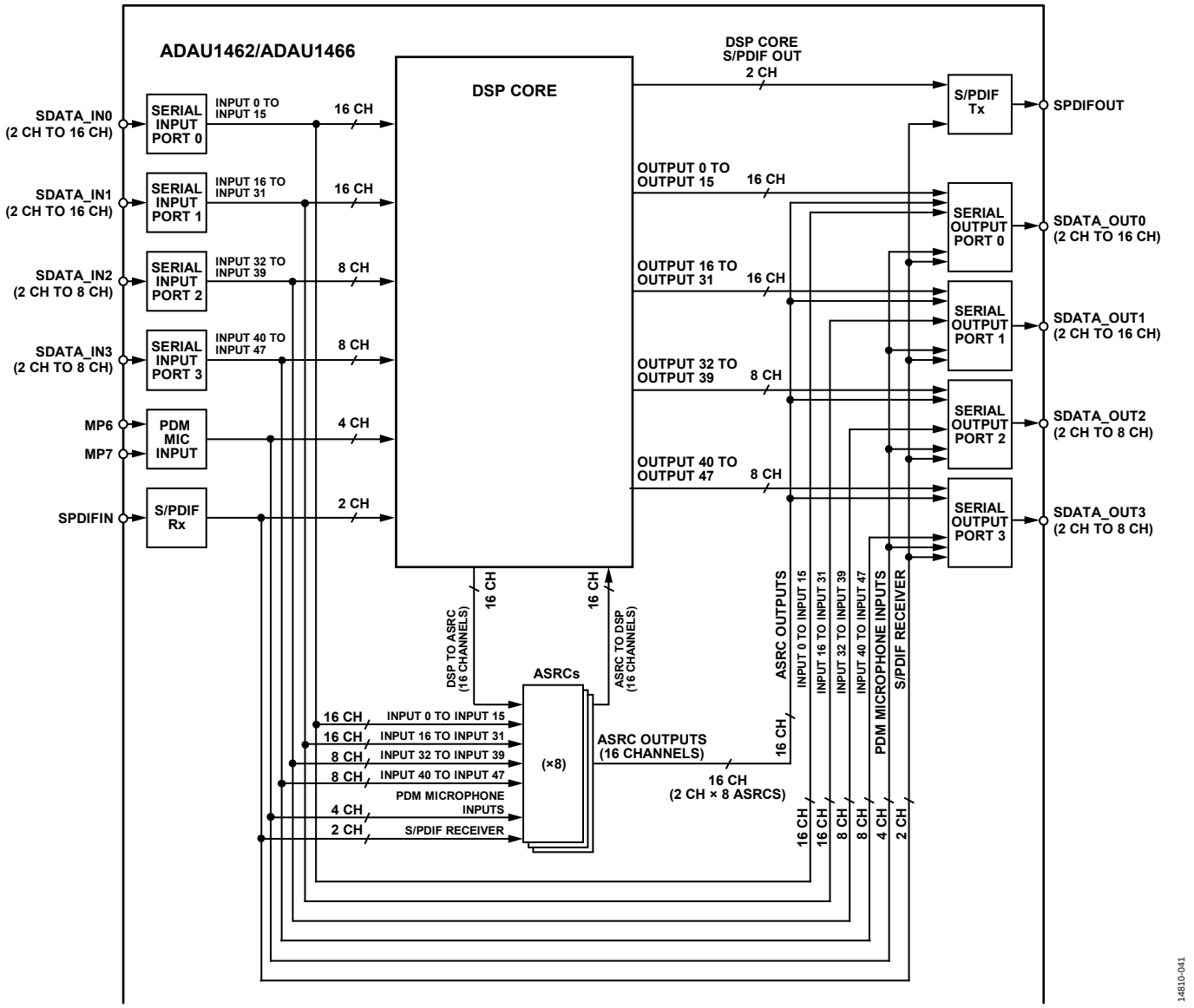

Figure 43. Audio Routing Overview

Serial Input Ports

There is a one to one mapping between the serial input ports and the audio input channels in the DSP and the ASRC input selectors, which is described in Table 42.

Table 42. Relationship Between Serial Input Port and Corresponding Channel Numbers on the DSP and ASRC Inputs

Serial Port	Audio Input Channels in the DSP and ASRC
Serial Input 0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
Serial Input 1	16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31
Serial Input 2	32, 33, 34, 35, 36, 37, 38, 39
Serial Input 3	40, 41, 42, 43, 44, 45, 46, 47

If a serial input port is configured using the SERIAL_BYTE_x_0 registers, Bits[2:0] (TDM_MODE) for a number of channels that is less than its maximum channel count, the unused channels carry zero data. For example, if Serial Input 0 is set in 8-channel (TDM8) mode, the first eight channels (Channel 0 to Channel 7) carry data, and the unused channels (Channel 8 to Channel 15) carry no data.

There are four options for the word length of each serial input port: 24 bits, 16 bits, 32 bits, or flexible TDM. The flexible TDM option is described in the Flexible TDM Input section.

In 32-bit mode (see Figure 67), the 32 bits received on the serial input are mapped directly to a 32-bit word in the DSP core. To use 32-bit mode, the 32-bit input cells must be used in SigmaStudio.

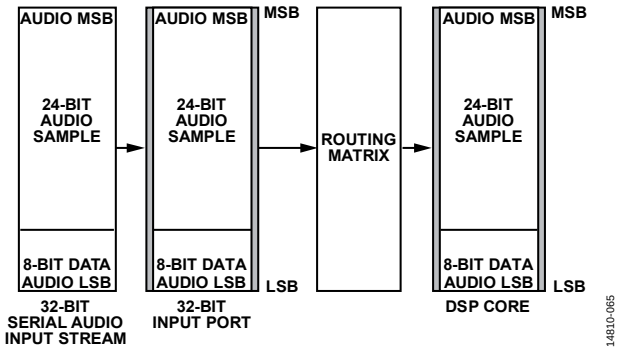

Figure 67. 32-Bit Serial Input Example

In 24-bit mode (see Figure 69), the 24-bit audio sample (in 1.23 format) is padded with eight zeros below its LSB (in 1.31 format) as it is input to the routing matrix. Then, the audio data is shifted such that the audio sample has 7 sign-extended zeros on top, 1 padded zero on the bottom, and 24 bits of data in the middle (8.24 format).

Whereas 16-bit mode is similar to 24-bit mode, the 16-bit audio data has 16 zeros below its LSB instead of just 8 zeros (in the 24-bit case). The resulting 8.24 sample, therefore, has 7 sign-extended zeros on top, 9 padded zeros on the bottom, and 16 bits of data in the middle (8.24 format).

Serial Output Ports

There is a one-to-one mapping between the serial output ports and the output audio channels in the DSP (see Table 43).

Table 43. Relationship Between Serial Input Port and Corresponding DSP Output Channel Numbers

Serial Input Port	Audio Output Channels from the DSP
Serial Output 0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
Serial Output 1	16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31
Serial Output 2	32, 33, 34, 35, 36, 37, 38, 39
Serial Output 3	40, 41, 42, 43, 44, 45, 46, 47

If a serial output port is configured using the SERIAL_BYTE_x_0 registers, Bits[2:0] (TDM_MODE), for a number of channels that is less than its maximum channel count, the unused channels are ignored. For example, if Serial Output Port 0 is set in 8-channel (TDM8) mode, and data is routed to it from the DSP, the first eight DSP output channels (Channel 0 through Channel 7) are output on SDATA_OUT0, but the remaining channels (Channel 8 through Channel 15) are not output from the device.

There are four options for the word length of each serial output port: 24 bits, 16 bits, 32 bits, or flexible TDM. See the Flexible TDM Output section for more information.

In 32-bit mode (see Figure 68), all 32 bits from the 8.24 word in the DSP core are copied directly to the serial output. To use 32-bit mode, the 32-bit output cells must be used in SigmaStudio.

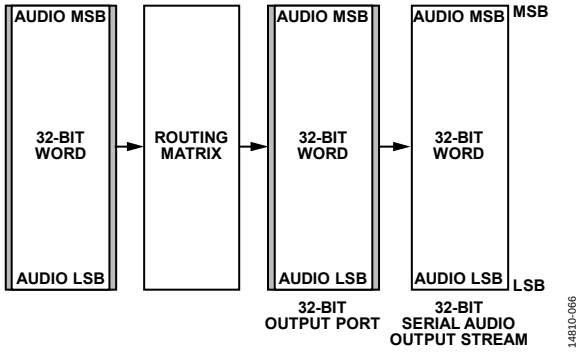

Figure 68. 32-Bit Serial Output Example

In 24-bit mode, the top 7 MSBs of the 8.24 audio word in the DSP core are saturated, and the resulting 1.23 word is output from the serial port, with 8 zeros padded under the LSB (see Figure 70).

In 16-bit mode, the top 7 MSBs of the 8.24 audio word in the DSP core are saturated, and the resulting 1.23 word is then truncated to a 1.15 word by removing the 8 LSBs. The resulting 1.15 word is then zero padded with 16 zeros under the LSB and output from the serial port.

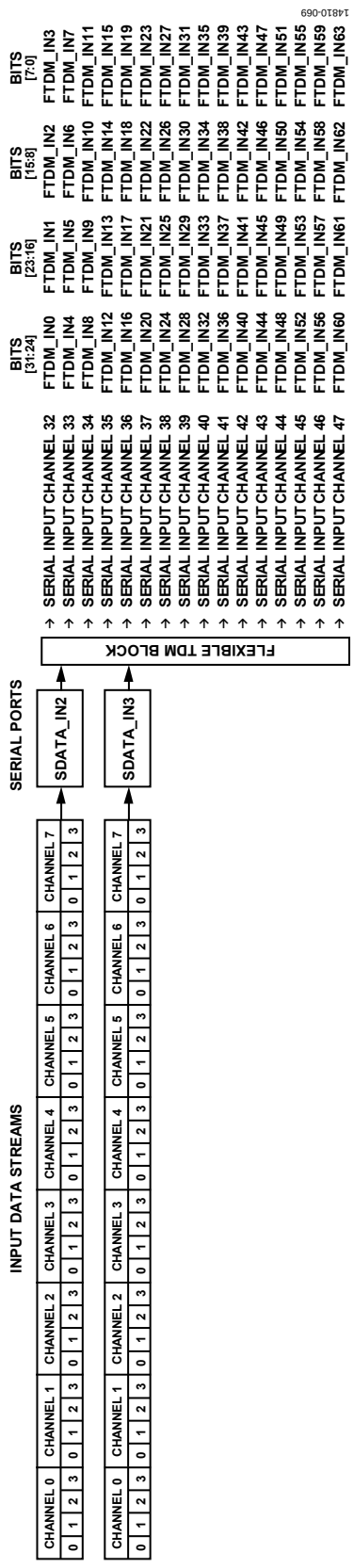

Figure 71. Flexible TDM Input Mapping

Address	Register	Description
0xF330	FTDM_IN48	FTDM mapping for the serial inputs (Channel 44, Bits[31:24])
0xF331	FTDM_IN49	FTDM mapping for the serial inputs (Channel 44, Bits[23:16])
0xF332	FTDM_IN50	FTDM mapping for the serial inputs (Channel 44, Bits[15:8])
0xF333	FTDM_IN51	FTDM mapping for the serial inputs (Channel 44, Bits[7:0])
0xF334	FTDM_IN52	FTDM mapping for the serial inputs (Channel 45, Bits[31:24])
0xF335	FTDM_IN53	FTDM mapping for the serial inputs (Channel 45, Bits[23:16])
0xF336	FTDM_IN54	FTDM mapping for the serial inputs (Channel 45, Bits[15:8])
0xF337	FTDM_IN55	FTDM mapping for the serial inputs (Channel 45, Bits[7:0])
0xF338	FTDM_IN56	FTDM mapping for the serial inputs (Channel 46, Bits[31:24])
0xF339	FTDM_IN57	FTDM mapping for the serial inputs (Channel 46, Bits[23:16])
0xF33A	FTDM_IN58	FTDM mapping for the serial inputs (Channel 46, Bits[15:8])
0xF33B	FTDM_IN59	FTDM mapping for the serial inputs (Channel 46, Bits[7:0])
0xF33C	FTDM_IN60	FTDM mapping for the serial inputs (Channel 47, Bits[31:24])
0xF33D	FTDM_IN61	FTDM mapping for the serial inputs (Channel 47, Bits[23:16])
0xF33E	FTDM_IN62	FTDM mapping for the serial inputs (Channel 47, Bits[15:8])
0xF33F	FTDM_IN63	FTDM mapping for the serial inputs (Channel 47, Bits[7:0])
0xF380	FTDM_OUT0	FTDM mapping for the serial outputs (Port 2, Channel 0, Bits[31:24])
0xF381	FTDM_OUT1	FTDM mapping for the serial outputs (Port 2, Channel 0, Bits[23:16])
0xF382	FTDM_OUT2	FTDM mapping for the serial outputs (Port 2, Channel 0, Bits[15:8])
0xF383	FTDM_OUT3	FTDM mapping for the serial outputs (Port 2, Channel 0, Bits[7:0])
0xF384	FTDM_OUT4	FTDM mapping for the serial outputs (Port 2, Channel 1, Bits[31:24])
0xF385	FTDM_OUT5	FTDM mapping for the serial outputs (Port 2, Channel 1, Bits[23:16])
0xF386	FTDM_OUT6	FTDM mapping for the serial outputs (Port 2, Channel 1, Bits[15:8])
0xF387	FTDM_OUT7	FTDM mapping for the serial outputs (Port 2, Channel 1, Bits[7:0])
0xF388	FTDM_OUT8	FTDM mapping for the serial outputs (Port 2, Channel 2, Bits[31:24])
0xF389	FTDM_OUT9	FTDM mapping for the serial outputs (Port 2, Channel 2, Bits[23:16])
0xF38A	FTDM_OUT10	FTDM mapping for the serial outputs (Port 2, Channel 2, Bits[15:8])
0xF38B	FTDM_OUT11	FTDM mapping for the serial outputs (Port 2, Channel 2, Bits[7:0])
0xF38C	FTDM_OUT12	FTDM mapping for the serial outputs (Port 2, Channel 3, Bits[31:24])
0xF38D	FTDM_OUT13	FTDM mapping for the serial outputs (Port 2, Channel 3, Bits[23:16])
0xF38E	FTDM_OUT14	FTDM mapping for the serial outputs (Port 2, Channel 3, Bits[15:8])
0xF38F	FTDM_OUT15	FTDM mapping for the serial outputs (Port 2, Channel 3, Bits[7:0])
0xF390	FTDM_OUT16	FTDM mapping for the serial outputs (Port 2, Channel 4, Bits[31:24])
0xF391	FTDM_OUT17	FTDM mapping for the serial outputs (Port 2, Channel 4, Bits[23:16])
0xF392	FTDM_OUT18	FTDM mapping for the serial outputs (Port 2, Channel 4, Bits[15:8])
0xF393	FTDM_OUT19	FTDM mapping for the serial outputs (Port 2, Channel 4, Bits[7:0])
0xF394	FTDM_OUT20	FTDM mapping for the serial outputs (Port 2, Channel 5, Bits[31:24])
0xF395	FTDM_OUT21	FTDM mapping for the serial outputs (Port 2, Channel 5, Bits[23:16])
0xF396	FTDM_OUT22	FTDM mapping for the serial outputs (Port 2, Channel 5, Bits[15:8])
0xF397	FTDM_OUT23	FTDM mapping for the serial outputs (Port 2, Channel 5, Bits[7:0])
0xF398	FTDM_OUT24	FTDM mapping for the serial outputs (Port 2, Channel 6, Bits[31:24])
0xF399	FTDM_OUT25	FTDM mapping for the serial outputs (Port 2, Channel 6, Bits[23:16])
0xF39A	FTDM_OUT26	FTDM mapping for the serial outputs (Port 2, Channel 6, Bits[15:8])
0xF39B	FTDM_OUT27	FTDM mapping for the serial outputs (Port 2, Channel 6, Bits[7:0])
0xF39C	FTDM_OUT28	FTDM mapping for the serial outputs (Port 2, Channel 7, Bits[31:24])
0xF39D	FTDM_OUT29	FTDM mapping for the serial outputs (Port 2, Channel 7, Bits[23:16])
0xF39E	FTDM_OUT30	FTDM mapping for the serial outputs (Port 2, Channel 7, Bits[15:8])
0xF39F	FTDM_OUT31	FTDM mapping for the serial outputs (Port 2, Channel 7, Bits[7:0])
0xF3A0	FTDM_OUT32	FTDM mapping for the serial outputs (Port 3, Channel 0, Bits[31:24])
0xF3A1	FTDM_OUT33	FTDM mapping for the serial outputs (Port 3, Channel 0, Bits[23:16])
0xF3A2	FTDM_OUT34	FTDM mapping for the serial outputs (Port 3, Channel 0, Bits[15:8])
0xF3A3	FTDM_OUT35	FTDM mapping for the serial outputs (Port 3, Channel 0, Bits[7:0])
0xF3A4	FTDM_OUT36	FTDM mapping for the serial outputs (Port 3, Channel 1, Bits[31:24])

The four multipliers are 64-bit double precision, capable of multiplying an 8.56 format number by an 8.24 number. The multiply accumulators consist of 16 registers, with a depth of 80 bits. The core can access RAM with a load/store width of 256 bits (eight 32-bit words per frame). The two ALUs have an 80-bit width and operate on numbers in 24.56 format. The 24.56-bit format provides more than 42 dB of headroom.

It is possible to create combinations of time domain and frequency domain processing, using block and sample frame interrupts. Sixteen data address generator (DAG) registers are available, and circular buffer addressing is possible.

Many of the signal processing functions are coded using full, 64-bit, double precision arithmetic. The serial port input and output word lengths are 24 bits; however, eight extra headroom bits are used in the processor to allow internal gains of up to 48 dB without clipping. Additional gains can be achieved by initially scaling down the input signal in the DSP signal flow.

Numeric Formats

DSP systems commonly use a standard numeric format. Fractional number systems are specified by an A.B format, where A is the number of bits to the left of the decimal point and B is the number of bits to the right of the decimal point.

The same numeric format is used for both the parameter and data values.

A digital clipper circuit is used within the DSP core before outputting to the serial port outputs, ASRCs, and S/PDIF. This circuit clips the top seven bits (and the least significant bit) of the signal to produce a 24-bit output with a range of +1.0 (minus 1 LSB) to −1.0. Figure 80 shows the maximum signal levels at each point in the data flow in both binary and decibel levels.

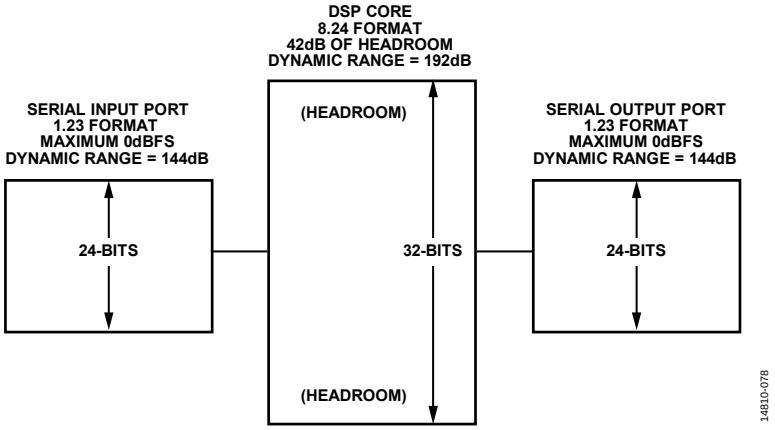

Figure 80. Signal Range for 1.23 Format (Serial Ports, ASRCs) and 8.24 Format (DSP Core)

Power Enable 1 Register

Address: 0xF051, Reset: 0x0000, Name: POWER_ENABLE1

For the purpose of power savings, this register allows the PDM microphone interfaces, S/PDIF interfaces, and auxiliary ADCs to be disabled when not in use. When these functional blocks are disabled, the current draw on the corresponding supply pins decreases.

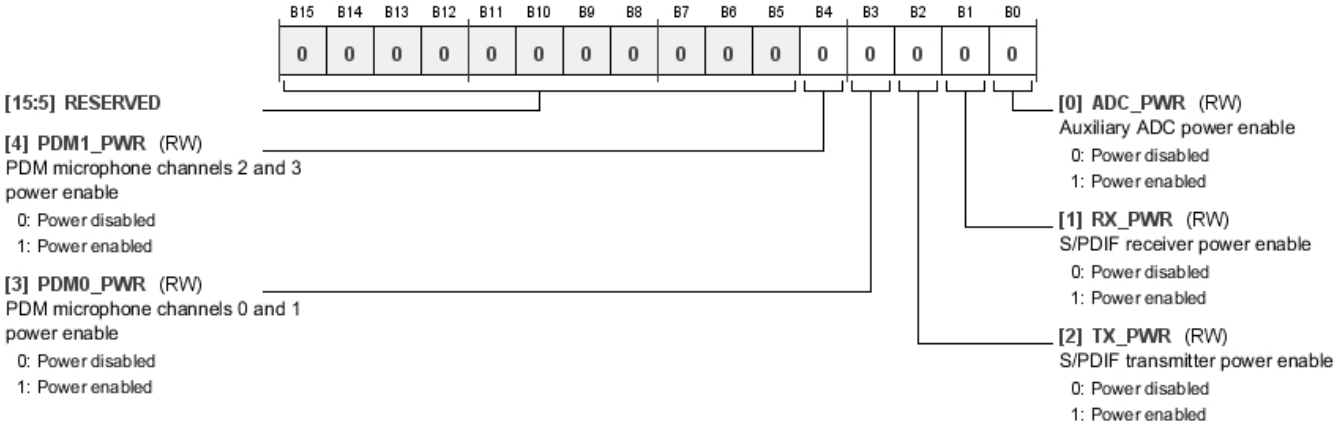

Table 75. Bit Descriptions for POWER_ENABLE1

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	RESERVED			0x0	RW
4	PDM1_PWR		PDM Microphone Channel 2 and PDM Microphone Channel 3 power enable. When this bit is disabled, PDM Microphone Channel 2 and PDM Microphone Channel 3 and their associated circuitry are disabled, and their data values cease to update.	0x0	RW
		0	Power disabled		
		1	Power enabled		
3	PDM0_PWR		PDM Microphone Channel 0 and PDM Microphone Channel 1 power enable. When this bit is disabled, PDM Microphone Channel 0 and PDM Microphone Channel 1 and their associated circuitry are disabled, and their data values cease to update.	0x0	RW
		0	Power disabled		
		1	Power enabled		
2	TX_PWR		S/PDIF transmitter power enable. This bit disables the S/PDIF transmitter circuit. Clock and data ceases to output from the S/PDIF transmitter pin, and the output is held at logic low as long as this bit is disabled.	0x0	RW
		0	Power disabled		
		1	Power enabled		
1	RX_PWR		S/PDIF receiver power enable. This bit disables the S/PDIF receiver circuit. Clock and data recovery from the S/PDIF input stream ceases until this bit is reenabled.	0x0	RW
		0	Power disabled		
		1	Power enabled		
0	ADC_PWR		Auxiliary ADC power enable. When this bit is disabled, the auxiliary ADCs are powered down, their outputs cease to update, and they hold their last value.	0x0	RW
		0	Power disabled		
		1	Power enabled		

SERIAL PORT CONFIGURATION REGISTERS

Serial Port Control 0 Register

Address: 0xF200 to 0xF21C (Increments of 0x4), Reset: 0x0000, Name: SERIAL_BYTE_x_0

These eight registers configure several settings for the corresponding serial input and serial output ports. Channel count, MSB position, data-word length, clock polarity, clock sources, and clock type are configured using these registers. On the input side, Register 0xF200 (SERIAL_BYTE_0_0) corresponds to SDATA_IN0; Register 0xF204 (SERIAL_BYTE_1_0) corresponds to SDATA_IN1; Register 0xF208 (SERIAL_BYTE_2_0) corresponds to SDATA_IN2; and Register 0xF20C (SERIAL_BYTE_3_0) corresponds to SDATA_IN3. On the output side, Register 0xF210 (SERIAL_BYTE_4_0) corresponds to SDATA_OUT0; Register 0xF214 (SERIAL_BYTE_5_0) corresponds to SDATA_OUT1; Register 0xF218 (SERIAL_BYTE_6_0) corresponds to SDATA_OUT2; and Register 0xF21C (SERIAL_BYTE_7_0) corresponds to SDATA_OUT3.

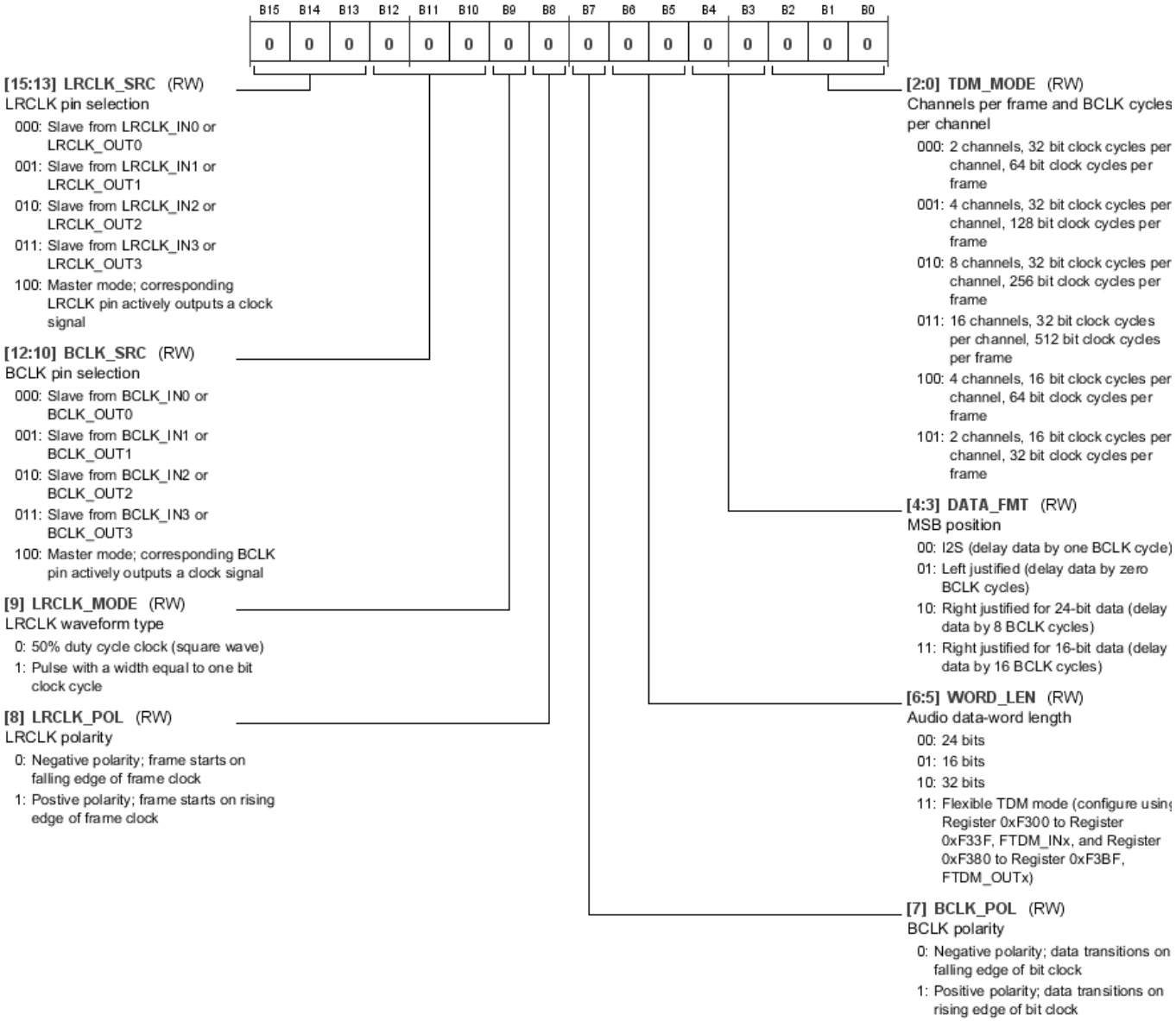

Table 85. Bit Descriptions for START_PULSE

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	RESERVED			0x0	RW
[4:0]	START_PULSE		Start pulse selection.	0x02	RW
		00000	Base sample rate ÷ 4 (12 kHz for 48 kHz base sample rate) (1/4 output of Clock Generator 1)		
		00001	Base sample rate ÷ 2 (24 kHz for 48 kHz base sample rate) (1/2 output of Clock Generator 1)		
		00010	Base sample rate (48 kHz for 48 kHz base sample rate) (×1 output of Clock Generator 1)		
		00011	Base sample rate × 2 (96 kHz for 48 kHz base sample rate) (×2 output of Clock Generator 1)		
		00100	Base sample rate × 4 (192 kHz for 48 kHz base sample rate) (×4 output of Clock Generator 1)		
		00101	Base sample rate ÷ 6 (8 kHz for 48 kHz base sample rate) (1/4 output of Clock Generator 2)		
		00110	Base sample rate ÷ 3 (16 kHz for 48 kHz base sample rate) (1/2 output of Clock Generator 2)		
		00111	2× base sample rate ÷ 3 (32 kHz for 48 kHz base sample rate) (×1 output of Clock Generator 2)		
		01000	Serial Input Port 0 sample rate (Register 0xF201 (SERIAL_BYTE_0_1), Bits[4:0])		
		01001	Serial Input Port 1 sample rate (Register 0xF205 (SERIAL_BYTE_1_1), Bits[4:0])		
		01010	Serial Input Port 2 sample rate (Register 0xF209 (SERIAL_BYTE_2_1), Bits[4:0])		
		01011	Serial Input Port 3 sample rate (Register 0xF20D (SERIAL_BYTE_3_1), Bits[4:0])		
		01100	Serial Output Port 0 sample rate (Register 0xF211 (SERIAL_BYTE_4_1), Bits[4:0])		
		01101	Serial Output Port 1 sample rate (Register 0xF215 (SERIAL_BYTE_5_1), Bits[4:0])		
		01110	Serial Output Port 2 sample rate (Register 0xF219 (SERIAL_BYTE_6_1), Bits[4:0])		
		01111	Serial Output Port 3 sample rate (Register 0xF21D (SERIAL_BYTE_7_1), Bits[4:0])		
		10000	S/PDIF receiver sample rate (derived from the S/PDIF input stream)		

Instruction to Start the Core Register

Address: 0xF402, Reset: 0x0000, Name: START_CORE

Enables the DSP core and initiates the program counter, which then begins incrementing through the program memory and executing instruction codes. This register is edge triggered, meaning that a rising edge on Bit 0 (START_CORE), that is, a transition from 0b0 to 0b1, initiates the program counter. A falling edge on Bit 0 (START_CORE), that is, a transition from 0b1 to 0b0, has no effect. To stop the DSP core, use Register 0xF400 (HIBERNATE), Bit 0 (HIBERNATE).

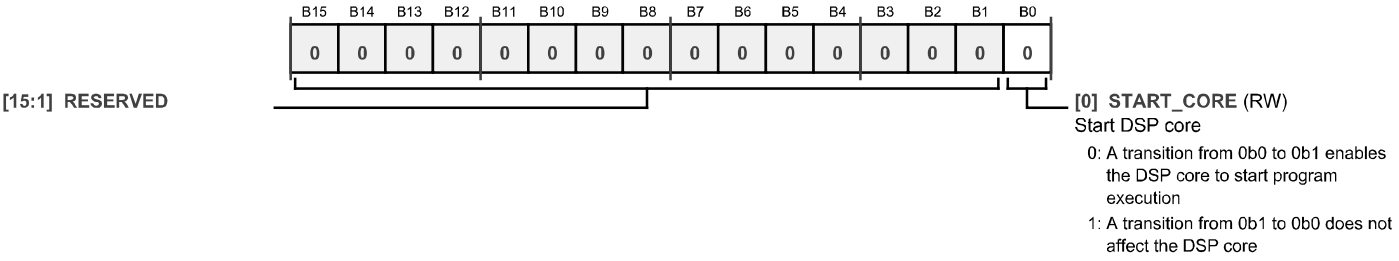

Table 86. Bit Descriptions for START_CORE

Bits	Bit Name	Settings	Description	Reset	Access
[15:1]	RESERVED			0x0	RW
0	START_CORE		A transition of this bit from 0b0 to 0b1 enables the DSP core to start executing its program. A transition from 0b1 to 0b0 does not affect the DSP core.	0x0	RW
		0	A transition from 0b0 to 0b1 enables the DSP core to start program execution		
		1	A transition from 0b1 to 0b0 does not affect the DSP core		

Core Status Register

Address: 0xF405, Reset: 0x0000, Name: CORE_STATUS

This read only register allows the user to check the status of the DSP core. To manually modify the core status, use Register 0xF400 (HIBERNATE), Register 0xF402 (START_CORE), and Register 0xF403 (KILL_CORE).

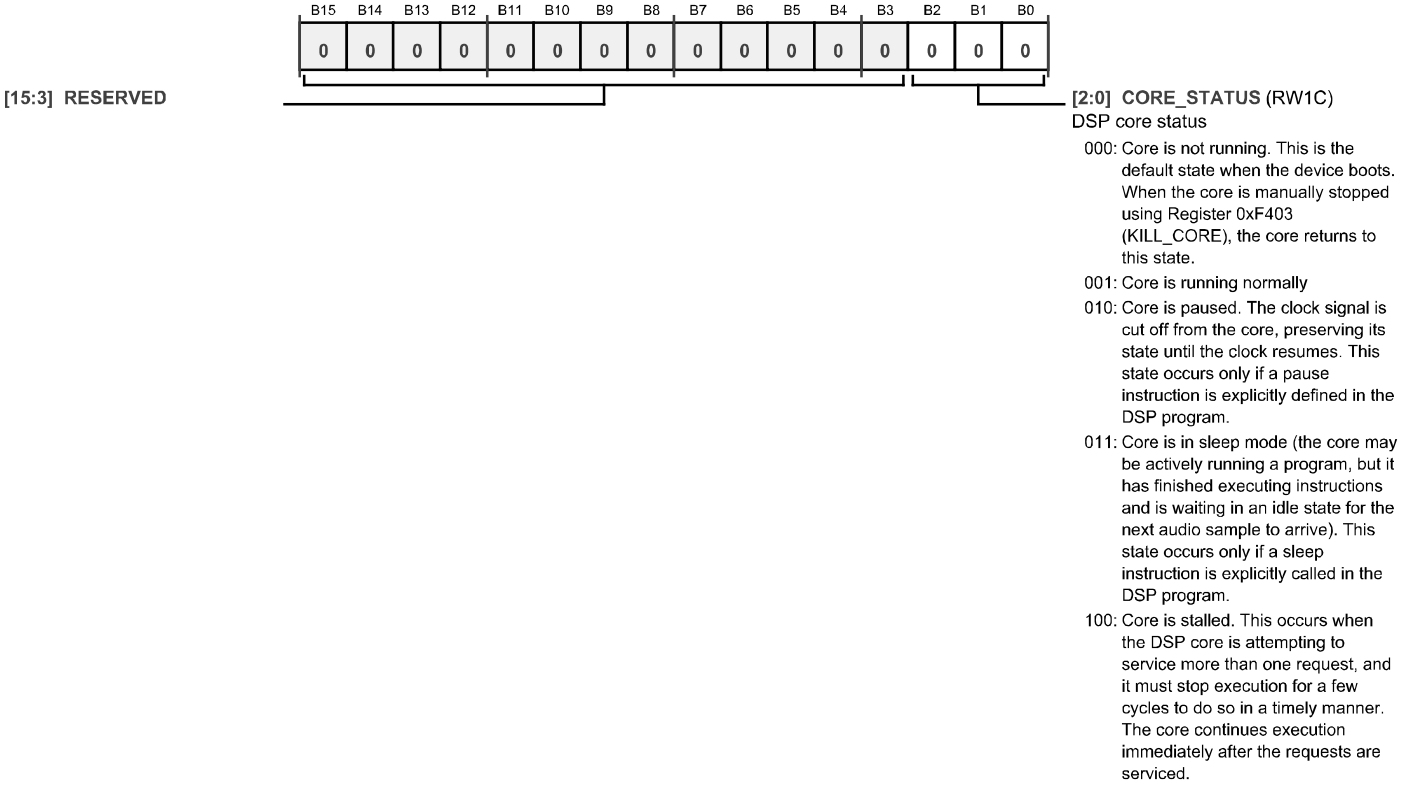

Table 89. Bit Descriptions for CORE_STATUS

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	RESERVED			0x0	RW
[2:0]	CORE_STATUS		DSP core status. These bits display the status of the DSP core at the moment the value is read.	0x0	RW
		000	Core is not running. This is the default state when the device boots. When the core is manually stopped using Register 0xF403 (KILL_CORE), the core returns to this state.		
		001	Core is running normally.		
		010	Core is paused. The clock signal is cut off from the core, preserving its state until the clock resumes. This state occurs only if a pause instruction is explicitly defined in the DSP program.		
		011	Core is in sleep mode (the core may be actively running a program, but it has finished executing instructions and is waiting in an idle state for the next audio sample to arrive). This state occurs only if a sleep instruction is explicitly called in the DSP program.		
		100	Core is stalled. This occurs when the DSP core is attempting to service more than one request, and it must stop execution for a few cycles to do so in a timely manner. The core continues execution immediately after the requests are serviced.		

Panic Parity Register

Address: 0xF422, Reset: 0x0003, Name: PANIC_PARITY_MASK

The panic manager checks and reports memory parity mask errors. Register 0xF422 (PANIC_PARITY_MASK) allows the user to configure which memories, if any, are subject to error reporting.

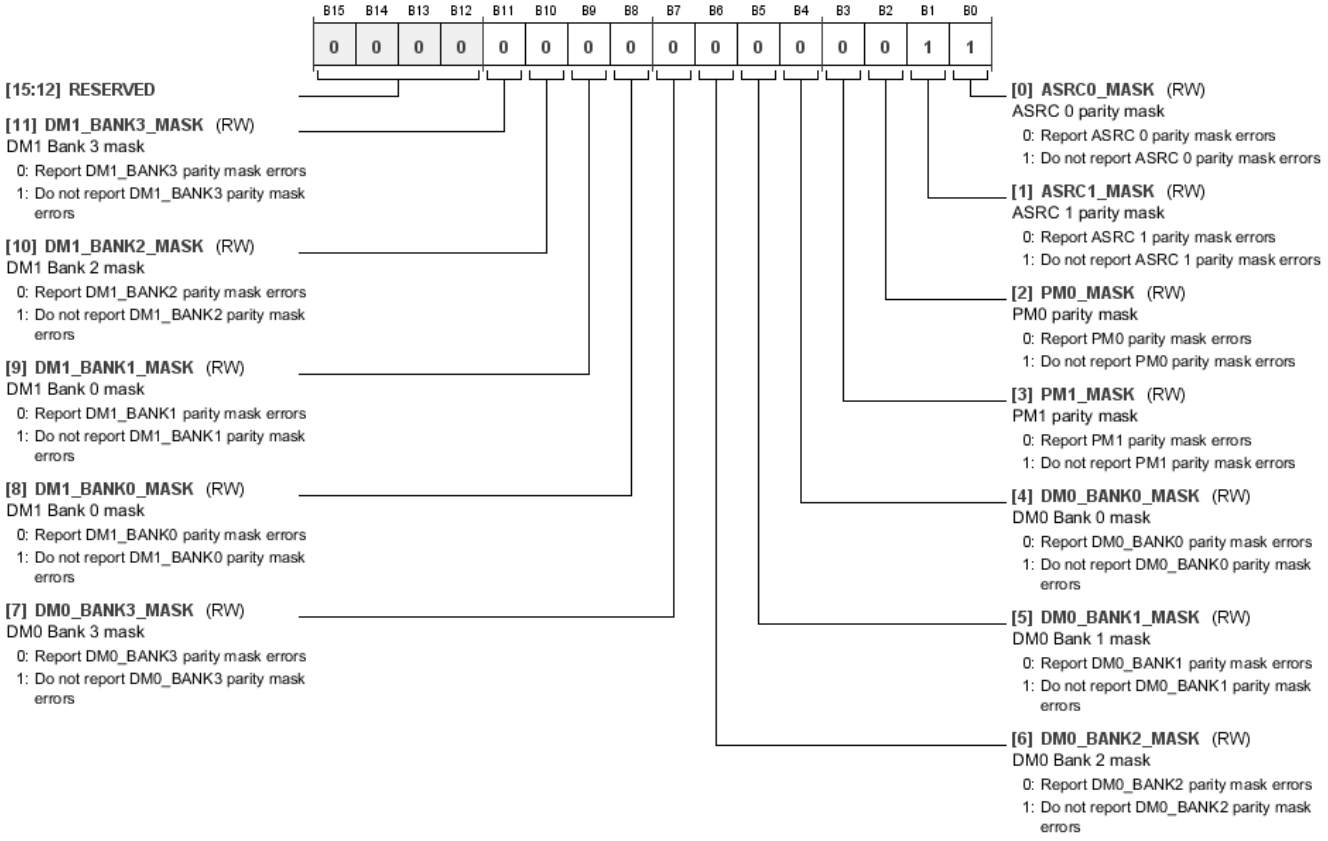

Table 91. Bit Descriptions for PANIC_PARITY_MASK

Bits	Bit Name	Settings	Description	Reset	Access
[15:12]	RESERVED			0x0	RW
11	DM1_BANK3_MASK		DM1 Bank 3 mask.	0x0	RW
		0	Report DM1_BANK3 parity mask errors		
		1	Do not report DM1_BANK3 parity mask errors		
10	DM1_BANK2_MASK		DM1 Bank 2 mask.	0x0	RW
		0	Report DM1_BANK2 parity mask errors		
		1	Do not report DM1_BANK2 parity mask errors		
9	DM1_BANK1_MASK		DM1 Bank 1 mask.	0x0	RW
		0	Report DM1_BANK1 parity mask errors		
		1	Do not report DM1_BANK1 parity mask errors		
8	DM1_BANK0_MASK		DM1 Bank 0 mask.	0x0	RW
		0	Report DM1_BANK0 parity mask errors		
		1	Do not report DM1_BANK0 parity mask errors		
7	DM0_BANK3_MASK		DM0 Bank 3 mask.	0x0	RW
		0	Report DM0_BANK3 parity mask errors		
		1	Do not report DM0_BANK3 parity mask errors		
6	DM0_BANK2_MASK		DM0 Bank 2 mask.	0x0	RW
		0	Report DM0_BANK2 parity mask errors		
		1	Do not report DM0_BANK2 parity mask errors		

DSP PROGRAM EXECUTION REGISTERS

Enable Block Interrupts Register

Address: 0xF450, Reset: 0x0000, Name: BLOCKINT_EN

This register enables block interrupts, which are necessary when frequency domain processing is required in the audio processing program. If block processing algorithms are used in SigmaStudio, SigmaStudio automatically sets this register accordingly. The user does not need to manually change the value of this register after SigmaStudio has configured it.

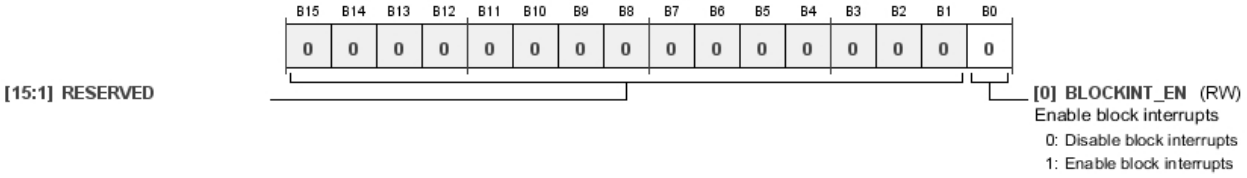

Table 103. Bit Descriptions for BLOCKINT_EN

Bits	Bit Name	Settings	Description	Reset	Access
[15:1]	RESERVED			0x0	RW
0	BLOCKINT_EN		Enable block interrupts.	0x0	RW
		0	Disable block interrupts		
		1	Enable block interrupts		

Value for the Block Interrupt Counter Register

Address: 0xF451, Reset: 0x0000, Name: BLOCKINT_VALUE

This 16-bit register controls the duration in audio frames of a block. A counter increments each time a new frame start pulse is received by the DSP core. When the counter reaches the value determined by this register, a block interrupt is generated and the counter is reset. If block processing algorithms are used in SigmaStudio, SigmaStudio automatically sets this register accordingly. The user does not need to manually change the value of this register after SigmaStudio has configured it.

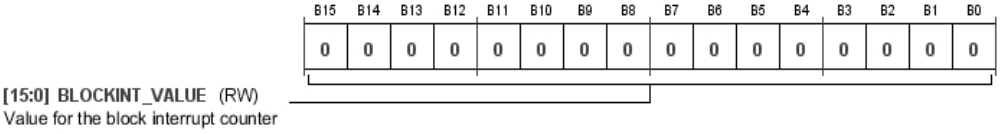

Table 104. Bit Descriptions for BLOCKINT_VALUE

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	BLOCKINT_VALUE		Value for the block interrupt counter.	0x0000	RW

Program Counter, Bits[23:16] Register

Address: 0xF460, Reset: 0x0000, Name: PROG_CNTR0

This register, in combination with Register 0xF461 (PROG_CNTR1), stores the current value of the program counter.

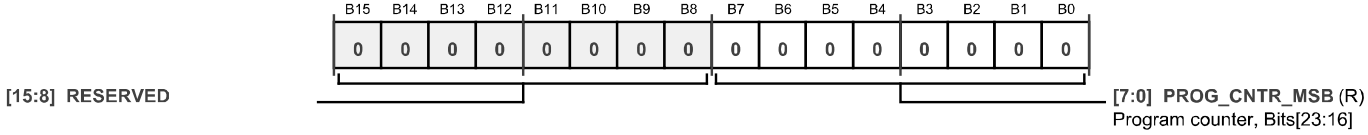

Table 105. Bit Descriptions for PROG_CNTR0

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	RESERVED			0x0	RW
[7:0]	PROG_CNTR_MSB		Program counter, Bits[23:16].	0x00	R

Bits	Bit Name	Settings	Description	Reset	Access
11	PM_BANK1_SUBBANK3_MASK		Bank 1 Subbank 3 mask.	0x0	RW
		0	Report Bank 1 Subbank 3 parity errors		
		1	Ignore Bank 1 Subbank 3 parity errors		
10	PM_BANK1_SUBBANK2_MASK		Bank 1 Subbank 2 mask.	0x0	RW
		0	Report Bank 1 Subbank 2 parity errors		
		1	Ignore Bank 1 Subbank 2 parity errors		
9	PM_BANK1_SUBBANK1_MASK		Bank 1 Subbank 1 mask.	0x0	RW
		0	Report Bank 1 Subbank 1 parity errors		
		1	Ignore Bank 1 Subbank 1 parity errors		
8	PM_BANK1_SUBBANK0_MASK		Bank 1 Subbank 0 mask.	0x0	RW
		0	Report Bank 1 Subbank 0 parity errors		
		1	Ignore Bank 1 Subbank 0 parity errors		
[7:6]	RESERVED		Reserved.	0x0	RW
5	PM_BANK0_SUBBANK5_MASK		Bank 0 Subbank 5 mask.	0x0	RW
		0	Report Bank 0 Subbank 5 parity errors		
		1	Ignore Bank 0 Subbank 5 parity errors		
4	PM_BANK0_SUBBANK4_MASK		Bank 0 Subbank 4 mask.	0x0	RW
		0	Report Bank 0 Subbank 4 parity errors		
		1	Ignore Bank 0 Subbank 4 parity errors		
3	PM_BANK0_SUBBANK3_MASK		Bank 0 Subbank 3 mask.	0x0	RW
		0	Report Bank 0 Subbank 3 parity errors		
		1	Ignore Bank 0 Subbank 3 parity errors		
2	PM_BANK0_SUBBANK2_MASK		Bank 0 Subbank 2 mask.	0x0	RW
		0	Report Bank 0 Subbank 2 parity errors		
		1	Ignore Bank 0 Subbank 2 parity errors		
1	PM_BANK0_SUBBANK1_MASK		Bank 0 Subbank 1 mask.	0x0	RW
		0	Report Bank 0 Subbank 1 parity errors		
		1	Ignore Bank 0 Subbank 1 parity errors		
0	PM_BANK0_SUBBANK0_MASK		Bank 0 Subbank 0 mask.	0x0	RW
		0	Report Bank 0 Subbank 0 parity errors		
		1	Ignore Bank 0 Subbank 0 parity errors		

Table 120. Bit Descriptions for PANIC_CODE4

Bits	Bit Name	Settings	Description	Reset	Access
[15:13]	RESERVED		Reserved.	0x0	RW
12	ERR_DM1B3SB4		Error in Bank 3 Subbank 4.	0x0	R
		0	No error in Bank 3 Subbank 4		
		1	Error in Bank 3 Subbank 4		
11	ERR_DM1B3SB3		Error in Bank 3 Subbank 3.	0x0	R
		0	No error in Bank 3 Subbank 3		
		1	Error in Bank 3 Subbank 3		
10	ERR_DM1B3SB2		Error in Bank 3 Subbank 2.	0x0	R
		0	No error in Bank 3 Subbank 2		
		1	Error in Bank 3 Subbank 2		
9	ERR_DM1B3SB1		Error in Bank 3 Subbank 1.	0x0	R
		0	No error in Bank 3 Subbank 1		
		1	Error in Bank 3 Subbank 1		
8	ERR_DM1B3SB0		Error in Bank 3 Subbank 0.	0x0	R
		0	No error in Bank 3 Subbank 0		
		1	Error in Bank 3 Subbank 0		
[7:5]	RESERVED		Reserved.	0x0	RW
4	ERR_DM1B2SB4		Error in Bank 2 Subbank 4.	0x0	R
		0	No error in Bank 2 Subbank 4		
		1	Error in Bank 2 Subbank 4		
3	ERR_DM1B2SB3		Error in Bank 2 Subbank 3.	0x0	R
		0	No error in Bank 2 Subbank 3		
		1	Error in Bank 2 Subbank 3		
2	ERR_DM1B2SB2		Error in Bank 2 Subbank 2.	0x0	R
		0	No error in Bank 2 Subbank 2		
		1	Error in Bank 2 Subbank 2		
1	ERR_DM1B2SB1		Error in Bank 2 Subbank 1.	0x0	R
		0	No error in Bank 2 Subbank 1		
		1	Error in Bank 2 Subbank 1		
0	ERR_DM1B2SB0		Error in Bank 2 Subbank 0.	0x0	R
		0	No error in Bank 2 Subbank 0		
		1	Error in Bank 2 Subbank 0		

Digital PDM Microphone Control Register

Address: 0xF560 to 0xF561 (Increments of 0x1), Reset: 0x4000, Name: DMIC_CTRLx

These registers configure the digital PDM microphone interface. Two registers are used to control up to four PDM microphones: Register 0xF560 (DMIC_CTRL0) configures PDM Microphone Channel 0 and PDM Microphone Channel 1, and Register 0xF561 (DMIC_CTRL1) configures PDM Microphone Channel 2 and PDM Microphone Channel 3.

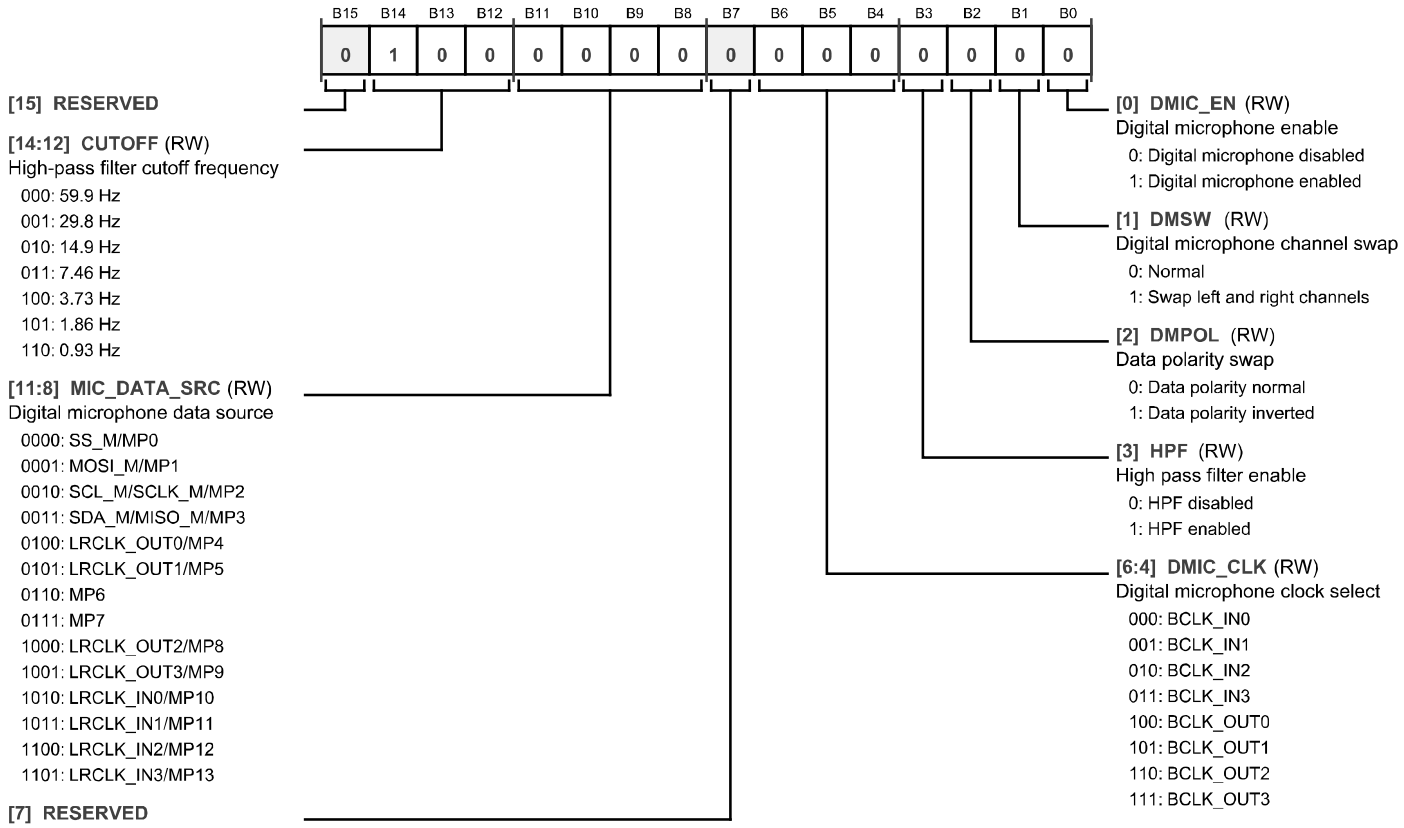

Table 125. Bit Descriptions for DMIC_CTRLx

Bits	Bit Name	Settings	Description	Reset	Access
15	RESERVED			0x0	RW
[14:12]	CUTOFF		High-pass filter cutoff frequency. These bits configure the cutoff frequency of an optional high-pass filter designed to remove dc components from the microphone data signal(s). To use these bits, Bit 3 (HPF), must be enabled.	0x4	RW
		000	59.9 Hz		
		001	29.8 Hz		
		010	14.9 Hz		
		011	7.46 Hz		
		100	3.73 Hz		
		101	1.86 Hz		
		110	0.93 Hz		

S/PDIF Receiver Auxiliary Outputs Enable Register

Address: 0xF608, Reset: 0x0000, Name: SPDIF_AUX_EN

The S/PDIF receiver on the ADAU1466 and ADAU1462 decodes embedded nonaudio data bits on the incoming data stream, including channel status, user data, validity bits, and parity bits. This information, together with the decoded audio data, can optionally be output on one of the SDATA_OUTx pins using Register 0xF608 (SPDIF_AUX_EN). The serial output port selected by Bits[3:0] (TDMOUT) outputs an 8-channel TDM stream containing this decoded information.

Channel 0 in the TDM8 stream contains the 24 audio bits from the left S/PDIF input channel, followed by eight zero bits.

Channel 1 in the TDM8 stream contains 20 zero bits, the parity bit, validity bit, user data bit, and the channel status bit from the left S/PDIF input channel, followed by eight zero bits.

Channel 2 in the TDM8 stream contains 22 zero bits, followed by the compression type bit (0b0 represents AC3 and 0b1 represents DTS) and the audio type bit (0b0 represents PCM and 0b1 represents compressed), followed by eight zero bits.

Channel 3 in the TDM8 stream contains 32 zero bits.

Channel 4 in the TDM8 stream contains the 24 audio bits from the right S/PDIF input channel, followed by eight zero bits.

Channel 5 in the TDM8 stream contains 20 zero bits followed by the parity bit, validity bit, user data bit, and channel status bit from the right S/PDIF input channel, followed by eight zero bits.

Channel 6 in the TDM8 stream contains 32 zero bits.

Channel 7 in the TDM8 stream contains 23 zero bits, the block start bit, and eight zero bits.

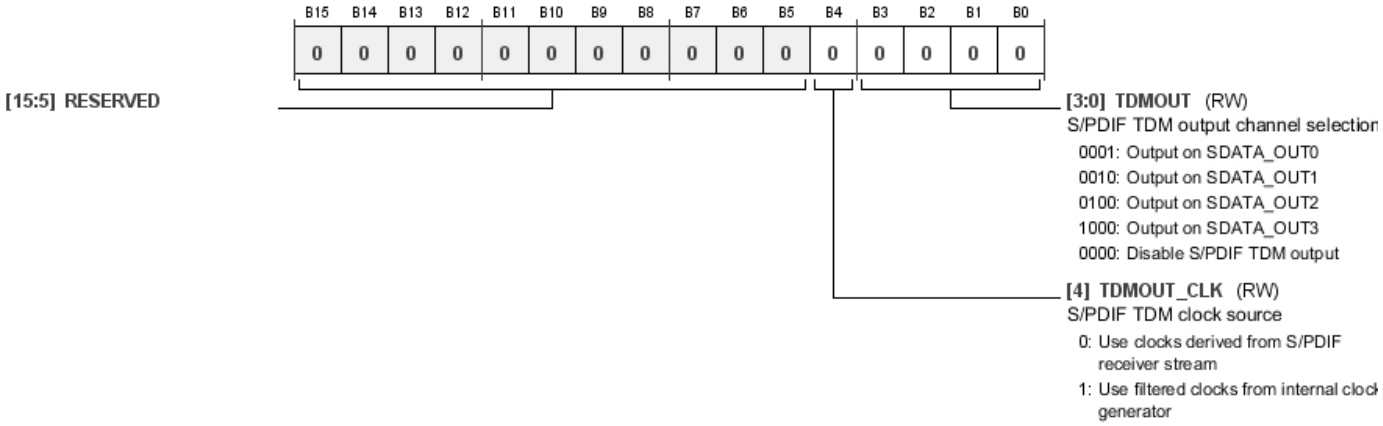

Table 138. Bit Descriptions for SPDIF_AUX_EN

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	RESERVED			0x0	RW
4	TDMOUT_CLK		S/PDIF TDM clock source. When Bits[3:0] (TDMOUT) are configured to output S/PDIF receiver data on one of the SDATA_OUTx pins, the corresponding serial port must be set in master mode; and Bit 4 (TDMOUT_CLK) configures which clock signals are used on the corresponding BCLK_OUTx and LRCLK_OUTx pins. If Bit 4 (TDMOUT_CLK) = 0b0, the clock signals recovered from the S/PDIF input signal are used to clock the serial output. If Bit 4 (TDMOUT_CLK) = 0b1, the output of Clock Generator 3 is used to clock serial output; and Register 0xF026 (CLK_GEN3_SRC), Bits[3:0] (FREF_PIN), must be 0b1110, and Register 0xF026 (CLK_GEN3_SRC), Bit 4 (CLK_GEN3_SRC), must be 0b1.	0x0	RW
		0	Use clocks derived from S/PDIF receiver stream		
		1	Use filtered clocks from internal clock generator		
[3:0]	TDMOUT		S/PDIF TDM output channel selection.	0x0	RW
		0001	Output on SDATA_OUT0		
		0010	Output on SDATA_OUT1		
		0100	Output on SDATA_OUT2		
		1000	Output on SDATA_OUT3		
		0000	Disable S/PDIF TDM output		

CLKOUT Pin Drive Strength and Slew Rate Register

Address: 0xF7A3, Reset: 0x0008, Name: CLKOUT_PIN

This register configures the drive strength, slew rate, and pull resistors for the CLKOUT pin.

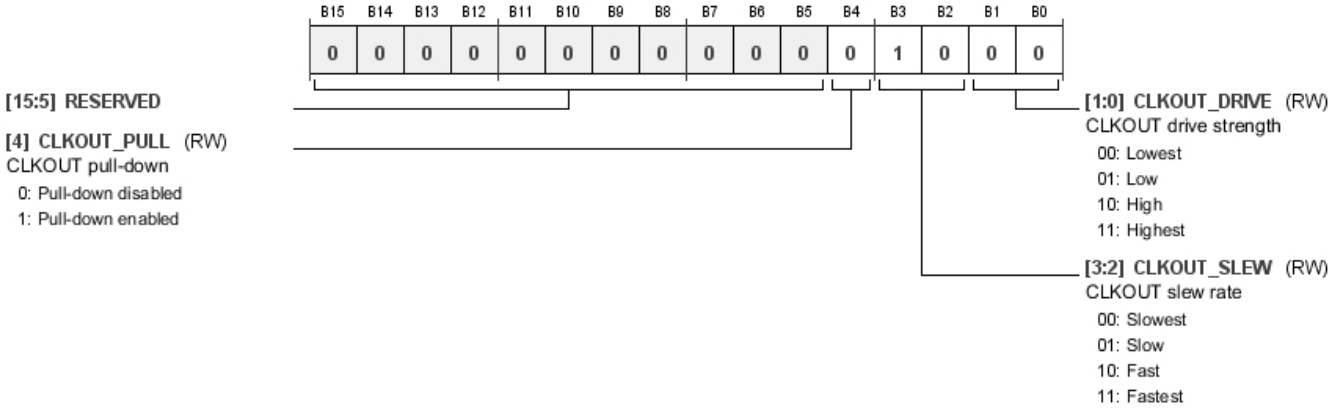

Table 176. Bit Descriptions for CLKOUT_PIN

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	RESERVED			0x0	RW
4	CLKOUT_PULL		CLKOUT pull-down.	0x0	RW
		0	Pull-down disabled		
		1	Pull-down enabled		
[3:2]	CLKOUT_SLEW		CLKOUT slew rate.	0x2	RW
		00	Slowest		
		01	Slow		
		10	Fast		
		11	Fastest		
[1:0]	CLKOUT_DRIVE		CLKOUT drive strength.	0x0	RW
		00	Lowest		
		01	Low		
		10	High		
		11	Highest		