DS125DF1610 Programming Guide

Reference Manual for Register Configuration

Texas Instruments

ti.com/sigcon

2014

Table of Contents

Revision Table

Revision Number	Date	Description/Comments
1.0	April 30, 2014	Initial release.
1.1	Aug 21, 2014	Corrected Typos in Rate/Subrate table

SMBus Slave Mode Operation

SMBus Address Configuration

The DS125DF1610 can be configured to respond to one of the sixteen SMBus addresses listed in Table 1. GPIO1 and GPIO0 are configured to be four level inputs immediately after the device powers on. Logic 0 can be set by tying the pin to ground through a 1kΩ or smaller resistor. Logic R is set by tying the pin to ground through a 20kΩ resistor. Logic F is set by floating the pin. Logic 1 is set by tying the pin to VCC = 2.5V through a 1kΩ or smaller resistor.

ADDR1 (pin D5)	ADDR0 (pin B6)	Address
0	0	7'h18
0	R	7'h19
0	F	7'h1A
0	1	7'h1B
R	0	7'h1C
R	R	7'h1D
R	F	7'h1E
R	1	7'h1F
F	0	7'h20
F	R	7'h21
F	F	7'h22
F	1	7'h23
1	0	7'h24
1	R	7'h25
1	F	7'h26
1	1	7'h27

READ_EN

The READ_EN pin should be pulled low when the DS125DF1610 is used in SMBus slave mode.

Note: this pin has an internal pull down.

When in SMBus slave mode, if this pin is pulled high it will force the device's address to be 7'h18.

Register Types

The DS125DF1610 register set has three types of registers:

1. Global Registers
2. Shared Registers
3. Channel Registers

Global registers can be accessed from the shared register page and also the channel register pages. There are four global registers in the DS125DF1610:

1. Register 0xFC
2. Register 0xFD
3. Register 0xFE
4. Register 0xFF

Registers 0xFC and 0xFD are used to select the channel registers to be written to. To select a channel write a 1 to its corresponding bit in these global registers. Note more than one channel may be written to by setting multiple bits in registers 0xFC and 0xFD. However, when performing an SMBus read transaction only one channel can be selected at a time. If multiple channels are selected when attempting to perform an SMBus read, the device will return 0x00.

Global Register	Bit	Description	CDR/TX Pin Assignment
0xFD	7	Channel 15– Quad 3 Channel 3– Cross Point Ch D	TX_7B
	6	Channel 14– Quad 3 Channel 2 – Cross Point Ch C	TX_7A
	5	Channel 13– Quad 3 Channel 1 – Cross Point Ch B	TX_6B
	4	Channel 12– Quad 3 Channel 0 – Cross Point Ch A	TX_6A
	3	Channel 11– Quad 2 Channel 3– Cross Point Ch D	TX_5B
	2	Channel 10– Quad 2 Channel 2 – Cross Point Ch C	TX_5A
	1	Channel 9– Quad 2 Channel 1 – Cross Point Ch B	TX_4B
	0	Channel 8– Quad 2 Channel 0 – Cross Point Ch A	TX_4A
0xFC	7	Channel 7– Quad 1 Channel 3– Cross Point Ch D	TX_3B
	6	Channel 6– Quad 1 Channel 2 – Cross Point Ch C	TX_3A
	5	Channel 5– Quad 1 Channel 1 – Cross Point Ch B	TX_2B
	4	Channel 4– Quad 1 Channel 0 – Cross Point Ch A	TX_2A
	3	Channel 3– Quad 0 Channel 3 – Cross Point Ch D	TX_1B
	2	Channel 2– Quad 0 Channel 2 – Cross Point Ch C	TX_1A
	1	Channel 1– Quad 0 Channel 1 – Cross Point Ch B	TX_0B
	0	Channel 0 – Quad 0 Channel 0 – Cross Point Ch A	TX_0A

Register 0xFE will always read back 0x03. This is useful to verify that there is a good SMBus connection between the SMBus master and the DS125DF1610.

Global Register	Description
0xFE	Read only register, contains value 0x03

Register 0xFF bit 0 is used select the shared register page or the channel register page for the channels selected in registers 0xFC and 0xFD.

Global Register	Write Value	Mask
0xFF	1: Broadcast write to all channels, 0xFF[0] must be set to 1. 0: Normal operation, select channel registers as defined in 0xFC and 0xFD.	0x02
	1: Select Channel Registers 0: Select Share Registers	0x01

Register 0xFF bit 1 can be used to perform broadcast register writes to all channels. Users can perform single channel read-modify broadcast write type commands by setting register 0xFF to 0x03 and selecting a single channel in the 0xFC or 0xFD registers. The example below shows how a user can configure the device to read back values from channel 15 (ch 7B) and then broadcast write back to all channels.

Global Register	Write Value	Mask
0xFC	0x00	0xFF
0xFD	0x80	0xFF
0xFF	0x03	0x03

1. Shared Register Commands

The commands listed in this section involve manipulating the shared registers to configure device level features of the DS125DF1610. For information on how to configure channel specific features see section 2) Channel Register Commands.

1.1. Resetting the Shared Registers

The shared registers can be reset to their default values by writing to bit 6 in share register 0x04. This bit is self-clearing, so once it is set high to initiate a shared register reset it is automatically cleared.

Share Register	Write Value	Mask
0x04	0x40	0x40

1.2. Limiting the Number of Channels Allowed to Lock Simultaneously

By default, DS125DF1610 is programmed to allow 8 channels to lock simultaneously. Since lock acquisition draw more power than when a channel is locked it is often desirable to limit the number of channels that are allowed to lock at any given time. To adjust this limit write to share register 0x05, bits 3-0.

Share Register	Write Value	Mask
0x05	Desired max # of channels to lock simultaneously	0x0F

1.3. Prevent Unused Channels from Attempting to Acquire Lock

If it is known that some channels in the DS125DF1610 will be unused, these unused channels can be programmed such that they will never attempt to acquire lock. This feature is useful in systems that experience high levels of crosstalk on the receiver inputs. In some cases large amounts of cross talk can trigger the signal detect circuit of a victim channel, which causes the victim channel to attempt to lock to the noise.

To prevent these unused channels from attempting to acquire lock, write a 0 to the unused channels specified in shared registers 0x0F and 0x10. By default registers 0x0F and 0x10 are set to 0xFF, which means that all channels are allowed to attempt lock if a signal is detected.

Share Register	Bit	Write Value
0x0F	7	Channel 15 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	6	Channel 14 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	5	Channel 13 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	4	Channel 12 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	3	Channel 11 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	2	Channel 10 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	1	Channel 9 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	0	Channel 8 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock

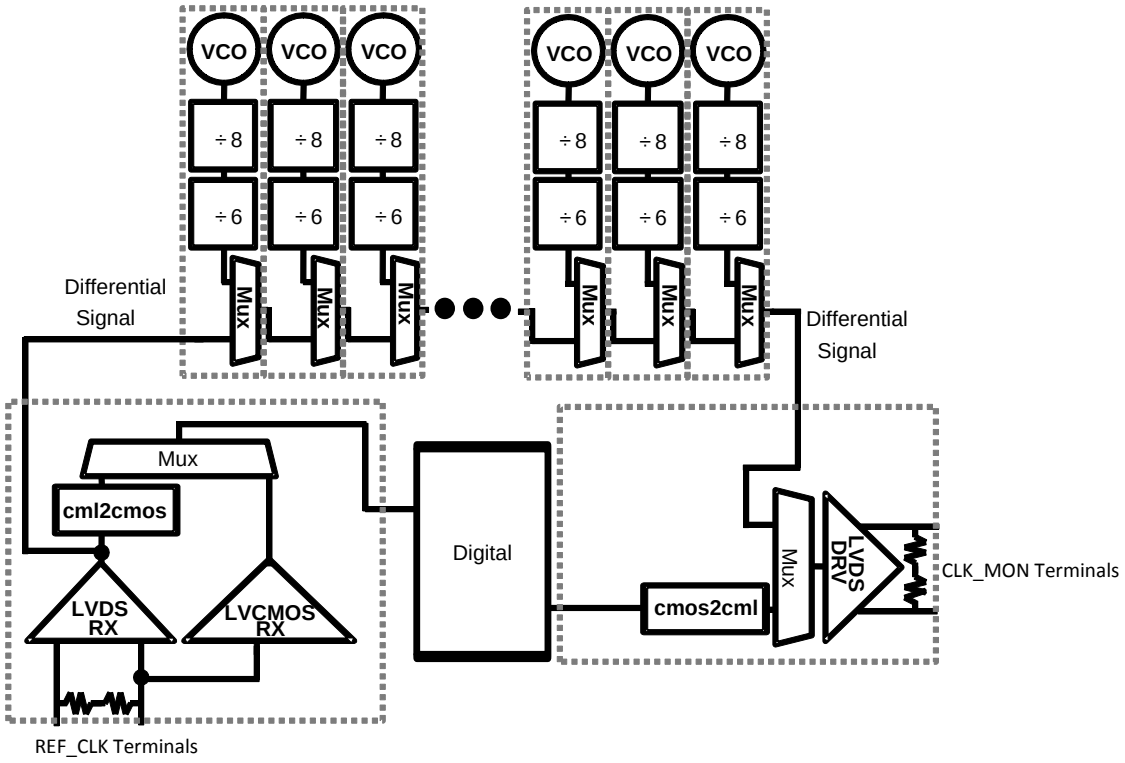
Share Register	Bit	Write Value
0x10	7	Channel 7 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	6	Channel 6 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	5	Channel 5 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	4	Channel 4 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	3	Channel 3 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	2	Channel 2 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	1	Channel 1 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock
	0	Channel 0 1: Allow channel to attempt lock if signal is detected 0: Prevent channel from acquiring lock

1.4. Reference Clock Daisy Chaining with CLK_MON pins

The DS125DF1610 is able to output a copy of the input reference clock through the CLK_MON pins. This is useful for applications that have multiple DS125DF1610 devices on a single PCB and a clock fan out buffer is not used. When using the CLK_MON terminals to daisy chain a reference clock across multiple devices it is recommended to use the divided clock output so that a 25 MHz clock is transmitted out of the first device in the chain.

Looping the reference clock through the device can be done in two ways:

1. Differential path chained through each channel's analog circuitry
2. Single-ended path through the shared digital circuitry of the device



If the path through analog is chosen, a differential reference clock must be supplied to the REF_CLK terminals. If the path through the digital is chosen either a differential or single-ended clock can be applied to the REF_CLK terminals.

Note: The individual channel's VCO divided clock is not available for use.

The registers used for looping the reference clock out through the CLK_MON terminals are shown below.

Shared Register	Bit	Description
0x04	7	1: Reference clock going to CLK_MON terminals is the same frequency as the divided version used by the channels 0: Undivided reference clock goes out to CLK_MON terminals

Shared Register	Bit	Description
0x0A	7	1: Select reference clock from the digital path 0: Select reference clock from the differential analog path(requires that all channels have reference clock mux enabled)
	0	1: CLK_MON outputs are powered down 0: CLK_MON outputs are enabled

Channel Register	Bit	Description
0x8E	5	1: Enable the reference clock mux within the channel(this bit must be set in all 16 channels if the analog path is chosen for looping the reference clock out through the CLK_MON terminals). 0: Disable the reference clock mux

An example set of register writes for enabling each path is shown below.

Path Through the Analog			
Channel Register	**Write Value**	**Mask**	**Description**
0xFF	**0x03**	**0x03**	Write to all channels, read from channel 1 (0A)
0xFD	**0x01**	**0xFF**	
0xFC	**0x00**	**0xFF**	
0x8E	**0x20**	**0x20**	Enable the reference clock mux in all channels
0xFF	**0x00**	**0x03**	Select share registers
0x0A	**0x00**	**0x81**	Select the analog loop through path and enable the CLK_MON outputs
0x04	**0x80**	**0x80**	Output the divided reference clock

Path Through the Digital			
Channel Register	**Write Value**	**Mask**	**Description**
0xFF	**0x00**	**0x03**	Select share registers
0x0A	**0x80**	**0x81**	Select the analog loop through path and enable the CLK_MON outputs
0x04	**0x80**	**0x80**	Output the divided reference clock

1.5. Reference Clock Detect

A reference clock is required for normal device operation. The DS125DF1610 device reports if a signal is present on the reference clock inputs in shared register 0x0B[6].

Shared Register	Bit	Description
0x0B	6	1: Reference clock is detected 0: No reference clock detected

1.6. Selecting the Reference Clock Divider

By default, the DS125DF1610 is configured to expect a 25 MHz differential reference clock. The internal reference clock divider can be adjusted to allow the DS125DF1610 to also accept either 125MHz or 312.5 MHz reference clocks. To select one of these alternate reference clock frequencies write to shared register 0x02 bits 6 and 5.

Share Register	Bit	Write Value
0x02	6 5	2'b00: 25 MHz reference clock (default) 2'b01: 125 MHz reference clock 2'b10: 312.5MHz reference clock 2'b11: reserved

Example write to select 312.5 MHz as the reference clock frequency:

Channel Register	Write Value	Mask	Description
0x02	0x40	0x60	Select 312.5 MHz for the reference clock frequency

2. Channel Register Commands

The commands listed in this section involve manipulating the channel registers to configure channel specific features of the DS125DF1610. For information on how to configure device level features see the Shared Register Commands section.

2.1. Resetting Channel Registers

The channel register set for individual channels can be reset to their default values by writing to bit 3 in channel register 0x00. This bit is self-clearing, so once it is set high to initiate a channel register reset it is automatically cleared.

Channel Register	Bit	Description
0x00	3	1: Reset channel registers, bit is self clearing 0: Normal operation

2.2. Signal Detect

The signal detect circuit monitors the energy level on the receiver inputs and powers on or off the rest of the high speed data path if a signal is detected. By default, each channel allows the signal detect circuit to automatically power on or off the rest of the high speed data path depending on if a signal is present. Users can obtain manual control over the signal detect block through the SMBus channel registers. This can be useful if it is desired manually force channels to be disabled.

The status of the signal detect circuit can be read back in two channel register locations:

Channel Register	Bit	Description
0x01	7	1: Signal is detected 0: No Signal detected
0x78	5	1: Signal is detected 0: No Signal detected

Typically, users will read back register 0x78, because 0x78[4] is the CDR lock status bit. So users can perform a single register read to get the CDR lock and signal detect status.

2.3. Force Signal Detect

Users can force the signal detect circuit to manually enable or disable channels. The force enable signal is located at channel register 0x14[7]. The force disable signal is located at channel register 0x14[6]. These signals should not be set at the same time.

Channel Register	Bit	Description
0x14	7	1: Signal is forced HIGH, channel is enabled 0: Normal operation
	6	1: Signal is detected 0: No Signal detected

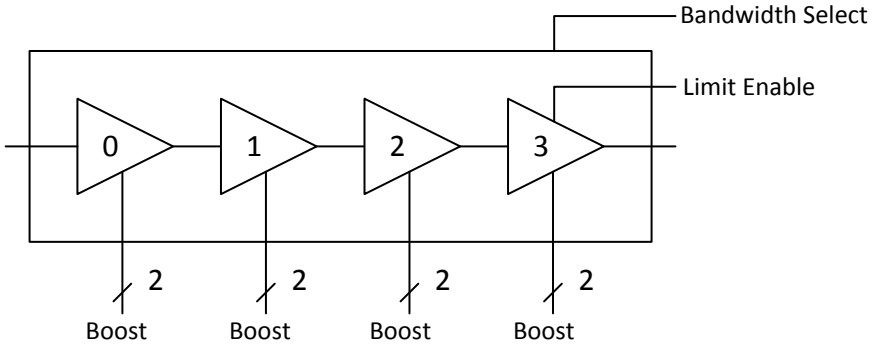
2.4. Signal Detect Threshold Adjustments

The signal detect assert and de-assert threshold can be adjusted individually in channel register 0x14. Adjusting these thresholds will change the amount of energy required to assert or de-assert signal detect. Since this circuit operates by detecting energy at the high speed input pins examples of corresponding voltage levels by data pattern are shown below for 10.3125 Gbps operation:

Channel Register	Bit	Description
0x14	5:4	Differential Signal Detect Assert Thresholds 00: 50mVpp (PRBS-31), 103mVpp (1T) **(default)** 01: default -10mVpp 10: default +10mVpp 11: default +15mVpp
	3:2	Signal Detect Assert Thresholds 00: 44mVpp (PRBS-31), 67mVpp (1T) **(default)** 01: default -10mVpp 10: default +10mVpp 11: default +15mVpp

2.5. CTLE Boost Curves

The CTLE consists of 4 stages with 2-bit boost control per stage. The overall boost is typically determined by summing the boost level of the 4 stages. The various combinations of boost per stage that sum to the same number, result in slightly different shapes to the EQ curve (transfer function).



There are 256 possible boost curves per bandwidth setting. The equalizer has three bandwidth options as described in the CTLE Bandwidth Select section, Full Rate – FR, Half Rate – HR, and Mid Rate - MR. These settings are shown on the following pages. Settings highlighted in gray are default settings in the device. I

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-0-0-0 FR	0.29	2.5
0-0-0-1 FR	3.69	5.5
0-1-0-0 FR	4.04	6
0-0-1-0 FR	4.06	6
1-0-0-0 FR	4.55	6.5
0-0-0-2 FR	5.57	4.5
0-0-2-0 FR	6.01	5
0-2-0-0 FR	6.03	5
2-0-0-0 FR	6.68	5.5
0-0-0-3 FR	6.94	4.5
0-0-3-0 FR	7.46	5
0-3-0-0 FR	7.47	4.5
0-1-0-1 FR	8.04	6.5
0-0-1-1 FR	8.09	6.5
3-0-0-0 FR	8.26	5.5
0-1-1-0 FR	8.51	6.5
1-0-0-1 FR	8.60	7
1-1-0-0 FR	9.05	7
1-0-1-0 FR	9.17	7
0-0-1-2 FR	9.62	5.5
0-1-0-2 FR	9.63	5.5
0-2-0-1 FR	9.83	5.5
0-0-2-1 FR	9.86	5.5
1-0-0-2 FR	10.05	6
0-2-1-0 FR	10.14	5.5
0-1-2-0 FR	10.20	6
1-2-0-0 FR	10.60	6
2-0-0-1 FR	10.63	6
1-0-2-0 FR	10.70	6
0-0-1-3 FR	10.90	5.5
0-1-0-3 FR	10.91	5.5
2-1-0-0 FR	11.01	6.5
2-0-1-0 FR	11.10	6.5
0-3-0-1 FR	11.17	5
0-0-3-1 FR	11.26	5.5
1-0-0-3 FR	11.32	5.5
0-3-1-0 FR	11.48	5.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-1-3-0 FR	11.57	5.5
0-0-2-2 FR	11.63	5
0-2-0-2 FR	11.65	5
1-3-0-0 FR	11.89	5.5
1-0-3-0 FR	12.06	6
0-2-2-0 FR	12.11	5
3-0-0-1 FR	12.17	6
2-0-0-2 FR	12.28	5.5
3-1-0-0 FR	12.53	6
0-1-1-1 FR	12.57	7
3-0-1-0 FR	12.60	6.5
2-2-0-0 FR	12.79	5.5
2-0-2-0 FR	12.85	5.5
0-0-2-3 FR	12.96	5
0-2-0-3 FR	12.99	5
0-3-0-2 FR	13.06	5
0-0-3-2 FR	13.07	5
1-1-0-1 FR	13.12	7
1-0-1-1 FR	13.24	7
0-3-2-0 FR	13.50	5
0-2-3-0 FR	13.54	5
2-0-0-3 FR	13.58	5
1-1-1-0 FR	13.67	7
3-0-0-2 FR	13.88	5
0-1-1-2 FR	13.96	6
0-2-1-1 FR	14.12	6
2-3-0-0 FR	14.13	5.5
0-1-2-1 FR	14.18	6
2-0-3-0 FR	14.26	5.5
3-2-0-0 FR	14.37	5.5
0-3-0-3 FR	14.40	4.5
0-0-3-3 FR	14.40	5
3-0-2-0 FR	14.44	5.5
1-1-0-2 FR	14.44	6.5
1-0-1-2 FR	14.51	6.5
1-2-0-1 FR	14.59	6.5
1-0-2-1 FR	14.74	6.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-3-3-0 FR	14.92	5
1-2-1-0 FR	15.04	6.5
2-1-0-1 FR	15.06	6.5
1-1-2-0 FR	15.13	6.5
2-0-1-1 FR	15.14	6.5
0-1-1-3 FR	15.20	6
3-0-0-3 FR	15.21	5
0-3-1-1 FR	15.40	6
0-1-3-1 FR	15.52	6
2-1-1-0 FR	15.56	7
1-1-0-3 FR	15.65	6
1-0-1-3 FR	15.70	6.5
3-3-0-0 FR	15.74	5
0-2-1-2 FR	15.75	5.5
0-1-2-2 FR	15.78	5.5
3-0-3-0 FR	15.84	5.5
1-3-0-1 FR	15.86	6
0-2-2-1 FR	15.96	5.5
1-0-3-1 FR	16.05	6.5
1-2-0-2 FR	16.16	5.5
1-0-2-2 FR	16.23	6
1-3-1-0 FR	16.25	6.5
1-1-3-0 FR	16.43	6.5
2-1-0-2 FR	16.54	6
3-1-0-1 FR	16.55	6.5
2-0-1-2 FR	16.57	6
3-0-1-1 FR	16.64	6.5
2-2-0-1 FR	16.70	6
1-2-2-0 FR	16.73	6
2-0-2-1 FR	16.82	6
3-1-1-0 FR	17.03	6.5
0-2-1-3 FR	17.03	5.5
0-1-2-3 FR	17.06	5.5
2-2-1-0 FR	17.09	6
0-3-1-2 FR	17.09	5.5
2-1-2-0 FR	17.17	6
0-1-3-2 FR	17.17	5.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-3-2-1 FR	17.29	5.5
0-2-3-1 FR	17.34	5.5
1-2-0-3 FR	17.45	5.5
1-0-2-3 FR	17.48	5.5
1-3-0-2 FR	17.50	5.5
1-0-3-2 FR	17.60	5.5
0-2-2-2 FR	17.73	5
1-1-1-1 FR	17.74	7
2-1-0-3 FR	17.77	5.5
2-0-1-3 FR	17.81	6
1-3-2-0 FR	18.01	5.5
2-3-0-1 FR	18.01	5.5
3-1-0-2 FR	18.07	6
1-2-3-0 FR	18.08	6
3-0-1-2 FR	18.11	6
2-0-3-1 FR	18.18	6
3-2-0-1 FR	18.25	5.5
2-3-1-0 FR	18.36	6
3-0-2-1 FR	18.36	6
0-3-1-3 FR	18.37	5.5
2-2-0-2 FR	18.40	5.5
2-0-2-2 FR	18.45	5.5
0-1-3-3 FR	18.46	5.5
2-1-3-0 FR	18.52	6
3-2-1-0 FR	18.63	6
0-3-3-1 FR	18.67	5.5
3-1-2-0 FR	18.71	6
1-3-0-3 FR	18.79	5.5
1-0-3-3 FR	18.88	5.5
2-2-2-0 FR	18.93	5.5
1-1-1-2 FR	18.96	6.5
0-2-2-3 FR	19.06	5
1-2-1-1 FR	19.09	6.5
0-3-2-2 FR	19.12	5
0-2-3-2 FR	19.16	5
1-1-2-1 FR	19.17	6.5
3-1-0-3 FR	19.35	5.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
3-0-1-3 FR	19.36	5.5
1-3-3-0 FR	19.39	5.5
3-3-0-1 FR	19.59	5.5
2-1-1-1 FR	19.63	7
2-2-0-3 FR	19.69	5.5
3-0-3-1 FR	19.72	5.5
2-0-2-3 FR	19.73	5.5
2-3-0-2 FR	19.74	5
2-0-3-2 FR	19.85	5.5
3-3-1-0 FR	19.90	6
3-2-0-2 FR	19.98	5.5
3-0-2-2 FR	20.04	5.5
3-1-3-0 FR	20.06	6
1-1-1-3 FR	20.15	6.5
2-3-2-0 FR	20.26	5.5
1-3-1-1 FR	20.30	6.5
2-2-3-0 FR	20.32	5.5
0-3-2-3 FR	20.45	5
1-1-3-1 FR	20.47	6.5
0-2-3-3 FR	20.49	5
1-2-1-2 FR	20.50	6
3-2-2-0 FR	20.51	5.5
0-3-3-2 FR	20.55	5
1-1-2-2 FR	20.57	6
1-2-2-1 FR	20.72	6
2-1-1-2 FR	20.96	6.5
3-1-1-1 FR	21.08	7
2-3-0-3 FR	21.08	5
2-2-1-1 FR	21.10	6.5
2-0-3-3 FR	21.13	5.5
2-1-2-1 FR	21.19	6.5
3-2-0-3 FR	21.30	5
3-0-2-3 FR	21.32	5.5
3-3-0-2 FR	21.36	5
3-0-3-2 FR	21.44	5.5
2-3-3-0 FR	21.64	5.5
1-2-1-3 FR	21.74	6

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
1-3-1-2 FR	21.78	6
1-1-2-3 FR	21.80	6
3-3-2-0 FR	21.84	5.5
0-3-3-3 FR	21.88	5
3-2-3-0 FR	21.91	5.5
1-1-3-2 FR	21.91	6
1-3-2-1 FR	21.98	6
1-2-3-1 FR	22.06	6
2-1-1-3 FR	22.16	6.5
1-2-2-2 FR	22.29	5.5
2-3-1-1 FR	22.35	6
3-1-1-2 FR	22.46	6.5
2-1-3-1 FR	22.50	6
3-2-1-1 FR	22.61	6
2-2-1-2 FR	22.63	6
3-1-2-1 FR	22.69	6
3-3-0-3 FR	22.69	5
2-1-2-2 FR	22.70	6
3-0-3-3 FR	22.74	5
2-2-2-1 FR	22.85	6
1-3-1-3 FR	23.02	6
1-1-3-3 FR	23.15	6
3-3-3-0 FR	23.23	5
1-3-3-1 FR	23.32	6
1-2-2-3 FR	23.58	5.5
1-3-2-2 FR	23.62	5.5
1-2-3-2 FR	23.68	5.5
3-1-1-3 FR	23.69	6
3-3-1-1 FR	23.88	6
2-2-1-3 FR	23.88	5.5
2-3-1-2 FR	23.93	5.5
2-1-2-3 FR	23.94	6
3-1-3-1 FR	24.04	6
2-1-3-2 FR	24.05	6
2-3-2-1 FR	24.14	5.5
3-2-1-2 FR	24.17	5.5
2-2-3-1 FR	24.21	5.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
3-1-2-2 FR	24.24	6
3-2-2-1 FR	24.40	5.5
2-2-2-2 FR	24.54	5.5
1-3-2-3 FR	24.91	5.5
1-2-3-3 FR	24.97	5.5
1-3-3-2 FR	25.01	5.5
2-3-1-3 FR	25.22	5.5
2-1-3-3 FR	25.32	5.5
3-2-1-3 FR	25.46	5.5
3-3-1-2 FR	25.50	5.5
3-1-2-3 FR	25.51	5.5
2-3-3-1 FR	25.53	5.5
3-1-3-2 FR	25.62	5.5
3-3-2-1 FR	25.72	5.5
3-2-3-1 FR	25.79	5.5
2-2-2-3 FR	25.83	5.5
2-3-2-2 FR	25.87	5.5
2-2-3-2 FR	25.93	5.5
3-2-2-2 FR	26.12	5.5
1-3-3-3 FR	26.29	5.5
3-3-1-3 FR	26.79	5.5
3-1-3-3 FR	26.90	5.5
3-3-3-1 FR	27.11	5.5
2-3-2-3 FR	27.16	5.5
2-2-3-3 FR	27.22	5.5
2-3-3-2 FR	27.26	5.5
3-2-2-3 FR	27.41	5.5
3-3-2-2 FR	27.45	5.5
3-2-3-2 FR	27.51	5.5
2-3-3-3 FR	28.57	5
3-3-2-3 FR	28.76	5
3-2-3-3 FR	28.80	5
3-3-3-2 FR	28.86	5
3-3-3-3 FR	30.19	5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-0-0-0 MR	2.32	2.5
0-1-0-0 MR	5.22	4
1-0-0-0 MR	5.46	4
0-0-0-1 MR	5.88	3
0-0-1-0 MR	6.19	3.5
0-2-0-0 MR	7.57	4
0-0-0-2 MR	7.91	2.5
2-0-0-0 MR	7.95	4
0-0-2-0 MR	8.40	3
0-1-0-1 MR	9.05	4
0-3-0-0 MR	9.07	4
0-0-0-3 MR	9.22	2.5
1-0-0-1 MR	9.30	4
1-1-0-0 MR	9.38	5.5
0-1-1-0 MR	9.55	4.5
3-0-0-0 MR	9.64	4
0-0-3-0 MR	9.73	3
1-0-1-0 MR	9.84	4.5
0-0-1-1 MR	9.96	3.5
0-1-0-2 MR	10.71	3.5
1-0-0-2 MR	10.92	3.5
0-1-2-0 MR	11.37	4
0-2-0-1 MR	11.40	4
1-2-0-0 MR	11.46	5
1-0-2-0 MR	11.62	4
2-1-0-0 MR	11.72	5
2-0-0-1 MR	11.78	4
0-0-1-2 MR	11.80	3
0-1-0-3 MR	11.84	3.5
0-2-1-0 MR	11.87	4
0-0-2-1 MR	12.03	3
1-0-0-3 MR	12.05	3.5
2-0-1-0 MR	12.28	4.5
0-1-3-0 MR	12.57	3.5
1-0-3-0 MR	12.78	3.5
1-3-0-0 MR	12.90	4.5
0-3-0-1 MR	12.90	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-0-1-3 MR	13.02	3
0-2-0-2 MR	13.11	3.5
1-1-0-1 MR	13.16	5
3-1-0-0 MR	13.35	5
0-3-1-0 MR	13.36	4
0-0-3-1 MR	13.36	3
0-1-1-1 MR	13.37	4.5
2-0-0-2 MR	13.40	3.5
3-0-0-1 MR	13.47	4
1-0-1-1 MR	13.67	4.5
0-2-2-0 MR	13.72	3.5
1-1-1-0 MR	13.77	5
2-2-0-0 MR	13.88	4.5
3-0-1-0 MR	13.95	4.5
0-0-2-2 MR	14.04	3
2-0-2-0 MR	14.11	4
0-2-0-3 MR	14.24	3.5
1-1-0-2 MR	14.45	4.5
2-0-0-3 MR	14.53	3.5
0-3-0-2 MR	14.63	3.5
0-2-3-0 MR	14.96	3.5
0-1-1-2 MR	14.96	4
3-0-0-2 MR	15.10	3.5
0-1-2-1 MR	15.19	4
1-1-2-0 MR	15.21	4.5
1-0-1-2 MR	15.21	4
0-3-2-0 MR	15.24	3.5
0-0-2-3 MR	15.26	3
2-0-3-0 MR	15.27	4
1-2-0-1 MR	15.28	4.5
2-3-0-0 MR	15.33	4.5
0-0-3-2 MR	15.37	3
1-1-0-3 MR	15.43	4.5
1-0-2-1 MR	15.45	4
2-1-0-1 MR	15.51	5
3-2-0-0 MR	15.53	4.5
0-2-1-1 MR	15.70	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-3-0-3 MR	15.76	3.5
3-0-2-0 MR	15.80	4
1-2-1-0 MR	15.84	5
0-1-1-3 MR	16.01	4
2-0-1-1 MR	16.11	4.5
2-1-1-0 MR	16.12	5
3-0-0-3 MR	16.23	3.5
1-0-1-3 MR	16.26	4
1-1-3-0 MR	16.29	4.5
0-1-3-1 MR	16.35	4
0-3-3-0 MR	16.47	3.5
0-0-3-3 MR	16.59	3
1-0-3-1 MR	16.61	4
1-2-0-2 MR	16.70	4
1-3-0-1 MR	16.73	4.5
2-1-0-2 MR	16.88	4.5
0-1-2-2 MR	16.91	3.5
3-0-3-0 MR	16.96	3.5
3-3-0-0 MR	16.98	4.5
1-0-2-2 MR	17.12	3.5
3-1-0-1 MR	17.13	4.5
0-3-1-1 MR	17.19	4
1-3-1-0 MR	17.26	4.5
0-2-1-2 MR	17.31	4
1-2-2-0 MR	17.43	4.5
0-2-2-1 MR	17.53	4
1-1-1-1 MR	17.55	5
2-1-2-0 MR	17.65	4.5
2-0-1-2 MR	17.69	4
2-2-0-1 MR	17.71	4.5
3-1-1-0 MR	17.75	5
1-2-0-3 MR	17.75	4
3-0-1-1 MR	17.77	4
2-1-0-3 MR	17.88	4
2-0-2-1 MR	17.93	4
0-1-2-3 MR	18.04	3.5
0-1-3-2 MR	18.15	3.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
1-3-0-2 MR	18.20	4
2-2-1-0 MR	18.25	4.5
1-0-2-3 MR	18.25	3.5
1-0-3-2 MR	18.36	3.5
0-2-1-3 MR	18.39	3.5
1-2-3-0 MR	18.53	4
3-1-0-2 MR	18.54	4.5
0-2-3-1 MR	18.72	3.5
2-1-3-0 MR	18.73	4.5
2-0-1-3 MR	18.74	4
0-3-1-2 MR	18.81	4
1-1-1-2 MR	18.82	4.5
1-3-2-0 MR	18.87	4
0-3-2-1 MR	19.02	4
1-1-2-1 MR	19.04	4.5
2-0-3-1 MR	19.09	4
2-3-0-1 MR	19.16	4.5
2-2-0-2 MR	19.17	4
1-3-0-3 MR	19.25	4
0-1-3-3 MR	19.28	3.5
0-2-2-2 MR	19.30	3.5
3-1-2-0 MR	19.31	4.5
3-2-0-1 MR	19.37	4.5
3-0-1-2 MR	19.38	4
1-0-3-3 MR	19.49	3.5
3-1-0-3 MR	19.57	4
2-0-2-2 MR	19.60	3.5
3-0-2-1 MR	19.63	4
1-2-1-1 MR	19.64	4.5
2-3-1-0 MR	19.69	4.5
1-1-1-3 MR	19.80	4.5
2-2-2-0 MR	19.87	4.5
3-2-1-0 MR	19.90	4.5
2-1-1-1 MR	19.91	5
0-3-1-3 MR	19.91	3.5
1-3-3-0 MR	20.01	4
1-1-3-1 MR	20.12	4.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
2-2-0-3 MR	20.23	4
0-3-3-1 MR	20.24	3.5
3-1-3-0 MR	20.39	4.5
3-0-1-3 MR	20.43	4
0-2-2-3 MR	20.44	3.5
1-1-2-2 MR	20.49	4
0-2-3-2 MR	20.54	3.5
2-3-0-2 MR	20.67	4
2-0-2-3 MR	20.73	3.5
3-0-3-1 MR	20.78	4
3-3-0-1 MR	20.82	4.5
0-3-2-2 MR	20.82	3.5
2-0-3-2 MR	20.84	3.5
3-2-0-2 MR	20.86	4
2-2-3-0 MR	21.01	4
1-2-1-2 MR	21.06	4.5
1-3-1-1 MR	21.09	4.5
1-2-2-1 MR	21.26	4.5
2-1-1-2 MR	21.26	4.5
3-0-2-2 MR	21.30	3.5
3-3-1-0 MR	21.35	4.5
2-3-2-0 MR	21.35	4
2-1-2-1 MR	21.48	4.5
3-1-1-1 MR	21.53	5
1-1-2-3 MR	21.54	4
3-2-2-0 MR	21.55	4
1-1-3-2 MR	21.64	4
0-2-3-3 MR	21.67	3.5
2-3-0-3 MR	21.72	4
3-2-0-3 MR	21.91	4
0-3-2-3 MR	21.96	3.5
2-0-3-3 MR	21.97	3.5
1-2-1-3 MR	22.04	4
0-3-3-2 MR	22.06	3.5
2-2-1-1 MR	22.08	4.5
2-1-1-3 MR	22.24	4.5
3-3-0-2 MR	22.35	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
1-2-3-1 MR	22.36	4
3-0-2-3 MR	22.43	3.5
2-3-3-0 MR	22.49	4
1-3-1-2 MR	22.51	4.5
3-0-3-2 MR	22.54	3.5
2-1-3-1 MR	22.56	4.5
3-2-3-0 MR	22.70	4
1-3-2-1 MR	22.70	4
1-1-3-3 MR	22.70	4
1-2-2-2 MR	22.83	4
3-1-1-2 MR	22.92	4.5
2-1-2-2 MR	22.97	4
3-3-2-0 MR	23.03	4
3-1-2-1 MR	23.14	4.5
0-3-3-3 MR	23.19	3.5
3-3-0-3 MR	23.41	4
2-2-1-2 MR	23.49	4.5
2-3-1-1 MR	23.52	4.5
1-3-1-3 MR	23.54	4
3-0-3-3 MR	23.67	3.5
2-2-2-1 MR	23.70	4.5
3-2-1-1 MR	23.74	4.5
1-3-3-1 MR	23.85	4
1-2-2-3 MR	23.88	4
3-1-1-3 MR	23.90	4.5
1-2-3-2 MR	23.98	4
2-1-2-3 MR	24.02	4
2-1-3-2 MR	24.13	4
3-3-3-0 MR	24.18	4
3-1-3-1 MR	24.22	4.5
1-3-2-2 MR	24.32	4
2-2-1-3 MR	24.52	4
3-1-2-2 MR	24.66	4
2-2-3-1 MR	24.85	4
2-3-1-2 MR	24.96	4
1-2-3-3 MR	25.03	4
3-2-1-2 MR	25.15	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
3-3-1-1 MR	25.18	4.5
2-3-2-1 MR	25.18	4
2-1-3-3 MR	25.18	4
2-2-2-2 MR	25.31	4
1-3-2-3 MR	25.37	4
3-2-2-1 MR	25.38	4
1-3-3-2 MR	25.47	4
3-1-2-3 MR	25.71	4
3-1-3-2 MR	25.82	4
2-3-1-3 MR	26.01	4
3-2-1-3 MR	26.21	4
2-3-3-1 MR	26.33	4
2-2-2-3 MR	26.36	4
2-2-3-2 MR	26.46	4
1-3-3-3 MR	26.52	4
3-2-3-1 MR	26.53	4
3-3-1-2 MR	26.64	4
2-3-2-2 MR	26.80	4
3-3-2-1 MR	26.87	4
3-1-3-3 MR	26.87	4
3-2-2-2 MR	27.00	4
2-2-3-3 MR	27.51	4
3-3-1-3 MR	27.70	4
2-3-2-3 MR	27.85	4
2-3-3-2 MR	27.95	4
3-3-3-1 MR	28.02	4
3-2-2-3 MR	28.05	4
3-2-3-2 MR	28.15	4
3-3-2-2 MR	28.48	4
2-3-3-3 MR	29.00	4
3-2-3-3 MR	29.20	4
3-3-2-3 MR	29.54	4
3-3-3-2 MR	29.63	4
3-3-3-3 MR	30.69	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-0-0-0 HR	2.32	2.5
0-1-0-0 HR	5.22	4
1-0-0-0 HR	5.46	4
0-0-0-1 HR	5.88	3
0-0-1-0 HR	6.19	3.5
0-2-0-0 HR	7.57	4
0-0-0-2 HR	7.91	2.5
2-0-0-0 HR	7.95	4
0-0-2-0 HR	8.40	3
0-1-0-1 HR	9.05	4
0-3-0-0 HR	9.07	4
0-0-0-3 HR	9.22	2.5
1-0-0-1 HR	9.30	4
1-1-0-0 HR	9.38	5.5
0-1-1-0 HR	9.55	4.5
3-0-0-0 HR	9.64	4
0-0-3-0 HR	9.73	3
1-0-1-0 HR	9.84	4.5
0-0-1-1 HR	9.96	3.5
0-1-0-2 HR	10.71	3.5
1-0-0-2 HR	10.92	3.5
0-1-2-0 HR	11.37	4
0-2-0-1 HR	11.40	4
1-2-0-0 HR	11.46	5
1-0-2-0 HR	11.62	4
2-1-0-0 HR	11.72	5
2-0-0-1 HR	11.78	4
0-0-1-2 HR	11.80	3
0-1-0-3 HR	11.84	3.5
0-2-1-0 HR	11.87	4
0-0-2-1 HR	12.03	3
1-0-0-3 HR	12.05	3.5
2-0-1-0 HR	12.28	4.5
0-1-3-0 HR	12.57	3.5
1-0-3-0 HR	12.78	3.5
1-3-0-0 HR	12.90	4.5
0-3-0-1 HR	12.90	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-0-1-3 HR	13.02	3
0-2-0-2 HR	13.11	3.5
1-1-0-1 HR	13.16	5
3-1-0-0 HR	13.35	5
0-3-1-0 HR	13.36	4
0-0-3-1 HR	13.36	3
0-1-1-1 HR	13.37	4.5
2-0-0-2 HR	13.40	3.5
3-0-0-1 HR	13.47	4
1-0-1-1 HR	13.67	4.5
0-2-2-0 HR	13.72	3.5
1-1-1-0 HR	13.77	5
2-2-0-0 HR	13.88	4.5
3-0-1-0 HR	13.95	4.5
0-0-2-2 HR	14.04	3
2-0-2-0 HR	14.11	4
0-2-0-3 HR	14.24	3.5
1-1-0-2 HR	14.45	4.5
2-0-0-3 HR	14.53	3.5
0-3-0-2 HR	14.63	3.5
0-2-3-0 HR	14.96	3.5
0-1-1-2 HR	14.96	4
3-0-0-2 HR	15.10	3.5
0-1-2-1 HR	15.19	4
1-1-2-0 HR	15.21	4.5
1-0-1-2 HR	15.21	4
0-3-2-0 HR	15.24	3.5
0-0-2-3 HR	15.26	3
2-0-3-0 HR	15.27	4
1-2-0-1 HR	15.28	4.5
2-3-0-0 HR	15.33	4.5
0-0-3-2 HR	15.37	3
1-1-0-3 HR	15.43	4.5
1-0-2-1 HR	15.45	4
2-1-0-1 HR	15.51	5
3-2-0-0 HR	15.53	4.5
0-2-1-1 HR	15.70	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
0-3-0-3 HR	15.76	3.5
3-0-2-0 HR	15.80	4
1-2-1-0 HR	15.84	5
0-1-1-3 HR	16.01	4
2-0-1-1 HR	16.11	4.5
2-1-1-0 HR	16.12	5
3-0-0-3 HR	16.23	3.5
1-0-1-3 HR	16.26	4
1-1-3-0 HR	16.29	4.5
0-1-3-1 HR	16.35	4
0-3-3-0 HR	16.47	3.5
0-0-3-3 HR	16.59	3
1-0-3-1 HR	16.61	4
1-2-0-2 HR	16.70	4
1-3-0-1 HR	16.73	4.5
2-1-0-2 HR	16.88	4.5
0-1-2-2 HR	16.91	3.5
3-0-3-0 HR	16.96	3.5
3-3-0-0 HR	16.98	4.5
1-0-2-2 HR	17.12	3.5
3-1-0-1 HR	17.13	4.5
0-3-1-1 HR	17.19	4
1-3-1-0 HR	17.26	4.5
0-2-1-2 HR	17.31	4
1-2-2-0 HR	17.43	4.5
0-2-2-1 HR	17.53	4
1-1-1-1 HR	17.55	5
2-1-2-0 HR	17.65	4.5
2-0-1-2 HR	17.69	4
2-2-0-1 HR	17.71	4.5
3-1-1-0 HR	17.75	5
1-2-0-3 HR	17.75	4
3-0-1-1 HR	17.77	4
2-1-0-3 HR	17.88	4
2-0-2-1 HR	17.93	4
0-1-2-3 HR	18.04	3.5
0-1-3-2 HR	18.15	3.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
1-3-0-2 HR	18.20	4
2-2-1-0 HR	18.25	4.5
1-0-2-3 HR	18.25	3.5
1-0-3-2 HR	18.36	3.5
0-2-1-3 HR	18.39	3.5
1-2-3-0 HR	18.53	4
3-1-0-2 HR	18.54	4.5
0-2-3-1 HR	18.72	3.5
2-1-3-0 HR	18.73	4.5
2-0-1-3 HR	18.74	4
0-3-1-2 HR	18.81	4
1-1-1-2 HR	18.82	4.5
1-3-2-0 HR	18.87	4
0-3-2-1 HR	19.02	4
1-1-2-1 HR	19.04	4.5
2-0-3-1 HR	19.09	4
2-3-0-1 HR	19.16	4.5
2-2-0-2 HR	19.17	4
1-3-0-3 HR	19.25	4
0-1-3-3 HR	19.28	3.5
0-2-2-2 HR	19.30	3.5
3-1-2-0 HR	19.31	4.5
3-2-0-1 HR	19.37	4.5
3-0-1-2 HR	19.38	4
1-0-3-3 HR	19.49	3.5
3-1-0-3 HR	19.57	4
2-0-2-2 HR	19.60	3.5
3-0-2-1 HR	19.63	4
1-2-1-1 HR	19.64	4.5
2-3-1-0 HR	19.69	4.5
1-1-1-3 HR	19.80	4.5
2-2-2-0 HR	19.87	4.5
3-2-1-0 HR	19.90	4.5
2-1-1-1 HR	19.91	5
0-3-1-3 HR	19.91	3.5
1-3-3-0 HR	20.01	4
1-1-3-1 HR	20.12	4.5

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
2-2-0-3 HR	20.23	4
0-3-3-1 HR	20.24	3.5
3-1-3-0 HR	20.39	4.5
3-0-1-3 HR	20.43	4
0-2-2-3 HR	20.44	3.5
1-1-2-2 HR	20.49	4
0-2-3-2 HR	20.54	3.5
2-3-0-2 HR	20.67	4
2-0-2-3 HR	20.73	3.5
3-0-3-1 HR	20.78	4
3-3-0-1 HR	20.82	4.5
0-3-2-2 HR	20.82	3.5
2-0-3-2 HR	20.84	3.5
3-2-0-2 HR	20.86	4
2-2-3-0 HR	21.01	4
1-2-1-2 HR	21.06	4.5
1-3-1-1 HR	21.09	4.5
1-2-2-1 HR	21.26	4.5
2-1-1-2 HR	21.26	4.5
3-0-2-2 HR	21.30	3.5
3-3-1-0 HR	21.35	4.5
2-3-2-0 HR	21.35	4
2-1-2-1 HR	21.48	4.5
3-1-1-1 HR	21.53	5
1-1-2-3 HR	21.54	4
3-2-2-0 HR	21.55	4
1-1-3-2 HR	21.64	4
0-2-3-3 HR	21.67	3.5
2-3-0-3 HR	21.72	4
3-2-0-3 HR	21.91	4
0-3-2-3 HR	21.96	3.5
2-0-3-3 HR	21.97	3.5
1-2-1-3 HR	22.04	4
0-3-3-2 HR	22.06	3.5
2-2-1-1 HR	22.08	4.5
2-1-1-3 HR	22.24	4.5
3-3-0-2 HR	22.35	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
1-2-3-1 HR	22.36	4
3-0-2-3 HR	22.43	3.5
2-3-3-0 HR	22.49	4
1-3-1-2 HR	22.51	4.5
3-0-3-2 HR	22.54	3.5
2-1-3-1 HR	22.56	4.5
3-2-3-0 HR	22.70	4
1-3-2-1 HR	22.70	4
1-1-3-3 HR	22.70	4
1-2-2-2 HR	22.83	4
3-1-1-2 HR	22.92	4.5
2-1-2-2 HR	22.97	4
3-3-2-0 HR	23.03	4
3-1-2-1 HR	23.14	4.5
0-3-3-3 HR	23.19	3.5
3-3-0-3 HR	23.41	4
2-2-1-2 HR	23.49	4.5
2-3-1-1 HR	23.52	4.5
1-3-1-3 HR	23.54	4
3-0-3-3 HR	23.67	3.5
2-2-2-1 HR	23.70	4.5
3-2-1-1 HR	23.74	4.5
1-3-3-1 HR	23.85	4
1-2-2-3 HR	23.88	4
3-1-1-3 HR	23.90	4.5
1-2-3-2 HR	23.98	4
2-1-2-3 HR	24.02	4
2-1-3-2 HR	24.13	4
3-3-3-0 HR	24.18	4
3-1-3-1 HR	24.22	4.5
1-3-2-2 HR	24.32	4
2-2-1-3 HR	24.52	4
3-1-2-2 HR	24.66	4
2-2-3-1 HR	24.85	4
2-3-1-2 HR	24.96	4
1-2-3-3 HR	25.03	4
3-2-1-2 HR	25.15	4

Boost Stage	Max EQ Boost (db)	Frequency (GHz)
3-3-1-1 HR	25.18	4.5
2-3-2-1 HR	25.18	4
2-1-3-3 HR	25.18	4
2-2-2-2 HR	25.31	4
1-3-2-3 HR	25.37	4
3-2-2-1 HR	25.38	4
1-3-3-2 HR	25.47	4
3-1-2-3 HR	25.71	4
3-1-3-2 HR	25.82	4
2-3-1-3 HR	26.01	4
3-2-1-3 HR	26.21	4
2-3-3-1 HR	26.33	4
2-2-2-3 HR	26.36	4
2-2-3-2 HR	26.46	4
1-3-3-3 HR	26.52	4
3-2-3-1 HR	26.53	4
3-3-1-2 HR	26.64	4
2-3-2-2 HR	26.80	4
3-3-2-1 HR	26.87	4
3-1-3-3 HR	26.87	4
3-2-2-2 HR	27.00	4
2-2-3-3 HR	27.51	4
3-3-1-3 HR	27.70	4
2-3-2-3 HR	27.85	4
2-3-3-2 HR	27.95	4
3-3-3-1 HR	28.02	4
3-2-2-3 HR	28.05	4
3-2-3-2 HR	28.15	4
3-3-2-2 HR	28.48	4
2-3-3-3 HR	29.00	4
3-2-3-3 HR	29.20	4
3-3-2-3 HR	29.54	4
3-3-3-2 HR	29.63	4
3-3-3-3 HR	30.69	4

2.6. Read CTLE Boost

The CTLE boost level can be read out from channel register 0x8F.

Channel Register	Bit	Description
0x8F	7:6	Stage 0 boost [1:0]
	5:4	Stage 1 boost [1:0]
	3:2	Stage 2 boost [1:0]
	1:0	Stage 3 boost [1:0]

When typing out the boost values for each stage is common practice to write the boost as:

Stage0-Stage1-Stage2-Stage3.

For example if stages 1 through 3 had a boost of 0 but stage 0 had a boost of 1 then the boost would be written as 1-0-0-0.

2.7. Reprogramming the CTLE Adaption Table

The CTLE consists of 4 stages with 4 levels of boost per stage, resulting in 256 possible boost settings. The CTLE adaption table is only 32 indexes long. It is possible that the best possible boost value for a particular channel may not be populated in the adaption table by default. Users may reprogram the CTLE adaption table so long as the modified table meets the following criteria:

- The adaption table should increase in boost strength from the lowest index to the highest index
- Users should avoid having the same boost setting in consecutive indexes

The CTLE adaption table is located within the channel register set and starts at channel register 0x40 and stops at channel register 0x5F. Each index uses the following convention:

Channel Register	Bit	Description
0x40 through 0x5F	7:6	Stage 0 boost [1:0]
	5:4	Stage 1 boost [1:0]
	3:2	Stage 2 boost [1:0]
	1:0	Stage 3 boost [1:0]

Users can reprogram any index to their desired boost value by simply writing to that index register.

2.8. Change the CTLE Adaption Table Starting Index

It is sometimes useful to use a smaller CTLE adaption table. For example, there may be some cases where it is desired to constrain the CTLE to certain boost levels. Users can change the CTLE starting

index to limit the adaption table all the way down to 1 index from a maximum of 32 indexes and then reprogram the remaining indexes to the desired constrained boost values.

The CTLE adaption table's start index value is located at channel register 0x39[4:0]. To change the starting index user must program the desired index value ranging from 0 to 32. Then the index_ov bit should be set in channel register 0x2F[3]. If the CDR is already active, a CDR reset should be issued to allow for these new setting to take effect.

Channel Register	Bit	Description
0x39	4:0	Determines the start index, values range from 0 to 31.
0x2F	3	Start index override 1: Override the start index with the value in 0x39[4:0] 0: Normal operation, start index is 0.

2.9. Force CTLE Boost Value

The recommended method for forcing a CTLE boost value is to manipulate the adaption table start index to the begin at the last index, reprogram the last index's boost value and then enable the CTLE adaption for lower data rates.

Channel Register	Bit	Description
0x39	4:0	Determines the start index, values range from 0 to 31.
0x2F	3	Start index override 1: Override the start index with the value in 0x39[4:0] 0: Normal operation, start index is 0.
0x6F	7	1: Enables CTLE adaption for divider settings greater than 2 (divide by 4, and 8) 0: Normal operation, the CTLE is not adapted for dividers beyond 2. Instead the CTLE is set to the fixed boost level specified in channel register 0x3A

Example:

Channel Register	Write Value	Mask	Description
0x39	0x1F	0x1F	Set the starting index to 31
0x2F	0x08	0x08	Override the start index to the value specified in channel register 0x39
0x6F	0x80	0x80	Enable CTLE adaption for all divider settings

2.10. Change the Non-adapted CTLE Boost level for Slower Data Rates

By default, the CTLE is adapted only for divider settings of 1 and 2. In this case, when the channel operates at divider setting of 4, 8 or 16 the CTLE is set to the value specified by channel register 0x3A.

Channel Register	Bit	Description
0x3A	7:6	Fixed stage 0 boost [1:0] for dividers greater than 2
	5:4	Fixed stage 1 boost [1:0] for dividers greater than 2
	3:2	Fixed stage 2 boost [1:0] for dividers greater than 2
	1:0	Fixed stage 3 boost [1:0] for dividers greater than 2

If the device is operated at a low data rate, it is recommended that this register be reprogrammed to have minimal boost(0x00).

2.11. Enable CTLE Adaption for Slower Data Rates

By default, the CTLE is adapted only for divider settings of 1 and 2. Users can allow the CTLE to adapt for these lower data rate dividers by configuring channel register 0x6F[7].

Channel Register	Bit	Description
0x6F	7	1: Enables CTLE adaption for divider settings greater than 2 (divide by 4, 8 and 16) 0: Normal operation, the CTLE is not adapted for dividers beyond 2. Instead the CTLE is set to the fixed boost level specified in channel register 0x3A

2.12. Enable CTLE Limiting Mode

The fourth and final stage of the CTLE can be configured to operate in a linear mode or limiting mode. By default this stage is configured to operate in the linear mode. Linear mode is typically preferred when the DFE is enabled.

CTLE limiting mode can be used in several ways. Limiting mode can be enabled for some CTLE boost values, all CTLE boost values, or no boost values. Refer to the register information and truth table below to choose the desired limiting mode function.

Channel Register	Bit	Description
0x13	2	CTLE Limiting Mode – variable A for the truth table
0x30	6	CTLE search with Limiting Mode – variable B for the truth table

A	B	Function
0	0	CTLE is configured for linear mode
0	1	If last stage of CTLE is set to boost value of 2'b11, then limiting mode is enabled; otherwise limiting mode is disabled
1	0	CTLE is configured for linear mode
1	1	CTLE is configured for limiting mode for all boost values

2.13. Enable the DFE

The DFE must be enabled regardless of the selected adapt mode. To enable the DFE write 0 to channel register 0x1E[3] while the CDR is in reset.

Channel Register	Bit	Description
0x1E	3	1: DFE is powered down (default) 0: DFE is enabled

Here is an example register write sequence:

Channel Register	Write Value	Mask	Description
0x0A	0x0C	0x0C	Place the CDR into reset
0x31	0x40/60	0x60	Configure the device to be in adapt mode 2 or 3
0x1E	0x00	0x08	Enable the DFE
0x0A	0x00	0x0C	Take the CDR out of reset

2.14. Read DFE Tap Values

The DFE tap polarities and tap weights can be read back from channel register 0x71 through 0x75. Each DFE tap weight has a step size of 7mV. A tap polarity reading of 0 indicates that the DFE is attenuating the signal by the tap weight value. A tap polarity reading of 1 indicates that the DFE is boosting the signal by the tap weight value.

Note: DFE tap 1 has a 5 bit tap weight range, while all other taps have a 4 bit tap weight range

Channel Register	Bit	Description
0x71	5	Tap 1 polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	4:0	Tap 1 weight
0x72	4	Tap 2 polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	3:0	Tap 2 weight
0x73	4	Tap 3 polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	3:0	Tap 3 weight
0x74	4	Tap 4 polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	3:0	Tap 4 weight
0x75	4	Tap 5 polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	3:0	Tap 5 weight

2.15. Force DFE Tap Values

Registers 0x71 through 0x75 only allow for reading of the DFE tap weights and polarities. If it is desired to manually set the DFE taps weights and tap polarities then another set of registers must be used. Setting the tap polarities can be done with writes to channel registers 0x11[3:0] and 0x12[7]. Manual tap weights can be set in channel registers 0x12[4:0], 0x20[7:0], and 0x21[7:0]. To enable manual programming of the tap weights both channel register 0x15[7] and 0x2C[3:0] need to be properly programmed.

Channel Register	Bit	Description
0x15	7	1: When ch reg 0x2C[3:0] = 4'b0000, enable manual DFE tap weight programming in registers 0x11[3:0], 0x12[4:0], 0x20[7:0], and 0x21[7:0] 0: Normal operation, DFE will adapt as per the set adapt mode

Channel Register	Bit	Description
0x2C	3:0	0000: When ch reg 0x15[7] =1, enable manual DFE tap weight programming in registers 0x11[3:0], 0x12[4:0], 0x20[7:0], and 0x21[7:0]; this setting should only be used during manual DFE operation

Channel Register	Bit	Description
0x11	3	DFE Tap 2 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	2	DFE Tap 3 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	1	DFE Tap 4 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	0	DFE Tap 5 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated

Channel Register	Bit	Description
0x12	7	DFE Tap 1 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	4:0	DFE Tap 1 Weight LSB = 7mV

Channel Register	Bit	Description
0x20	7:4	DFE Tap 5 Weight LSB = 7mV
	3:0	DFE Tap 4 Weight LSB = 7mV

Channel Register	Bit	Description
0x21	7:4	DFE Tap 3 Weight LSB = 7mV
	3:0	DFE Tap 2 Weight LSB = 7mV

2.16. Change the DFE Adaption Constraints: Polarity Search

The DFE searches for the best values by incrementing the tap weight until a maximum figure of merit is found within the specified constraints for adaption. If the device cannot find a satisfactory figure of merit after searching through the allowable tap weights the DFE tap weights will reset, the tap polarity will invert and the device will increment through the tap weights again. Sometimes it can be beneficial to have the DFE search the opposite tap polarity first. To configure the polarity of the first tap weight search, write to channel registers 0x11[3:0] and 0x12[7].

Channel Register	Bit	Description
0x11	3	DFE Tap 2 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	2	DFE Tap 3 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	1	DFE Tap 4 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated
	0	DFE Tap 5 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated

Channel Register	Bit	Description
0x12	7	DFE Tap 1 Polarity 1: Negative polarity - high pass effect, signal is boosted 0: Positive polarity – low pass effect, signal is attenuated

2.17. Enable Continuous DFE Adaption

By default the DFE is configured to adapt during lock acquisition and then freeze; meaning that after the device acquires lock the DFE tap weights will not change. The DFE can be configured to adapt continuously by configuring channel register 0x7F[4].

Channel Register	Bit	Description
0x7F	4	1: Enable continuous DFE adaption 0: DFE adapts only during lock acquisition

2.18. Change the Continuous DFE Adaption Constraints: Allowable Change is Tap Weight

Users can constrain the allowable range between continuous DFE adaptions as well as the number of look ahead values.

The limit of allowable tap weight change from the previous base point is set by channel register 0x7E[7:4].

Channel Register	Bit	Description
0x7E	7:4	Sets the allowable tap weight change relative to the previous weight, for continuously adapting DFE

The number of look ahead settings used during continuous adaption is set by channel register 0x7F[2:0]. Users should be wary of making the look ahead range too large. If the DFE allowable tap weight change is large and the look ahead range is large, user may see that the DFE causes signal integrity to degrade while it tries these values. Under most circumstances the look ahead range can be left at its default setting of 2.

Channel Register	Bit	Description
0x7F	2:0	Sets the look ahead range for the DFE continuous adaption.

2.19. VGA Configuration

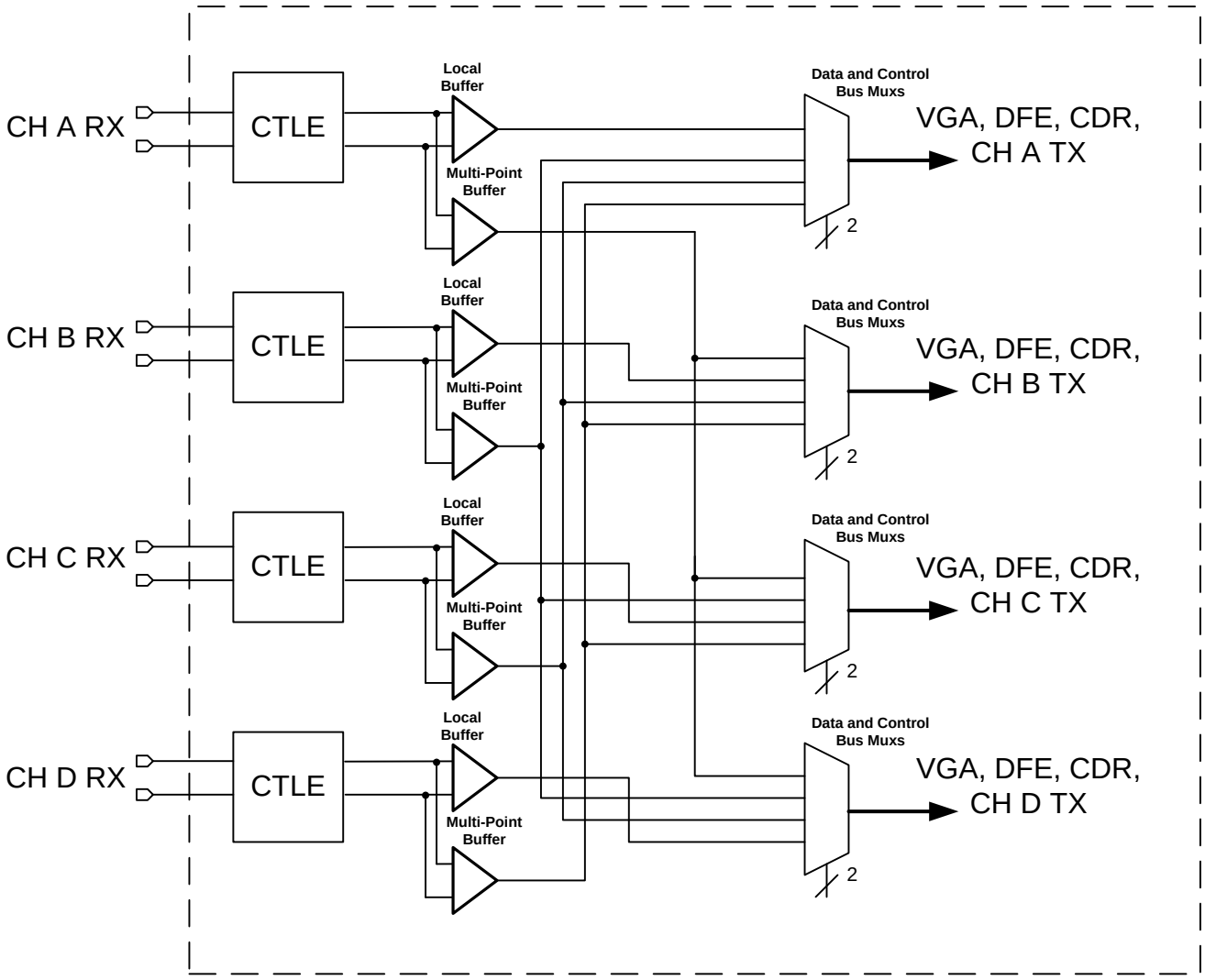
The variable gain amplifier (VGA) can be enabled to allow the DS125DF1610 to support very small input signals. This feature is not typically required for operation, but it can be enabled to provide additional margin in some applications.

The VGA is configured in channel register 0x8E[1:0].

Channel Register	Bit	Description
0x8E	1:0	00: 0 dB 01: 6 dB 10: 6 dB 11: 12 dB

2.20. Cross Point Switch Configuration

The DS125DF1610 features a full non-blocking 4x4 cross point switch for each quad within the device. The cross point switch allows for all combinations of routing within each quad, including simple channel switching or 1:n broadcasting – where a single input can be broadcast out to up to 4 channels simultaneously. To set the cross point switch to the desired functionality, configure channel registers 0x96 and 0x9B for each channel that will utilize the cross point routing.



The cross point switch is located between CTLE (EQ) and VGA/DFE blocks within each channel, as shown in the figure above. The EQ of each channel has a local buffer and a multi-point buffer. The local buffer is used when the cross point passes data from the EQ of one channel to that same channel's VGA/DFE, CDR and driver blocks. The multi-point buffer should be enabled whenever the EQ of one channel must transmit data to another channel's VGA/DFE, CDR and driver blocks.

If the cross point is used to broadcast data from one EQ to multiple channels at the same time then one channel must be assigned to have master control over the EQ for the purposes of EQ and DFE adaption. All other channels receiving data from the EQ should be configured to operate as a slave to the master channel's adaption. To configure a channels CDR to be either a master or slave for the purposes of adaption with cross point routing, write to 0x96 bit 5.

Channel Register	Bit	Write Value
0x96	5	Cross Point Master/Slave Select 1: Sets the channel's CDR to behave as a slave for EQ/DFE adaption – (**DO NOT SET)** 0: Sets the channel's CDR to behave as a master for EQ/DFE adaption (**RECOMMENDED SETTING)**
	4	Cross Point Enable 1: Cross Point Switching is enabled for this channel. 0: Cross Point Switching is disabled for this channel. Cross Point configuration bits have no effect when this bit is set to 0.
	3 2	EQ Buffer Select 11: Both buffers ON **(RECOMMENDED SETTING)** 10: Local buffer OFF, multi point buffer ON 01: Local buffer ON, multi point buffer OFF 00: Both buffers OFF
	1 0	Channel's Mux Setting to Select EQ Data 00: channel A 01: Channel B 10: Channel C 11: Channel D Channel A = 0,4,8,12 Channel B = 1,5,9,13 Channel C = 2,6,10,14 Channel D = 3,7,11,15
0x9B	1 0	Channel's Mux Setting to Select EQ Control Bus 00: channel A 01: Channel B 10: Channel C 11: Channel D Channel A = 0,4,8,12 Channel B = 1,5,9,13 Channel C = 2,6,10,14 Channel D = 3,7,11,15

Below is an example to show the register writes necessary to configure Quad 0 such that the input of channel 0 is routed to the output of channel 1, while the input of channel 1 is routed to the outputs of channel 0.

Share Register	Write Value	Mask	Comment
0xFC	0x01	0xFF	Select channel registers of Quad 0 channel 0
0xFD	0x00	0xFF	
0xFF	0x01	0x01	
0x9B	0x01	0x03	Set the CDR of channel 0 to interface with the control bus for the EQ in channel 1
0x96	0x1D	0x3F	Enable cross point switching, enable the local buffer, enable multi point buffer. Set CDR of channel 0 to receive data from the EQ in channel 1
0xFC	0x02	0xFF	Select channel registers of Quad 0 channel 1. Note global registers 0xFD and 0xFF are still properly configured
0x9B	0x00	0x03	Set channel 1 ‘s mux to receive the EQ control bus from the EQ on channel 0
0x96	0x1C	0x3F	Enable cross point switching, enable the local buffer, enable multi point buffer. Configure the channel 1 mux to receive data from the EQ on channel 0

Note that in this example channel 1 was assigned master control over channel 0’s EQ. Since channel 1 is the only channel connected to the EQ of channel 0, it must have master control over the EQ.

Continuing with this example, if it was desired to switch back to normal operation where channel 0 gets its data from the EQ on channel 0, then the cross point enable bit can simply be disabled. Users should avoid turning off both the local buffer and multi-drive buffer when using the cross point. When configuring for a point to point switch, users should have both the local and multi-drive buffer enabled.

2.21. CTLE and DFE Adaption Mode Select

The DS125DF1610 has 3 adaption modes and the option perform no CTLE/DFE adaption so that these blocks can be configured manually. The adapt mode options are:

1. Mode 0: No adaption. CTLE and DFE are configured only through manual programming
2. Mode 1: CTLE only. CTLE is adapted to an optimal level.
3. Mode 2: CTLE and DFE. CTLE is adapted to an optimal level, then the DFE is adapted to an optimal level.
4. Mode 3: CTLE and DFE. CTLE is adapted to a minimum figure of merit level, next the DFE is adapted to an optimal level. Then the CTLE is re-adapted to an optimal level.

To configure the adapt mode write to channel register 0x31, bits 6 and 5.

Channel Register	Write Value	Mask
0x31	0x60: Adapt Mode 3 0x40: Adapt Mode 2 0x20: Adapt Mode 1 0x00: Adapt Mode 0	0x60

As described in, Enable the DFE, the DFE is powered down by default and must be manually powered on for the applicable adapt modes.

Channel Register	Write Value	Mask
0x1E	0x08: DFE is powered down (default) 0x00: DFE is powered on	0x08

The DFE can further be configured to adapt during lock acquisition then freeze or continuously adapt. To select between freeze after lock and continuous adaption write to channel register 0x7F bit 4.

Channel Register	Write Value	Mask
0x7F	0x10: Continuous DFE adaption 0x00: DFE settings freeze after adaption during lock acquisition	0x10

2.22. Figure of Merit for Adaption

The CTLE and DFE circuits are adapted according to a figure of merit (FoM). The figure of merit is the result of a calculation involving the measured HEO and VEO for each particular adaption setting. The CTLE and DFE use their own FoM for adaption.

There are two FoM equations available for use.

- Original FoM – allows for adapting HEO only, VEO only or HEO and VEO
- New/Alt FoM – allows for scaling the weight of either HEO or VEO for the overall calculation.

The original FoM for the DFE is configured in channel register 0x2C.

Channel Register	Bit	Description
0x2C	5	DFE Original FoM 00: Reserved 01: HEO only 10: VEO only 11: HEO and VEO
	4	
This register is ignored if the alternate FoM is used, channel register 0x6E[6] =1		

The original FoM for the CTLE is configured in channel register 0x31.

Channel Register	Bit	Description
0x31	5	CTLE Original FoM 00: Reserved 01: HEO only 10: VEO only 11: HEO and VEO
	4	
This register is ignored if the alternate FoM is used, channel register 0x6E[7] =1		

The alternate FoM for the DFE and CTLE are enabled in channel register 0x6E.

Channel Register	Bit	Description
0x6E	7	CTLE Alternate FoM 1: Select alternate FoM 0: Use FoM as defined in ch reg 0x31[5:4]
	6	DFE Alternate FoM 1: Select alternate FoM 0: Use FoM as defined in ch reg 0x31[5:4]

The equations for scaling HEO versus VEO for the alternate FoM are shown below.

- HEO_ALT = (HEO – B) x A x 2
- VEO_ALT = (VEO – C) x (1 – A) x 2
- FoM_ALT = HEO_ALT + VEO_ALT

Variables A, B, and C for the alternate FoM calculation are set in channel register 0x6B, 0x6C, and 0x6D.

Channel Register	Bit	Description
0x6B	7:0	Variable A for Alternate FoM A = 0x6B[7:0] ÷ 128 The max value of this register should be 128

Channel Register	Bit	Description
0x6C	7:0	Variable B for Alternate FoM HEO adjustment for the alternate FoM

Channel Register	Bit	Description
0x6D	7:0	Variable C for Alternate FoM VEO adjustment for the alternate FoM

2.23. CTLE Bandwidth Select

The bandwidth of each EQ in the DS125DF1610 can be independently set to one of three values. The EQ defaults to the highest bandwidth setting, so if the channel will operate at slower data rates then it is recommended to consider lower the bandwidth of the EQ to achieve a greater signal to noise ratio. To change the bandwidth of a CTLE, write to channel register 0x8D bits 6 and 2.

Channel Register	Write Value	Mask
0x8D	0x44: EQ half rate mode. Lowest bandwidth 0x04: EQ mid-rate mode. 0x00: EQ full rate mode. Highest bandwidth	0x44

2.24. Power Down Driver

The driver for each channel is normally powered on or off along with the other functional block in a channel depending on if the receiver detect a signal or not. However, it is possible to manually power down the output driver by writing to channel register 0x15 bit 3.

Channel Register	Bit	Description

0x15	3	1: Power down output driver 0: normal operation, driver will be enabled is a signal detect is high.

2.25. Driver Edge Rate Select

The output driver edge rate for each channel in the DS125DF1610 can be independently configured by writing to channel register 0x8E bits 4, 3 and 2. For fast data rates, it is recommended to leave the edge rate at its default, fastest setting. For slower data rates slower edge rates may be desired to help reduce EMI.

Channel Register	Bit	Description
0x8E	4	111: minimum edge rate, fastest setting, 35 ps TYP (default) 110: 40 ps TYP 101: 45 ps TYP 100: 50 ps TYP 011: 55 ps TYP 010: 60 ps TYP 001: 65ps TYP 000: maximum edge rate, slowest setting 70 ps TYP
	3	
	2	

2.26. Simple Driver VOD Control

The output differential voltage (VOD) of each driver can be set two ways. For the best performance see Advanced Driver VOD Control. The simple method requires configuring the five drv_sel_vod bits. These bits are located in channel registers 0x0D bits 5 and 4, and 0x2D bits 2, 1, and 0. When using only these drv_sel_vod bits, the step size between VOD levels is approximately 43mVp-p. Max VOD is 1400mVp-p TYP and is set by writing a 1 to all drv_sel_vod bits. Min VOD is 53mVp-p and is set by writing a 0 to all drv_sel_vod bits.

drv_sel_vod [4:0] Setting	VOD
00000	53.3
00001	105.3
00010	153.3
00011	193.3
00100	237.3
00101	273.3
00110	317.3
00111	349.3
01000	389.3
01001	437.3
01010	489.3
01011	529.3
01100	577.3
01101	633.3
01110	669.3
01111	721.3

drv_sel_vod [4:0] Setting	VOD
10000	753.3
10001	801.3
10010	845.3
10011	889.3
10100	921.3
10101	973.3
10110	1013.3
10111	1057.3
11000	1101.3
11001	1145.3
11010	1189.3
11011	1233.3
11100	1277.3
11101	1313.3
11110	1353.3
11111	1393.3

2.27. Advanced Driver VOD Control

For the best performance, the output VOD can be configured by manipulating the FIR filter, the drv_sel_vod bits and the pre driver degeneration bits (drv_dem). For the best performance, the VOD bits are set to the high setting possible, then the pre driver degeneration bits are used in conjunction with the FIR main cursor to bring the VOD down to the desired level. Small amounts of post cursor are implemented to optimize for jitter.

Channel Register	Bit	Description
0x0D	5	drv_sel_vod(4)
	4	drv_sel_vod(3)
0x2D	2	drv_sel_vod(2)
	1	drv_sel_vod(1)
	0	drv_sel_vod(0)

Channel Register	Bit	Description
0x15	1:0	DEM Internal VOD degeneration 00: 0dB 01: -1dB 10: -2dB 11: -3dB

Channel Register	Bit	Description
0x3D	6	FIR Main cursor sign bit 1: Negative 0: Positive
	5:0	FIR Main Cursor Setting Bit5 MSB Bit 0 LSB

Channel Register	Bit	Description
0x3F	6	FIR Post cursor sign bit 1: Negative 0: Positive
	5:0	FIR Post Cursor Setting Bit5 MSB Bit 0 LSB

Channel Register	Bit	Description
0x3E	6	FIR Pre cursor sign bit 1: Negative 0: Positive
	5:0	FIR Pre Cursor Setting Bit5 MSB Bit 0 LSB

The table below contains some example configurations for commonly used VOD levels. Note that depending on application circumstances it may be desirable to have additional amounts of pre or post cursor boost to achieve better signal integrity across the channel. For information about how to use the FIR filter for transmit equalization see the FIR Filter Configuration section.

Approximate VOD	DEM Setting	VOD Setting	FIR		
			Pre	Main	Post
1200	0	31	0	56	-4
1150	0	31	0	52	-4
1100	0	31	0	49	-4
1050	0	31	0	45	-4
1000	2	31	0	54	-3
950	3	31	0	56	-2
900	3	31	0	52	-2
850	3	31	0	46	-1
800	3	31	0	42	-1
750	3	25	0	56	-3
700	3	25	0	46	-2
650	3	25	0	40	-1
600	3	21	0	50	-2
550	3	19	0	50	-3
500	3	17	0	52	-3
450	3	15	0	50	-3
400	3	13	0	52	-3
350	3	12	0	51	-3
300	3	10	0	51	-3
250	3	8	0	55	-3
200	3	6	0	56	-2
150	3	4	0	57	-3

Below is an example register write sequence to set the VOD for channel 0 to the 800mVp-p setting described in the table above.

Channel Register	Write Value	Mask	Comments
0xFC	0x01	0xFF	Select channel 1
0xFD	0x00	0xFF	
0xFF	0x01	0x01	Select the channel registers for the channels in 0xFC and 0xFD
0x0D	0x30	0x30	Set the drv_sel_vod bits to 31 decimal
0x2D	0x07	0x07	
0x15	0x03	0x03	Set the drv_dem bits to 3 decimal
0x3D	0x2A	0x3F	Set the main cursor to +42
0x3F	0x01	0x3F	Set the post cursor to -1

2.28. CDR Reset

The CDR of each channel in the DS125DF1610 can be manually reset by writing to channel register 0x0A, bits 4, 3, and 2. Note that these bits are not self-clearing, so they must be cleared in order for the CDR to exit the reset state. It is recommend to perform a CDR reset after changing any settings which involves the CDR. Some of these features include:

- adapt mode,
- reference mode,
- cross point switch configuration,
- rate/sub-rate configuration,
- etc.

An example for performing a CDR reset and reset -release on channel 0 is described below.

Channel Register	Write Value	Mask	Comments
0xFC	0x01	0xFF	Select Channel 0
0xFD	0x00	0xFF	
0xFF	0x01	0x01	Select the channel registers for the channels selected in 0xFC and 0xFD
0x0A	0x1C	0x1C	Reset the CDR, then return to normal operation
0x0A	0x00	0x1C	

2.29. Reference Mode Select

By default the DS125DF1610 is configured to operate in reference mode 3. This is the recommended reference mode. If a reference clock is not available, the DS125DF1610 can be made to operate in a reference-less mode, reference mode 0. Reference mode 0 can be used for simple laboratory check and experiments, but it is recommended to use reference mode 3 in all applications. Note if the reference mode is changed at any time, a CDR Reset must be initiated. To change the reference mode for a particular channel, write to channel register 0x36 bits 5 and 4.

Channel Register	Bit	Description
0x36	5	00: Reference-less operation. To be used only for debug, not recommended for normal use 11: Recommended setting. Reference clock required.
	4	

2.30. Rate/Sub-rate Select

Reference mode 3 operation requires that the DS125DF1610 be preprogrammed with the expected data rates. Users can program the rate/sub-rate table to quickly configure the DS125DF1610 for any data rates listed in the table. If an application requires a data rate that exists within the VCO range or the divide by 2, 4 or 8 VCO ranges, but is not listed in the rate sub rate table then those data rate parameters must be manually programmed into the device. For instructions on how to manually program data the DS125DF1610 to operate at data rates not listed in the rate/sub-rate table, please see the Manual Data Rate Programming.

The rate/sub-rate settings are programmed by writing to channel register 0x2F bits 7, 6, 5 and 4.

Channel Register	Write Value	Mask
0x2F	0x60: 11.5Gbps 0x70: 12.5Gbps, 6.25Gbps, 3.125Gbps 0x80: 9.8304Gbps, 4.9152Gbps, 2.4576Gbps 0x90: 6.144Gbps, 3.072Gbps 0xA0: 10Gbps, 5Gbps, 2.5Gbps 0xB0: 10.3125Gbps, 1.25Gbps	0xF0

2.31. Manual Data Rate Programming

Reference mode 3 operation requires that the DS125DF1610 be preprogrammed with the expected data rates. If the desired application data rate does not exist in the rate/sub-rate options for register 0x2F, then the parameters for the desired data rate must be programmed into the DS125DF1610 manually. Manual programming of application data rates includes selecting divider groups or forcing a specific divider, and setting the PPM offset error threshold.

The following instructions for calculating the necessary values and programming the registers for manual data rate configuration can also be found in the DS125DF1610 datasheet.

- Select a divider grouping from the Rate/Sub-rate Table and program that value to channel register 0x2F. When manually programming the data rate into the device, other rate/sub-rate values may be used to allow for different divider and group combinations. A list of all preprogrammed divider, group combinations is shown in the table below.

Channel Register 0x2F[7:4] Setting	First Group Divider Settings	Second Group Divider Settings
0x0	2, 4	2, 4
0x1	1	1
0x2	1, 2, 4	1, 2, 4
0x3	1, 2, 4	1, 2, 4
0x4	1	1
0x5	1	1
0x6	1	1
0x7	1, 2, 4	1, 2, 4
0x8	1, 2, 4	1, 2, 4
0x9	2, 4	2, 4
0xA	1, 2, 4	1, 2, 4
0xB	8	1
0xC	8	1
0xD	1, 2, 4	1
0xE	1	1
0xF	1, 2	1, 2

- To calculate the settings for the first group, use the table below.

Parameter	Value/Equation	Comment
Reference Clock	F0 = 25e6	Internally the reference clock always operates at 25 MHz
Desired VCO Frequency	F1	F1 is the frequency of the VC0 which is equal to the Desired VCO Frequency F1 desired data rate. If the desired data rate uses dividers, be sure to multiply the data rate by the divide setting to get the correct VCO frequency
Number of Reference Clocks	N = 1024	
VCO Freq ÷ 32	F2 = F1 ÷ 32	
Counts of VCO Freq ÷ 32 required	F3 = F2 x N ÷ F0	
Counts of VCO Freq ÷ 32 required rounded	F4	Round F3 to the nearest integer value. Convert this value to binary. Program the upper 8 bits to ch register 0x61 and the lower 8 bits to ch register 0x60. Be sure to set channel register 0x61[7] to 1 to enable the override function for manual programming.
PPM error due to rounding	Err = 1e6 x (F4 – F3) ÷ F3	
Required PPM tolerance	T	Enter the desired PPM tolerance. (Example T = 1000 PPM)
VCO Freq ÷ 32 +PPM tolerance	F5 = (1+ T÷1e6) x F2	
Rounded Counts of the VCO Freq ÷ 32 +PPM tolerance required	F6 = F5 x N ÷ F0	Round F6 to the nearest integer value
PPM Counts delta	F7 = F6 – F3	Convert this value to binary. Program the most significant bit channel register 0x67[7] and the rest of the bits to channel register 0x64[7:4]

- To calculate the settings for the second group, use the table below.

Parameter	Value/Equation	Comment
Reference Clock	F0 = 25e6	Internally the reference clock always operates at 25 MHz
Desired VCO Frequency	F1	F1 is the frequency of the VC0 which is equal to the Desired VCO Frequency F1 desired data rate. If the desired data rate uses dividers, be sure to multiply the data rate by the divide setting to get the correct VCO frequency
Number of Reference Clocks	N = 1024	
VCO Freq ÷ 32	F2 = F1 ÷ 32	
Counts of VCO Freq ÷ 32 required	F3 = F2 x N ÷ F0	
Counts of VCO Freq ÷ 32 required rounded	F4	Round F3 to the nearest integer value. Convert this value to binary. Program the upper 8 bits to ch register 0x63 and the lower 8 bits to ch register 0x62. Be sure to set channel register 0x63[7] to 1 to enable the override function for manual programming.
PPM error due to rounding	Err = 1e6 x (F4 – F3) ÷ F3	
Required PPM tolerance	T	Enter the desired PPM tolerance. (Example T = 1000 PPM)
VCO Freq ÷ 32 +PPM tolerance	F5 = (1+ T÷1e6) x F2	
Rounded Counts of the VCO Freq ÷ 32 +PPM tolerance required	F6 = F5 x N ÷ F0	Round F6 to the nearest integer value
PPM Counts delta	F7 = F6 – F3	Convert this value to binary. Program the most significant bit channel register 0x67[6] and the rest of the bits to channel register 0x64[3:0]

An example for setting the first and second groups to 11.3 Gbps is shown in the table below.

Channel Register (Hex)	Value
0x60	0x80
0x61	0xB8
0x62	0x80
0x63	0xB8
0x64	0xEE
0x67[7:6]	2'b00

2.32. FIR Filter Configuration

The DS125DF1610 uses a transmit FIR filter to pre-distort the output waveform for the purposes of transmit equalization. The FIR filter consists of three taps: 1) Main cursor, 2) Pre cursor, and 3) Post cursor. Each tap can have a range from 0 to 63 with either a positive or negative polarity. For normal operation the main cursor should be set to a positive value with the pre and post cursors set to a negative value. If the polarity of the output waveform needs to be inverted then the main cursor polarity can be set to negative, while the pre and post cursor polarity should be set to positive.

When using the FIR for transmit equalization it is important to first know which method of driver VOD control is being used. If the FIR is used transmit equalization with simple VOD control then the absolute values of each tap should sum to 63.

$$|Pre| + |Main| + |Post| = 63$$

If transmit equalization is required when using advanced VOD control, then the absolute values for each tap should sum to the default sum for that particular VOD.

$$|Pre| + |Main| + |Post| = default\ sum$$

For example if the advanced VOD control method is used to configure the drive for 800mVp-p VOD, then the default sum is 43.

Typically pre and post cursor boost is added at the cost of the main cursor; meaning that for each additional amount of pre and post cursor added for boost, the same amount is subtracted from the main cursor so that the absolute value of each tap sums to the default sum.

Channel Register	Bit	Description
0x3D	6	FIR Main cursor sign bit 1: Negative 0: Positive
	5:0	FIR Main Cursor Setting Bit 5 MSB Bit 0 LSB

Channel Register	Bit	Description
0x3F	6	FIR Post cursor sign bit 1: Negative 0: Positive
	5:0	FIR Post Cursor Setting Bit 5 MSB Bit 0 LSB

Channel Register	Bit	Description
0x3E	6	FIR Pre cursor sign bit 1: Negative 0: Positive
	5:0	FIR Pre Cursor Setting Bit 5 MSB Bit 0 LSB

3. Debug Features

3.1. PRBS Pattern Generator

The PRBS pattern generator can be enabled to generate PRBS-7, 9, 15, or 31.The PRBS generator requires that a high speed signal be present at the inputs of the desired channel and the CDR to be locked. The PRBS generator in each channel is independent from the other channels and can be enabled at any time.

To configure the PRBS generator use the following registers:

Channel Register	Bit	Description
0x1E	7 6 5	100: Configures CDR to output PRBS data
	4	1: Enable the serializer 0: Disable the serializer
0x09	5	1: Override enable, enables settings in 0x1E[7:5] to take effect 0: Normal operation
0x79	6	1: Enable PRBS Checker, should not be set if PRBS generator is enabled 0: Disable PRBS Checker
	5	1: Enable PRBS generator, should not be set if PRBS checker is enabled 0: Disable PRBS generator
0x30	3	1: Enable digital clock 0: Disable digital clock
	1 0	00: PRBS-7 01: PRBS-9 10: PRBS-15 11: PRBS-31

In order to change the pattern of the PRBS generator the digital clock must also be disabled and enabled.

3.2. User Pattern Generator

The user pattern generator, like the PRBS generator, requires that a high speed signal be present at the inputs of the desired channel. Users can define a 16-bit sequence to be looped back continuously. Users also have the option to have the user pattern generator automatically invert every other 16-bit word for simple DC balance purposes. These features allow users to create various clock patterns, including the common 8T pattern (8-ones and 8-zeros) or various other customer 16-bit sequences.

To configure the user pattern generator use the following registers:

Channel Register	Bit	Description
0x1E	7	100: Configures CDR to output user pattern data
	6	
	5	
	4	1: Enable the serializer 0: Disable the serializer
0x09	5	1: Override enable, enables settings in 0x1E[7:5] to take effect 0: Normal operation
0x7C	7:0	Lower 8-bits of the user defined pattern (LSB)
0x97	7:0	Upper 8-bits of the user defined pattern (MSB)
0x30	5	1: Invert every other 16-bit word 0: Normal operation, 16-bit sequence is shifted out without auto inversion
	2	1: Enables the user defined pattern generator 0: Disables the user defined pattern generator

3.3. PRBS Pattern Checker

The PRBS checker has a 47-bit data counter and 11-bit error counter that are used to collect bit error statistics. The PRBS checker will automatically lock to PRBS-7, 9, 15, or 31. The PRBS checker will also automatically detect the polarity of the incoming PRBS pattern. Users can run the PRBS checker for as short or as long as they wish, so long as the run time does not cause the data counter to max out.

Note: The PRBS checker and PRBS generator should not be enabled at the same time within the same channel. It is OK to have various PRBS generators and checker enabled at the same time, as long as they are in different channels.

To configure the PRBS checker use the following registers:

Channel Register	Bit	Description
0x0D	7	0: Turn on the deserializer 1: Turn off the deserializer
0x79	6	1: Enable PRBS Checker 0: Disable PRBS Checker
	5	1: Enable generator 0: Disable generator
0x30	3	1: Enable digital clock 0: Disable digital clock

To read the PRBS checker results use the following registers:

Channel Register	Bit	Description
0x82	7	1: Freeze PRBS checker counters 0: PRBS counters are free to increment
0x83	2:0	Error count MSBs
0x84	7:0	Error count LSBs
0x85	7:0	Total bit count [46:0] 0x85 contains the MSBs 0x8A contains the LSBs
0x86	7:0	
0x87	7:0	
0x88	7:0	
0x89	7:0	
0x8A	7:0	

The PRBS checker must be enabled for PRBS pattern detection to function. Users can find the detected PRBS pattern and polarity from the PRBS checker in channel register 0x01.

Channel Register	Bit	Description
0x01	6	1: PRBS pattern polarity inversion detected 0: PRBS pattern is true, no polarity inversion detected
	4	1: PRBS-31 pattern detected 0: PRBS-31 pattern not detected
	3	1: PRBS-15 pattern detected 0: PRBS-15 pattern not detected
	2	1: PRBS-9 pattern detected 0: PRBS-9 pattern not detected
	1	1: PRBS-7 pattern detected 0: PRBS-7 pattern not detected

If the PRBS pattern changes while the PRBS checker is enabled, the checker will start accumulating errors. The user will need to reset the checker's digital clock in order for the checker to relock to the new pattern and update the pattern detection status in channel register 0x01.

3.4. Reading HEO/VEO

Horizontal Eye Opening and Vertical Eye Opening statistics are updated approximately every 10ms. These results are available in channel register 0x27 (HEO) and 0x28 (VEO). HEO is measured at the 0V eye crossing. VEO is measured at the 0.5UI phase.

To convert the channel register 0x27 information to HEO use the following formula:

2. ch_reg_0x27_val / 64 = HEO [UI]

To convert the channel register 0x28 information to VEO use the following formula:

3. ch_reg_0x28_val x 3.125 = VEO [mV]

3.5. Plotting an Eye Diagram

The Eye Opening Monitor is capable of plotting an equalized, but non-retimed eye to represent the quality of signal that is going into the CDR block. This feature is useful for diagnosing gross signal integrity errors. The eye plot generated by the EOM cannot be used to extrapolate performance to bit error rates beyond $1e^{-7}$.

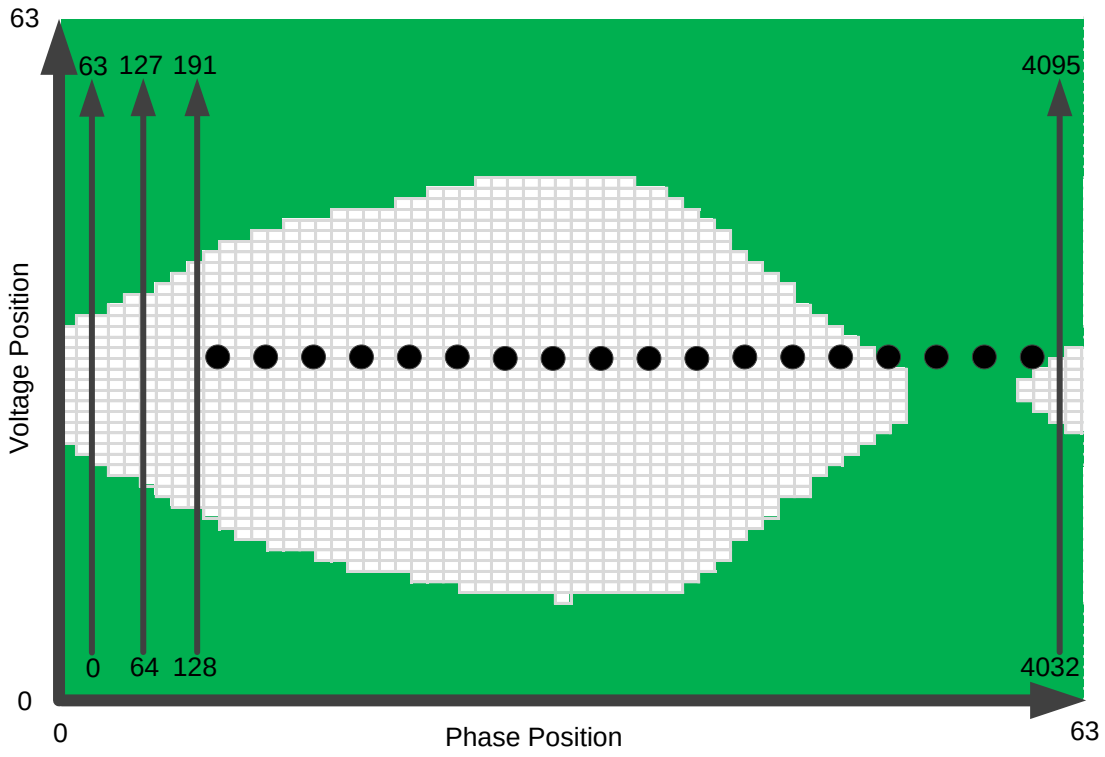
The EOM generates the eye plot according to a 64 x 64 matrix. Each grid space in the matrix consists of a 16-bit word that represents the number of times the incoming data stream touches that particular phase

and voltage level. The EOM under-samples the incoming data stream by recording the total hit count for each combination of phase and voltage, one grid space at a time.

To configure the EOM to plot the eye diagram, sometimes called fast eye capture, use the following registers.

Step	Register[bits]	Value	Description
1	0x67[5]	0	Disable lock EOM lock monitoring
2	0x2C[6] 0x11[7:6]	0 2'b xx	Set the desired EOM vertical range as 2'b xx
3	0x11[5]	0	Power on the EOM
4	0x24[7]	1	Enable fast EOM
5	0x24[0] 0x25 0x26	1	When 24[0] is 1 begin read out of the 64 x 64 array, discard first 4 bytes This bit is self-clearing.
6	0x24[0] 0x25 0x26	1	Continue reading information until the 64 x 64 array is complete.
7	0x67[5] 0x2C[6] 0x11[5] 0x24[7]	1 1 1 0	Return the EOM to its original state. Undo steps 1-4

The data recorded from the Eye Opening Monitor (EOM) begins at (X, Y) position (0, 0) and proceeds to position (0, 63). Next, the Y-value is reset to 0 and the X-value is incremented. This process is repeated until the entire 64 x 64 matrix is read out.



3.6. Interrupts

The DS125DF1610 is capable of generating several different interrupts. These interrupt signals behave as “sticky” bits to signal to a user that an event has occurred. If an interrupt request is generated, the interrupt signal will remain logic HIGH until it has been serviced. Each interrupt has its own enable bit and interrupt status bit in the channel register set. Listed below are the available interrupts along with their status and enable bit locations.

Interrupt	Channel Register	Bit	Description
Signal Detect	0x79	0	Signal Detect Interrupt Enable 1: Enable signal detect interrupt 0: Disable signal detect interrupt
	0x78	2	1: Indicates that the status of signal detect has changed at least once. Can indicate loss of signal detect or signal has been detected. 0: No change in signal detect status
CDR_lock_loss	0x79	1	CDR Lock Interrupt Enable 1: Enables CDR lock loss interrupt 0: Disables CDR lock loss interrupt
	0x78	3	1: CDR has lost lock at least once 0: CDR has not lost lock
PRBS	0x31	7	PRBS Error Interrupt Enable 1: Enables interrupt for PRBS errors 0: Disable PRBS error interrupts
	0x71	7	1: PRBS error is detected, cleared on reading. Subsequent interrupts indicate PRBS errors have occurred. This bit is also set when signal detect is lost 0: No PRBS error detected
Heo_veo	0x36	6	HEO/VEO Interrupt Enable 1: Enables interrupt on HEO/VEO threshold violation; thresholds specified in register 0x32 0: Disable HEO/VEO interrupts
	0x78	0	1: HEO/VEO measurements have dropped below the specified thresholds 0: HEO/VEO still within thresholds

The INTERR_IO# pin of the DS125DF1610 is an open drain pin. This allows multiple devices to connect to the same interrupt line to save on the master ASIC's pin count.

Within the DS125DF1610, each channel is capable of generating its own interrupts independently of other channels. To determine which channel generated the interrupt request, see shared registers 0x08 and 0x09.

Shared Register	Bit	Description	CDR/TX Pin Assignment
0x08	7	Channel 7– Quad 1 Channel 3	TX_3B
	6	Channel 6– Quad 1 Channel 2	TX_3A
	5	Channel 5– Quad 1 Channel 1	TX_2B
	4	Channel 4– Quad 1 Channel 0	TX_2A
	3	Channel 3– Quad 0 Channel 3	TX_1B
	2	Channel 2– Quad 0 Channel 2	TX_1A
	1	Channel 1– Quad 0 Channel 1	TX_0B
	0	Channel 0 – Quad 0 Channel 0	TX_0A
0x09	7	Channel 15– Quad 3 Channel 3	TX_7B
	6	Channel 14– Quad 3 Channel 2	TX_7A
	5	Channel 13– Quad 3 Channel 1	TX_6B
	4	Channel 12– Quad 3 Channel 0	TX_6A
	3	Channel 11– Quad 2 Channel 3	TX_5B
	2	Channel 10– Quad 2 Channel 2	TX_5A
	1	Channel 9– Quad 2 Channel 1	TX_4B
	0	Channel 8– Quad 2 Channel 0	TX_4A

4. SMBus Master Mode

4.1. Configuration by EEPROM

The DS125DF1610 can configure itself by acting as an SMBus master and loading the desired setting from an external EEPROM. To configure the DS125DF1610 SMBus master mode, tie the EN_SMB pin to V_{DD} through a 1kΩ resistor. To enable the DS125DF1610 to load from the EEPROM, the READ_EN pin must be pulled low.

The table below shows the expected parameters of the external EEPROM requirements.

Parameter	EEPROM Spec
Max EEPROM Size Supported	2048 Bytes
Min EEPROM Size for Single DS125DF1610 with Individual Channel Configuration	1417 Bytes
EEPROM SMBus Slave Address	0xA0 (8-bit address)
EEPROM Clock Frequency	400 kHz

4.2. EEPROM Bit Map

The EEPROM configuration bits are extracted from the DS125DF1610 registers sequentially. The table below summarizes the bit names and register positions. For more information about each bit, please refer to the DS125DF1610 datasheet.

Channel Registers										
EEPROM Address Byte			**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Description		0	EQ_BST0[1]	EQ_BST0[0]	EQ_BST1[1]	EQ_BST1[0]	EQ_BST2[1]	EQ_BST2[0]	EQ_BST3[1]	EQ_BST3[0]
Value	0x00		0	0	0	0	0	0	0	0
Description		1	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
Value	0x30		0	0	1	1	0	0	0	0
Description		2	RESERVED	REG_VCO_CAP_OV	REG_SET_CP_LVL_LPF_OV	REG_BYPASS_PFD_OV	REG_EN_FD_PD_VCO_PDIQ_OV	REG_EN_PD_CP_OV	REG_DIVSEL_OV	REG_EN_FLD_OV
Value	0x00		0	0	0	0	0	0	0	0
Description		3	REG_PFD_LOCK_MODE_SM	REG_SBT_EN	REG_EN_IDAC_PD_CP_OV_AND_REG_EN_IDAC_FD_CP_OV	REG_DAC_LPF_HIGH_PHASE_OV_AND_REG_DAC_LPF_LOW_PHASE_OV	REG_EN150_LPF_OV	RESERVED	RESERVED	RESERVED
Value	0x2B		0	0	1	0	1	0	1	1
Description		4	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	SINGLE_BIT_LIMIT_CHECK_ON	EN_IDAC_FD_CP[3]	EN_IDAC_PD_CP[3]
Value	0x7E		0	1	1	1	1	1	1	0
Description		5	DRV_SEL_VOD[4]	DRV_SEL_VOD[3]	FIR_RLOAD_MAX	FIR_SEL_NEG_GM	RESERVED	RESERVED	RESERVED	RESERVED
Value	0XD3		1	1	0	1	0	0	1	1
Description		6	RESERVED	RESERVED	RESERVED	RESERVED	EOM_SEL_VRANGE[1]	EOM_SEL_VRANGE[0]	EOM_PD	DFE_TAP2_POL
Value	0XA2		1	0	1	0	0	0	1	0
Description		7	DFE_TAP3_POL	DFE_TAP4_POL	DFE_TAP5_POL	DFE_TAP1_POL	DFE_SEL_NEG_GM	DFE_WT1[4]	DFE_WT1[3]	DFE_WT1[2]
Value	0x18		0	0	0	1	1	0	0	0

Description		8	DFE_WT1[1]	DFE_WT1[0]	EQ_EN_DC_OFF	RESERVED	EQ_LIMIT_EN	EQ_SD_PRESET	EQ_SD_RESET	EQ_REFA_SEL[1]
Value	0x40		0	0	1	0	0	0	0	0
Description		9	EQ_REFA_SEL[0]	EQ_REFD_SEL[1]	EQ_REFD_SEL[0]	RESERVED	RESERVED	RESERVED	RESERVED	DRV_DEM[1]
Value	0x01		0	0	0	0	0	0	0	1
Description		A	DRV_DEM[0]	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
Value	0x3D		0	0	1	1	1	1	0	1
Description		B	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
Value	0x1B		0	0	0	1	1	0	1	1
Description		C	RESERVED	PDIQ_SEL_DIV[2]	PDIQ_SEL_DIV[1]	PDIQ_SEL_DIV[0]	RESERVED	RESERVED	RESERVED	RESERVED
Value	0x48		0	1	0	0	1	0	0	0
Description		D	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	DRV_SEL_VCM[1]	DRV_SEL_VCM[0]
Value	0x28		0	0	1	0	1	0	0	0
Description		E	CP_EN_CP_PD	CP_EN_CP_FD	CP_EN_IDAC_PD[2]	CP_EN_IDAC_PD[1]	CP_EN_IDAC_PD[0]	CP_EN_IDAC_FD[2]	CP_EN_IDAC_FD[1]	CP_EN_IDAC_FD[0]
Value	0XE4		1	1	1	0	0	1	0	0
Description		F	SBT_EN	PFD_SEL_DATA_MUX[2]	PFD_SEL_DATA_MUX[1]	PFD_SEL_DATA_MUX[0]	DFE_PD	PFD_PD_PD	PFD_EN_FLD	PFD_EN_FD
Value	0x79		0	1	1	1	1	0	0	1
Description		10	RESERVED	RESERVED	DFE_WT5[3]	DFE_WT5[2]	DFE_WT5[1]	DFE_WT5[0]	DFE_WT4[3]	DFE_WT4[2]
Value	0x40		0	1	0	0	0	0	0	0
Description		11	DFE_WT4[1]	DFE_WT4[0]	DFE_WT3[3]	DFE_WT3[2]	DFE_WT3[1]	DFE_WT3[0]	DFE_WT2[3]	DFE_WT2[2]
Value	0x00		0	0	0	0	0	0	0	0

Description		12	DFE_WT2[1]	DFE_WT2[0]	DFE_OV	EOM_TIMER_THR[7]	EOM_TIMER_THR[6]	EOM_TIMER_THR[5]	EOM_TIMER_THR[4]	EOM_TIMER_THR[3]
Value	0x26		0	0	1	0	0	1	1	0
Description		13	EOM_TIMER_THR[2]	EOM_TIMER_THR[1]	EOM_TIMER_THR[0]	RESERVED	RESERVED	EOM_MIN_REQ_HITS[3]	EOM_MIN_REQ_HITS[2]	EOM_MIN_REQ_HITS[1]
Value	0x07		0	0	0	0	0	1	1	1
Description		14	EOM_MIN_REQ_HITS[0]	VEO_SCALE	DFE_SM_FOM[1]	DFE_SM_FOM[0]	DFE_ADAPT_COUTNER[3]	DFE_ADAPT_COUTNER[2]	DFE_ADAPT_COUTNER[1]	DFE_ADAPT_COUTNER[0]
Value	0XF2		1	1	1	1	0	0	1	0
Description		15	PD_SCP	SD_EN_FAST_OOB	SD_REF_HIGH	SD_GAIN	REG_EQ_BST_OV	DRV_SEL_VOD[2]	DRV_SEL_VOD[1]	DRV_SEL_VOD[0]
Value	0x04		0	0	0	0	0	1	0	0
Description		16	RATE[1]	RATE[0]	SUBRATE[1]	SUBRATE[0]	INDEX_OV	EN_PPM_CHECK	EN_FLD_CHECK	EQ_SEARCH_OV_EN
Value	0x36		0	0	1	1	0	1	1	0
Description		17	ADAPT_MODE[1]	ADAPT_MODE[0]	EQ_SM_FOM[1]	EQ_SM_FOM[0]	HEO_INT_THRESH[3]	HEO_INT_THRESH[2]	HEO_INT_THRESH[1]	HEO_INT_THRESH[0]
Value	0x41		0	1	0	0	0	0	0	1

Description		18	VEO_INT_THRESH[3]	VEO_INT_THRESH[2]	VEO_INT_THRESH[1]	VEO_INT_THRESH[0]	HEO_THRESH[3]	HEO_THRESH[2]	HEO_THRESH[1]	HEO_THRESH[0]
Value	0x18		0	0	0	1	1	0	0	0
Description		19	VEO_THRESH[3]	VEO_THRESH[2]	VEO_THRESH[1]	VEO_THRESH[0]	LOW_POWER_MODE_DISABLE	LOCK_COUNTER[1]	LOCK_COUNTER[0]	DFE_MAX_TAP_2_5[3]
Value	0x87		1	0	0	0	0	1	1	1
Description		1A	DFE_MAX_TAP_2_5[2]	DFE_MAX_TAP_2_5[1]	DFE_MAX_TAP_2_5[0]	DATA_LOCK_PPM[1]	DATA_LOCK_PPM[0]	DFE_MAX_TAP_1[4]	DFE_MAX_TAP_1[3]	DFE_MAX_TAP_1[2]
Value	0XE7		1	1	1	0	0	1	1	1
Description		1B	DFE_MAX_TAP_1[1]	DFE_MAX_TAP_1[0]	REF_MODE[1]	REF_MODE[0]	MR_EN_64B66B	MR_CDR_CAP_DAC_RNG_OV	MR_EOM_RATE[1]	MR_EOM_RATE[0]
Value	0XF0		1	1	1	1	0	0	0	0
Description		1C	START_INDEX[4]	START_INDEX[3]	START_INDEX[2]	START_INDEX[1]	START_INDEX[0]	FIXED_EQ_BST0[1]	FIXED_EQ_BST0[0]	FIXED_EQ_BST1[1]
Value	0x00		0	0	0	0	0	0	0	0
Description		1D	FIXED_EQ_BST1[0]	FIXED_EQ_BST2[1]	FIXED_EQ_BST2[0]	FIXED_EQ_BST3[1]	FIXED_EQ_BST3[0]	FIR_PD_PD	FIR_C0_SGN	FIR_C0[5]
Value	0x01		0	0	0	0	0	0	0	1
Description		1E	FIR_C0[4]	FIR_C0[3]	FIR_C0[2]	FIR_C0[1]	FIR_C0[0]	FIR_PD_TX	FIR_CN1_SGN	FIR_CN1[5]
Value	0xAA		1	0	1	0	1	0	1	0
Description		1F	FIR_CN1[4]	FIR_CN1[3]	FIR_CN1[2]	FIR_CN1[1]	FIR_CN1[0]	FIR_SEL_I_MAX	FIR_CP1_SGN	FIR_CP1[5]
Value	0x1E		0	0	0	1	1	1	1	0

Description		20	FIR_CP1[4]	FIR_CP1[3]	FIR_CP1[2]	FIR_CP1[1]	FIR_CP1[0]	EQ_ARRAY_INDEX_0_BST0[1]	EQ_ARRAY_INDEX_0_BST0[0]	EQ_ARRAY_INDEX_0_BST1[1]
Value	0x38		0	0	1	1	1	0	0	0
Description		21	EQ_ARRAY_INDEX_0_BST1[0]	EQ_ARRAY_INDEX_0_BST2[1]	EQ_ARRAY_INDEX_0_BST2[0]	EQ_ARRAY_INDEX_0_BST3[1]	EQ_ARRAY_INDEX_0_BST3[0]	EQ_ARRAY_INDEX_1_BST0[1]	EQ_ARRAY_INDEX_1_BST0[0]	EQ_ARRAY_INDEX_1_BST1[1]
Value	0x00		0	0	0	0	0	0	0	0
Description		22	EQ_ARRAY_INDEX_1_BST1[0]	EQ_ARRAY_INDEX_1_BST2[1]	EQ_ARRAY_INDEX_1_BST2[0]	EQ_ARRAY_INDEX_1_BST3[1]	EQ_ARRAY_INDEX_1_BST3[0]	EQ_ARRAY_INDEX_2_BST0[1]	EQ_ARRAY_INDEX_2_BST0[0]	EQ_ARRAY_INDEX_2_BST1[1]
Value	0x08		0	0	0	0	1	0	0	0
Description		23	EQ_ARRAY_INDEX_2_BST1[0]	EQ_ARRAY_INDEX_2_BST2[1]	EQ_ARRAY_INDEX_2_BST2[0]	EQ_ARRAY_INDEX_2_BST3[1]	EQ_ARRAY_INDEX_2_BST3[0]	EQ_ARRAY_INDEX_3_BST0[1]	EQ_ARRAY_INDEX_3_BST0[0]	EQ_ARRAY_INDEX_3_BST1[1]
Value	0x20		0	0	1	0	0	0	0	0
Description		24	EQ_ARRAY_INDEX_3_BST1[0]	EQ_ARRAY_INDEX_3_BST2[1]	EQ_ARRAY_INDEX_3_BST2[0]	EQ_ARRAY_INDEX_3_BST3[1]	EQ_ARRAY_INDEX_3_BST3[0]	EQ_ARRAY_INDEX_4_BST0[1]	EQ_ARRAY_INDEX_4_BST0[0]	EQ_ARRAY_INDEX_4_BST1[1]
Value	0x82		1	0	0	0	0	0	1	0
Description		25	EQ_ARRAY_INDEX_4_BST1[0]	EQ_ARRAY_INDEX_4_BST2[1]	EQ_ARRAY_INDEX_4_BST2[0]	EQ_ARRAY_INDEX_4_BST3[1]	EQ_ARRAY_INDEX_4_BST3[0]	EQ_ARRAY_INDEX_5_BST0[1]	EQ_ARRAY_INDEX_5_BST0[0]	EQ_ARRAY_INDEX_5_BST1[1]
Value	0x00		0	0	0	0	0	0	0	0
Description		26	EQ_ARRAY_INDEX_5_BST1[0]	EQ_ARRAY_INDEX_5_BST2[1]	EQ_ARRAY_INDEX_5_BST2[0]	EQ_ARRAY_INDEX_5_BST3[1]	EQ_ARRAY_INDEX_5_BST3[0]	EQ_ARRAY_INDEX_6_BST0[1]	EQ_ARRAY_INDEX_6_BST0[0]	EQ_ARRAY_INDEX_6_BST1[1]
Value	0x40		0	1	0	0	0	0	0	0
Description		27	EQ_ARRAY_INDEX_6_B	EQ_ARRAY_INDEX_6_B	EQ_ARRAY_INDEX_6_B	EQ_ARRAY_INDEX_6_B	EQ_ARRAY_INDEX_6_B	EQ_ARRAY_INDEX_7_B	EQ_ARRAY_INDEX_7_B	EQ_ARRAY_INDEX_7_B

			ST1[0]	ST2[1]	ST2[0]	ST3[1]	ST3[0]	ST0[1]	ST0[0]	ST1[1]
Value	0x14		0	0	0	1	0	1	0	0
Description		28	EQ_ARRAY_INDEX_7_BST1[0]	EQ_ARRAY_INDEX_7_BST2[1]	EQ_ARRAY_INDEX_7_BST2[0]	EQ_ARRAY_INDEX_7_BST3[1]	EQ_ARRAY_INDEX_7_BST3[0]	EQ_ARRAY_INDEX_8_BST0[1]	EQ_ARRAY_INDEX_8_BST0[0]	EQ_ARRAY_INDEX_8_BST1[1]
Value	0x00		0	0	0	0	0	0	0	0
Description		29	EQ_ARRAY_INDEX_8_BST1[0]	EQ_ARRAY_INDEX_8_BST2[1]	EQ_ARRAY_INDEX_8_BST2[0]	EQ_ARRAY_INDEX_8_BST3[1]	EQ_ARRAY_INDEX_8_BST3[0]	EQ_ARRAY_INDEX_9_BST0[1]	EQ_ARRAY_INDEX_9_BST0[0]	EQ_ARRAY_INDEX_9_BST1[1]
Value	0x18		0	0	0	1	1	0	0	0
Description		2A	EQ_ARRAY_INDEX_9_BST1[0]	EQ_ARRAY_INDEX_9_BST2[1]	EQ_ARRAY_INDEX_9_BST2[0]	EQ_ARRAY_INDEX_9_BST3[1]	EQ_ARRAY_INDEX_9_BST3[0]	EQ_ARRAY_INDEX_10_BST0[1]	EQ_ARRAY_INDEX_10_BST0[0]	EQ_ARRAY_INDEX_10_BST1[1]
Value	0x61		0	1	1	0	0	0	0	1
Description		2B	EQ_ARRAY_INDEX_10_BST1[0]	EQ_ARRAY_INDEX_10_BST2[1]	EQ_ARRAY_INDEX_10_BST2[0]	EQ_ARRAY_INDEX_10_BST3[1]	EQ_ARRAY_INDEX_10_BST3[0]	EQ_ARRAY_INDEX_11_BST0[1]	EQ_ARRAY_INDEX_11_BST0[0]	EQ_ARRAY_INDEX_11_BST1[1]
Value	0x82		1	0	0	0	0	0	1	0
Description		2C	EQ_ARRAY_INDEX_11_BST1[0]	EQ_ARRAY_INDEX_11_BST2[1]	EQ_ARRAY_INDEX_11_BST2[0]	EQ_ARRAY_INDEX_11_BST3[1]	EQ_ARRAY_INDEX_11_BST3[0]	EQ_ARRAY_INDEX_12_BST0[1]	EQ_ARRAY_INDEX_12_BST0[0]	EQ_ARRAY_INDEX_12_BST1[1]
Value	0x0A		0	0	0	0	1	0	1	0
Description		2D	EQ_ARRAY_INDEX_12_BST1[0]	EQ_ARRAY_INDEX_12_BST2[1]	EQ_ARRAY_INDEX_12_BST2[0]	EQ_ARRAY_INDEX_12_BST3[1]	EQ_ARRAY_INDEX_12_BST3[0]	EQ_ARRAY_INDEX_13_BST0[1]	EQ_ARRAY_INDEX_13_BST0[0]	EQ_ARRAY_INDEX_13_BST1[1]
Value	0x86		1	0	0	0	0	1	1	0
Description		2E	EQ_ARRAY_INDEX_13_BST1[0]	EQ_ARRAY_INDEX_13_BST2[1]	EQ_ARRAY_INDEX_13_BST2[0]	EQ_ARRAY_INDEX_13_BST3[1]	EQ_ARRAY_INDEX_13_BST3[0]	EQ_ARRAY_INDEX_14_BST0[1]	EQ_ARRAY_INDEX_14_BST0[0]	EQ_ARRAY_INDEX_14_BST1[1]
Value	0x03		0	0	0	0	0	0	1	1

Description		2F	EQ_ARRAY_INDEX_14_BST1[0]	EQ_ARRAY_INDEX_14_BST2[1]	EQ_ARRAY_INDEX_14_BST2[0]	EQ_ARRAY_INDEX_14_BST3[1]	EQ_ARRAY_INDEX_14_BST3[0]	EQ_ARRAY_INDEX_15_BST0[1]	EQ_ARRAY_INDEX_15_BST0[0]	EQ_ARRAY_INDEX_15_BST1[1]
Value	0x04		0	0	0	0	0	1	0	0
Description		30	EQ_ARRAY_INDEX_15_BST1[0]	EQ_ARRAY_INDEX_15_BST2[1]	EQ_ARRAY_INDEX_15_BST2[0]	EQ_ARRAY_INDEX_15_BST3[1]	EQ_ARRAY_INDEX_15_BST3[0]	EQ_ARRAY_INDEX_16_BST0[1]	EQ_ARRAY_INDEX_16_BST0[0]	EQ_ARRAY_INDEX_16_BST1[1]
Value	0x84		1	0	0	0	0	1	0	0
Description		31	EQ_ARRAY_INDEX_16_BST1[0]	EQ_ARRAY_INDEX_16_BST2[1]	EQ_ARRAY_INDEX_16_BST2[0]	EQ_ARRAY_INDEX_16_BST3[1]	EQ_ARRAY_INDEX_16_BST3[0]	EQ_ARRAY_INDEX_17_BST0[1]	EQ_ARRAY_INDEX_17_BST0[0]	EQ_ARRAY_INDEX_17_BST1[1]
Value	0x44		0	1	0	0	0	1	0	0
Description		32	EQ_ARRAY_INDEX_17_BST1[0]	EQ_ARRAY_INDEX_17_BST2[1]	EQ_ARRAY_INDEX_17_BST2[0]	EQ_ARRAY_INDEX_17_BST3[1]	EQ_ARRAY_INDEX_17_BST3[0]	EQ_ARRAY_INDEX_18_BST0[1]	EQ_ARRAY_INDEX_18_BST0[0]	EQ_ARRAY_INDEX_18_BST1[1]
Value	0x15		0	0	0	1	0	1	0	1
Description		33	EQ_ARRAY_INDEX_18_BST1[0]	EQ_ARRAY_INDEX_18_BST2[1]	EQ_ARRAY_INDEX_18_BST2[0]	EQ_ARRAY_INDEX_18_BST3[1]	EQ_ARRAY_INDEX_18_BST3[0]	EQ_ARRAY_INDEX_19_BST0[1]	EQ_ARRAY_INDEX_19_BST0[0]	EQ_ARRAY_INDEX_19_BST1[1]
Value	0x02		0	0	0	0	0	0	1	0
Description		34	EQ_ARRAY_INDEX_19_BST1[0]	EQ_ARRAY_INDEX_19_BST2[1]	EQ_ARRAY_INDEX_19_BST2[0]	EQ_ARRAY_INDEX_19_BST3[1]	EQ_ARRAY_INDEX_19_BST3[0]	EQ_ARRAY_INDEX_20_BST0[1]	EQ_ARRAY_INDEX_20_BST0[0]	EQ_ARRAY_INDEX_20_BST1[1]
Value	0x32		0	0	1	1	0	0	1	0
Description		35	EQ_ARRAY_INDEX_20_BST1[0]	EQ_ARRAY_INDEX_20_BST2[1]	EQ_ARRAY_INDEX_20_BST2[0]	EQ_ARRAY_INDEX_20_BST3[1]	EQ_ARRAY_INDEX_20_BST3[0]	EQ_ARRAY_INDEX_21_BST0[1]	EQ_ARRAY_INDEX_21_BST0[0]	EQ_ARRAY_INDEX_21_BST1[1]
Value	0x94		1	0	0	1	0	1	0	0
Description		36	EQ_ARRAY_INDEX_21_BST1[0]	EQ_ARRAY_INDEX_21_BST2[1]	EQ_ARRAY_INDEX_21_BST2[0]	EQ_ARRAY_INDEX_21_BST3[1]	EQ_ARRAY_INDEX_21_BST3[0]	EQ_ARRAY_INDEX_22_BST0[1]	EQ_ARRAY_INDEX_22_BST0[0]	EQ_ARRAY_INDEX_22_BST1[1]

Value	0x65		0	1	1	0	0	1	0	1
Description		37	EQ_ARRAY_INDEX_22_BST1[0]	EQ_ARRAY_INDEX_22_BST2[1]	EQ_ARRAY_INDEX_22_BST2[0]	EQ_ARRAY_INDEX_22_BST3[1]	EQ_ARRAY_INDEX_22_BST3[0]	EQ_ARRAY_INDEX_23_BST0[1]	EQ_ARRAY_INDEX_23_BST0[0]	EQ_ARRAY_INDEX_23_BST1[1]
Value	0x16		0	0	0	1	0	1	1	0
Description		38	EQ_ARRAY_INDEX_23_BST1[0]	EQ_ARRAY_INDEX_23_BST2[1]	EQ_ARRAY_INDEX_23_BST2[0]	EQ_ARRAY_INDEX_23_BST3[1]	EQ_ARRAY_INDEX_23_BST3[0]	EQ_ARRAY_INDEX_24_BST0[1]	EQ_ARRAY_INDEX_24_BST0[0]	EQ_ARRAY_INDEX_24_BST1[1]
Value	0x42		0	1	0	0	0	0	1	0
Description		39	EQ_ARRAY_INDEX_24_BST1[0]	EQ_ARRAY_INDEX_24_BST2[1]	EQ_ARRAY_INDEX_24_BST2[0]	EQ_ARRAY_INDEX_24_BST3[1]	EQ_ARRAY_INDEX_24_BST3[0]	EQ_ARRAY_INDEX_25_BST0[1]	EQ_ARRAY_INDEX_25_BST0[0]	EQ_ARRAY_INDEX_25_BST1[1]
Value	0xBA		1	0	1	1	1	0	1	0
Description		3A	EQ_ARRAY_INDEX_25_BST1[0]	EQ_ARRAY_INDEX_25_BST2[1]	EQ_ARRAY_INDEX_25_BST2[0]	EQ_ARRAY_INDEX_25_BST3[1]	EQ_ARRAY_INDEX_25_BST3[0]	EQ_ARRAY_INDEX_26_BST0[1]	EQ_ARRAY_INDEX_26_BST0[0]	EQ_ARRAY_INDEX_26_BST1[1]
Value	0xEB		1	1	1	0	1	0	1	1
Description		3B	EQ_ARRAY_INDEX_26_BST1[0]	EQ_ARRAY_INDEX_26_BST2[1]	EQ_ARRAY_INDEX_26_BST2[0]	EQ_ARRAY_INDEX_26_BST3[1]	EQ_ARRAY_INDEX_26_BST3[0]	EQ_ARRAY_INDEX_27_BST0[1]	EQ_ARRAY_INDEX_27_BST0[0]	EQ_ARRAY_INDEX_27_BST1[1]
Value	0x4B		0	1	0	0	1	0	1	1
Description		3C	EQ_ARRAY_INDEX_27_BST1[0]	EQ_ARRAY_INDEX_27_BST2[1]	EQ_ARRAY_INDEX_27_BST2[0]	EQ_ARRAY_INDEX_27_BST3[1]	EQ_ARRAY_INDEX_27_BST3[0]	EQ_ARRAY_INDEX_28_BST0[1]	EQ_ARRAY_INDEX_28_BST0[0]	EQ_ARRAY_INDEX_28_BST1[1]
Value	0xAE		1	0	1	0	1	1	1	0
Description		3D	EQ_ARRAY_INDEX_28_BST1[0]	EQ_ARRAY_INDEX_28_BST2[1]	EQ_ARRAY_INDEX_28_BST2[0]	EQ_ARRAY_INDEX_28_BST3[1]	EQ_ARRAY_INDEX_28_BST3[0]	EQ_ARRAY_INDEX_29_BST0[1]	EQ_ARRAY_INDEX_29_BST0[0]	EQ_ARRAY_INDEX_29_BST1[1]
Value	0xAC		1	0	1	0	1	1	0	0
Description		3	EQ_ARRAY	EQ_ARRAY	EQ_ARRAY	EQ_ARRAY	EQ_ARRAY	EQ_ARRAY	EQ_ARRAY	EQ_ARRAY

		E	_INDEX_29_BST1[0]	_INDEX_29_BST2[1]	_INDEX_29_BST2[0]	_INDEX_29_BST3[1]	_INDEX_29_BST3[0]	_INDEX_30_BST0[1]	_INDEX_30_BST0[0]	_INDEX_30_BST1[1]
Value	0xCC		1	1	0	0	1	1	0	0
Description		3F	EQ_ARRAY_INDEX_30_BST1[0]	EQ_ARRAY_INDEX_30_BST2[1]	EQ_ARRAY_INDEX_30_BST2[0]	EQ_ARRAY_INDEX_30_BST3[1]	EQ_ARRAY_INDEX_30_BST3[0]	EQ_ARRAY_INDEX_31_BST0[1]	EQ_ARRAY_INDEX_31_BST0[0]	EQ_ARRAY_INDEX_31_BST1[1]
Value	0XB5		1	0	1	1	0	1	0	1
Description		40	EQ_ARRAY_INDEX_31_BST1[0]	EQ_ARRAY_INDEX_31_BST2[1]	EQ_ARRAY_INDEX_31_BST2[0]	EQ_ARRAY_INDEX_31_BST3[1]	EQ_ARRAY_INDEX_31_BST3[0]	GRP0_OV_CNT[7]	GRP0_OV_CNT[6]	GRP0_OV_CNT[5]
Value	0x28		0	0	1	0	1	0	0	0
Description		41	GRP0_OV_CNT[4]	GRP0_OV_CNT[3]	GRP0_OV_CNT[2]	GRP0_OV_CNT[1]	GRP0_OV_CNT[0]	CNT_DLTA_OV_0	GRP0_OV_CNT[14]	GRP0_OV_CNT[13]
Value	0x00		0	0	0	0	0	0	0	0
Description		42	GRP0_OV_CNT[12]	GRP0_OV_CNT[11]	GRP0_OV_CNT[10]	GRP0_OV_CNT[9]	GRP0_OV_CNT[8]	GRP1_OV_CNT[7]	GRP1_OV_CNT[6]	GRP1_OV_CNT[5]
Value	0x00		0	0	0	0	0	0	0	0

Description		43	GRP1_OV_CNT[4]	GRP1_OV_CNT[3]	GRP1_OV_CNT[2]	GRP1_OV_CNT[1]	GRP1_OV_CNT[0]	CNT_DLTA_OV_1	GRP1_OV_CNT[14]	GRP1_OV_CNT[13]
Value	0x00		0	0	0	0	0	0	0	0
Description		44	GRP1_OV_CNT[12]	GRP1_OV_CNT[11]	GRP1_OV_CNT[10]	GRP1_OV_CNT[9]	GRP1_OV_CNT[8]	GRP0_OV_DLTA[3]	GRP0_OV_DLTA[2]	GRP0_OV_DLTA[1]
Value	0x00		0	0	0	0	0	0	0	0
Description		45	GRP0_OV_DLTA[0]	GRP1_OV_DLTA[3]	GRP1_OV_DLTA[2]	GRP1_OV_DLTA[1]	GRP1_OV_DLTA[0]	GRP0_OV_DLTA[4]	GRP1_OV_DLTA[4]	HV_LOCKMON_EN
Value	0x01		0	0	0	0	0	0	0	1
Description		46	HV_LCKMON_CNT_MS[3]	HV_LCKMON_CNT_MS[2]	HV_LCKMON_CNT_MS[1]	HV_LCKMON_CNT_MS[0]	VEO_LCK_THRSH[3]	VEO_LCK_THRSH[2]	VEO_LCK_THRSH[1]	VEO_LCK_THRSH[0]
Value	0XA2		1	0	1	0	0	0	1	0
Description		47	HEO_LCK_THRSH[3]	HEO_LCK_THRSH[2]	HEO_LCK_THRSH[1]	HEO_LCK_THRSH[0]	FOM_A[7]	FOM_A[6]	FOM_A[5]	FOM_A[4]
Value	0x24		0	0	1	0	0	1	0	0
Description		48	FOM_A[3]	FOM_A[2]	FOM_A[1]	FOM_A[0]	FOM_B[7]	FOM_B[6]	FOM_B[5]	FOM_B[4]
Value	0x00		0	0	0	0	0	0	0	0
Description		49	FOM_B[3]	FOM_B[2]	FOM_B[1]	FOM_B[0]	FOM_C[7]	FOM_C[6]	FOM_C[5]	FOM_C[4]
Value	0x00		0	0	0	0	0	0	0	0
Description		4A	FOM_C[3]	FOM_C[2]	FOM_C[1]	FOM_C[0]	EN_NEW_FOM_CTLE	EN_NEW_FOM_DFE	MR_EN_LOW_DIVSEL_EQ	RESERVED
Value	0x00		0	0	0	0	0	0	0	0

Description		4B	RESERVED	EQ_LB_CNT[3]	EQ_LB_CNT[2]	EQ_LB_CNT[1]	EQ_LB_CNT[0]	POST_LOCK_VEO_THR[3]	POST_LOCK_VEO_THR[2]	POST_LOCK_VEO_THR[1]
Value	0x19		0	0	0	1	1	0	0	1
Description		4C	POST_LOCK_VEO_THR[0]	POST_LOCK_HEO_THR[3]	POST_LOCK_HEO_THR[2]	POST_LOCK_HEO_THR[1]	POST_LOCK_HEO_THR[0]	CDR_CAP_DAC_START1[5]	CDR_CAP_DAC_START0[5]	POST_LOCK_SBT_THR[4]
Value	0x11		0	0	0	1	0	0	0	1
Description		4D	POST_LOCK_SBT_THR[3]	POST_LOCK_SBT_THR[2]	POST_LOCK_SBT_THR[1]	POST_LOCK_SBT_THR[0]	ENABLE_K28_5	CAL_OVERRIDE	CONT_ADPT_HEO_CHNG_THRESHOLD[3]	CONT_ADPT_HEO_CHNG_THRESHOLD[2]
Value	0xA1		1	0	1	0	0	0	0	1
Description		4E	CONT_ADPT_HEO_CHNG_THRESHOLD[1]	CONT_ADPT_HEO_CHNG_THRESHOLD[0]	CONT_ADPT_VEO_CHNG_THRESHOLD[3]	CONT_ADPT_VEO_CHNG_THRESHOLD[2]	CONT_ADPT_VEO_CHNG_THRESHOLD[1]	CONT_ADPT_VEO_CHNG_THRESHOLD[0]	CONT_ADPT_TAP_INCRMNT[3]	CONT_ADPT_TAP_INCRMNT[2]
Value	0x20		0	0	1	0	0	0	0	0
Description		4F	CONT_ADPT_TAP_INCRMNT[1]	CONT_ADPT_TAP_INCRMNT[0]	CONT_ADPT_FOM_CHNG_THRRESHOLD[3]	CONT_ADPT_FOM_CHNG_THRRESHOLD[2]	CONT_ADPT_FOM_CHNG_THRRESHOLD[1]	CONT_ADPT_FOM_CHNG_THRRESHOLD[0]	MR_DIS_HV_CHK_FOR_CONT_ADAPT	EN_DFE_CONT_ADAPT
Value	0x4F		0	1	0	0	1	1	1	1
Description		50	CONT_ADPT_CMP_BOTH	CONT_ADPT_COUNT[2]	CONT_ADPT_COUNT[1]	CONT_ADPT_COUNT[0]	EQ_EN_HR_MODE	PFD_EN_HR_MODE	EOM_EN10	RESERVED
Value	0xA0		1	0	1	0	0	0	0	0
Description		51	EQ_EN_MR_MODE	SD_DC_EN	EQ_SEL_LOOP_OUT	EN_CLK_LOOPTHRU_LV	FIR_SEL_EDGE[2]	FIR_SEL_EDGE[1]	FIR_SEL_EDGE[0]	DFE_SEL_GAIN[1]
Value	0x4E		0	1	0	0	1	1	1	0

Description		52	DFE_SEL_GAIN[0]	K28P5_COMPR_PERIOD[3]	K28P5_COMPR_PERIOD[2]	K28P5_COMPR_PERIOD[1]	K28P5_COMPR_PERIOD[0]	MIN_K28P5_REQD[11]	MIN_K28P5_REQD[10]	MIN_K28P5_REQD[9]
Value	0x00		0	0	0	0	0	0	0	0
Description		53	MIN_K28P5_REQD[8]	MIN_K28P5_REQD[7]	MIN_K28P5_REQD[6]	MIN_K28P5_REQD[5]	MIN_K28P5_REQD[4]	MIN_K28P5_REQD[3]	MIN_K28P5_REQD[2]	MIN_K28P5_REQD[1]
Value	0x00		0	0	0	0	0	0	0	0
Description		54	MIN_K28P5_REQD[0]	XPNT_SLAVE	XPNT_EN	EQ_BUFFER_EN[1]	EQ_BUFFER_EN[0]	EQ_DATA_MUX_IN[1]	EQ_DATA_MUX_IN[0]	RESERVED
Value	0x08		0	0	0	0	1	0	0	0
Description		55	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	MR_DIVSEL_START1_OV	MR_DIVSEL_STOP1_OV	DIVSEL_START1[2]
Value	0x61		0	1	1	0	0	0	0	1
Description		56	DIVSEL_START1[1]	DIVSEL_START1[0]	DIVSEL_STOP1[2]	DIVSEL_STOP1[1]	DIVSEL_STOP1[0]	MR_DIVSEL_START0_OV	MR_DIVSEL_STOP0_OV	DIVSEL_START0[2]
Value	0xF9		1	1	1	1	1	0	0	1
Description		57	DIVSEL_START0[1]	DIVSEL_START0[0]	DIVSEL_STOP0[2]	DIVSEL_STOP0[1]	DIVSEL_STOP0[0]	RESERVED	EQ_CTRL_MUX_IN[1]	EQ_CTRL_MUX_IN[0]
Value	0xF8		1	1	1	1	1	0	0	0

Shared Registers										
EEPROM Address Byte			**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Description		0	RESERVED	REFCLK_SEL[1]	REFCLK_SEL[0]	RESERVED	PDREFCLK_TX	REFCLK_SINGLE_END	MR_ENABLE_CH_LOCK[15]	MR_ENABLE_CH_LOCK[14]
Value	0x4B		0	1	0	0	1	0	1	1
Description		1	MR_ENABLE_CH_LOCK[13]	MR_ENABLE_CH_LOCK[12]	MR_ENABLE_CH_LOCK[11]	MR_ENABLE_CH_LOCK[10]	MR_ENABLE_CH_LOCK[9]	MR_ENABLE_CH_LOCK[8]	MR_ENABLE_CH_LOCK[7]	MR_ENABLE_CH_LOCK[6]
Value	0x30		0	0	1	1	0	0	0	0
Description		2	MR_ENABLE_CH_LOCK[5]	MR_ENABLE_CH_LOCK[4]	MR_ENABLE_CH_LOCK[3]	MR_ENABLE_CH_LOCK[2]	MR_ENABLE_CH_LOCK[1]	MR_ENABLE_CH_LOCK[0]	RESERVED	RESERVED
Value	0x00		0	0	0	0	0	0	0	0

4.3. EEPROM File Format

The EEPROM file consists of several groups of bytes. The table below explains the byte order and special bit settings for these bytes.

Byte (Decimal)[bit]	Function
0-2	Base Headers
0[7]	CRC Enable 0: Disable 1: Enable
0[6]	Address Map Enable 1: Enable 0: Disable
0[5]	EEPROM > 256 1: Yes 0: No
0[4]	Common Channel 1: Yes, all channels will load the same settings 0: No, channels will load individual settings
1, 2	EEPROM burst size to provide the fast reading of the configuration from the EEPROM Set to 00, 10
3 - 50	Address Map Header 16 possible slave device addresses. Three bytes for each address. Example for Address = 0x30: (00, 30, 00)
51 – 139 (Common Channel = 1)	88 bytes for channel configuration
51 – 1459 (Common Channel = 0)	1408 bytes for configuration of 88 bytes x 16 channels
140 – 143 (Common Channel = 1)	3 bytes for shared register configuration
1460 – 1462 (Common Channel = 0)	3 bytes for shared register configuration

4.4. EEPROM File Example, Common Channel = 1

The text below shows an example EEPROM file for the DS125DF1610 with the Common Channel bit set 1. In this example the EEPROM size is 1024 bytes.

```
:200000007F0010003300003300003300003300003300003300003300003300003300003353
:20002000000033000033000033000033000033000033000030002B7CD3A21820053D1B4865
:2000400028E47940002607F20436411887E7F00001AA1E3800082082004014001861820AC7
:20006000860304844415023294658642BAEB4BAEACCCB5280000000001A2240000001911 3D
:20008000A1204FA04E00000861F9F80BFFFCFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF14
:2000A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF60
:2000A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF60
:2000C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF40
:2000E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF20
:20010000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
:20012000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDF
:20014000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBF
:20016000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9F
:20018000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7F
:2001A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5F
:2001C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3F
:2001E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1F
:20020000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE
:20022000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDE
:20024000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBE
:20026000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9E
:20028000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7E
:2002A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5E
:2002C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3E
:2002E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1E
:20030000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFD
:20032000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDD
:20034000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBD
:20036000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9D
:20038000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7D
:2003A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5D
:2003C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3D
:2003E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1D
:00000001FF
```

4.5. EEPROM File Example, Common Channel = 0

The text below shows an example EEPROM file for the DS125DF1610 with the Common Channel bit set 1. In this example the EEPROM size is 16k bits. This example sets channel 15 to rate/sub-rate code 0x8 for 9.8304G and selects the reference clock to be 125 MHz.

```
:200000006F0010003300003300003300003300003300003300003300003300003300003363
:20002000000033000033000033000033000033000030002B7CD3A21820053D1B4865
:2000400028E47940002607F20436411887E7F00001AA1E3800082082004014001861820AC7
:20006000860304844415023294658642BAEB4BAEACCCB5280000000001A22400000019113D
:20008000A1204FA04E00000861F9F80030002B7CD3A21820053D1B4828E47940002607F2FB
:2000A0000436411887E7F00001AA1E3800082082004014001861820A8603048444150232AD
:2000C00094658642BAEB4BAEACCCB5280000000001A2240000001911A1204FA04E00000875
:2000E00061F9F80030002B7CD3A21820053D1B4828E47940002607F20436411887E7F000B0
:2001000001AA1E3800082082004014001861820A860304844415023294658642BAEB4BAEDE
:20012000ACCCB5280000000001A2240000001911A1204FA04E00000861F9F80030002B7C4A
:20014000D3A21820053D1B4828E47940002607F20436411887E7F00001AA1E3800082082CD
:20016000004014001861820A860304844415023294658642BAEB4BAEACCCB52800000000D4
:2001800001A2240000001911A1204FA04E00000861F9F80030002B7CD3A21820053D1B48ED
:2001A00028E47940002607F20436411887E7F00001AA1E3800082082004014001861820A66
:2001C000860304844415023294658642BAEB4BAEACCCB5280000000001A2240000001911DC
:2001E000A1204FA04E00000861F9F80030002B7CD3A21820053D1B4828E47940002607F29A
:200200000436411887E7F00001AA1E3800082082004014001861820A86030484441502324B
:2002200094658642BAEB4BAEACCCB5280000000001A2240000001911A1204FA04E00000813
:2002400061F9F80030002B7CD3A21820053D1B4828E47940002607F20436411887E7F0004E
:2002600001AA1E3800082082004014001861820A860304844415023294658642BAEB4BAE7D
:20028000ACCCB5280000000001A2240000001911A1204FA04E00000861F9F80030002B7CE9
:2002A000D3A21820053D1B4828E47940002607F20436411887E7F00001AA1E38000820826C
:2002C000004014001861820A860304844415023294658642BAEB4BAEACCCB5280000000073
:2002E00001A2240000001911A1204FA04E00000861F9F80030002B7CD3A21820053D1B488C
:2003000028E47940002607F20436411887E7F00001AA1E3800082082004014001861820A04
:20032000860304844415023294658642BAEB4BAEACCCB5280000000001A22400000019117A
:20034000A1204FA04E00000861F9F80030002B7CD3A21820053D1B4828E47940002607F238
:200360000436411887E7F00001AA1E3800082082004014001861820A8603048444150232EA
:2003800094658642BAEB4BAEACCCB5280000000001A2240000001911A1204FA04E000008B2
:2003A00061F9F80030002B7CD3A21820053D1B4828E47940002607F20436411887E7F000ED
:2003C00001AA1E3800082082004014001861820A860304844415023294658642BAEB4BAE1C
:2003E000ACCCB5280000000001A2240000001911A1204FA04E00000861F9F80030002B7C88

:20040000D3A21820053D1B4828E47940002607F20436411887E7F00001AA1E380008208200A
:20042000004014001861820A860304844415023294658642BAEB4BAEACCCB5280000000011
:2004400001A2240000001911A1204FA04E00000861F9F80030002B7CD3A21820053D1B482A
:2004600028E47940002607F20436411887E7F00001AA1E3800082082004014001861820AA3
:20048000860304844415023294658642BAEB4BAEACCCB5280000000001A224000000191119
:2004A000A1204FA04E00000861F9F80030002B7CD3A21820053D1B4828E47940002607F2D7
:2004C0000436411887E7F00001AA1E3800082082004014001861820A860304844415023289
:2004E00094658642BAEB4BAEACCCB5280000000001A2240000001911A1204FA04E00000851
:2005000061F9F80030002B7CD3A21820053D1B4828E47940002607F20436411887E7F0008B
```

```
:2005200001AA1E380008208200401400186182 0A86030484441502329465864 2BAEB4BAEBA
:20054000ACCCB5280000000001A2240000001911A1204FA04E00000861F9F80030002B7C26
:20056000D3A21820053D1B4828E47940002607F20486411887E7F00001AA1E380008208259
:2005800000401400186182 0A86030484441502329465864 2BAEB4BAEACCCB52800000000B0
:2005A00001A2240000001911A1204FA04E00000861F9F82BFFFCFFFFFFFFFFFFFFFFFFD6
:2005C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3B
:2005E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1B
:20060000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFA
:20062000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDA
:20064000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBA
:20066000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9A
:20068000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7A
:2006A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5A
:2006C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3A
:2006E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1A
:20070000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9
:20072000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFD9
:20074000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFB9
:20076000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF99
:20078000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF79
:2007A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF59
:2007C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF39
:2007E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF19
:00000001FF
```

5. JTAG Boundary Scan

5.1. BSDL File

The following text is the Boundary Scan Description Language (BSDL) file. This file describes the available pins and pin usage for the purposes of JTAG boundary scan.

DS125DF1610 BSDL File:

```
--------------------------------------------------------------------------
--  Texas Instruments Inc.
--  DS125DF1610 BSDL file
--------------------------------------------------------------------------
entity DS125DF1610 is

-- Generic Parameter
generic (PHYSICAL_PIN_MAP : string := "DS125DF1610_PKG");

-- Logical Port Description
port (
   TCK_IO, TMS_IO, TDI_IO, TRST_IO                                          :
in bit;
   TDO_IO                                                                   :
out bit;
   GPIO0, GPIO1, GPIO2, GPIO3                                               :
inout bit;
   TX_0A_P, TX_0A_N, TX_0B_P, TX_0B_N, TX_1A_P, TX_1A_N, TX_1B_P, TX_1B_N :
buffer bit;
   TX_2A_P, TX_2A_N, TX_2B_P, TX_2B_N, TX_3A_P, TX_3A_N, TX_3B_P, TX_3B_N :
buffer bit;
   TX_4A_P, TX_4A_N, TX_4B_P, TX_4B_N, TX_5A_P, TX_5A_N, TX_5B_P, TX_5B_N :
buffer bit;
   TX_6A_P, TX_6A_N, TX_6B_P, TX_6B_N, TX_7A_P, TX_7A_N, TX_7B_P, TX_7B_N :
buffer bit;
   SCL_IO, SDA_IO                                                           :
inout bit;
   RESET_IO                                                                 :
in bit;
   CLK_MON_P, CLK_MON_N                                                     :
buffer bit;
   REF_CLK_P, REF_CLK_N                                                     :
in bit;
   SCAN_MODE                                                                :
in bit;
   INTERR_IO                                                                :
inout bit;
   RX_0A_P, RX_0A_N, RX_0B_P, RX_0B_N, RX_1A_P, RX_1A_N, RX_1B_P, RX_1B_N :
in bit;
   RX_2A_P, RX_2A_N, RX_2B_P, RX_2B_N, RX_3A_P, RX_3A_N, RX_3B_P, RX_3B_N :
in bit;
```

```
   RX_4A_P, RX_4A_N, RX_4B_P, RX_4B_N, RX_5A_P, RX_5A_N, RX_5B_P, RX_5B_N :
in bit;
   RX_6A_P, RX_6A_N, RX_6B_P, RX_6B_N, RX_7A_P, RX_7A_N, RX_7B_P, RX_7B_N :
in bit;
   VSS                                                                    :
linkage bit_vector (1 to 84);
   VDD                                                                    :
linkage bit_vector (1 to 20);
   NC                                                                     :
linkage bit_vector (1 to 10)
);

-- Use Statements
use STD_1149_1_2001.all;
use STD_1149_6_2003.all;

-- Compliance Conformance Statement
attribute COMPONENT_CONFORMANCE of DS125DF1610 : entity is "STD_1149_1_2001";

-- Device Package Pin Mappings
attribute PIN_MAP of DS125DF1610 : entity is PHYSICAL_PIN_MAP;
constant PACK: PIN_MAP_STRING :=
   "TCK_IO    : D6, " &
   "TMS_IO    : B7, " &
   "TDI_IO    : D7, " &
   "TRST_IO   : C8, " &
   "TDO_IO    : C7, " &
   "CLK_MON_P : A7, " &
   "CLK_MON_N : A8, " &
   "GPIO0     : B6, " &
   "GPIO1     : D5, " &
   "GPIO2     : G5, " &
   "GPIO3     : L5, " &
   "TX_1A_P   : A1, " &
   "TX_1A_N   : B1, " &
   "TX_0B_P   : A3, " &
   "TX_0B_N   : B3, " &
   "TX_0A_P   : A5, " &
   "TX_0A_N   : B5, " &
   "TX_2A_P   : C2, " &
   "TX_2A_N   : D2, " &
   "TX_1B_P   : C4, " &
   "TX_1B_N   : D4, " &
   "TX_3A_P   : E1, " &
   "TX_3A_N   : F1, " &
   "TX_2B_P   : E3, " &
   "TX_2B_N   : F3, " &
   "TX_4A_P   : G2, " &
   "TX_4A_N   : H2, " &
```

```
"TX_3B_P   : G4, " &
"TX_3B_N   : H4, " &
"TX_4B_P   : J1, " &
"TX_4B_N   : K1, " &
"TX_5A_P   : J3, " &
"TX_5A_N   : K3, " &
"TX_5B_P   : L2, " &
"TX_5B_N   : M2, " &
"TX_6A_P   : L4, " &
"TX_6A_N   : M4, " &
"TX_6B_P   : N1, " &
"TX_6B_N   : P1, " &
"TX_7A_P   : N3, " &
"TX_7A_N   : P3, " &
"TX_7B_P   : N5, " &
"TX_7B_N   : P5, " &
"SCL_IO    : L6, " &
"SDA_IO    : M7, " &
"RESET_IO  : L8, " &
"REF_CLK_P : P7, " &
"REF_CLK_N : P8, " &
"SCAN_MODE : N8, " &
"INTERR_IO : M8, " &
"RX_1A_P   : A14, " &
"RX_1A_N   : B14, " &
"RX_0B_P   : A12, " &
"RX_0B_N   : B12, " &
"RX_0A_P   : A10, " &
"RX_0A_N   : B10, " &
"RX_2A_P   : C13, " &
"RX_2A_N   : D13, " &
"RX_1B_P   : C11, " &
"RX_1B_N   : D11, " &
"RX_3A_P   : E14, " &
"RX_3A_N   : F14, " &
"RX_2B_P   : E12, " &
"RX_2B_N   : F12, " &
"RX_4A_P   : G13, " &
"RX_4A_N   : H13, " &
"RX_3B_P   : G11, " &
"RX_3B_N   : H11, " &
"RX_4B_P   : J14, " &
"RX_4B_N   : K14, " &
"RX_5A_P   : J12, " &
"RX_5A_N   : K12, " &
"RX_5B_P   : L13, " &
"RX_5B_N   : M13, " &
"RX_6A_P   : L11, " &
"RX_6A_N   : M11, " &
```

```
   "RX_6B_P   : N14, " &
   "RX_6B_N   : P14, " &
   "RX_7A_P   : N12, " &
   "RX_7A_N   : P12, " &
   "RX_7B_P   : N10, " &
   "RX_7B_N   : P10, " &
   "VSS       : (A2, A4, A6, A9, A11, A13, B2, B4, B9, B11, " &
                "B13, C1, C3, C5, C10, C12, C14, D1, D3, D10, " &
            "D12, D14, E2, E4, E6, E8, E11, E13, F2, F4, F7, " &
            "F9, F11, F13, G1, G3, G6, G8, G10, G12, G14, H1, " &
            "H3, H5, H7, H9, H10, H12, H14, J2, J4, J6, J8, J11, " &
            "J13, K2, K4, K7, K9, K11, K13, L1, L3, L10, L12, L14, " &
            "M1, M3, M5, M10, M12, M14, N2, N4, N6, N9, N11, N13, P2, " &
            "P4, P6, P9, P11, P13)," &
   "VDD       : (E5, E7, E9, E10, F5, F6, F8, F10, G7, G9, H6, H8, J5, " &
                "J7, J9, J10, K5, K6, K8, K10)," &
   "NC        : (B8, C6, C9, D8, D9, L7, L9, M6, M9, N7)" ;


-- Grouped Port Identification
attribute PORT_GROUPING of DS125DF1610 : entity is
   "Differential_Voltage (" &
   "(CLK_MON_P, CLK_MON_N), "&
   "(TX_0A_P, TX_0A_N), "&
   "(TX_0B_P, TX_0B_N), "&
   "(TX_1A_P, TX_1A_N), "&
   "(TX_1B_P, TX_1B_N), "&
   "(TX_2A_P, TX_2A_N), "&
   "(TX_2B_P, TX_2B_N), "&
   "(TX_3A_P, TX_3A_N), "&
   "(TX_3B_P, TX_3B_N), "&
   "(TX_4A_P, TX_4A_N), "&
   "(TX_4B_P, TX_4B_N), "&
   "(TX_5A_P, TX_5A_N), "&
   "(TX_5B_P, TX_5B_N), "&
   "(TX_6A_P, TX_6A_N), "&
   "(TX_6B_P, TX_6B_N), "&
   "(TX_7A_P, TX_7A_N), "&
   "(TX_7B_P, TX_7B_N), "&
   "(RX_0A_P, RX_0A_N), "&
   "(RX_0B_P, RX_0B_N), "&
   "(RX_1A_P, RX_1A_N), "&
   "(RX_1B_P, RX_1B_N), "&
   "(RX_2A_P, RX_2A_N), "&
   "(RX_2B_P, RX_2B_N), "&
   "(RX_3A_P, RX_3A_N), "&
   "(RX_3B_P, RX_3B_N), "&
   "(RX_4A_P, RX_4A_N), "&
   "(RX_4B_P, RX_4B_N), "&
```

```
   "(RX_5A_P, RX_5A_N), "&
   "(RX_5B_P, RX_5B_N), "&
   "(RX_6A_P, RX_6A_N), "&
   "(RX_6B_P, RX_6B_N), "&
   "(RX_7A_P, RX_7A_N), "&
   "(RX_7B_P, RX_7B_N), "&
   "(REF_CLK_P, REF_CLK_N) )";

-- Scan Port Identification
attribute TAP_SCAN_RESET of TRST_IO : signal is true;
attribute TAP_SCAN_CLOCK of TCK_IO  : signal is (10.0e6, BOTH);
attribute TAP_SCAN_MODE  of TMS_IO  : signal is true;
attribute TAP_SCAN_IN    of TDI_IO  : signal is true;
attribute TAP_SCAN_OUT   of TDO_IO  : signal is true;

-- Instruction Register Description
attribute INSTRUCTION_LENGTH of DS125DF1610 : entity is 3;
attribute INSTRUCTION_OPCODE of DS125DF1610 : entity is
   -- IEEE Std 1149.1
   --   BYPASS gets all unused codes
   "BYPASS       (111)," &
   "EXTEST       (000)," &
   "SAMPLE       (001)," &
   "PRELOAD      (001)," &
   "IDCODE       (010)," &
   "HIGHZ        (011)," &
   "CLAMP        (100)," &
   -- IEEE Std 1149.6
   "EXTEST_PULSE (101)," &
   "EXTEST_TRAIN (110)";

attribute INSTRUCTION_CAPTURE of DS125DF1610 : entity is "001";

-- Optional Register Description
attribute IDCODE_REGISTER of DS125DF1610 : entity is
   "0001"             & -- version
   "1011100110110000" & -- part number
   "00000010111"      & -- manufacturer's identity
   "1";                 -- required by 1149.1

-- Register Access Description
attribute REGISTER_ACCESS of DS125DF1610 : entity is
   "BOUNDARY (EXTEST_PULSE, EXTEST_TRAIN), " &
   "BYPASS   (CLAMP, HIGHZ)";

-- Boundary-Scan Register Description
attribute BOUNDARY_LENGTH of DS125DF1610 : entity is 68;
attribute BOUNDARY_REGISTER of DS125DF1610 : entity is
-- num  cell      port       function      safe   [ccell disval rslt]
```

```
"67 (AC_SelU,  *,          INTERNAL,       X)," &
"66 (AC_2,     CLK_MON_P,  OUTPUT2,        X)," &
"65 (BC_7,     GPIO0,      BIDIR,          X,     64,    0,    Z)," &
"64 (BC_2,     *,          CONTROL,        0)," &
"63 (BC_7,     GPIO1,      BIDIR,          X,     62,    0,    Z)," &
"62 (BC_2,     *,          CONTROL,        0)," &
"61 (AC_2,     TX_1A_P,    OUTPUT2,        X)," &
"60 (AC_2,     TX_0B_P,    OUTPUT2,        X)," &
"59 (AC_2,     TX_0A_P,    OUTPUT2,        X)," &
"58 (AC_2,     TX_2A_P,    OUTPUT2,        X)," &
"57 (AC_2,     TX_1B_P,    OUTPUT2,        X)," &
"56 (AC_2,     TX_3A_P,    OUTPUT2,        X)," &
"55 (AC_2,     TX_2B_P,    OUTPUT2,        X)," &
"54 (AC_2,     TX_4A_P,    OUTPUT2,        X)," &
"53 (AC_2,     TX_3B_P,    OUTPUT2,        X)," &
"52 (AC_2,     TX_4B_P,    OUTPUT2,        X)," &
"51 (AC_2,     TX_5A_P,    OUTPUT2,        X)," &
"50 (AC_2,     TX_5B_P,    OUTPUT2,        X)," &
"49 (AC_2,     TX_6A_P,    OUTPUT2,        X)," &
"48 (AC_2,     TX_6B_P,    OUTPUT2,        X)," &
"47 (AC_2,     TX_7A_P,    OUTPUT2,        X)," &
"46 (AC_2,     TX_7B_P,    OUTPUT2,        X)," &
"45 (BC_7,     GPIO3,      BIDIR,          X,     44,    0,    Z)," &
"44 (BC_2,     *,          CONTROL,        0)," &
"43 (BC_7,     GPIO2,      BIDIR,          X,     42,    0,    Z)," &
"42 (BC_2,     *,          CONTROL,        0)," &
"41 (BC_7,     SCL_IO,     BIDIR,          X,     40,    0,    Z)," &
"40 (BC_2,     *,          CONTROL,        0)," &
"39 (BC_7,     SDA_IO,     BIDIR,          X,     38,    0,    Z)," &
"38 (BC_2,     *,          CONTROL,        0)," &
"37 (BC_4,     SCAN_MODE,  OBSERVE_ONLY,   X)," &
"36 (BC_7,     INTERR_IO,  BIDIR,          X,     35,    0,    Z)," &
"35 (BC_2,     *,          CONTROL,        0)," &
"34 (BC_4,     RESET_IO,   OBSERVE_ONLY,   X)," &
"33 (BC_4,     *,          INTERNAL,       X)," &
"32 (BC_4,     *,          INTERNAL,       X)," &
"31 (BC_4,     *,          INTERNAL,       X)," &
"30 (BC_4,     *,          INTERNAL,       X)," &
"29 (BC_4,     *,          INTERNAL,       X)," &
"28 (BC_4,     *,          INTERNAL,       X)," &
"27 (BC_4,     *,          INTERNAL,       X)," &
"26 (BC_4,     *,          INTERNAL,       X)," &
"25 (BC_4,     *,          INTERNAL,       X)," &
"24 (BC_4,     *,          INTERNAL,       X)," &
"23 (BC_4,     *,          INTERNAL,       X)," &
"22 (BC_4,     *,          INTERNAL,       X)," &
"21 (BC_4,     *,          INTERNAL,       X)," &
"20 (BC_4,     *,          INTERNAL,       X)," &
"19 (BC_4,     *,          INTERNAL,       X)," &
```

```
   "18 (BC_4,     *,          INTERNAL,       X)," &
   "17 (BC_4,     *,          INTERNAL,       X)," &
   "16 (BC_4,     *,          INTERNAL,       X)," &
   "15 (BC_4,     *,          INTERNAL,       X)," &
   "14 (BC_4,     *,          INTERNAL,       X)," &
   "13 (BC_4,     *,          INTERNAL,       X)," &
   "12 (BC_4,     *,          INTERNAL,       X)," &
   "11 (BC_4,     *,          INTERNAL,       X)," &
   "10 (BC_4,     *,          INTERNAL,       X)," &
   " 9 (BC_4,     *,          INTERNAL,       X)," &
   " 8 (BC_4,     *,          INTERNAL,       X)," &
   " 7 (BC_4,     *,          INTERNAL,       X)," &
   " 6 (BC_4,     *,          INTERNAL,       X)," &
   " 5 (BC_4,     *,          INTERNAL,       X)," &
   " 4 (BC_4,     *,          INTERNAL,       X)," &
   " 3 (BC_4,     *,          INTERNAL,       X)," &
   " 2 (BC_4,     *,          INTERNAL,       X)," &
   " 1 (BC_4,     *,          INTERNAL,       X)," &
   " 0 (BC_4,     *,          INTERNAL,       X)";

-- Advanced I/O Description
attribute AIO_COMPONENT_CONFORMANCE of DS125DF1610 : entity is
   "STD_1149_6_2003";

attribute AIO_Pin_Behavior of DS125DF1610: entity is
   "TX_0A_P,TX_0B_P,TX_1A_P,TX_1B_P, " &
   "TX_2A_P,TX_2B_P,TX_3A_P,TX_3B_P, " &
   "TX_4A_P,TX_4B_P,TX_5A_P,TX_5B_P, " &
   "TX_6A_P,TX_6B_P,TX_7A_P,TX_7B_P, " &
   "CLK_MON_P : AC_Select=67 " ;

end DS125DF1610;
```

6. Recommendations and Debug Hints

6.1. Device Initialization

The DS125DF1610 is a high performance retimer device with very strong analog circuitry. It is recommended to initialize the device after the signal data transmitted to the DS125DF1610 has stabilized.

A typical system initialization sequence would be:

1) System power on
2) DS125DF1610 retimers put into CDR reset or signal detect forced low
3) CPU/ASIC initialization
4) CPU/ASIC high speed I/O stable and transmitting data
5) DS125DF1610 retimers programmed for desired application settings
6) DS125DF1610 retimer released from CDR reset or signal detect force low is undone

If the DS125DF1610 is programmed before the CPU/ASIC's outputs are stable, it will still try to lock to the detected signal. Issues can arise if the CPU/ASIC's outputs glitch or are not stable in frequency while the DS125DF1610 is trying to lock to the detect signal. In this scenario the CDR may need to be reset and released from reset in order to lock and adapt to optimal settings.

6.2. Temperature Lock Range

The DS125DF1610 is designed to stay in lock over the supported temperature range of -20°C to +85°C. In some cases it has been shown to be beneficial to set REG_EN_PD_CP_OV = 1. This function is found in channel register 0x09[3].

Shared Register	Bit	Description
0x09	3	Setting this bit to 1 allows for manual enable/disable of the charge pumps. With default values in register 0x1B[1:0], this will enable both charge pumps which can assist in increased temperature lock range performance in some applications.

6.3. Debug Hints

This section addresses some common problems that hardware and software engineers may encounter during development or system bring up.

Before beginning debug, ensure that there is a communication link to the DS125DF1610 through the SMBus interface. One simple way to verify the SMBus link to the DS125DF1610 is to access register 0xFE and ensure that a value of 0x03 is read back.

Problem	Debug Checklist
Cannot communicate to the DS125DF1610 through SMBus	• Check power to the DS125DF1610 • Check pull up resistors on SMBus line for proper value and voltage measurement • Check the ADDR1 and ADDR0 pins for selection of SMBus Address • Check READ_EN and EN_SMB pins for proper setting • Check that the DS125DF1610 does not have the same address as any other device on the SMBus lines
CDR no lock	• Check power to the DS125DF1610 • Check that the reference clock is detected • Check the rate/sub-rate table data rate selection • Check register 0x78 for signal detect and CDR lock • Try ref mode 0, if CDR locks then the data rate being used is different than the rate/sub-rate selection or there is a problem with the reference clock. Be sure to address these issues and return the device to reference mode 3 for the actual implementation
Cannot use PRBS Generator and Checker at the same time in the same channel	• This is expected. The PRBS generator and check share resources and are not able to operate at the same time in the same channel.
Reference clock not detected	• Check share register 0x02 to make sure that the DS125DF1610 is programmed to operate with the desired reference clock frequency

Bit errors detected at cold temp, but not at hot temp	• Use time domain reflectometry (TDR) the connections to the DS125DF1610 and the CPU/ASIC's inputs and outputs. Check to make sure that there are no cracked or cold solder joints • Check power connection to the DS125DF1610 at hot and cold temp
No signal detected	• Check power to the DS125DF1610 • Check the datasheet pin names and register channel assignments versus the system schematics
Reference clock only detected in the first device in the reference clock daisy chain	• Check that the CLK_MON outputs are enabled in register share register 0x0A • Check that the 25 MHz signal is selected to be output for reference clock daisy chaining • Check that all subsequent DS125DF1610 devices are programmed to expect the correct 25 MHz reference clock frequency for daisy chaining in shared register 0x02
Register values do not make sense compared to the datasheet	• Before reading or write to share or channel registers, check to make sure that register 0xFF is configured to select the shared or channel registers • Check to make sure a channel is selected in register 0xFC or 0xFD before reading or writing to the channel registers
All register reads return 0x00	• Check registers 0xFC and 0xFD to make sure only a single channel is selected. It is OK to write to more than one channel at a time. The DS125DF1610 does not support reading from multiple channels at the same time and will return 0x00 if more than one channel is selected

Before contacting TI field or factory application support collect the following information:

- System block diagram
- Desired data rates
- Expected channel lengths, loss and material type
- DS125DF1610 register dump for the shared and channel registers
- Schematic pages for the DS125DF1610