Application Report
SNLAxxx – October 2015

Understanding EEPROM Programming for 10G to 12.5G Retimers

Michael Lu *Signal & Datapath Solutions*

ABSTRACT

System designers often use EEPROM (Electrically Erasable Programmable Read-Only Memory) to program a set of customized start-up settings that are different from the default. Using the information here will make EEPROM configuration and programming easy to implement and understand for 10-12.5 Gbps retimer devices. With a complete understanding of how to program and interpret EEPROM hex files for TI's 10-12.5 Gbps retimers, system designers are better equipped to generate their own customized hex files and increase the efficiency of their final designs.

Contents

1 Introduction

EEPROM is non-volatile memory used in electronic devices to store data that must be saved when power is removed. This non-volatile memory is particularly important when an application requires different start-up configurations than the factory default settings. Upon device power-up, data saved in the EEPROM will load automatically to the device. If EEPROM is not used, interface system designs require external access to the SMBus SDA and SCL line in order to set individual registers after each power-up. With EEPROM, designers eliminate the requirement for an external microprocessor or software driver to provide their desired register settings.

Programming EEPROM for TI's 10-12.5 Gbps retimers requires an understanding of the relationship between EEPROM bits and Slave Mode register bits. There are several design challenges that are unique when programming the 10-12.5 Gbps retimers compared to their 8-12.5 Gbps redriver and mux buffer counterparts. Some of these challenges are listed below:

- SMBus-to-EEPROM bit mapping incorporate multiple register pages per device.
- Retimer EEPROMs feature the option of using Common Channel Configuration to reduce the overall EEPROM size.

This application note is applicable to the following 10-12.5 Gbps retimer products:

Table 1. Applicable Retimer Devices

Device Type	Retimer Devices Included
2-Channel	DS110DF111, DS125DF111
4-Channel	DS100DF410, DS110DF410, DS125DF410
16-Channel	DS110DF1610, DS125DF1610, DS150DF1610

As a prerequisite, it is assumed that the reader is already familiar with the following high speed device EEPROM topics:

- How to configure high speed devices to operate in EEPROM Master Mode
- How to read EEPROM hex format
- How to calculate the CRC-8 value from a given bit stream of values
- Difference between Number of Slots v. Number of Devices

If the aforementioned topics are unfamiliar to the reader, please reference "Understanding EEPROM Programming for High Speed Repeaters and Mux Buffers" (SNLA228) for more details.

2 EEPROM Device Data Fundamentals

Every EEPROM file contains one Base Header. Depending on the system design, an Address Map Header may follow the Base Header. The following subsections explain the contents of these headers and other key fundamentals as they relate to the 10-12.5 Gbps retimer devices.

2.1 Base Header

2.1.1 Base Header Definitions

The first three Bytes define the Base Header. Table 2 explains the meaning of the Base Header Bytes.

Table 2. Base Header Information

Byte	Bit Number	Bit Name	Description
0	7	CRC_EN	1 = CRC enable. If enabled, each device slot will have a CRC value calculated from all the Bytes used by the retimer to load from EEPROM.
	6	ADDR Map Enable	1 = Address Map Header enable. If enabled, a 2 or 3 Byte Address Map Header will be placed after the Base Header to indicate the start address of each device's EEPROM.
	5	EEPROM > 256 Bytes	1 = Required EEPROM size is more than 256 Bytes. This must be enabled if a retimer must load from an EEPROM memory location greater than 0xFF.
	4	COMMON_CHANNEL	1 = Common Channel Configuration enable. If enabled, the settings for all channels are referenced from one Channel Register's settings.
	3:0	DEVICE COUNT	(Total number of Devices) - 1. **Note:** This value is not used by the device when the EEPROM loads data, though it is a useful debugging reference.
1	7:0	RES	Reserved. Set bits to 0.
2	7:0	Max EEPROM Burst Size	Maximum number of Bytes that are read during a burst read operation. A value of 0x10 is suitable for all EEPROMs.

2.1.2 Common Channel Configuration

When Common Channel Configuration is enabled from the Base Header Byte 0, Bit 4, EEPROM data for one Channel Register page is used as the universal channel settings for all Channel Registers in the device slot. The primary purpose of Common Channel Configuration is to save EEPROM memory space, because only one Channel Register setting is needed to represent all the Channel Register pages.

For example, the DS125DF410 contains 74 Bytes of EEPROM data per Channel Register page and 2 additional Bytes from the Share Register page. Without Common Channel Configuration, the total EEPROM size for the device slot per device is (4 x 74) + 2 = 298 Bytes. In contrast, with Common Channel Configuration, the total EEPROM size per device slot becomes only 74 + 2 = 76 Bytes. By using Common Channel Configuration, the EEPROM size is reduced by 224 Bytes per device slot for the quad-channel DS125DF410.

2.2 Address Map Header

2.2.1 Address Map Header Definitions

If Base Header Byte 0, Bit 6 = 1, Address Map Headers are used. The Address Map Header specifies the memory location where each retimer begins reading its programmed data settings. Table 3 explains the meaning of the Address Map Header Bytes.

Table 3. Address Map Header Information

Byte	Bit Number	Bit Name	Description
0	7:0	CRC Value	8-Bit CRC value for each device. CRC is computed from all the Bytes used by the retimer to load from EEPROM.

1	7:0	Device EEPROM Start Address	Start address for Reg 0x03 of device EEPROM. Recall that Reg 0x00-0x02 of device EEPROM is stored in the Base Header.
2	7:3	RES	Reserved. Set bits to 0.
	2:0	Device EEPROM Start Address MSBs	These bits are only set if EEPROM Size > 256 Bytes. Up to 3 MSB bits can be appended to the front of the EEPROM start address indicated in Byte 1.

Note: Byte 2 is present only if the EEPROM > 256 Bytes Enable Bit is set by asserting Base Header Byte 0, Bit 5 = 1. For example, if the EEPROM start address is located at Address 0x1F4, 9 bits are required. Thus, Address Map Header Byte 1 = 0xF4, and Address Map Header Byte 2 = 0x01. If EEPROM ≤ 256 Bytes, then the Address Map Header will be 2 Bytes, *not* 3 Bytes.

2.2.2 Page Addressing

If the EEPROM size > 256 Bytes, it is possible that data must be read from memory locations that exceed Address 0xFF. Address Map Header Byte 2, Bits [2:0] are used for EEPROM page addressing in order for the retimer to access these memory locations beyond Address 0xFF.

In the Atmel AT24C01C/02C/04C/08C/16C EEPROMs, the first four MSB bits of the EEPROM device address consist of a mandatory “1010” pattern. The next three bits are reserved for either hard-wire addressing on EEPROM pins A0, A1, and A2, or page addressing for memory page bits P0, P1, and P2. If the EEPROM size requires access to a memory address greater than 0xFF, memory page bits P0 to P2 are used for device addressing instead of the logic applied to pins A0 to A2. This is illustrated in Table 4.

Table 4. EEPROM Device Addressing[1]

EEPROM Size (bits)	Bit 7 (MSB)	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0 (LSB)
1K/2K	1	0	1	0	A2	A1	A0	R/W
4K	1	0	1	0	A2	A1	P0	R/W
8K	1	0	1	0	A2	P1	P0	R/W
16K	1	0	1	0	P2	P1	P0	R/W

The EEPROM will only replace pin logic on A0 to A2 with the minimum number of page bits required to support its size. This concept is detailed further in Table 5.

Table 5. Page Address Effect on A0-A2 EEPROM Pins

EEPROM Size (bits)	Max EEPROM Memory Location	Comments
2K	Byte 0xFF	- No Page addressing is required. - Pin logic on A0, A1, and A2 are used for EEPROM device addressing.
4K	Byte 0x1FF	- 1 bit is required for page addressing. - Page address bit replaces A0 pin logic. - Only pin logic on A1 and A2 are used for EEPROM device addressing.

[1] For each P0, P1, or P2 device address bit that is used, the corresponding A0, A1, or A2 pin is treated as no-connect and can be tied low to ground.

EEPROM Size (bits)	Max EEPROM Memory Location	Comments
8K	Byte 0x3FF	- 2 bits are required for page addressing. - Page address bits replace A0 and A1 pin logic. - Only pin logic on A2 is used for EEPROM device addressing.
16K	Byte 0x7FF	- 3 bits are required for page addressing. - Page address bits replace A0, A1, and A2 pin logic. - No A0, A1, or A2 logic pins are used for EEPROM device addressing.

As a result of Page Addressing, the maximum EEPROM size that can be supported for retimer programming is 16 Kbits (2048 Bytes). Likewise, EEPROMs with a size greater than 256 Bytes must support page addressing in order to be used properly with TI high speed devices.

2.2.3 EEPROM Configuration without Address Map Headers

If Address Map Headers are not used, the retimer device start address is not stored in a programmable location. Instead, the retimer computes a fixed starting memory location for each device's data. To do this, the retimer first determines its SMBus address index, I_{SMB_ADDR}, in the array of permissible SMBus Write Address Bytes for the retimer, shown in Table 6.

Table 6. I_{SMB_ADDR} Mapping

Retimer SMBus Write Address	SMBus Address Index I_{SMB_Addr}
0x30	0
0x32	1
0x34	2
0x36	3
0x38	4
0x3A	5
0x3C	6
0x3E	7
0x40	8
0x42	9
0x44	10
0x46	11
0x48	12
0x4A	13
0x4C	14
0x4E	15

The retimer then computes the length of the data slot, N_{DATA_SLOT}. The length of each data slot is determined by the number of channels in the device IC and whether the Common Channel bit is enabled in the Base Header. As an example, consider the case where the DS125DF410 quad retimer is used.

Table 7. N_{DATA_SLOT} Calculation for DS125DF410

	COMMON_CHANNEL = 1	**COMMON_CHANNEL = 0**
Channel Register Bytes	74 x 1 Channel = 74	74 x 4 Channels = 296
Share Register Bytes	2	2
CRC Byte[2]	1	1
N_{DATA_SLOT} (Total Bytes per Data Slot)	**77**	**299**

With knowledge of I_{SMB_ADDR} and N_{DATA_SLOT}, the retimer computes its data slot start address as follows:

$ADDR_{DATA\_START} = 3 + (I_{SMB\_ADDR} \times N_{DATA\_SLOT})$

Note: An offset of 3 accounts for the Base Header Bytes.

As an example, consider the case where Common Channel Configuration is used and a DS125DF410 retimer with SMBus Address 0x34 attempts to load settings from EEPROM. Referencing Table 6 and Table 7, $N_{DATA\_SLOT} = 77$ and $I_{SMB\_ADDR} = 3$. Therefore, the device slot's start address $ADDR_{DATA\_START}$ can be calculated:

$ADDR_{DATA\_START} = 3 + (3 \times 77) = 234$ or 0xEA

When Address Map Headers are not used, the Base Header Byte 0, Bit 5 (EEPROM Size > 256 Bytes Enable Bit) must be set to 1 if the EEPROM memory space is greater than 256 Bytes. If this Base Header bit is not asserted, the retimer will not calculate the correct memory locations when the desired EEPROM address Byte exceeds Address 0xFF.

Although retimers can load from an EEPROM without an Address Map Header, this practice is not recommended, especially when multiple devices are used. There are several reasons for this.

Without Address Map Headers…

1. Each device must load from a unique EEPROM slot, so multiple devices sharing the same setting cannot reference the same memory location.
2. Each device slot starting and ending location is fixed.
3. EEPROM space is not usually conserved when multiple devices are used.

The benefits of Address Map Headers include not only simplifying EEPROM programming for retimers, but also more efficiently using EEPROM memory space.

2.3 Register Data Slot Settings

The SMBus register map architecture for 10-12.5 Gbps retimers uses one global Share Register page and up to 16 individual Channel Register pages[3]. When mapping the different SMBus register pages to EEPROM, the EEPROM data slot begins first with Channel 0 Register data, followed by Channel 1

[2] Without Address Map Headers, the CRC Byte is programmed as an extra Byte after the last Share Register Byte in each data slot. This Byte is still present even if CRC is not enabled in the Base Header.
[3] The total number of Channel Registers equals the number of channels in the IC. For example, the DS125DF111 contains two Channel Registers.

Register data, and so on. The Share Register data comes after the last Channel Register's data is programmed.

2.3.1 EEPROM Data for Retimers

While an EEPROM allows devices to start up with settings different than the factory default, the EEPROM only maps a subset of the SMBus register bits. SMBus register bits that are not stored in EEPROM cannot be changed from default at device startup. The number of Bytes mapped to EEPROM per Channel Register and Share Register Page varies depending on the device's channel width. The difference in Byte size per device type is summarized in Table 8.

Table 8. Number of Bytes Mapped to EEPROM per Retimer Device

Device Type	From Channel Register Page	From Share Register Page	Retimer Devices Included
2-Channel	61	2	DS110DF111, DS125DF111
4-Channel	74	2	DS100DF410, DS110DF410, DS125DF410
16-Channel	88	3	DS110DF1610, DS125DF1610, DS150DF1610

For detailed information about bit mapping from Channel Register and Share Register Pages to EEPROM, refer to the tables in Section 6.

2.3.2 CRC Calculation for Device Slots

The CRC for each data slot is calculated by evaluating the CRC-8 polynomial for all the Bytes the retimer uses when loading its device slot settings from EEPROM.

2.3.2.1 CRC Calculation with Address Map Header

If the Address Map Header is enabled by asserting Base Header 0, Bit 6 = 1, the CRC Byte for each device slot is computed using the following Bytes:

- Base Header (3 Bytes)
- Non-CRC Bytes of the Device Slot's Address Map Header (1 or 2 Bytes)
- Channel Register Data (61, 74, or 88 Bytes per Channel)
- Share Register Data (2 or 3 Bytes)

When the Address Map Header is enabled, the CRC Byte for each device slot is stored in that device slot's Address Map Header Byte 0.

2.3.2.2 CRC Calculation without Address Map Header

If the Address Map Header is disabled by asserting Base Header 0, Bit 6 = 0, the CRC Byte for each device slot is computed using the following Bytes:

- Base Header (3 Bytes)
- Channel Register Data (61, 74, or 88 Bytes per Channel)
- Share Register Data (2 or 3 Bytes)

When the Address Map Header is disabled, the CRC Byte is stored in the Byte location immediately following the last Share Register Byte of that particular device slot.

3 Example 1: EEPROM Hex File for 2 Devices, CRC Disabled

A simple case for programming EEPROM is shown here with two retimer devices. In this example, the following settings are desired:

- Enable Address Map Header.
- Enable Common Channel Configuration.
- Set EEPROM ≤ 256 Bytes.
- Since CRC is disabled, the first Address Map Header Byte value is arbitrary.

An example of a hex file with two DS125DF111 devices that meet these requirements is shown below:

```
:20000000510010003300720000000000000000000000000000000000000000000000000000DA
:20002000000000000000000000000000000000000000000000000000000083C93693A218100041E8B8
:20004000966E19238A000026007280198408C43F1F32000100204028604 82A50584AB4EADA
:20006000CCD2F37C8000001358B85EFFA900000000000000000083C93693A218100041E8962C
:200080006E19238A000026007280198408C43F1F32000100204028604 82A50584AB4EACC64
:2000A000D2F37C8000001358B85EFFA900000000000FFFFFFFFFFFFFFFFFFFFFFFFFFFFF65
:2000C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF40
:2000E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF20
:00000001FF
```

__ = EEPROM Base Header

__ = Device 0 Address Map Header + Data

__ = Device 1 Address Map Header + Data

In the DS125DF111 hex file, the Base Header Bytes are 0x510010. From Table 2, this means the following:

- CRC is disabled (Reg 0x00[7] = 0'b).
- Address Map Header is used (Reg 0x00[6] = 1'b).
- EEPROM ≤ 256 Bytes (Reg 0x00[5] = 0'b).
- Common Channel is enabled (Reg 0x00[4] = 1'b).
- DEVICE COUNT = 2 Devices (Reg 0x00[3:0] = 0001'b).
- Max EEPROM Burst size = 16 bits (Reg 0x02 = 0x10).

Because EEPROM size ≤ 256 Bytes, the Address Map Header is two Bytes long. The starting address location for the two devices can be determined by the non-CRC Bytes of these Address Map Headers:

- Device 0 [CRC, Start Address] = [0x00, 0x33]
- Device 1 [CRC, Start Address] = [0x00, 0x72]

When Common Channel Configuration is enabled, all channels in each device slot derive their device settings from only one set of Channel Register settings, followed immediately by the Share Register

Bytes. The Channel and Share Register EEPROM data for Device 0 in this example are shown in Table 9.

Table 9. Device 0 Slot Settings

Channel Register Data	0x00000083C93693A218100041E8966E19238A000026007280198408C43F1F32000100204028604 82A50584AB4EACCD2F37C8000001358B85EFFA9000000
Share Register Data	0x0000

4 Example 2: EEPROM Hex File for 1 Device, No Address Map Header

When programming a single device, an Address Map Header may not be necessary. In this example, the following settings are desired.

- Enable CRC Byte.
- Disable Address Map Header.
- Enable Common Channel Configuration.
- Set EEPROM ≤ 256 Bytes.

An example of a hex file with a DS125DF1610 that meets these requirements is shown below:

```
:2000000090001000030002B7CD3A21820053D1B4828E47940002607F20436411887E7F00042
:2000200001AA1E380008208200401400186182 0A8603048444150232946586 42BAEB4BAEBF
:20004000ACCCB52800000000001A2240000001911A1204FA04E00000861F9F80BFFFCEDFF10
:20006000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFA0
:20008000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF80
:2000A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF60
:2000C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF40
:2000E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF20
:00000001FF
```

__ = EEPROM Base Header

__ = Device 0 Address Map Header + Data

__ = Device 0 CRC Byte

In the DS125DF1610 hex file, the Base Header Bytes are 0x900010. From Table 2, this means the following:

- CRC is enabled (Reg 0x00[7] = 1’b).
- Address Map Header is not used (Reg 0x00[6] = 0’b).
- EEPROM ≤ 256 Bytes (Reg 0x00[5] = 0’b).
- Common Channel is enabled (Reg 0x00[4] = 1’b).
- DEVICE COUNT = 1 Device (Reg 0x00[3:0] = 0000’b).
- Max EEPROM Burst size = 16 bits (Reg 0x02 = 0x10).

Since an Address Map Header is not used, the data slot for the first device begins immediately after the Base Header. When Common Channel Configuration is enabled, all channels derive their device settings from only one set of Channel Register settings, followed immediately by the Share Register Bytes. The Channel and Share Register EEPROM data for Device 0 in this example are shown in

Table 10.

Table 10. Device 0 Slot Settings

Channel Register Data	0x0030002B7CD3A21820053D1B4828E47940002607F20436411887E7F000 01AA1E380008208200401400186182 0A8603048444150232946586 42BAEB 4BAEACCCB5280000000001A2240000001911A1204FA04E00000861F9F8
Share Register Data	0x0BFFFC

The CRC Byte is calculated from the Base Header and relevant device slot Bytes. The values used for the CRC calculation in this example are shown below:

900010 0030002B7CD3A21820053D1B4828E47940002607F20436411887E7F00001AA1E3800082082004 014001861820A8603048444150232946586 42BAEB4BAEACCCB5280000000001A2240000001911A1204F A04E00000861F9F80BFFFC

CRC-8 calculation of the above data Bytes for Device 0 yields CRC = 0xED.

5 Example 3: EEPROM Hex File for 2 Devices, CRC Enabled

When programming multiple EEPROM device slots, Address Map Headers are recommended. In this example, the following settings are desired:

- Enable CRC Byte.
- Enable Address Map Header.
- Disable Common Channel Configuration.
- Set EEPROM > 256 Bytes.

An example of a hex file for two DS125DF410 devices that meet these requirements is shown below:

```
:20000000E10010793300E95D010000000000000000000000000000000000000000000000FC
:20002000000000000000000000000000000000000000000000000083C93693A218180020F4C5
:200040006D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10F3
:200060005430182422220A81194A32C3215D75A5D756665A940000000002800000001000003B
:2000800083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80008E
:2000A0004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A940004E
:2000C000000002800000002000083C93693A218180020F46D230C91E500001300394000005D
:2000E000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C32A1
:2001000015D75A5D756665A9400000000028000000300083C93693A218180020F46D2335D
:200120000C91E500001300394000C104621F8F99014A80004104100200A000C30C1054301E
:2001400018242220A81194A32C3215D75A5D756665A940000000002800000000000040000DB
:2001600083C93693A218180020F46D230C91E500001300394000C104621F8F99014A8000AD
:200180004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A940006D
:2001A000000002800000005000083C93693A218180020F46D230C91E50000130039400079
:2001C000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C32C0
:2001E00015D75A5D756665A9400000000028000000600000 83C93693A218180020F46D237A
:200200000C91E500001300394000C104621F8F99014A80004104100200A000C30C1054303D
:2002200018242220A81194A32C3215D75A5D756665A940000000002800000007000083C9AB
:2002400036 93A218180020F46D230C91E500001300394000C104621F8F99014A80004104D3
:20026000100200A000C30C10543018242220A81194A32C3215D75A5D756665A940000000D1
:200280000028000000 0000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF4F
:2002A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5E
:2002C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3E
```

```
:2002E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1E
:20030000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFD
:20032000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDD
:20034000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBD
:20036000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9D
:20038000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7D
:2003A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5D
:2003C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3D
:2003E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1D
:00000001FF
```

= EEPROM Base Header

= Device 0 Address Map Header

= Device 1 Address Map Header

= Device 0, Channel 0 Register Data

= Device 1, Channel 0 Register Data

= Device 0, Channel 1 Register Data

= Device 1, Channel 1 Register Data

= Device 0, Channel 2 Register Data

= Device 1, Channel 2 Register Data

= Device 0, Channel 3 Register Data

= Device 1, Channel 3 Register Data

= Device 0, Share Register Data

= Device 1, Share Register Data

In the DS125DF410 hex file, the Base Header Bytes are 0xE10010. From Table 2, this means the following:

- CRC is enabled (Reg 0x00[7] = 1'b).
- Address Map Header is used (Reg 0x00[6] = 1'b).
- EEPROM > 256 Bytes (Reg 0x00[5] = 1'b).
- Common Channel is disabled (Reg 0x00[4] = 0'b).
- DEVICE COUNT = 2 Devices (Reg 0x00[3:0] = 0001'b).
- Max EEPROM Burst size = 16 bits (Reg 0x02 = 0x10).

Since EEPROM size > 256 Bytes, three Bytes are used in the Address Map Header. Recall that when the Address Map Header is three Bytes, the 3 LSBs of the Address Map Header Byte 2 become the 3 MSBs of the EEPROM start address. The start address for the two DS125DF410 devices can be derived from their Address Map Header values as shown below.

- Device 0 [CRC, Start Address] = [0x79, 0x0033]
- Device 1 [CRC, Start Address] = [0xE9, 0x015D]

Since Common Channel Configuration is disabled, four Channel Register data settings are programmed into the EEPROM, followed immediately by the Share Register Bytes. Comparing this

example with Example 1, it is easy to see that the EEPROM size increases noticeably when Common Channel Configuration is not used. The Channel and Share Register EEPROM data for Device 1 in this example are shown in Table 11.

Table 11. Device 1 Slot Settings

Channel 0 Register Data	0x04000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000
Channel 1 Register Data	0x05000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000
Channel 2 Register Data	0x06000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000
Channel 3 Register Data	0x07000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000
Share Register Data	0x0000

The CRC Byte for each device slot is calculated from the Base Header, non-CRC Bytes of the Address Map Header, and the relevant Channel Register and Share Register data Bytes. The values used for the CRC calculation in Device 1 are shown below:

E1001055D0104000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000**05000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000****06000083C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000****070000**83C93693A218180020F46D230C91E500001300394000C104621F8F99014A80004104100200A000C30C10543018242220A81194A32C3215D75A5D756665A9400000000028000000**0000**

CRC-8 calculation of the above data Bytes yields CRC = 0xE9.

6 SMBus-to-EEPROM Register Maps

The SMBus-to-EEPROM register maps for the 10-12.5 Gbps retimer devices are provided in the following tables, depending on the device type.

Table 12. EEPROM Register Map Lookup Table

Device Type	Channel Register Mapping	Share Register Mapping
2-Channel	Table 13	Table 14
4-Channel	Table 15	Table 16
16-Channel	Table 17	Table 18

To read each table, the blue column represents the EEPROM Address Byte, while the remaining columns to the right show Bits 7:0 for the corresponding EEPROM Byte. The matching SMBus register bit for each EEPROM address bit is shown in green. As an example, in Table 13, EEPROM Address Byte 0x05, Bit 4 maps to DS1xxDF111 SMBus Reg 0x0E[0], which is 1 by default.

Note: Share Register EEPROM data follows the last channel's Channel Register EEPROM data for each device's data slot. The actual number of Channel Registers used per EEPROM device slot depends on the retimer device type (2-channel, 4-channel, or 16-channel) and whether Common Channel Configuration is enabled. The EEPROM Register Maps presented in this section assume that Common Channel Configuration has been enabled.

Table 13. Default EEPROM Map for DS1xxDF111 Channel Register Data

EEPROM Address Byte		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	0 (0x00)								
SMB Register		0x03 [7]	0x03 [6]	0x03 [5]	0x03 [4]	0x03 [3]	0x03 [2]	0x03 [1]	0x03 [0]
Value	0x00	0	0	0	0	0	0	0	0
	1 (0x01)								
SMB Register		0x08 [4]	0x08 [3]	0x08 [2]	0x08 [1]	0x08 [0]	0x09 [7]	0x09 [6]	0x09 [5]
Value	0x00	0	0	0	0	0	0	0	0
	2 (0x02)								
SMB Register		0x09 [4]	0x09 [3]	0x09 [2]	0x09 [1]	0x09 [0]	0x0A [7]	0x0A [6]	0x0A [5]
Value	0x00	0	0	0	0	0	0	0	0
	3 (0x03)								
SMB Register		0x0A [4]	0x0A [3]	0x0A [2]	0x0A [1]	0x0A [0]	0x0B [4]	0x0B [3]	0x0B [2]
Value	0x83	1	0	0	0	0	0	1	1
	4 (0x04)								
SMB Register		0x0B [1]	0x0B [0]	0x0C [2]	0x0D [5]	0x0E [7]	0x0E [6]	0x0E [5]	0x0E [4]
Value	0xC9	1	1	0	0	1	0	0	1
	5 (0x05)								
SMB Register		0x0E [3]	0x0E [2]	0x0E [1]	0x0E [0]	0x0F [7]	0x0F [6]	0x0F [5]	0x0F [4]
Value	0x36	0	0	1	1	0	1	1	0
	6 (0x06)								
SMB Register		0x0F [3]	0x0F [2]	0x0F [1]	0x0F [0]	0x10 [7]	0x10 [6]	0x10 [5]	0x10 [4]
Value	0x93	1	0	0	1	0	0	1	1
	7 (0x07)								
SMB Register		0x10 [3]	0x10 [2]	0x10 [1]	0x10 [0]	0x11 [7]	0x11 [6]	0x11 [5]	0x11 [3]
Value	0xA2	1	0	1	0	0	0	1	0
	8 (0x08)								
SMB Register		0x11 [2]	0x11 [1]	0x11 [0]	0x12 [7]	0x12 [5]	0x12 [4]	0x12 [3]	0x12 [2]
Value	0x18	0	0	0	1	1	0	0	0
	9 (0x09)								
SMB Register		0x12 [1]	0x12 [0]	0x13 [6]	0x13 [4]	0x13 [3]	0x13 [2]	0x13 [1]	0x13 [0]
Value	0x10	0	0	0	1	0	0	0	0
	10 (0x0A)								
SMB Register		0x14 [7]	0x14 [6]	0x14 [5]	0x14 [4]	0x14 [3]	0x14 [2]	0x15 [7]	0x15 [6]
Value	0x00	0	0	0	0	0	0	0	0
	11 (0x0B)								
SMB Register		0x15 [5]	0x15 [4]	0x15 [3]	0x15 [2]	0x15 [1]	0x15 [0]	0x16 [7]	0x16 [6]
Value	0x41	0	1	0	0	0	0	0	1
	12 (0x0C)								
SMB Register		0x16 [5]	0x16 [4]	0x16 [3]	0x16 [2]	0x16 [1]	0x16 [0]	0x17 [7]	0x17 [6]
Value	0xE8	1	1	1	0	1	0	0	0
	13 (0x0D)								
SMB Register		0x17 [5]	0x17 [4]	0x17 [3]	0x17 [2]	0x17 [1]	0x17 [0]	0x18 [6]	0x18 [5]
Value	0x96	1	0	0	1	0	1	1	0
	14 (0x0E)								
SMB Register		0x18 [4]	0x19 [5]	0x19 [4]	0x19 [3]	0x19 [2]	0x19 [1]	0x19 [0]	0x1A [7]
Value	0x6E	0	1	1	0	1	1	1	0
	15 (0x0F)								
SMB Register		0x1A [6]	0x1A [5]	0x1A [4]	0x1B [1]	0x1B [0]	0x1C [7]	0x1C [6]	0x1C [5]
Value	0x19	0	0	0	1	1	0	0	1
	16 (0x10)								
SMB Register		0x1C [4]	0x1C [3]	0x1C [2]	0x1C [1]	0x1C [0]	0x1D [7]	0x1E [7]	0x1E [6]

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value	0x23		0	0	1	0	0	0	1	1
SMB Register		17 (0x11)	0x1E [5]	0x1E [3]	0x1E [2]	0x1E [1]	0x1E [0]	0x1F [7]	0x1F [6]	0x20 [7]
Value	0x8A		1	0	0	0	1	0	1	0
SMB Register		18 (0x12)	0x20 [6]	0x20 [5]	0x20 [4]	0x20 [3]	0x20 [2]	0x20 [1]	0x20 [0]	0x21 [7]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		19 (0x13)	0x21 [6]	0x21 [5]	0x21 [4]	0x21 [3]	0x21 [2]	0x21 [1]	0x21 [0]	0x22 [7]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		20 (0x14)	0x22 [6]	0x23 [7]	0x23 [6]	0x2A [7]	0x2A [6]	0x2A [5]	0x2A [4]	0x2A [3]
Value	0x26		0	0	1	0	0	1	1	0
SMB Register		21 (0x15)	0x2A [2]	0x2A [1]	0x2A [0]	0x2B [5]	0x2B [4]	0x2B [3]	0x2B [2]	0x2B [1]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		22 (0x16)	0x2B [0]	0x2C [6]	0x2C [5]	0x2C [4]	0x2C [3]	0x2C [2]	0x2C [1]	0x2C [0]
Value	0x72		0	1	1	1	0	0	1	0
SMB Register		23 (0x17)	0x2D [7]	0x2D [6]	0x2D [5]	0x2D [4]	0x2D [3]	0x2D [2]	0x2D [1]	0x2D [0]
Value	0x80		1	0	0	0	0	0	0	0
SMB Register		24 (0x18)	0x2E [5]	0x2E [2]	0x2F [7]	0x2F [6]	0x2F [5]	0x2F [4]	0x2F [3]	0x2F [2]
Value	0x19		0	0	0	1	1	0	0	1
SMB Register		25 (0x19)	0x2F [1]	0x30 [3]	0x30 [1]	0x30 [0]	0x31 [7]	0x31 [6]	0x31 [5]	0x31 [4]
Value	0x84		1	0	0	0	0	1	0	0
SMB Register		26 (0x1A)	0x31 [3]	0x32 [7]	0x32 [6]	0x32 [5]	0x32 [4]	0x32 [3]	0x32 [2]	0x32 [1]
Value	0x08		0	0	0	0	1	0	0	0
SMB Register		27 (0x1B)	0x32 [0]	0x33 [7]	0x33 [6]	0x33 [5]	0x33 [4]	0x33 [3]	0x33 [2]	0x33 [1]
Value	0xC4		1	1	0	0	0	1	0	0
SMB Register		28 (0x1C)	0x33 [0]	0x34 [6]	0x34 [5]	0x34 [4]	0x34 [3]	0x34 [2]	0x34 [1]	0x34 [0]
Value	0x3F		0	0	1	1	1	1	1	1
SMB Register		29 (0x1D)	0x35 [7]	0x35 [6]	0x35 [5]	0x35 [4]	0x35 [3]	0x35 [2]	0x35 [1]	0x35 [0]
Value	0x1F		0	0	0	1	1	1	1	1
SMB Register		30 (0x1E)	0x36 [7]	0x36 [6]	0x36 [5]	0x36 [4]	0x36 [2]	0x36 [1]	0x36 [0]	0x39 [6]
Value	0x32		0	0	1	1	0	0	1	0
SMB Register		31 (0x1F)	0x39 [5]	0x39 [4]	0x39 [3]	0x39 [2]	0x39 [1]	0x39 [0]	0x3A [7]	0x3A [6]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		32 (0x20)	0x3A [5]	0x3A [4]	0x3A [3]	0x3A [2]	0x3A [1]	0x3A [0]	0x3D [7]	0x3E [7]
Value	0x01		0	0	0	0	0	0	0	1
SMB Register		33 (0x21)	0x3F [7]	0x40 [7]	0x40 [6]	0x40 [5]	0x40 [4]	0x40 [3]	0x40 [2]	0x40 [1]

EEPROM Address Byte		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value 0x00		0	0	0	0	0	0	0	0
SMB Register	34 (0x22)	0x40 [0]	0x41 [7]	0x41 [6]	0x41 [5]	0x41 [4]	0x41 [3]	0x41 [2]	0x41 [1]
Value 0x20		0	0	1	0	0	0	0	0
SMB Register	35 (0x23)	0x41 [0]	0x42 [7]	0x42 [6]	0x42 [5]	0x42 [4]	0x42 [3]	0x42 [2]	0x42 [1]
Value 0x40		0	1	0	0	0	0	0	0
SMB Register	36 (0x24)	0x42 [0]	0x43 [7]	0x43 [6]	0x43 [5]	0x43 [4]	0x43 [3]	0x43 [2]	0x43 [1]
Value 0x28		0	0	1	0	1	0	0	0
SMB Register	37 (0x25)	0x43 [0]	0x44 [7]	0x44 [6]	0x44 [5]	0x44 [4]	0x44 [3]	0x44 [2]	0x44 [1]
Value 0x60		0	1	1	0	0	0	0	0
SMB Register	38 (0x26)	0x44 [0]	0x45 [7]	0x45 [6]	0x45 [5]	0x45 [4]	0x45 [3]	0x45 [2]	0x45 [1]
Value 0x48		0	1	0	0	1	0	0	0
SMB Register	39 (0x27)	0x45 [0]	0x46 [7]	0x46 [6]	0x46 [5]	0x46 [4]	0x46 [3]	0x46 [2]	0x46 [1]
Value 0x2A		0	0	1	0	1	0	1	0
SMB Register	40 (0x28)	0x46 [0]	0x47 [7]	0x47 [6]	0x47 [5]	0x47 [4]	0x47 [3]	0x47 [2]	0x47 [1]
Value 0x50		0	1	0	1	0	0	0	0
SMB Register	41 (0x29)	0x47 [0]	0x48 [7]	0x48 [6]	0x48 [5]	0x48 [4]	0x48 [3]	0x48 [2]	0x48 [1]
Value 0x58		0	1	0	1	1	0	0	0
SMB Register	42 (0x2A)	0x48 [0]	0x49 [7]	0x49 [6]	0x49 [5]	0x49 [4]	0x49 [3]	0x49 [2]	0x49 [1]
Value 0x4A		0	1	0	0	1	0	1	0
SMB Register	43 (0x2B)	0x49 [0]	0x4A [7]	0x4A [6]	0x4A [5]	0x4A [4]	0x4A [3]	0x4A [2]	0x4A [1]
Value 0xB4		1	0	1	1	0	1	0	0
SMB Register	44 (0x2C)	0x4A [0]	0x4B [7]	0x4B [6]	0x4B [5]	0x4B [4]	0x4B [3]	0x4B [2]	0x4B [1]
Value 0xEA		1	1	1	0	1	0	1	0
SMB Register	45 (0x2D)	0x4B [0]	0x4C [7]	0x4C [6]	0x4C [5]	0x4C [4]	0x4C [3]	0x4C [2]	0x4C [1]
Value 0xCC		1	1	0	0	1	1	0	0
SMB Register	46 (0x2E)	0x4C [0]	0x4D [7]	0x4D [6]	0x4D [5]	0x4D [4]	0x4D [3]	0x4D [2]	0x4D [1]
Value 0xD2		1	1	0	1	0	0	1	0
SMB Register	47 (0x2F)	0x4D [0]	0x4E [7]	0x4E [6]	0x4E [5]	0x4E [4]	0x4E [3]	0x4E [2]	0x4E [1]
Value 0xF3		1	1	1	1	0	0	1	1
SMB Register	48 (0x30)	0x4E [0]	0x4F [7]	0x4F [6]	0x4F [5]	0x4F [4]	0x4F [3]	0x4F [2]	0x4F [1]
Value 0x7C		0	1	1	1	1	1	0	0
SMB Register	49 (0x31)	0x4F [0]	0x50 [6]	0x50 [4]	0x50 [3]	0x50 [2]	0x50 [1]	0x50 [0]	0x51 [7]
Value 0x80		1	0	0	0	0	0	0	0
SMB Register	50 (0x32)	0x51 [6]	0x51 [5]	0x51 [4]	0x51 [3]	0x51 [2]	0x51 [1]	0x51 [0]	0x55 [6]

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		51 (0x33)	0x55 [5]	0x55 [3]	0x55 [2]	0x55 [1]	0x55 [0]	0x56 [3]	0x56 [2]	0x56 [1]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		52 (0x34)	0x56 [0]	0x57 [7]	0x57 [6]	0x57 [5]	0x57 [4]	0x57 [3]	0x57 [2]	0x57 [1]
Value	0x13		0	0	0	1	0	0	1	1
SMB Register		53 (0x35)	0x57 [0]	0x58 [7]	0x58 [6]	0x58 [5]	0x58 [4]	0x58 [3]	0x58 [2]	0x58 [1]
Value	0x58		0	1	0	1	1	0	0	0
SMB Register		54 (0x36)	0x58 [0]	0x59 [7]	0x59 [6]	0x59 [5]	0x59 [4]	0x59 [3]	0x59 [2]	0x59 [1]
Value	0xB8		1	0	1	1	1	0	0	0
SMB Register		55 (0x37)	0x59 [0]	0x5A [7]	0x5A [6]	0x5A [5]	0x5A [4]	0x5A [3]	0x5A [2]	0x5A [1]
Value	0x5E		0	1	0	1	1	1	1	0
SMB Register		56 (0x38)	0x5A [0]	0x5B [7]	0x5B [6]	0x5B [5]	0x5B [4]	0x5B [3]	0x5B [2]	0x5B [1]
Value	0xFF		1	1	1	1	1	1	1	1
SMB Register		57 (0x39)	0x5B [0]	0x5E [5]	0x60 [3]	0x60 [2]	0x60 [1]	0x60 [0]	0x62 [7]	0x62 [6]
Value	0xA9		1	0	1	0	1	0	0	1
SMB Register		58 (0x3A)	0x62 [5]	0x62 [4]	0x62 [3]	0x62 [2]	0x62 [1]	0x62 [0]	0x63 [7]	0x63 [6]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		59 (0x3B)	0x63 [5]	0x63 [4]	0x63 [3]	0x63 [2]	0x63 [1]	0x63 [0]	0x64 [7]	0x64 [6]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		60 (0x3C)	0x64 [5]	0x64 [4]	0x64 [3]	0x64 [2]	0x64 [1]	0x64 [0]	0x65 [7]	0x65 [6]
Value	0x00		0	0	0	0	0	0	0	0

Table 14. Default EEPROM Map for DS1xxDF111 Share Register Data

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMB Register		61 (0x3D)	0x03 [7]	0x03 [6]	0x03 [5]	0x03 [4]	0x03 [3]	0x06 [7]	0x06 [6]	0x06 [4]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		62 (0x3E)	0x06 [3]	0x06 [2]	0x06 [1]	0x06 [0]	0x07 [1]	0x07 [0]	0xFF [7]	0xFF [6]
Value	0x00		0	0	0	0	0	0	0	0

Table 15. Default EEPROM Map for DS1xxDF410 Channel Register Data

EEPROM Address Byte		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	0 (0x00)								
SMB Register		0x03 [7]	0x03 [6]	0x03 [5]	0x03 [4]	0x03 [3]	0x03 [2]	0x03 [1]	0x03 [0]
Value	0x00	0	0	0	0	0	0	0	0
	1 (0x01)								
SMB Register		0x08 [4]	0x08 [3]	0x08 [2]	0x08 [1]	0x08 [0]	0x09 [7]	0x09 [6]	0x09 [5]
Value	0x00	0	0	0	0	0	0	0	0
	2 (0x02)								
SMB Register		0x09 [4]	0x09 [3]	0x09 [2]	0x09 [1]	0x09 [0]	0x0A [7]	0x0A [6]	0x0A [5]
Value	0x00	0	0	0	0	0	0	0	0
	3 (0x03)								
SMB Register		0x0A [4]	0x0A [3]	0x0A [2]	0x0A [1]	0x0A [0]	0x0B [4]	0x0B [3]	0x0B [2]
Value	0x83	1	0	0	0	0	0	1	1
	4 (0x04)								
SMB Register		0x0B [1]	0x0B [0]	0x0C [2]	0x0D [5]	0x0E [7]	0x0E [6]	0x0E [5]	0x0E [4]
Value	0xC9	1	1	0	0	1	0	0	1
	5 (0x05)								
SMB Register		0x0E [3]	0x0E [2]	0x0E [1]	0x0E [0]	0x0F [7]	0x0F [6]	0x0F [5]	0x0F [4]
Value	0x36	0	0	1	1	0	1	1	0
	6 (0x06)								
SMB Register		0x0F [3]	0x0F [2]	0x0F [1]	0x0F [0]	0x10 [7]	0x10 [6]	0x10 [5]	0x10 [4]
Value	0x93	1	0	0	1	0	0	1	1
	7 (0x07)								
SMB Register		0x10 [3]	0x10 [2]	0x10 [1]	0x10 [0]	0x11 [7]	0x11 [6]	0x11 [5]	0x11 [3]
Value	0xA2	1	0	1	0	0	0	1	0
	8 (0x08)								
SMB Register		0x11 [2]	0x11 [1]	0x11 [0]	0x12 [7]	0x12 [5]	0x12 [4]	0x12 [3]	0x12 [2]
Value	0x18	0	0	0	1	1	0	0	0
	9 (0x09)								
SMB Register		0x12 [1]	0x12 [0]	0x13 [6]	0x13 [5]	0x13 [4]	0x13 [3]	0x13 [2]	0x13 [1]
Value	0x18	0	0	0	1	1	0	0	0
	10 (0x0A)								
SMB Register		0x13 [0]	0x14 [7]	0x14 [6]	0x14 [5]	0x14 [4]	0x14 [3]	0x14 [2]	0x15 [7]
Value	0x00	0	0	0	0	0	0	0	0
	11 (0x0B)								
SMB Register		0x15 [6]	0x15 [5]	0x15 [4]	0x15 [3]	0x15 [2]	0x15 [1]	0x15 [0]	0x16 [7]
Value	0x20	0	0	1	0	0	0	0	0
	12 (0x0C)								
SMB Register		0x16 [6]	0x16 [5]	0x16 [4]	0x16 [3]	0x16 [2]	0x16 [1]	0x16 [0]	0x17 [7]
Value	0xF4	1	1	1	1	0	1	0	0
	13 (0x0D)								
SMB Register		0x17 [6]	0x17 [5]	0x17 [4]	0x17 [3]	0x17 [2]	0x17 [1]	0x17 [0]	0x18 [6]
Value	0x6D	0	1	1	0	1	1	0	1
	14 (0x0E)								
SMB Register		0x18 [5]	0x18 [4]	0x19 [5]	0x19 [4]	0x19 [3]	0x19 [2]	0x19 [1]	0x19 [0]
Value	0x23	0	0	1	0	0	0	1	1
	15 (0x0F)								
SMB Register		0x1A [7]	0x1A [6]	0x1A [5]	0x1A [4]	0x1B [1]	0x1B [0]	0x1C [7]	0x1C [6]
Value	0x0C	0	0	0	0	1	1	0	0
	16 (0x10)								
SMB Register		0x1C [5]	0x1C [4]	0x1C [3]	0x1C [2]	0x1C [1]	0x1C [0]	0x1D [7]	0x1E [7]

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value	0x91		1	0	0	1	0	0	0	1
SMB Register		17 (0x11)	0x1E [6]	0x1E [5]	0x1E [3]	0x1E [2]	0x1E [1]	0x1E [0]	0x1F [7]	0x1F [6]
Value	0xE5		1	1	1	0	0	1	0	1
SMB Register		18 (0x12)	0x20 [7]	0x20 [6]	0x20 [5]	0x20 [4]	0x20 [3]	0x20 [2]	0x20 [1]	0x20 [0]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		19 (0x13)	0x21 [7]	0x21 [6]	0x21 [5]	0x21 [4]	0x21 [3]	0x21 [2]	0x21 [1]	0x21 [0]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		20 (0x14)	0x22 [7]	0x22 [6]	0x23 [7]	0x23 [6]	0x2A [7]	0x2A [6]	0x2A [5]	0x2A [4]
Value	0x13		0	0	0	1	0	0	1	1
SMB Register		21 (0x15)	0x2A [3]	0x2A [2]	0x2A [1]	0x2A [0]	0x2B [5]	0x2B [4]	0x2B [3]	0x2B [2]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		22 (0x16)	0x2B [1]	0x2B [0]	0x2C [6]	0x2C [5]	0x2C [4]	0x2C [3]	0x2C [2]	0x2C [1]
Value	0x39		0	0	1	1	1	0	0	1
SMB Register		23 (0x17)	0x2C [0]	0x2D [7]	0x2D [6]	0x2D [5]	0x2D [4]	0x2D [3]	0x2D [2]	0x2D [1]
Value	0x40		0	1	0	0	0	0	0	0
SMB Register		24 (0x18)	0x2D [0]	0x2E [5]	0x2E [2]	0x2F [7]	0x2F [6]	0x2F [5]	0x2F [4]	0x2F [3]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		25 (0x19)	0x2F [2]	0x2F [1]	0x30 [3]	0x30 [1]	0x30 [0]	0x31 [7]	0x31 [6]	0x31 [5]
Value	0xC1		1	1	0	0	0	0	0	1
SMB Register		26 (0x1A)	0x31 [4]	0x31 [3]	0x32 [7]	0x32 [6]	0x32 [5]	0x32 [4]	0x32 [3]	0x32 [2]
Value	0x04		0	0	0	0	0	1	0	0
SMB Register		27 (0x1B)	0x32 [1]	0x32 [0]	0x33 [7]	0x33 [6]	0x33 [5]	0x33 [4]	0x33 [3]	0x33 [2]
Value	0x62		0	1	1	0	0	0	1	0
SMB Register		28 (0x1C)	0x33 [1]	0x33 [0]	0x34 [6]	0x34 [5]	0x34 [4]	0x34 [3]	0x34 [2]	0x34 [1]
Value	0x1F		0	0	0	1	1	1	1	1
SMB Register		29 (0x1D)	0x34 [0]	0x35 [7]	0x35 [6]	0x35 [5]	0x35 [4]	0x35 [3]	0x35 [2]	0x35 [1]
Value	0x8F		1	0	0	0	1	1	1	1
SMB Register		30 (0x1E)	0x35 [0]	0x36 [7]	0x36 [6]	0x36 [5]	0x36 [4]	0x36 [2]	0x36 [1]	0x36 [0]
Value	0x99		1	0	0	1	1	0	0	1
SMB Register		31 (0x1F)	0x39 [6]	0x39 [5]	0x39 [4]	0x39 [3]	0x39 [2]	0x39 [1]	0x39 [0]	0x3A [7]
Value	0x01		0	0	0	0	0	0	0	1
SMB Register		32 (0x20)	0x3A [6]	0x3A [5]	0x3A [4]	0x3A [3]	0x3A [2]	0x3A [1]	0x3A [0]	0x3D [7]
Value	0x4A		0	1	0	0	1	0	1	0
		33 (0x21)								
SMB Register			0x3E [7]	0x3F [7]	0x40 [7]	0x40 [6]	0x40 [5]	0x40 [4]	0x40 [3]	0x40 [2]

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value	0x80		1	0	0	0	0	0	0	0
SMB Register		34 (0x22)	0x40 [1]	0x40 [0]	0x41 [7]	0x41 [6]	0x41 [5]	0x41 [4]	0x41 [3]	0x41 [2]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		35 (0x23)	0x41 [1]	0x41 [0]	0x42 [7]	0x42 [6]	0x42 [5]	0x42 [4]	0x42 [3]	0x42 [2]
Value	0x41		0	1	0	0	0	0	0	1
SMB Register		36 (0x24)	0x42 [1]	0x42 [0]	0x43 [7]	0x43 [6]	0x43 [5]	0x43 [4]	0x43 [3]	0x43 [2]
Value	0x04		0	0	0	0	0	1	0	0
SMB Register		37 (0x25)	0x43 [1]	0x43 [0]	0x44 [7]	0x44 [6]	0x44 [5]	0x44 [4]	0x44 [3]	0x44 [2]
Value	0x10		0	0	0	1	0	0	0	0
SMB Register		38 (0x26)	0x44 [1]	0x44 [0]	0x45 [7]	0x45 [6]	0x45 [5]	0x45 [4]	0x45 [3]	0x45 [2]
Value	0x02		0	0	0	0	0	0	1	0
SMB Register		39 (0x27)	0x45 [1]	0x45 [0]	0x46 [7]	0x46 [6]	0x46 [5]	0x46 [4]	0x46 [3]	0x46 [2]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		40 (0x28)	0x46 [1]	0x46 [0]	0x47 [7]	0x47 [6]	0x47 [5]	0x47 [4]	0x47 [3]	0x47 [2]
Value	0xA0		1	0	1	0	0	0	0	0
SMB Register		41 (0x29)	0x47 [1]	0x47 [0]	0x48 [7]	0x48 [6]	0x48 [5]	0x48 [4]	0x48 [3]	0x48 [2]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		42 (0x2A)	0x48 [1]	0x48 [0]	0x49 [7]	0x49 [6]	0x49 [5]	0x49 [4]	0x49 [3]	0x49 [2]
Value	0xC3		1	1	0	0	0	0	1	1
SMB Register		43 (0x2B)	0x49 [1]	0x49 [0]	0x4A [7]	0x4A [6]	0x4A [5]	0x4A [4]	0x4A [3]	0x4A [2]
Value	0x0C		0	0	0	0	1	1	0	0
SMB Register		44 (0x2C)	0x4A [1]	0x4A [0]	0x4B [7]	0x4B [6]	0x4B [5]	0x4B [4]	0x4B [3]	0x4B [2]
Value	0x10		0	0	0	1	0	0	0	0
SMB Register		45 (0x2D)	0x4B [1]	0x4B [0]	0x4C [7]	0x4C [6]	0x4C [5]	0x4C [4]	0x4C [3]	0x4C [2]
Value	0x54		0	1	0	1	0	1	0	0
SMB Register		46 (0x2E)	0x4C [1]	0x4C [0]	0x4D [7]	0x4D [6]	0x4D [5]	0x4D [4]	0x4D [3]	0x4D [2]
Value	0x30		0	0	1	1	0	0	0	0
SMB Register		47 (0x2F)	0x4D [1]	0x4D [0]	0x4E [7]	0x4E [6]	0x4E [5]	0x4E [4]	0x4E [3]	0x4E [2]
Value	0x18		0	0	0	1	1	0	0	0
SMB Register		48 (0x30)	0x4E [1]	0x4E [0]	0x4F [7]	0x4F [6]	0x4F [5]	0x4F [4]	0x4F [3]	0x4F [2]
Value	0x24		0	0	1	0	0	1	0	0
SMB Register		49 (0x31)	0x4F [1]	0x4F [0]	0x50 [7]	0x50 [6]	0x50 [5]	0x50 [4]	0x50 [3]	0x50 [2]
Value	0x22		0	0	1	0	0	0	1	0
		50 (0x32)								
SMB Register			0x50 [1]	0x50 [0]	0x51 [7]	0x51 [6]	0x51 [5]	0x51 [4]	0x51 [3]	0x51 [2]

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value	0x20		0	0	1	0	0	0	0	0
SMB Register		51 (0x33)	0x51 [1]	0x51 [0]	0x52 [7]	0x52 [6]	0x52 [5]	0x52 [4]	0x52 [3]	0x52 [2]
Value	0xA8		1	0	1	0	1	0	0	0
SMB Register		52 (0x34)	0x52 [1]	0x52 [0]	0x53 [7]	0x53 [6]	0x53 [5]	0x53 [4]	0x53 [3]	0x53 [2]
Value	0x11		0	0	0	1	0	0	0	1
SMB Register		53 (0x35)	0x53 [1]	0x53 [0]	0x54 [7]	0x54 [6]	0x54 [5]	0x54 [4]	0x54 [3]	0x54 [2]
Value	0x94		1	0	0	1	0	1	0	0
SMB Register		54 (0x36)	0x54 [1]	0x54 [0]	0x55 [7]	0x55 [6]	0x55 [5]	0x55 [4]	0x55 [3]	0x55 [2]
Value	0xA3		1	0	1	0	0	0	1	1
SMB Register		55 (0x37)	0x55 [1]	0x55 [0]	0x56 [7]	0x56 [6]	0x56 [5]	0x56 [4]	0x56 [3]	0x56 [2]
Value	0x2C		0	0	1	0	1	1	0	0
SMB Register		56 (0x38)	0x56 [1]	0x56 [0]	0x57 [7]	0x57 [6]	0x57 [5]	0x57 [4]	0x57 [3]	0x57 [2]
Value	0x32		0	0	1	1	0	0	1	0
SMB Register		57 (0x39)	0x57 [1]	0x57 [0]	0x58 [7]	0x58 [6]	0x58 [5]	0x58 [4]	0x58 [3]	0x58 [2]
Value	0x15		0	0	0	1	0	1	0	1
SMB Register		58 (0x3A)	0x58 [1]	0x58 [0]	0x59 [7]	0x59 [6]	0x59 [5]	0x59 [4]	0x59 [3]	0x59 [2]
Value	0xD7		1	1	0	1	0	1	1	1
SMB Register		59 (0x3B)	0x59 [1]	0x59 [0]	0x5A [7]	0x5A [6]	0x5A [5]	0x5A [4]	0x5A [3]	0x5A [2]
Value	0x5A		0	1	0	1	1	0	1	0
SMB Register		60 (0x3C)	0x5A [1]	0x5A [0]	0x5B [7]	0x5B [6]	0x5B [5]	0x5B [4]	0x5B [3]	0x5B [2]
Value	0x5D		0	1	0	1	1	1	0	1
SMB Register		61 (0x3D)	0x5B [1]	0x5B [0]	0x5C [7]	0x5C [6]	0x5C [5]	0x5C [4]	0x5C [3]	0x5C [2]
Value	0x75		0	1	1	1	0	1	0	1
SMB Register		62 (0x3E)	0x5C [1]	0x5C [0]	0x5D [7]	0x5D [6]	0x5D [5]	0x5D [4]	0x5D [3]	0x5D [2]
Value	0x66		0	1	1	0	0	1	1	0
SMB Register		63 (0x3F)	0x5D [1]	0x5D [0]	0x5E [7]	0x5E [6]	0x5E [5]	0x5E [4]	0x5E [3]	0x5E [2]
Value	0x65		0	1	1	0	0	1	0	1
SMB Register		64 (0x40)	0x5E [1]	0x5E [0]	0x5F [7]	0x5F [6]	0x5F [5]	0x5F [4]	0x5F [3]	0x5F [2]
Value	0xA9		1	0	1	0	1	0	0	1
SMB Register		65 (0x41)	0x5F [1]	0x5F [0]	0x60 [7]	0x60 [6]	0x60 [5]	0x60 [4]	0x60 [3]	0x60 [2]
Value	0x40		0	1	0	0	0	0	0	0
SMB Register		66 (0x42)	0x60 [1]	0x60 [0]	0x61 [7]	0x61 [6]	0x61 [5]	0x61 [4]	0x61 [3]	0x61 [2]
Value	0x00		0	0	0	0	0	0	0	0
		67 (0x43)								
SMB Register			0x61 [1]	0x61 [0]	0x62 [7]	0x62 [6]	0x62 [5]	0x62 [4]	0x62 [3]	0x62 [2]

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Value	0x00		0	0	0	0	0	0	0	0
		68 (0x44)								
SMB Register			0x62 [1]	0x62 [0]	0x63 [7]	0x63 [6]	0x63 [5]	0x63 [4]	0x63 [3]	0x63 [2]
Value	0x00		0	0	0	0	0	0	0	0
		69 (0x45)								
SMB Register			0x63 [1]	0x63 [0]	0x64 [7]	0x64 [6]	0x64 [5]	0x64 [4]	0x64 [3]	0x64 [2]
Value	0x00		0	0	0	0	0	0	0	0
		70 (0x46)								
SMB Register			0x64 [1]	0x64 [0]	0x69 [3]	0x69 [2]	0x69 [1]	0x69 [0]	0x6B [7]	0x6B [6]
Value	0x28		0	0	1	0	1	0	0	0
		71 (0x47)								
SMB Register			0x6B [5]	0x6B [4]	0x6B [3]	0x6B [2]	0x6B [1]	0x6B [0]	0x6C [7]	0x6C [6]
Value	0x00		0	0	0	0	0	0	0	0
		72 (0x48)								
SMB Register			0x6C [5]	0x6C [4]	0x6C [3]	0x6C [2]	0x6C [1]	0x6C [0]	0x6D [7]	0x6D [6]
Value	0x00		0	0	0	0	0	0	0	0
		73 (0x49)								
SMB Register			0x6D [5]	0x6D [4]	0x6D [3]	0x6D [2]	0x6D [1]	0x6D [0]	0x6E [7]	0x6E [6]
Value	0x00		0	0	0	0	0	0	0	0

Table 16. Default EEPROM Map for DS1xxDF410 Share Register Data

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		74 (0x4A)								
SMB Register			0x03 [7]	0x03 [6]	0x03 [5]	0x03 [4]	0x03 [3]	0x03 [2]	0x03 [1]	0x03 [0]
Value	0x00		0	0	0	0	0	0	0	0
		75 (0x4B)								
SMB Register			0x06 [7]	0x06 [6]	0x06 [5]	0x06 [4]	0x06 [3]	0x06 [2]	0x06 [1]	0x06 [0]
Value	0x00		0	0	0	0	0	0	0	0

Table 17. Default EEPROM Map for DS1xxDF1610 Channel Register Data

EEPROM Address Byte		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	0 (0x00)								
SMB Register		0x03 [7]	0x03 [6]	0x03 [5]	0x03 [4]	0x03 [3]	0x03 [2]	0x03 [1]	0x03 [0]
Value	0x00	0	0	0	0	0	0	0	0
	1 (0x01)								
SMB Register		0x04 [2]	0x08 [7]	0x08 [6]	0x08 [5]	0x08 [4]	0x08 [3]	0x08 [2]	0x08 [1]
Value	0x30	0	0	1	1	0	0	0	0
	2 (0x02)								
SMB Register		0x08 [0]	0x09 [7]	0x09 [6]	0x09 [5]	0x09 [4]	0x09 [3]	0x09 [2]	0x09 [1]
Value	0x00	0	0	0	0	0	0	0	0
	3 (0x03)								
SMB Register		0x09 [0]	0x0A [7]	0x0A [6]	0x0A [5]	0x0A [4]	0x0B [7]	0x0B [6]	0x0B [5]
Value	0x2B	0	0	1	0	1	0	1	1
	4 (0x04)								
SMB Register		0x0B [4]	0x0B [3]	0x0B [2]	0x0B [1]	0x0B [0]	0x0C [3]	0x0C [1]	0x0C [0]
Value	0x7C	0	1	1	1	1	1	0	0
	5 (0x05)								
SMB Register		0x0D [5]	0x0D [4]	0x0D [3]	0x0D [2]	0x10 [7]	0x10 [6]	0x10 [5]	0x10 [4]
Value	0xD3	1	1	0	1	0	0	1	1
	6 (0x06)								
SMB Register		0x10 [3]	0x10 [2]	0x10 [1]	0x10 [0]	0x11 [7]	0x11 [6]	0x11 [5]	0x11 [3]
Value	0xA2	1	0	1	0	0	0	1	0
	7 (0x07)								
SMB Register		0x11 [2]	0x11 [1]	0x11 [0]	0x12 [7]	0x12 [5]	0x12 [4]	0x12 [3]	0x12 [2]
Value	0x18	0	0	0	1	1	0	0	0
	8 (0x08)								
SMB Register		0x12 [1]	0x12 [0]	0x13 [4]	0x13 [3]	0x13 [2]	0x14 [7]	0x14 [6]	0x14 [5]
Value	0x20	0	0	1	0	0	0	0	0
	9 (0x09)								
SMB Register		0x14 [4]	0x14 [3]	0x14 [2]	0x15 [7]	0x15 [5]	0x15 [4]	0x15 [2]	0x15 [1]
Value	0x05	0	0	0	0	0	1	0	1
	10 (0x0A)								
SMB Register		0x15 [0]	0x16 [7]	0x16 [6]	0x16 [5]	0x16 [4]	0x16 [3]	0x16 [2]	0x16 [1]
Value	0x3D	0	0	1	1	1	1	0	1
	11 (0x0B)								
SMB Register		0x16 [0]	0x17 [7]	0x17 [6]	0x17 [5]	0x17 [4]	0x17 [3]	0x17 [2]	0x17 [1]
Value	0x1B	0	0	0	1	1	0	1	1
	12 (0x0C)								
SMB Register		0x17 [0]	0x18 [6]	0x18 [5]	0x18 [4]	0x19 [5]	0x19 [4]	0x19 [3]	0x19 [2]
Value	0x48	0	1	0	0	1	0	0	0
	13 (0x0D)								
SMB Register		0x19 [1]	0x19 [0]	0x1A [7]	0x1A [6]	0x1A [5]	0x1A [4]	0x1A [3]	0x1A [2]
Value	0x28	0	0	1	0	1	0	0	0
	14 (0x0E)								
SMB Register		0x1B [1]	0x1B [0]	0x1C [7]	0x1C [6]	0x1C [5]	0x1C [4]	0x1C [3]	0x1C [2]
Value	0xE4	1	1	1	0	0	1	0	0
	15 (0x0F)								
SMB Register		0x1D [7]	0x1E [7]	0x1E [6]	0x1E [5]	0x1E [3]	0x1E [2]	0x1E [1]	0x1E [0]
Value	0x79	0	1	1	1	1	0	0	1
	16 (0x10)								
SMB Register		0x1F [7]	0x1F [6]	0x20 [7]	0x20 [6]	0x20 [5]	0x20 [4]	0x20 [3]	0x20 [2]

Value	0x40		0	1	0	0	0	0	0	0
SMB Register		17 (0x11)	0x20 [1]	0x20 [0]	0x21 [7]	0x21 [6]	0x21 [5]	0x21 [4]	0x21 [3]	0x21 [2]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		18 (0x12)	0x21 [1]	0x21 [0]	0x23 [6]	0x2A [7]	0x2A [6]	0x2A [5]	0x2A [4]	0x2A [3]
Value	0x26		0	0	1	0	0	1	1	0
SMB Register		19 (0x13)	0x2A [2]	0x2A [1]	0x2A [0]	0x2B [5]	0x2B [4]	0x2B [3]	0x2B [2]	0x2B [1]
Value	0x07		0	0	0	0	0	1	1	1
SMB Register		20 (0x14)	0x2B [0]	0x2C [6]	0x2C [5]	0x2C [4]	0x2C [3]	0x2C [2]	0x2C [1]	0x2C [0]
Value	0xF2		1	1	1	1	0	0	1	0
SMB Register		21 (0x15)	0x2D [7]	0x2D [6]	0x2D [5]	0x2D [4]	0x2D [3]	0x2D [2]	0x2D [1]	0x2D [0]
Value	0x04		0	0	0	0	0	1	0	0
SMB Register		22 (0x16)	0x2F [7]	0x2F [6]	0x2F [5]	0x2F [4]	0x2F [3]	0x2F [2]	0x2F [1]	0x30 [6]
Value	0x36		0	0	1	1	0	1	1	0
SMB Register		23 (0x17)	0x31 [6]	0x31 [5]	0x31 [4]	0x31 [3]	0x32 [7]	0x32 [6]	0x32 [5]	0x32 [4]
Value	0x41		0	1	0	0	0	0	0	1
SMB Register		24 (0x18)	0x32 [3]	0x32 [2]	0x32 [1]	0x32 [0]	0x33 [7]	0x33 [6]	0x33 [5]	0x33 [4]
Value	0x18		0	0	0	1	1	0	0	0
SMB Register		25 (0x19)	0x33 [3]	0x33 [2]	0x33 [1]	0x33 [0]	0x34 [6]	0x34 [5]	0x34 [4]	0x34 [3]
Value	0x87		1	0	0	0	0	1	1	1
SMB Register		26 (0x1A)	0x34 [2]	0x34 [1]	0x34 [0]	0x35 [7]	0x35 [6]	0x35 [4]	0x35 [3]	0x35 [2]
Value	0xE7		1	1	1	0	0	1	1	1
SMB Register		27 (0x1B)	0x35 [1]	0x35 [0]	0x36 [5]	0x36 [4]	0x36 [3]	0x36 [2]	0x39 [6]	0x39 [5]
Value	0xF0		1	1	1	1	0	0	0	0
SMB Register		28 (0x1C)	0x39 [4]	0x39 [3]	0x39 [2]	0x39 [1]	0x39 [0]	0x3A [7]	0x3A [6]	0x3A [5]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		29 (0x1D)	0x3A [4]	0x3A [3]	0x3A [2]	0x3A [1]	0x3A [0]	0x3D [7]	0x3D [6]	0x3D [5]
Value	0x01		0	0	0	0	0	0	0	1
SMB Register		30 (0x1E)	0x3D [4]	0x3D [3]	0x3D [2]	0x3D [1]	0x3D [0]	0x3E [7]	0x3E [6]	0x3E [5]
Value	0xAA		1	0	1	0	1	0	1	0
SMB Register		31 (0x1F)	0x3E [4]	0x3E [3]	0x3E [2]	0x3E [1]	0x3E [0]	0x3F [7]	0x3F [6]	0x3F [5]
Value	0x1E		0	0	0	1	1	1	1	0
SMB Register		32 (0x20)	0x3F [4]	0x3F [3]	0x3F [2]	0x3F [1]	0x3F [0]	0x40 [7]	0x40 [6]	0x40 [5]
Value	0x38		0	0	1	1	1	0	0	0
SMB Register		33 (0x21)	0x40 [4]	0x40 [3]	0x40 [2]	0x40 [1]	0x40 [0]	0x41 [7]	0x41 [6]	0x41 [5]
Value	0x00		0	0	0	0	0	0	0	0
		34 (0x22)								

SMB Register		0x41 [4]	0x41 [3]	0x41 [2]	0x41 [1]	0x41 [0]	0x42 [7]	0x42 [6]	0x42 [5]
Value	0x08	0	0	0	0	1	0	0	0
SMB Register	35 (0x23)	0x42 [4]	0x42 [3]	0x42 [2]	0x42 [1]	0x42 [0]	0x43 [7]	0x43 [6]	0x43 [5]
Value	0x20	0	0	1	0	0	0	0	0
SMB Register	36 (0x24)	0x43 [4]	0x43 [3]	0x43 [2]	0x43 [1]	0x43 [0]	0x44 [7]	0x44 [6]	0x44 [5]
Value	0x82	1	0	0	0	0	0	1	0
SMB Register	37 (0x25)	0x44 [4]	0x44 [3]	0x44 [2]	0x44 [1]	0x44 [0]	0x45 [7]	0x45 [6]	0x45 [5]
Value	0x00	0	0	0	0	0	0	0	0
SMB Register	38 (0x26)	0x45 [4]	0x45 [3]	0x45 [2]	0x45 [1]	0x45 [0]	0x46 [7]	0x46 [6]	0x46 [5]
Value	0x40	0	1	0	0	0	0	0	0
SMB Register	39 (0x27)	0x46 [4]	0x46 [3]	0x46 [2]	0x46 [1]	0x46 [0]	0x47 [7]	0x47 [6]	0x47 [5]
Value	0x14	0	0	0	1	0	1	0	0
SMB Register	40 (0x28)	0x47 [4]	0x47 [3]	0x47 [2]	0x47 [1]	0x47 [0]	0x48 [7]	0x48 [6]	0x48 [5]
Value	0x00	0	0	0	0	0	0	0	0
SMB Register	41 (0x29)	0x48 [4]	0x48 [3]	0x48 [2]	0x48 [1]	0x48 [0]	0x49 [7]	0x49 [6]	0x49 [5]
Value	0x18	0	0	0	1	1	0	0	0
SMB Register	42 (0x2A)	0x49 [4]	0x49 [3]	0x49 [2]	0x49 [1]	0x49 [0]	0x4A [7]	0x4A [6]	0x4A [5]
Value	0x61	0	1	1	0	0	0	0	1
SMB Register	43 (0x2B)	0x4A [4]	0x4A [3]	0x4A [2]	0x4A [1]	0x4A [0]	0x4B [7]	0x4B [6]	0x4B [5]
Value	0x82	1	0	0	0	0	0	1	0
SMB Register	44 (0x2C)	0x4B [4]	0x4B [3]	0x4B [2]	0x4B [1]	0x4B [0]	0x4C [7]	0x4C [6]	0x4C [5]
Value	0x0A	0	0	0	0	1	0	1	0
SMB Register	45 (0x2D)	0x4C [4]	0x4C [3]	0x4C [2]	0x4C [1]	0x4C [0]	0x4D [7]	0x4D [6]	0x4D [5]
Value	0x86	1	0	0	0	0	1	1	0
SMB Register	46 (0x2E)	0x4D [4]	0x4D [3]	0x4D [2]	0x4D [1]	0x4D [0]	0x4E [7]	0x4E [6]	0x4E [5]
Value	0x03	0	0	0	0	0	0	1	1
SMB Register	47 (0x2F)	0x4E [4]	0x4E [3]	0x4E [2]	0x4E [1]	0x4E [0]	0x4F [7]	0x4F [6]	0x4F [5]
Value	0x04	0	0	0	0	0	1	0	0
SMB Register	48 (0x30)	0x4F [4]	0x4F [3]	0x4F [2]	0x4F [1]	0x4F [0]	0x50 [7]	0x50 [6]	0x50 [5]
Value	0x84	1	0	0	0	0	1	0	0
SMB Register	49 (0x31)	0x50 [4]	0x50 [3]	0x50 [2]	0x50 [1]	0x50 [0]	0x51 [7]	0x51 [6]	0x51 [5]
Value	0x44	0	1	0	0	0	1	0	0
SMB Register	50 (0x32)	0x51 [4]	0x51 [3]	0x51 [2]	0x51 [1]	0x51 [0]	0x52 [7]	0x52 [6]	0x52 [5]
Value	0x15	0	0	0	1	0	1	0	1
SMB Register	51 (0x33)	0x52 [4]	0x52 [3]	0x52 [2]	0x52 [1]	0x52 [0]	0x53 [7]	0x53 [6]	0x53 [5]
Value	0x02	0	0	0	0	0	0	1	0

SMB Register	52 (0x34)	0x53 [4]	0x53 [3]	0x53 [2]	0x53 [1]	0x53 [0]	0x54 [7]	0x54 [6]	0x54 [5]
Value 0x32		0	0	1	1	0	0	1	0
SMB Register	53 (0x35)	0x54 [4]	0x54 [3]	0x54 [2]	0x54 [1]	0x54 [0]	0x55 [7]	0x55 [6]	0x55 [5]
Value 0x94		1	0	0	1	0	1	0	0
SMB Register	54 (0x36)	0x55 [4]	0x55 [3]	0x55 [2]	0x55 [1]	0x55 [0]	0x56 [7]	0x56 [6]	0x56 [5]
Value 0x65		0	1	1	0	0	1	0	1
SMB Register	55 (0x37)	0x56 [4]	0x56 [3]	0x56 [2]	0x56 [1]	0x56 [0]	0x57 [7]	0x57 [6]	0x57 [5]
Value 0x86		1	0	0	0	0	1	1	0
SMB Register	56 (0x38)	0x57 [4]	0x57 [3]	0x57 [2]	0x57 [1]	0x57 [0]	0x58 [7]	0x58 [6]	0x58 [5]
Value 0x42		0	1	0	0	0	0	1	0
SMB Register	57 (0x39)	0x58 [4]	0x58 [3]	0x58 [2]	0x58 [1]	0x58 [0]	0x59 [7]	0x59 [6]	0x59 [5]
Value 0xBA		1	0	1	1	1	0	1	0
SMB Register	58 (0x3A)	0x59 [4]	0x59 [3]	0x59 [2]	0x59 [1]	0x59 [0]	0x5A [7]	0x5A [6]	0x5A [5]
Value 0xEB		1	1	1	0	1	0	1	1
SMB Register	59 (0x3B)	0x5A [4]	0x5A [3]	0x5A [2]	0x5A [1]	0x5A [0]	0x5B [7]	0x5B [6]	0x5B [5]
Value 0x4B		0	1	0	0	1	0	1	1
SMB Register	60 (0x3C)	0x5B [4]	0x5B [3]	0x5B [2]	0x5B [1]	0x5B [0]	0x5C [7]	0x5C [6]	0x5C [5]
Value 0xAE		1	0	1	0	1	1	1	0
SMB Register	61 (0x3D)	0x5C [4]	0x5C [3]	0x5C [2]	0x5C [1]	0x5C [0]	0x5D [7]	0x5D [6]	0x5D [5]
Value 0xAC		1	0	1	0	1	1	0	0
SMB Register	62 (0x3E)	0x5D [4]	0x5D [3]	0x5D [2]	0x5D [1]	0x5D [0]	0x5E [7]	0x5E [6]	0x5E [5]
Value 0xCC		1	1	0	0	1	1	0	0
SMB Register	63 (0x3F)	0x5E [4]	0x5E [3]	0x5E [2]	0x5E [1]	0x5E [0]	0x5F [7]	0x5F [6]	0x5F [5]
Value 0xB5		1	0	1	1	0	1	0	1
SMB Register	64 (0x40)	0x5F [4]	0x5F [3]	0x5F [2]	0x5F [1]	0x5F [0]	0x60 [7]	0x60 [6]	0x60 [5]
Value 0x28		0	0	1	0	1	0	0	0
SMB Register	65 (0x41)	0x60 [4]	0x60 [3]	0x60 [2]	0x60 [1]	0x60 [0]	0x61 [7]	0x61 [6]	0x61 [5]
Value 0x00		0	0	0	0	0	0	0	0
SMB Register	66 (0x42)	0x61 [4]	0x61 [3]	0x61 [2]	0x61 [1]	0x61 [0]	0x62 [7]	0x62 [6]	0x62 [5]
Value 0x00		0	0	0	0	0	0	0	0
SMB Register	67 (0x43)	0x62 [4]	0x62 [3]	0x62 [2]	0x62 [1]	0x62 [0]	0x63 [7]	0x63 [6]	0x63 [5]
Value 0x00		0	0	0	0	0	0	0	0
SMB Register	68 (0x44)	0x63 [4]	0x63 [3]	0x63 [2]	0x63 [1]	0x63 [0]	0x64 [7]	0x64 [6]	0x64 [5]
Value 0x00		0	0	0	0	0	0	0	0
SMB Register	69 (0x45)	0x64 [4]	0x64 [3]	0x64 [2]	0x64 [1]	0x64 [0]	0x67 [7]	0x67 [6]	0x67 [5]

Value	0x01		0	0	0	0	0	0	0	1
SMB Register		70 (0x46)	0x69 [3]	0x69 [2]	0x69 [1]	0x69 [0]	0x6A [7]	0x6A [6]	0x6A [5]	0x6A [4]
Value	0xA2		1	0	1	0	0	0	1	0
SMB Register		71 (0x47)	0x6A [3]	0x6A [2]	0x6A [1]	0x6A [0]	0x6B [7]	0x6B [6]	0x6B [5]	0x6B [4]
Value	0x24		0	0	1	0	0	1	0	0
SMB Register		72 (0x48)	0x6B [3]	0x6B [2]	0x6B [1]	0x6B [0]	0x6C [7]	0x6C [6]	0x6C [5]	0x6C [4]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		73 (0x49)	0x6C [3]	0x6C [2]	0x6C [1]	0x6C [0]	0x6D [7]	0x6D [6]	0x6D [5]	0x6D [4]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		74 (0x4A)	0x6D [3]	0x6D [2]	0x6D [1]	0x6D [0]	0x6E [7]	0x6E [6]	0x6F [7]	0x6F [6]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		75 (0x4B)	0x6F [5]	0x70 [3]	0x70 [2]	0x70 [1]	0x70 [0]	0x76 [7]	0x76 [6]	0x76 [5]
Value	0x19		0	0	0	1	1	0	0	1
SMB Register		76 (0x4C)	0x76 [4]	0x76 [3]	0x76 [2]	0x76 [1]	0x76 [0]	0x77 [6]	0x77 [5]	0x77 [4]
Value	0x11		0	0	0	1	0	0	0	1
SMB Register		77 (0x4D)	0x77 [3]	0x77 [2]	0x77 [1]	0x77 [0]	0x79 [3]	0x79 [2]	0x7D [7]	0x7D [6]
Value	0xA1		1	0	1	0	0	0	0	1
SMB Register		78 (0x4E)	0x7D [5]	0x7D [4]	0x7D [3]	0x7D [2]	0x7D [1]	0x7D [0]	0x7E [7]	0x7E [6]
Value	0x20		0	0	1	0	0	0	0	0
SMB Register		79 (0x4F)	0x7E [5]	0x7E [4]	0x7E [3]	0x7E [2]	0x7E [1]	0x7E [0]	0x7F [5]	0x7F [4]
Value	0x4F		0	1	0	0	1	1	1	1
SMB Register		80 (0x50)	0x7F [3]	0x7F [2]	0x7F [1]	0x7F [0]	0x8D [6]	0x8D [5]	0x8D [4]	0x8D [3]
Value	0xA0		1	0	1	0	0	0	0	0
SMB Register		81 (0x51)	0x8D [2]	0x8D [1]	0x8D [0]	0x8E [5]	0x8E [4]	0x8E [3]	0x8E [2]	0x8E [1]
Value	0x4E		0	1	0	0	1	1	1	0
SMB Register		82 (0x52)	0x8E [0]	0x90 [7]	0x90 [6]	0x90 [5]	0x90 [4]	0x90 [3]	0x90 [2]	0x90 [1]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		83 (0x53)	0x90 [0]	0x91 [7]	0x91 [6]	0x91 [5]	0x91 [4]	0x91 [3]	0x91 [2]	0x91 [1]
Value	0x00		0	0	0	0	0	0	0	0
SMB Register		84 (0x54)	0x91 [0]	0x96 [5]	0x96 [4]	0x96 [3]	0x96 [2]	0x96 [1]	0x96 [0]	0x98 [5]
Value	0x08		0	0	0	0	1	0	0	0
SMB Register		85 (0x55)	0x98 [4]	0x98 [3]	0x98 [2]	0x98 [1]	0x98 [0]	0x99 [7]	0x99 [6]	0x99 [5]
Value	0x61		0	1	1	0	0	0	0	1
SMB Register		86 (0x56)	0x99 [4]	0x99 [3]	0x99 [2]	0x99 [1]	0x99 [0]	0x9A [7]	0x9A [6]	0x9A [5]
Value	0xF9		1	1	1	1	1	0	0	1
		87 (0x57)								

SMB Register			0x9A [4]	0x9A [3]	0x9A [2]	0x9A [1]	0x9A [0]	0x9B [2]	0x9B [1]	0x9B [0]
Value	0xF8		1	1	1	1	1	0	0	0

Table 18. Default EEPROM Map for DS1xxDF1610 Share Register Data

EEPROM Address Byte			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMB Register		88 (0x58)	0x00 [3]	0x02 [6]	0x02 [5]	0x03 [2]	0x0A [0]	0x0B [4]	0x0F [7]	0x0F [6]
Value	0x0B		0	0	0	0	1	0	1	1
SMB Register		89 (0x59)	0x0F [5]	0x0F [4]	0x0F [3]	0x0F [2]	0x0F [1]	0x0F [0]	0x10 [7]	0x10 [6]
Value	0xFF		1	1	1	1	1	1	1	1
SMB Register		90 (0x5A)	0x10 [5]	0x10 [4]	0x10 [3]	0x10 [2]	0x10 [1]	0x10 [0]	0xFF [3]	0xFF [2]
Value	0xFC		1	1	1	1	1	1	0	0

7 Summary

In this application note, EEPROM programming is explained for TI's 10-12.5 Gbps retimers. To program an EEPROM efficiently and accurately, system designers must decide when to use programming features for retimers such as Common Channel Configuration, Address Map Headers, and CRC Bytes. With a complete understanding of how to program and interpret EEPROM hex files for TI's 10-12.5 Gbps retimers, system designers are better equipped to generate their own customized hex files and increase the efficiency of their final designs.

8 References

1. DS125DF410 Datasheet (SNLS398G)
2. DS125DF111 Datasheet (SNLS450A)
3. DS100DF410EVK, DS110DF410EVK and DS125DF410EVM Evaluation Board Software Installation, Setup, and Operating Guide (SNLU126A)