NXP Semiconductors
Application Note

Document Number: AN5132
Rev. 1.0, 7/2016

PF3000 OTP programming instructions

1 Introduction

This application note provides a detailed description of the PF3000's One-Time Programmable (OTP) function. It outlines the system requirements and the instructions needed to program the internal fuses for a selected power-up configuration. All examples assume the customer is using silicon revision P1.1 or higher.

NXP analog ICs are manufactured using the SMARTMOS process, a combinational BiCMOS manufacturing flow that integrates precision analog, power functions and dense CMOS logic together on a single cost-effective die.

Contents

2 Hardware considerations

The minimum system requirements for programming the OTP fuses are:

1. An I^2C communication bridge for communicating with the PF3000
2. A 1.7 V to 3.6 V power supply at VDDIO (power to I^2C block and pull-up resistors for the SCL and SDA lines).
3. A 9.5 V, 100 mA power supply at VDDOTP bypassed by 2 x 10 mF capacitors. See section OTP Programming Example for details.
4. An input voltage of 3.3 V at the VIN pin

The KITPF3000FRDMPGM programming board includes all of these features in a USB standalone solution and is plug-and-play compatible with the PF3000 GUI software.

Figure 1 shows the minimum requirements for programming the PF3000. For programming the PF3000 on an application board, observe the hardware constraints outline in Section 3 "Programming the PF3000 on an application board".

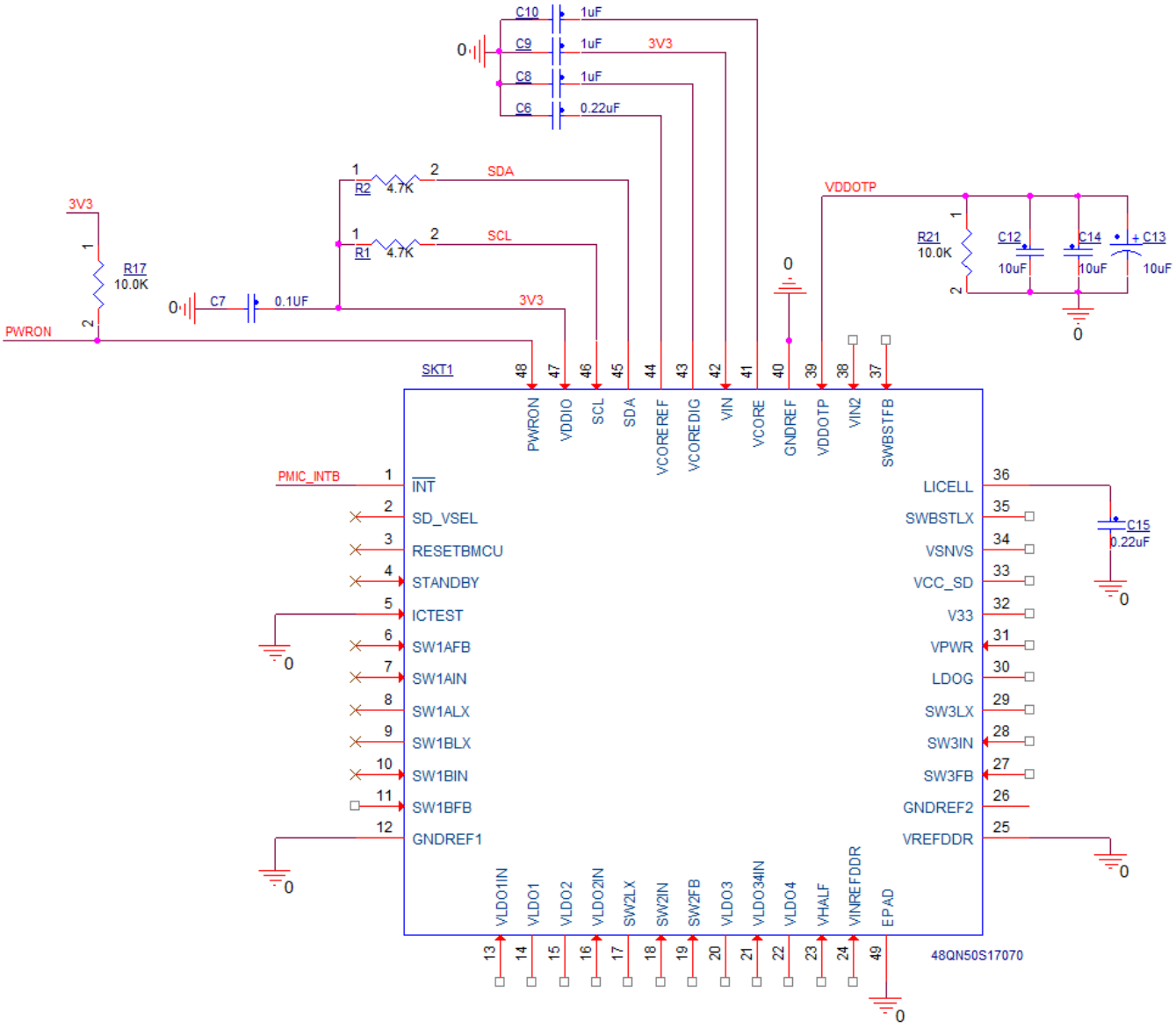

Figure 1. Minimum OTP programming requirements diagram

3 Programming the PF3000 on an application board

When programming the PF3000 in an application board, you must apply voltages at the VIN, VDDIO, and VDDOTP pins. Make the necessary adjustments to assure that voltages are applied on these rails in a fully populated system board.

3.1 Isolating SCL/SDA

During OTP programming, the PF3000 is connected to an I^2C communications bridge (such as the FRDM-KL25Z) and receives commands via the SCL and SDA pins. In a typical application, the SCL and SDA pins of the PF3000 are connected to the communication ports of an I^2C master, (ie.the processor). Depending on how the ports in the processor are designed, it may or may not be valid to communicate with the PF3000 using an external dongle while the SCL/SDA pins are still connected to the processor. This is particularly true when the processor is unpowered due to a yet-to-be-programmed PF3000.

Use an external programmer to isolate the SCL/SDA lines going to the processor while communicating with the PF3000. (See the example in Figure 2.) In the normally closed position of the analog switch (NLAS3158 or similar), SCL and SDA on the PF3000 are connected to the processor. When the signal Programmer_Select_O/P is high, SCL and SDA on the PF3000 are connected to the external programming interface. The Programmer_Select_O/P signal can be generated by the programming interface as well.

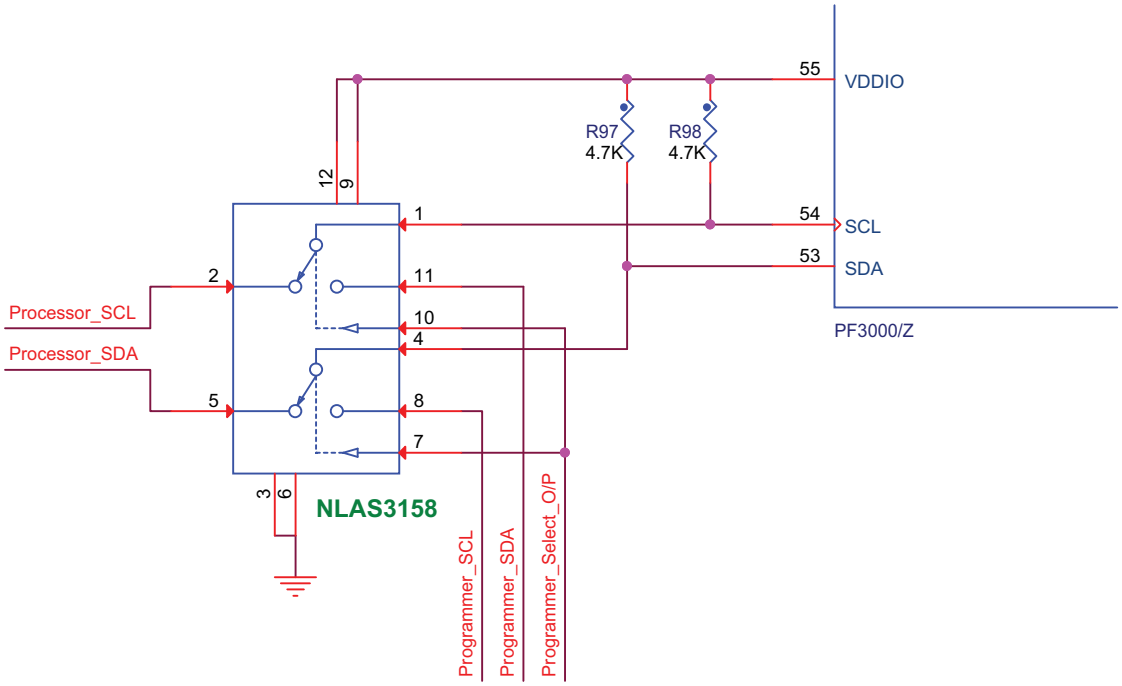

Figure 2. Isolating SCL and SDA using an analog switch

NOTE

Using the analog switch may not be the most cost effective way to isolate the I^2C bus. Similar functionality can be achieved by using solder shorts or 0 Ω resistors. However, minor rework of the board would be required once OTP programming is completed.

3.2 Programming using the KITPF3000FRDMGM

The NXP KITPF3000FRDMPGM board provides an ideal platform for programming the PF3000's OTP. The board integrates a 3.3 V LDO to power the PF3000 and a boost converter with a 9.5 V output voltage to generate the OTP programming voltage. An integrated USB-to-I^{2}C converter allows PC communication with the PF3000 using a NXP supplied GUI. See Figure 3 for a block diagram of the KITPF3000FRDMPGM.

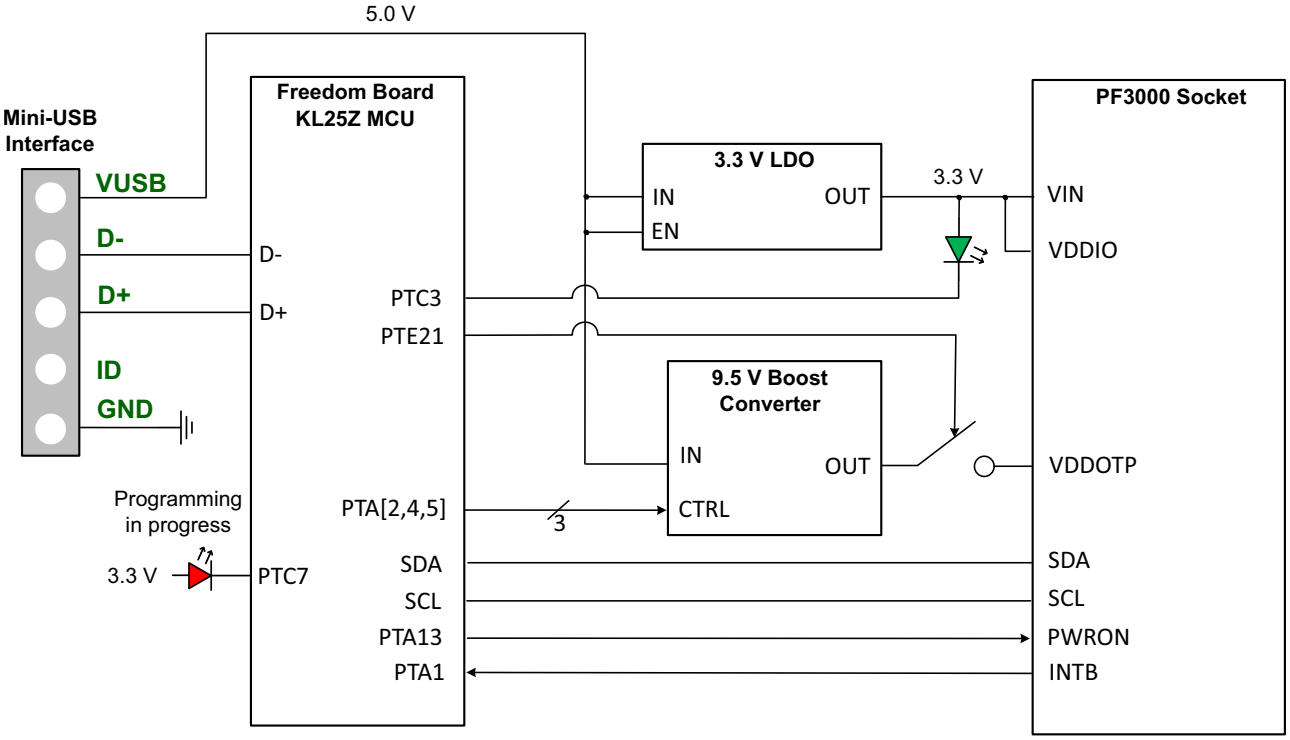

Figure 3. KITPF3000FRDMPGM Rev. D block diagram

3.3 Programming using a generic programmer

Figure 4 shows how to interface a generic programmer with the PF3000 in an application board. For applications that use a single rail for VIN and VDDIO, the connection is straightforward as shown in Figure 4. However, other loads connected to the 3.3 V rail must not surpass the current rating of the LDO. If that is the case, isolation in the form of an analog switch, a solder short, or a 0 Ω resistor is required.

The following requirements apply when a generic programmer board is used:

1. VIN power supply: 3.3 V, 100 mA
2. VDDIO power supply: 1.8 V to 3.3 V, 10 mA
3. I2C Master
4. GPO signal to control PF3000's PWRON pin
5. GPO signal to control analog switch (Programmer_Select_O/P)
6. .5 V, 100 mA power supply at VDDOTP bypassed by 2 x 10 µF capacitors. The voltage depends on the silicon revision used. See section OTP Programming Example for details.

Figure 4 illustrates a typical configuration using a generic programmer.

NOTE

Using the analog switch may not be the most cost effective way to isolate the I^{2}C bus. Similar functionality can be achieved by using solder shorts or 0 Ω resistors. However, minor rework of the board would be required once OTP programming is completed.

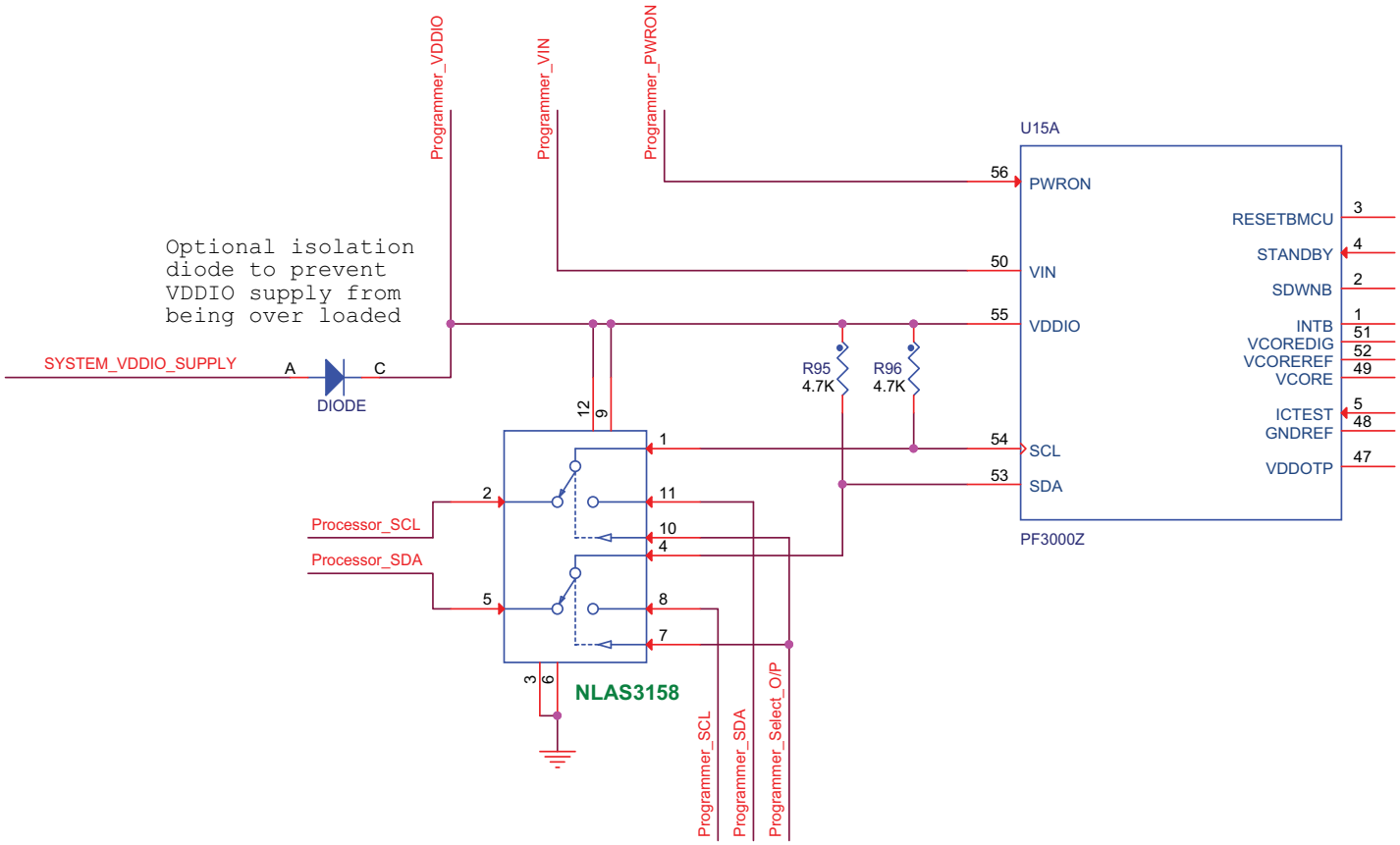

Figure 4. Interfacing a generic programmer to the PF3000 in an application board

4 OTP overview

The regulators in the PF3000 are configurable and are designed for flexibility in a wide variety of applications. One-Time-Programmable (OTP) fuses in the PF3000 demonstrate this flexibility. Key startup parameters and regulator configuration information can be programmed into the PF3000 to enable it to power the system. These parameters are:

- **General**
  - I^2C slave address
  - PWRON pin configuration
  - regulator start-up sequence and timing
  - RESETBMCU configuration
- **Buck regulators**
  - Output voltage
  - single phase or independent mode configuration
  - switching frequency
  - soft start ramp rate
- **Boost regulator and LDOs**
  - Output voltage

PF3000 starts up based on the contents of the TBBOTP registers. You can load the TBBOTP registers from different sources, as shown in Table 1. The default setting is hard-coded in the PF3000 and is available in all non-programmed and programmed PF3000 devices. Once you complete OTP programming, you can load TBBOTP either from the default values or from the OTP fuses.

The OTP block in the PF3000 also features a 'Try-Before-Buy' (TBB) mode which allows you to experiment with different voltages and sequences of the regulators. In the TBB mode, you can write to the TBBOTP registers directly and use them for startup of the PF3000. To maintain the contents of the TBBOTP registers in the absence of the main input supply (VIN), use a coin cell at the LICELL pin.

4.1 Power-up configuration

The PF3000 powers up based on the contents of the TBBOTP registers. Depending on certain pin and bit settings, the TBBOTP registers load from different sources as shown in Table 1.

Table 1. Start-up configuration source and conditions

Source	Condition	Power-up configuration
ROM	**VDDOTP = VCOREDIG(1)**	**PF3000 starts up using the factory default settings**
TBBOTP Registers	**VDDOTP = 0 V and TBB_POR = 1**	**PF3000 starts up from current values of TBBOTP registers. This is referred to as the 'Try-Before-Buy' mode**
OTP Fuses	**VDDOTP = 0 V and TBB_POR = 0**	**The PF3000 starts up from the OTP fuse values**

Notes
1. Pull-up VDDOTP to VCOREDIG with a 100 kΩ resistor

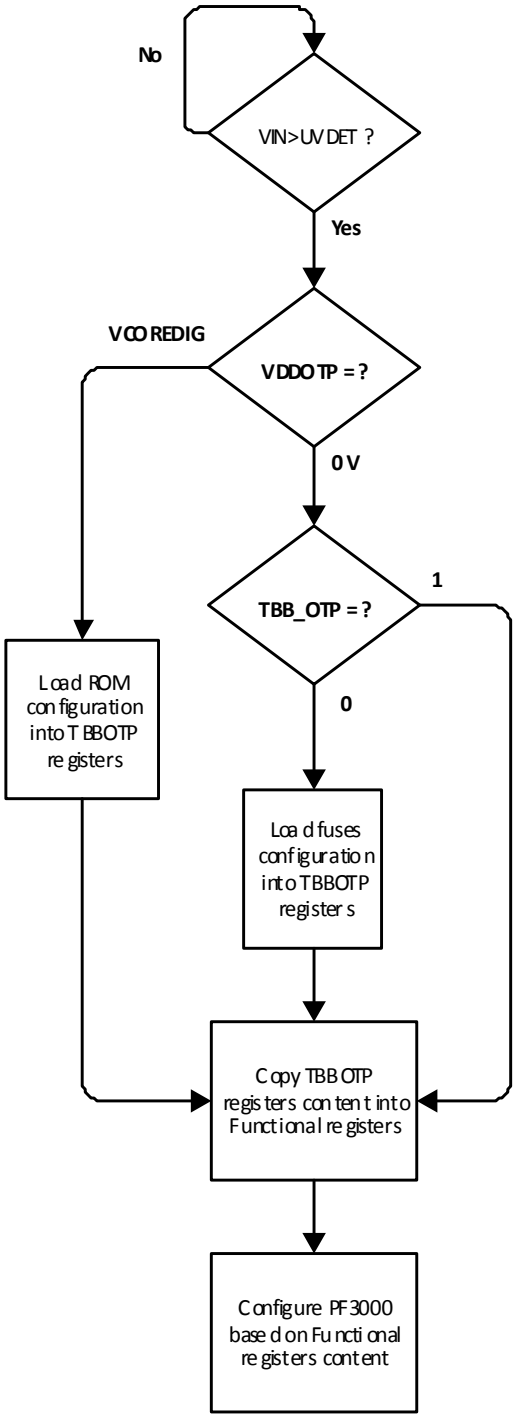

Figure 5. PF3000 power-up flow chart

The TBBOTP registers serve as temporary storage for any of the following:

- The values to be written to the fuses
- The values read from the fuses
- The values to start from during TBB development
- The values read from the default configuration.

The TBBOTP registers reside within the Extended Page 1 of the PF3000 register map.

During a power-up, the TBBOTP registers behave as follows:

- The contents of the TBBOTP registers initialize to zero when a valid VIN is first applied.
- The values that then load into the TBBOTP registers depend on the setting of the VDDOTP pin, and on the value of the TBB_POR.
  - If VDDOTP = VCOREDIG (1.5 V), the TBBOTP values are loaded from ROM.
  - If VDDOTP = 0 V and TBB_POR = 1, the PF3000 starts up from current values of TBBOTP registers.
  - If VDDOTP = 0 V and TBB_POR = 0, the TBBOTP values are loaded from the fuses.

Notice that the initial value of TBB_POR is always "0".

The contents of the TBBOTP registers may be modified by I^2C. To communicate with I^2C, VIN must be valid and VDDIO (to which SDA and SCL are pulled up) must be powered by a 1.7 V to 3.6 V supply. VIN or the coin cell voltage must be valid to maintain the contents of the TBBOTP registers. To power on with the contents of the TBBOTP registers, a valid turn-on event must occur with the following conditions:

- a valid VIN
- optional LICELL
- VDDOTP = 0 V
- TBB_POR = 1

4.2 OTP programming example

The One-Time-Programmable memory is control by fuses. The startup configuration programmed into the PF3000 depends on the state of these fuses as defined during the OTP programming process.

There are 5 banks of fuses. Each bank consists of 26 fuses. Of the 26 fuses in a bank, 20 are programmable by the user. The remaining 6 are redundant fuses that allow implementation of Error Correction. An Error Correction Code within the PF3000 corrects single bit errors if they occur in the bank.

The programming voltage should have a tolerance of +/-3% and OTP programming should be done at room temperature. For reliability reasons, do not OTP program a given part more than once.

NOTE

All code examples in this document represent a script using the KITPF3000FRDMPGM and the associated GUI. Command syntax may vary if the user utilizes a different tool for communication.

The following is an example of programming the PF3000.

```
//--------------------------------------------------------------------------
// A1 - Sample Configuration (i.MX 7 with DDR3L)
// Set VDDOTP = 0 V, PWRON = HIGH, LICELL = 3.0 V (Optional), VIN = 3.3 V, VDDIO = 3.3 V
//--------------------------------------------------------------------------
//==========================================================================
// Commands with Green background come from customer based on their requirements
//==========================================================================
WRITE_I2C:7F:01 // Access PF3000 EXT Page1 (OTP Registers)
//[Extended Page 1 Registers: 0xA0 - 0xAF] ---------------------------------
WRITE_I2C:A0:10 // SW1A OTP Vout = 1.100V
WRITE_I2C:A1:01 // SW1A OTP Sequence = 1
WRITE_I2C:A2:0D // SW1A OTP SwConfig = A|B INDEPENDENT; SW1A OTP Frequency = 2.0MHz
WRITE_I2C:A8:0C // SW1B OTP Vout = 1.000V
WRITE_I2C:A9:01 // SW1B OTP Sequence = 1
WRITE_I2C:AC:06 // SW2 OTP Vout = 1.800V
WRITE_I2C:AD:02 // SW2 OTP Sequence = 2
WRITE_I2C:AE:01 // SW2 OTP Frequency = 2.0MHz
//[Extended Page 1 Registers: 0xB0 - 0xBF] ---------------------------------
WRITE_I2C:B0:09 // SW3 OTP Vout = 1.350V
WRITE_I2C:B1:05 // SW3 OTP Sequence = 5
WRITE_I2C:B2:01 // SW3 OTP Frequency = 2.0MHz
WRITE_I2C:BC:00 // SWBST OTP Vout = 5.000 Volts
WRITE_I2C:BD:00 // SWBST OTP Sequence = 0
//[Extended Page 1 Registers: 0xC0 - 0xCF] ---------------------------------
WRITE_I2C:C0:06 // VSNVS OTP Vout = 3.000 Volts
WRITE_I2C:C8:00 // VLDO1 OTP Vout = 1.800 Volts
WRITE_I2C:C9:04 // VLDO1 OTP Sequence = 4
WRITE_I2C:CC:0E // VLDO2 OTP Vout = 1.500 Volts
WRITE_I2C:CD:04 // VLDO2 OTP Sequence = 4
//[Extended Page 1 Registers: 0xD0 - 0xDF] ---------------------------------
WRITE_I2C:D0:03 // VCC_SD OTP Vout = 3.300 Volts
WRITE_I2C:D1:04 // VCC_SD OTP Sequence = 4
WRITE_I2C:D4:03 // V33 OTP Vout = 3.300 Volts
WRITE_I2C:D5:03 // V33 OTP Sequence = 3
WRITE_I2C:D8:0F // VLDO3 OTP Vout = 3.300 Volts
WRITE_I2C:D9:03 // VLDO3 OTP Sequence = 3
WRITE_I2C:DC:0F // VLDO4 OTP Vout = 3.300 Volts
WRITE_I2C:DD:03 // VLDO4 OTP Sequence = 3
```

```
//[Extended Page 1 Registers: 0xE0 - 0xEF] ---------------------------------

WRITE_I2C:E0:00 // PWRON_CFG=0 (PWRON is level sensitive); OTP_SWDVS_CLK=0 (25 mV step each 2.0 us);
OTP_SEQ_CLK_SPEED=0 (Clock speed 500 us increments)

WRITE_I2C:E4:00 // TBB_POR=0 (Try Before Buy disabled)

WRITE_I2C:E8:00 // OTP_PG_EN=0 (RESETBMCU default mode. After start-up, RESETBMCU will be released
2.0 ms after the last regulator in enabled)

//[Extended Page 1 Registers: 0xF0 - 0xFF] ---------------------------------

WRITE_I2C:FF:08 // I2C_SLV_ADDR[3]=1 (hard coded); I2C_SLV_ADDR[2:0] = 000 (I2C Device Address is
0x08)

//=============================================================================

// Commands with Purple background are for actual OTP programming

//=============================================================================

// PROGRAMMING COMMANDS FOLLOW

//=============================================================================================

//---------------------------------------------------------------------------

WRITE_I2C:7F:02 // Access PF3000 EXT Page2

WRITE_I2C:D0:FF

WRITE_I2C:D8:46

//---------------------------------------------------------------------------

VPGM:ON // Turn ON 9.5V Supply at VDDOTP

DELAY:50 // Adds 50msec delay to allow VPGM time to ramp up

//---------------------------------------------------------------------------

WRITE_I2C:D9:FF // Command to blow fuses

//---------------------------------------------------------------------------

DELAY:750 // Adds 750msec delay to allow programming to complete

VPGM:OFF // Turn off 9.5V Boost Supply

DELAY:500 // Adds delay to allow VPGM to bleed off

PWRON:LOW // PWRON LOW to reload new OTP data

DELAY:500

PWRON:HIGH
```

After completing OTP programming by following the above steps, read the registers 0xA0 to 0xE8 in Extended Page 1 and compare them to the required register values as in the script. Additionally, read the ECC Interrupt bit, OTP_ECCI in register 0x0E. If there is an error in the programmed values or if the OTP_ECCI bit is set to 1, reject the part because the programming process resulted in errors.

4.3 Try-before-buy mode example

As shown in Table 1, start the PF3000 directly from the TBBOTP registers without actually programming the part. The following example illustrates illustrates a typical implementation of the Try-Before-Buy mode.

NOTE

All code examples in this document represent a script using the KITPF3000FRDMPGM and the associated GUI. Command syntax may vary if the user utilizes a different tool for communication.

```
//=============================================================================
// Commands with Green background come from customer based on their requirements
//=============================================================================
WRITE_I2C:7F:01 // Access PF3000 EXT Page1 (OTP Registers)
//[Extended Page 1 Registers: 0xA0 - 0xAF] ---------------------------------
WRITE_I2C:A0:10 // SW1A OTP Vout = 1.100V
WRITE_I2C:A1:01 // SW1A OTP Sequence = 1
WRITE_I2C:A2:0D // SW1A OTP SwConfig = A|B INDEPENDENT; SW1A OTP Frequency = 2.0MHz
WRITE_I2C:A8:0C // SW1B OTP Vout = 1.000V
WRITE_I2C:A9:01 // SW1B OTP Sequence = 1
WRITE_I2C:AC:06 // SW2 OTP Vout = 1.800V
WRITE_I2C:AD:02 // SW2 OTP Sequence = 2
WRITE_I2C:AE:01 // SW2 OTP Frequency = 2.0MHz
//[Extended Page 1 Registers: 0xB0 - 0xBF] ---------------------------------
WRITE_I2C:B0:09 // SW3 OTP Vout = 1.350V
WRITE_I2C:B1:05 // SW3 OTP Sequence = 5
WRITE_I2C:B2:01 // SW3 OTP Frequency = 2.0MHz
WRITE_I2C:BC:00 // SWBST OTP Vout = 5.000 Volts
WRITE_I2C:BD:00 // SWBST OTP Sequence = 0
//[Extended Page 1 Registers: 0xC0 - 0xCF] ---------------------------------
WRITE_I2C:C0:06 // VSNVS OTP Vout = 3.000 Volts
WRITE_I2C:C8:00 // VLDO1 OTP Vout = 1.800 Volts
WRITE_I2C:C9:04 // VLDO1 OTP Sequence = 4
WRITE_I2C:CC:0E // VLDO2 OTP Vout = 1.500 Volts
WRITE_I2C:CD:04 // VLDO2 OTP Sequence = 4
//[Extended Page 1 Registers: 0xD0 - 0xDF] ---------------------------------
WRITE_I2C:D0:03 // VCC_SD OTP Vout = 3.300 Volts
WRITE_I2C:D1:04 // VCC_SD OTP Sequence = 4
WRITE_I2C:D4:03 // V33 OTP Vout = 3.300 Volts
WRITE_I2C:D5:03 // V33 OTP Sequence = 3
WRITE_I2C:D8:0F // VLDO3 OTP Vout = 3.300 Volts
WRITE_I2C:D9:03 // VLDO3 OTP Sequence = 3
WRITE_I2C:DC:0F // VLDO4 OTP Vout = 3.300 Volts
WRITE_I2C:DD:03 // VLDO4 OTP Sequence = 3
```

```
//[Extended Page 1 Registers: 0xE0 - 0xEF] ---------------------------------
WRITE_I2C:E0:00 // PWRON_CFG=0 (PWRON is level sensitive); OTP_SWDVS_CLK=0 (25 mV step each 2.0 us);
OTP_SEQ_CLK_SPEED=0 (Clock speed 500 us increments)
WRITE_I2C:E4:00 // TBB_POR=0 (Try Before Buy disabled)
WRITE_I2C:E8:00 // OTP_PG_EN=0 (RESETBMCU default mode. After start-up, RESETBMCU will be released
2.0 ms after the last regulator in enabled)
//[Extended Page 1 Registers: 0xF0 - 0xFF] ---------------------------------
WRITE_I2C:FF:08 // I2C_SLV_ADDR[3]=1 (hard coded); I2C_SLV_ADDR[2:0] = 000 (I2C Device Address is
0x08)

//=============================================================================
// Commands with Blue background are for entering the TBB mode.
//=============================================================================
// TRY-BEFORE-BUY COMMANDS FOLLOW
//=============================================================================
//-----------------------------------------------------------------------------
WRITE_I2C:E4:80 // TBB POR=1 (This Enables TBB Mode)
//-----------------------------------------------------------------------------
PWRON:LOW // PWRON LOW
DELAY:500
PWRON:HIGH // PWRON HIGH to Start PMIC from desired TBB Configuration
```

4.4 OTP register descriptions

The PF3000 OTP registers consist of 130 fuses arranged in five banks. Each bank contains 26 fuses. Each fuse represents one bit of the TBBOTP register map. Table 2 to Table 6 show the banks, their fuses and the corresponding bits in the register map.

Table 2. Bank 1

Fuses	OTP Register Name	Register bits	Description
4:0	SW1A VOLT	OTP_SW1A_VOLT[4:0]	SW1A Power-up voltage
7:5	SW1A SEQ	OTP_SW1A_SEQ[2:0]	SW1A Power-up sequence
12:8	SW1B VOLT	OTP_SW1B_VOLT[4:0]	SW1B Power-up voltage
15:13	SW1B SEQ	OTP_SW1B_SEQ[2:0]	SW1B Power-up sequence
17:16	SW1 FREQ	OTP_SW1_FREQ[1:0]	SW1 Frequency
19:18	SW1 CONFIG	OTP_SW1_CONFIG[1:0]	SW1 Single Phase or Independent mode
25:20	—	—	ECC check bits for fuse bank 1

Table 3. Bank 2

Fuses	OTP register name	Register bits	Description
3:0	SW2 VOLT	OTP_SW2_VOLT[3:0]	SW2 Power-up voltage
6:4	SW2 SEQ	OTP_SW2_SEQ[2:0]	SW2 Power-up sequence
10:7	SW3 VOLT	OTP_SW3_VOLT[4:0]	SW3 Power-up voltage
13:11	SW3 SEQ	OTP_SW3_SEQ[2:0]	SW3 Power-up sequence
15:14	SW2 FREQ	OTP_SW2_FREQ[1:0]	SW2 Frequency
17:16	SW3 FREQ	OTP_SW3_ FREQ [1:0]	SW3 Frequency
19:18	Reserved	Reserved	—
25:20	—	—	ECC check bits for fuse bank 2

Table 4. Bank 3

Fuses	OTP register name	Register bits	Description
3:0	VLDO1 VOLT	OTP_VLDO1_VOLT[3:0]	VLDO1 Power-up voltage
7:4	VLDO1 SEQ	OTP_VLDO1_SEQ[3:0]	VLDO1 Power-up sequence
11:8	VLDO2 VOLT	OTP_VLDO2_VOLT[3:0]	VLDO2 Power-up voltage
15:12	VLDO2 SEQ	OTP_VLDO2_SEQ[3:0]	VLDO2 Power-up sequence
18:16	VSNVS VOLT	OTP_VSNVS_VOLT[2:0]	VSNVS Power-up voltage
19	Reserved	Reserved	—
25:20	—	—	ECC check bits for fuse bank 3

Table 5. Bank 4

Fuses	OTP register name	Register bits	Description
3:0	VLDO3 VOLT	OTP_VLDO3_VOLT[3:0]	VLDO3 Power-up voltage
7:4	VLDO3 SEQ	OTP_VLDO3_SEQ[3:0]	VLDO3 Power-up sequence
11:8	VLDO4 VOLT	OTP_VLDO4_VOLT[3:0]	VLDO4 Power-up voltage
15:12	VLDO4 SEQ	OTP_VLDO4_SEQ[3:0]	VLDO4 Power-up sequence
18:16	I2C SLAVE ADDR	OTP_I2C_SLAVE_ADDR[2:0]	PF3000 I2C Address
19	OTP PWRGD EN	OTP_PWRGD_EN[0]	Fault mode enable
25:20	—	—	ECC check bits for fuse bank 4

Table 6. Bank 5

Fuses	OTP register name	Register bits	Description
1:0	SWBST VOLT	OTP_SWBST_VOLT[1:0]	SWBST Power-up voltage
4:2	SWBST SEQ	OTP_SWBST_SEQ[2:0]	SWBST Power-up sequence
6:5	V33 VOLT	OTP_V33_VOLT[3:0]	V33 Power-up voltage
9:7	V33 SEQ	OTP_V33_SEQ[2:0]	V33 Power-up sequence
11:10	VCC_SD VOLT	OTP_VCC_SD_VOLT[1:0]	VCC_SD Power-up voltage

Table 6. Bank 5 (continued)

Fuses	OTP register name	Register bits	Description
14:12	VCC SD SEQ	OTP_VCC_SD_SEQ[2:0]	VCC_SD Power-up sequence
15	SEQ CLK FREQ	OTP_SEQ_CLK_FREQ[0]	SEQ Clock Frequency selection
16	SWDVS CLK	OTP_SWDVS_CLK[0]	DVS Clock selection
17	PWRON CFG	OTP_PRWON_CFG[0]	PWRON Level/Edge Configuration
18	Reserved	Reserved	—
19	OTP BLOWN	OTP_BLOWN[0]	OTP Fuses blown status
25:20	—	—	ECC check bits for fuse bank 2

The TBBOTP registers store data for programming the fuses. These registers are written to and read from using the I^2C interface.

Once the TBBOTP registers are loaded with the correct values, the fuses can then be programmed. Before discussing the programming process, some salient features of the OTP function are described.

4.4.1 TBBOTP registers description

The TBBOTP registers for configuring the switching regulators are listed in Table 12 and Table 13 to Table 18 provide a general description of the TBBOTP registers for all the switching regulators.

Table 7. OTP switching regulators register summary

Register	Address	Output
OTP SW1A VOLT	0xA0	SW1A OTP Output voltage set point
OTP SW1A SEQ	0xA1	SW1A OTP power-up sequence selection
OTP SW1x CONFIG	0xA2	SW1A and SW1B OTP operation mode and frequency selection
OTP SW1B VOLT	0xA8	SW1B OTP Output voltage set point
OTP SW1B SEQ	0xA9	SW1B OTP power-up sequence selection
OTP SW2 VOLT	0xAC	SW2 OTP Output voltage set point
OTP SW2 SEQ	0xAD	SW2 OTP power-up sequence selection
OTP SW2 FREQ	0xAE	SW2 OTP frequency selection
OTP SW3 VOLT	0xB0	SW3 OTP Output voltage set point
OTP SW3A SEQ	0xB1	SW3 OTP power-up sequence selection
OTP SWBST VOLT	0xBC	SWBST OTP output voltage set point
OTP SWBST SEQ	0xBD	SWBST OTP power-up sequence selection

Table 8. OTP SW1A VOLT register description

Name	Bit #	Description
OTP_SW1A_VOLT	4:0	Sets the SW1A output voltage to be programmed on the OTP fuses and loaded during power-up. Refer to SW1A output voltage configuration table on Data Sheet for all possible configurations.
UNUSED	7:5	UNUSED

Table 9. OTP SW1A SEQ register description

Name	Bit #	Description
OTP_SW1A_SEQ	2:0	Assigns the power-up sequence slot 0-7 for SW1A
UNUSED	7:3	UNUSED

Table 10. OTP SW1x CONFIG register description

Name	Bit #	Description
OTP_SW1x_FREQ	1:0	SW1x OTP Frequency configuration 00 = 1.0 MHz 01 = 2.0 MHz 10 = 4.0 MHz 11 = Reserved
OTP_SW1x_CONFIG	3:2	SW1A/B 00 = Unused 01 = SW1A and SW1B in Single Phase mode 10 = Unused 11 = SW1A and SW1B in Independent mode
UNUSED	7:4	UNUSED

Table 11. OTP SW2 VOLT register description

Name	Bit #	Description
OTP_SW2_VOLT	2:0	Sets the SWx output voltage to be programmed on the OTP fuses and loaded during power-up. Refer to the respective SWx output voltage configuration table on datasheet for all possible configurations.
OTP_SW2_HI	3	0: low output voltage range selected (1.50 V to 1.85 V) 1: high output voltage range selected (2.50 V to 3.3 V)
UNUSED	7:4	UNUSED

Table 12. OTP SW2 SEQ register description

Name	Bit #	Description
OTP_SW2_SEQ	2:0	Assigns the power-up sequence slot 07 for SW2
UNUSED	7:3	UNUSED

Table 13. OTP SW2 CONFIG register description

Name	Bit #	Description
OTP_SW2_FREQ	1:0	SW2 OTP Frequency configuration 00 = 1.0 MHz 01 = 2.0 MHz 10 = 4.0 MHz 11 = Reserved
UNUSED	7:2	UNUSED

Table 14. OTP SW3 VOLT register description

Name	Bit #	Description
OTP_SW3_VOLT	3:0	Sets the SW3 output voltage to be programmed on the OTP fuses and loaded during power-up. Refer to SW3 output voltage configuration table on Data Sheet for all possible configurations.
UNUSED	7:4	UNUSED

Table 15. OTP SW3 SEQ register description

Name	Bit #	Description
OTP_SW3_SEQ	2:0	Assigns the power-up sequence slot 0-7 for SW3
UNUSED	7:3	UNUSED

Table 16. OTP SW3 CONFIG register description

Name	Bit #	Description
OTP_SW3_FREQ	1:0	SW3 OTP Frequency configuration 00 = 1.0 MHz 01 = 2.0 MHz 10 = 4.0 MHz 11 = Reserved
UNUSED	7:2	UNUSED

Table 17. OTP SWBST VOLT register description

Name	Bit #	Description
OTP_SWBST_VOLT	1:0	SWBST OTP output voltage setpoint 00 = 5.00 V 01 = 5.05 V 10 = 5.10 V 11 = 5.15 V
UNUSED	7:2	UNUSED

Table 18. OTP SWBST SEQ register description

Name	Bit #	Description
SWBST_SEQ	4:0	Assign the power-up sequence slot 0-31 for SWBST
UNUSED	7:5	UNUSED

Table 19 shows a summary of all the registers related to the linear regulators, and Table 20 to Table 22 provide a general bit description of the linear regulator OTP registers.

Table 19. OTP linear regulators register summary

Register	Address	Output
OTP VSNVS VOLT	0xC0	VSNVS OTP Output voltage set point
OTP VLDO1 VOLT	0xC8	VLDO1 OTP Output voltage set point
OTP VLDO1 SEQ	0xC9	VLDO1 OTP power-up sequence selection
OTP VLDO2 VOLT	0xCC	VLDO2 OTP Output voltage set point
OTP VLDO2 SEQ	0xCD	VLDO2 OTP power-up sequence selection
OTP VCC_SD VOLT	0xD0	VCC_SD OTP Output voltage set point

Table 19. OTP linear regulators register summary (continued)

Register	Address	Output
OTP VCC_SD SEQ	0xD1	VCC_SD OTP power-up sequence selection
OTP V33 VOLT	0xD4	V33 OTP Output voltage set point
OTP V33 SEQ	0xD5	V33 OTP power-up sequence selection
OTP VLDO3 VOLT	0xD8	VLDO3 OTP Output voltage set point
OTP VLDO3 SEQ	0xD9	VLDO3 OTP power-up sequence selection
OTP VLDO4 VOLT	0xDC	VLDO4 OTP Output voltage set point
OTP VLDO4 SEQ	0xDD	VLDO4 OTP power-up sequence selection

Table 20. OTP VSNVS VOLT register description

Name	Bit #	Description
OTP_VSNVS_VOLT	2:0	Sets the VSNVS output voltage to be programmed on the OTP fuses and loaded during power-up 000 = RSVD 001 = RSVD 010 = RSVD 011 = RSVD 100 = RSVD 101 = RSVD 110 = 3.0 V 111 = RSVD
UNUSED	7:3	UNUSED

Table 21. OTP VLDOx VOLT register description (VLDO1, VLDO2, VLDO3 and VLDO4)

Name	Bit #	Description
OTP_VLDOx_VOLT	3:0	Sets the VLDOx output voltage to be programmed on the OTP fuses and loaded during power-up. Refer to the VLDOx output voltage configuration table on Data Sheet for all possible configurations.
UNUSED	7:4	UNUSED

Table 22. OTP VLDOx SEQ register description

Name	Bit #	Description
OTP_VLDOx_SEQ	3:0	Assign the power-up sequence slot 0-31 for the specific linear regulator
UNUSED	7:4	UNUSED

Table 23. OTP VCC_SD VOLT register description

Name	Bit #	Description
OTP_VCC_SD_VOLT	1:0	Sets VCC_SD output voltage to be programmed on the OTP fuses and loaded during power-up. Refer to the VCC_SD output voltage configuration table on Data Sheet for all possible configurations.
UNUSED	7:2	UNUSED

Table 24. OTP VCC_SD SEQ register description

Name	Bit #	Description
OTP_VCC_SD_SEQ	3:0	Assign the power-up sequence slot 0-31 for the specific linear regulator
UNUSED	7:4	UNUSED

Table 25. OTP V33 VOLT register description

Name	Bit #	Description
OTP_V33_VOLT	1:0	Sets V33output voltage to be programmed on the OTP fuses and loaded during power-up. Refer to the V33 output voltage configuration table on Data Sheet for all possible configurations.
UNUSED	7:2	UNUSED

Table 26. OTP V33 SEQ register description

Name	Bit #	Description
OTP_V33_SEQ	3:0	Assign the power-up sequence slot 0-31 for the specific linear regulator
UNUSED	7:4	UNUSED

4.4.2 Other OTP bits

Table 27. OTP PU CONFIG1 bits definition

Bit	Name	Description
0	OTP_SEQ_CLK_SPEED	Sequence delay between steps 0 = 500 µs 1 = 2000 µs
1	RSVD	Reserved
2	OTP_SWDVS_CLK	Start-up slew rate 0 = 25 mV step each 2.0 µs 1 = 25 mV step each 4.0 µs
3	RSVD	Reserved
4	OTP_PWRON_CFG	Set the power on button initial configuration 0 = Power button is level sensitive 1 = Power button is edge sensitive and turn-off is based on time held low
7:5	RSVD	Reserved

Table 28. OTP_FUSE_POR1 bits definition

Bit	Name	Description
6:0	RSVD	Reserved
7	TBB_POR	Prototyping enable bit 0 = Prototyping disabled 1 = Prototyping enabled

Table 29. OTP PWRGD EN bits definition

Bit	Name	Description
0	OTP_PG_EN	Fault mode enable bit 0 = Fault mode disabled 1 = Fault mode enabled
7:1	RSVD	Reserved

Table 30. OTP BLOWN bits definition

Bit	Name	Description
0	OTP_BLOWN	OTP fuses blown status 0 = OTP fuses not blown 1 = OTP fuses blown
7:1	RSVD	Reserved

4.4.3 TBBOTP register reloading without turn-on event

After the fuses are programmed, their values may be loaded into the digital control logic without toggling VIN or PWRON. To update the TBBOTP registers by reloading the fuse values automatically, set bits in the OTP LOAD MASK register depending on the functionality required. Refer to Table 31 for a description of the OTP LOAD MASK register.

Table 31. OTP load mask register

Extended Page 1		I^2C Data Bits							
Addr	**Reg Name**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
84	OTP LOAD MASK	START	RL PWRTN	FORCE PWRCTL	RL PWRCTL	RL OTP	RL OTP ECC	RL OTP FUSE	RSVD
		0	0	0	0	0	0	0	0

Table 32. OTP reload mask register bit description

Bit	Name	Description
0	RSVD	Reserved
1	RL_OTP_FUSE	Reload the OTP fuse latch from the analog fuse bit 0 = Disable loading 1 = Enable loading
2	RL_OTP_ECC	Reload the OTP ECC registers. Set this bit irrespective of whether ECC is enabled or disabled. 0 = Disable loading 1 = Enable loading
3	RL_OTP	Reload the TBBOTP registers from the fuses 0 = Disable loading 1 = Enable loading of fuses if ECC is disabled. Enable loading of ECC corrected fuses if ECC is enabled.
4	RL_PWRCTL	Reload the power control registers from the TBBOTP registers 0 = Disable loading 1 = Enable loading
5	FORCE_PWRCTL	Forces the power control registers to be reloaded if they are being used to control the regulators 0 = No reload forced 1 = Power control register value affects regulators when the reload sequence is enabled and RL PWRCTL bit is enabled.This is needed when changing output voltage of switching regulators from low-voltage range to high-voltage range
6	RL_PWRTN	Reloads the register that controls how the PWRON button works 0 = PWRON configuration setting does not change until a shutdown and restart event 1 = PWRON behavior switch to new OTP PWRON button configuration when START bit is enabled
7	START	Reload sequence start bit 0 = Reload sequence disabled 1 = Starts the reload sequence, when the sequence is done all of the OTP_LOAD_MASK bits are reset

Often only bits 1, 2, and 3 need to be set, as well as the START bit, to reload the TBBOTP registers after the fuses are programmed. Then, check the TBBOTP register values to make sure that the correct values have been loaded from the fuses. Setting bits 4 and 5, updates the regulator parameters immediately. This should be done with caution if PWRON is already asserted. A PWRON event triggers a complete reload using the same logic. When a '1' is written to Bit 7 of the OTP_LOAD_MASK registers, the PF3000 turns off momentarily and then turns back on to reload the fuses. To reload the fuses without first turning off the PF3000, clear Bit 0 of the PWRCTRL_OTP_CTRL register prior to writing to the OTP_LOAD_MASK register. Note that the OTP_LOAD_MASK is register 0x84 in Extended Page 1 whereas the PWRCTRL_OTP_CTRL is register 0x88 in Extended Page 2.

4.4.4 Direct OTP fuse read

Setting the OTP_FUSE_READ_EN bit to HIGH allows you to read uncorrected fuse values. If ECC is not enabled, or there is no programming error, the values loaded into the TBBOTP registers are identical to the fuse values. If ECC is enabled and a single-bit error occurs during programming, the fuse values may be different from the values loaded into the TBBOTP registers. The values loaded into the TBBOTP registers are the error-corrected values. Table 33 shows the OTP FUSE READ EN register

.

Table 33. OTP fuse read enable register

Extended page 1		I^2C data bits							
Addr	**Register Name**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
80	OTP FUSE READ EN	—	—	—	—	—	—	—	OTP_FUSE_READ_EN
		0	0	0	0	0	0	0	0

4.5 Fuse programming and error correction code (ECC)

4.5.1 OTP fuse control register

Section 4.2 "OTP programming example" provides an example of a typical OTP programming script. You must write to the OTP_FUSE_CTLx registers, located in the Extended Page 2, in order to program fuses. There are ten such registers, one for each bank, Refer to Table 34 and Table 35 for a description of the registers.

.

Table 34. General OTP fuse control register bits

Extended page 2	I^2C data bits							
Reg Name	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
OTP_FUSE_CTLx	—	—	—	—	ANTIFUSEx_EN	ANTIFUSEx_ LOAD	ANTIFUSEx_ RW	BYPASSx

Table 35. OTP fuse control bits description

Bit	Name	Description
0	BYPASSx	Multiplexor that selects between the value stored in the digital fuse latch and the value on the TBBOTP register 0 = Select from digital latch 1 = Select from TBBOTP register
1	ANTIFUSEx_RW	Allows programming the fuse bank when VDDOTP is 9.5 V 0 = Disable program fuse 1 = Enable program fuse
2	ANTIFUSEx_LOAD	Clock input to the digital latch that stores the state of the analog fuse cell, it is active high and is pulsed while the ANTIFUSE_EN bit is high to load the value of the analog fuse state into the digital latch.
3	ANTIFUSEx_EN	Turns on the bias to the analog fuse cell so that it can be written to or read from 0 = Analog bias disabled 1 = Analog bias enabled
4-7	Not used	Not used

4.5.2 Error correction code (ECC)

Error correction is set to OFF by default. However, NXP recommends that it be enabled (set to ON) for all programming operations. When error correction is enabled, a single-bit error per fuse bank is both reported and corrected. A double-bit error per fuse bank is reported but not corrected. Fuses may be programmed without using ECC. However, after verifying that the part is configured properly, ECC may enabled and the error check bits programmed.

Double bit errors can prevent regulators from powering up, or can result in a configuration that does not match the external components. Although such occurrences are rare, it is still a good practice to employ ECC to at least alert the user when this occurs.

NOTE

The desired function of the redundant bits must be determined when ECC is configured and its bits programmed, or the ECC logic attempts to correct the newly programmed redundant bits.

Section 4.5.2.1 through Section 4.5.2.3 are for advanced users. For a simple script that enables ECC, see Section 4.2 "OTP programming example".

4.5.2.1 ECC interrupt

With ECC enabled, if a single fuse in a bank has the wrong value, the ECC logic corrects that bit and loads the corrected value into the TBBOTP register for that bank. The single-error bit for that bank is set and also the main interrupt ECC bit is set. If two or more bits are in error, in a bank, the ECC is not able to correct them. The double-error bit error for that bank is set and the ECC interrupt bit is set. The single-error and double-error bits may be read from registers 0x8A to 0x8D in the Extended Page1 of the register map. The ECC interrupt bit may be read from register, 0xE, on the functional page of the register map.

Table 36. ECC error detection registers

Extended Page 1		I^2C data bits							
Addr	**Reg Name**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
8A	OTP ECC SE1	—	—	—	ECC5_SE	ECC4_SE	ECC3_SE	ECC2_SE	ECC1_SE
		x	x	x	0	0	0	0	0
8B	OTP ECC SE2	—	—	—	—	—	—	—	—
		x	x	x	x	x	x	x	x
8C	OTP ECC DE1	—	—	—	ECC5_DE	ECC4_DE	ECC3_DE	ECC2_DE	ECC1_DE
		x	x	x	0	0	0	0	0
8D	OTP ECC DE2	—	—	—	—	—	—	—	—
		x	x	x	x	x	x	x	x

Table 37. OTP ECC SE1 and 2 register description

Bit	Name	Default	Description
OTP ECC SE1			
0	ECC1_SE	0	Single error detection in fuse bank 1 0 = No single error detected 1 = Single error detected
1	ECC2_SE	0	Single error detection in fuse bank 2 0 = No single error detected 1 = Single error detected
2	ECC3_SE	0	Single error detection in fuse bank 3 0 = No single error detected 1 = Single error detected
3	ECC4_SE	0	Single error detection in fuse bank 4 0 = No single error detected 1 = Single error detected
4	ECC5_SE	0	Single error detection in fuse bank 5 0 = No single error detected 1 = Single error detected
7:5	RSVD	0	Reserved

Table 38. OTP ECC DE1 and 2 register description

Bit	Name	Default	Description
OTP ECC DE1			
0	ECC1_DE	0	Dual error detection in fuse bank 1 0 = No single error detected 1 = Single error detected
1	ECC2_DE	0	Dual error detection in fuse bank 2 0 = No single error detected 1 = Single error detected
2	ECC3_DE	0	Dual error detection in fuse bank 3 0 = No single error detected 1 = Single error detected
3	ECC4_DE	0	Dual error detection in fuse bank 4 0 = No single error detected 1 = Single error detected
4	ECC5_DE	0	Dual error detection in fuse bank 5 0 = No single error detected 1 = Single error detected
7:5	RSVD	0	Reserved

All interrupts are masked by default. Therefore, after programming the fuses, you should unmask the ECC interrupt with ECC enabled to determine if single- or double-bit errors exist in any of the banks. To read the error bits, see Table 36 for their location in the registers.

4.5.2.2 Analyzing a Single Bit ECC Error

When a single bit error occurs, the ECC check bits indicate which fuse in a given bank is in error. If you require ECC error information, read the check bits for each bank from bits[5:0], in registers 0xE1 to 0xEA, in the Extended Page 2. (See Table 39.) For example, if there is an error in bit[5] of fuse bank 3, reading bits[5:0] of register 0xE3 yields a hexadecimal code of 0x15. Refer to Table 42 and Table 43 for a description of the error control registers

Table 39. ECC error location coding

Bit in error	ECC check bit code
0	07
1	0B
2	0D
3	0E
4	13
5	15
6	16
7	19
8	1A
9	1C
10	23
11	25
12	26
13	29
14	2A
15	2C
16	31
17	32
18	34
19	38
20	01
21	02
22	04
23	08
24	10
25	20

4.5.2.3 Fuse Programming with ECC

To program fuses with ECC, enable bits in the following registers:

- OTP EN ECC0 and OTP EN ECC1 in the Extended Page 1
- OTP AUTO ECC0 and OTP AUTO ECC1 in the Extended Page 2.

The ECC enable registers are shown in Table 40. To enable error correction for any bank, set the appropriate bit. Bits in the OTP EN ECCx registers are programmed (not just set) in software.

Table 41 shows the OTP AUTO ECC registers. After programming the fuses, load their values into the TBBOTP registers. Load the error-corrected values if there was a single bit error in any bank. To view the uncorrected or raw fuse values, see Section 4.4.4 "Direct OTP fuse read". To determine if an error occurred while programming fuses, do any of the following:

- Check the fuse values against what was written.
- Monitor the INTB signal, but first the ECC interrupt must be unmasked.
- Read bits[5:0] from the OTP ECC CTRLx registers in the Extended Page 2. See Table 39 to decipher single bit error codes and Table 43 for a description of the ECC registers.

Table 40. ECC Enable Registers

Extended Page 1		I^2C Data Bits							
Addr	**Name**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
F0	OTP EN ECC0	—	—	—	EN_ECC_BANK5	EN_ECC_BANK4	EN_ECC_BANK3	EN_ECC_BANK2	EN_ECC_BANK1
		—	—	—	0	0	0	0	0
F1	OTP EN ECC1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—

Table 41. Automatic ECC mode enable registers

Extended Page 2		I^2C Data Bits							
Addr	**Name**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
D0	OTP AUTO ECC0	—	—	—	AUTO_ECC_BANK5	AUTO_ECC_BANK4	AUTO_ECC_BANK3	AUTO_ECC_BANK2	AUTO_ECC_BANK1
		—	—	—	0	0	0	0	0

Table 42. ECC control registers in the extended page 2

Addr	Name	D7	D6	D5	D4	D3	D2	D1	D0
E1	ECC_CTRL1	RSVD	ECC1_CALC_CIN	ECC1_CIN_TBB[5:0]					
		0	0	0	0	0	0	0	0
E2	ECC_CTRL2	RSVD	ECC2_CALC_CIN	ECC2_CIN_TBB[5:0]					
		0	0	0	0	0	0	0	0
E3	ECC_CTRL3	RSVD	ECC3_CALC_CIN	ECC3_CIN_TBB[5:0]					
		0	0	0	0	0	0	0	0
E4	ECC_CTRL4	RSVD	ECC4_CALC_CIN	ECC4_CIN_TBB[5:0]					
		0	0	0	0	0	0	0	0
E5	ECC_CTRL5	RSVD	ECC5_CALC_CIN	ECC5_CIN_TBB[5:0]					
		0	0	0	0	0	0	0	0

Table 43. ECC_CTRLx Registers Description

Bit	Name	Default	Description
5:0	ECCx_CIN_TBB	0	ECC error location code See codes on Table 39
6	ECCx_CALC_CIN	0	Calculate the ECC check bit values 0 = Calculation disabled 1 = Calculation enabled
7	RSVD	0	Reserved

5 References

To obtain more information on NXP products and application solutions, go to the following are URLs:

Support pages	Description	URL
PF3000	Product Summary Page	http://www.nxp.com/webapp/sps/site/prod_summary.jsp?code=PF3000
Analog and Power Management	Home Page	http://www.nxp.com/webapp/sps/site/homepage.jsp?code=ANALOGHOME
Power Management Integrated Circuits (PMICs)	Home Page	http://www.nxp.com/PMIC

6 Revision history

Revision	Date	Description
1.0	5/2015	• Initial release
	7/2015	• Corrected document format
	7/2016	• Updated to NXP document form and style

How to Reach Us:

Home Page:
NXP.com

Web Support:
http://www.nxp.com/support

Document Number: AN5132
Rev. 1.0
7/2016