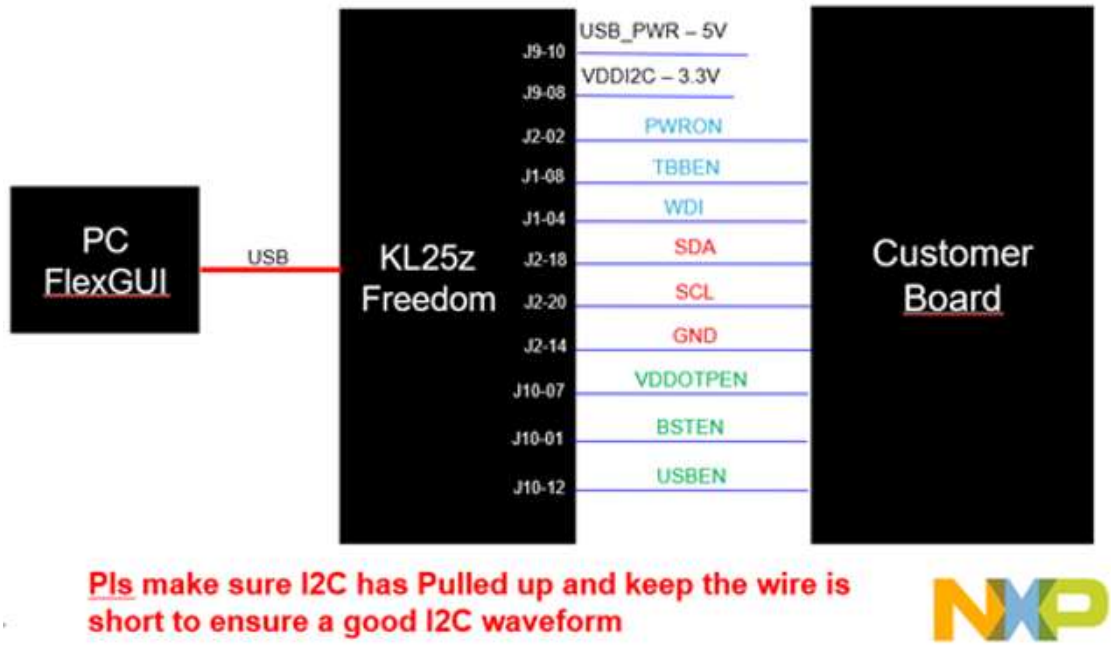
Use KL25Z Freedom board with PF502X board to do OTP

Some customers have not the programming board, or customer solder Blank Version on own board by mistake, they can still use KL25FRDM board to do the OTP burn process by the PF82/PF502X Flex GUI to get their desired PMIC configurations.

The connection structure between customer PMIC board and KL25Z board as below:



You don't need to connect VDDOTPEN, BSTEN and USBEN on board, and these pins are used to generate one 8V for VDDOTP Pin, which is the OTP burn process request.

Customer should provide 8V voltage from external supply for VDDOTP Pin, and then run the script which is generated from OTP excel.

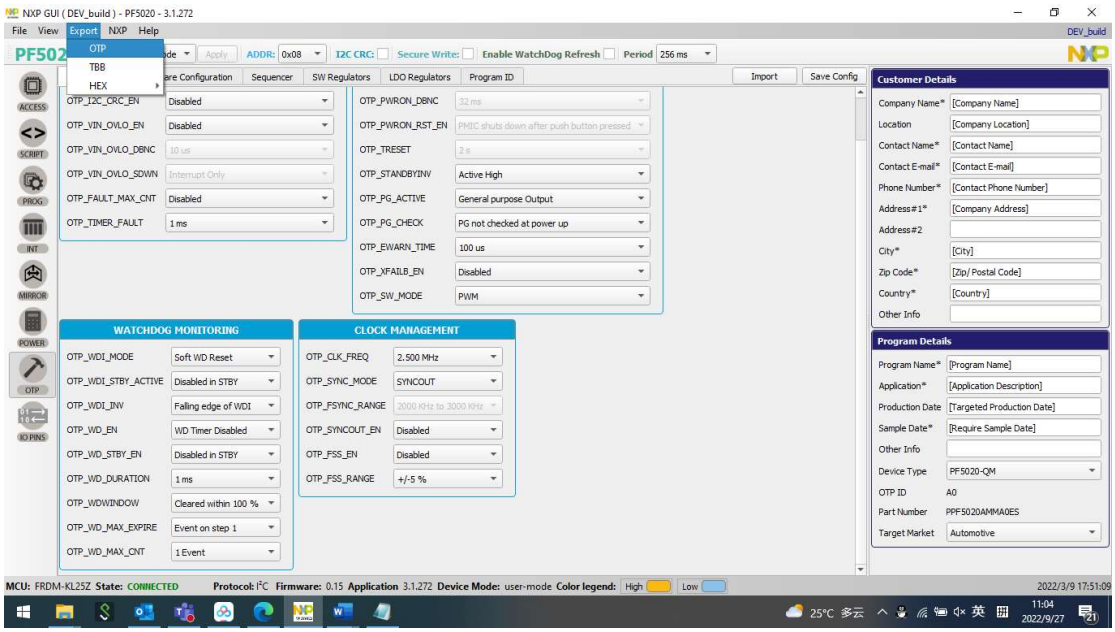
After others pins connection well, the OTP burn can also be achieved by this method on customer board.

Customer can also develop the OTP burn tool by themselves on their hardware product line, the code and operation process can refer to the script which is generated from the OTP request form.

For the detail you can below or your also can refer to UM11373 page22:

1:To program the OTP, the device must be operated in TBB mode. An OTP script can be created using the NXPGUI OTP section.

2:Select OTP from the Export tab in the menu and save the generated OTP file(.txt) in a known location with a desired filename.



3: To operate the board in TBB mode:

4: Remove jumper on J39 and connect an external supply of 8.0 V on the VDDOTP pin at J39, pin2 or TP103.

5: In the PROG section, use the Config Source button to select "script", and then click the Browse button to load the config (*.cfg) file saved before. Click Program to start programming the PF502x device.

6. After the script completes the programming, the device is automatically enabled with the selected OTP configuration. Remove the voltage on the VDDOTP pin (J39-2)

7. Put the jumper on J39 (1-2). Select user-mode in the GUI and click Apply to turn On the board with new settings. Or

8. Turn Off the power supply and turn back-on to reconnect the device.

9.Using the IO Pins tab in the main window of the GUI, set the PWRON pin high to enable the device with the programmed configuration. Ensure that the enable switches SW10, SW11, SW12, SW13 are set high for the regulators to turn On.

Appendix:

A:GUI download address:

GUI PMIC Software | Automotive | NXP Semiconductors

B:KL25ZFreedom board schematic download:

FRDM-KL25Z|Freedom Development Platform|Kinetis® MCU | NXP Semiconductors

C:Below is the example of OTP script:

```
//PF5020 - OTP Editor
//file generated on 周二 9 月 27 11:02:01 2022
//Device Type:PF5020-QM
//OTP ID:A0
//Customer:[Company Name]
//Part Number:PPF5020AMMA0ES
```

SET_DPIN:PF5020:PWRON:low
SET_DPIN:PF5020:WDI:low
SET_DPIN:PF5020:TBBEN:high

//MAIN_OTP
SET_REG:PF5020:OTP_MIRROR:OTP_I2C:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_CTRL1:0x08
SET_REG:PF5020:OTP_MIRROR:OTP_CTRL2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_CTRL3:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_FREQ_CTRL:0x80
SET_REG:PF5020:OTP_MIRROR:OTP_PWRON:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_WD_CONFIG:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_WD_EXPIRE:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_WD_COUNTER:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_FAULT_COUNTERS:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_FAULT_TIMERS:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_PWRDN_DLY1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_PWRDN_DLY2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_PWRUP_CTRL:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_RESETBMCU_PWRUP:0x0c
SET_REG:PF5020:OTP_MIRROR:OTP_PGOOD_PWRUP:0x10
SET_REG:PF5020:OTP_MIRROR:OTP_SW1_VOLT:0x19
SET_REG:PF5020:OTP_MIRROR:OTP_SW1_PWRUP:0x05
SET_REG:PF5020:OTP_MIRROR:OTP_SW1_CONFIG1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SW1_CONFIG2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SW2_VOLT:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SW2_PWRUP:0x0a
SET_REG:PF5020:OTP_MIRROR:OTP_SW2_CONFIG1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SW2_CONFIG2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SWND1_VOLT:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SWND1_PWRUP:0x0d
SET_REG:PF5020:OTP_MIRROR:OTP_SWND1_CONFIG1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SWND1_CONFIG2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_LDO1_VOLT:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_LDO1_PWRUP:0x0e
SET_REG:PF5020:OTP_MIRROR:OTP_LDO1_CONFIG:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_VSNVS_CONFIG:0x01
SET_REG:PF5020:OTP_MIRROR:OTP_OV_BYPASS1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_OV_BYPASS2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_UV_BYPASS1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_UV_BYPASS2:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_ILIM_BYPASS1:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_ILIM_BYPASS2:0x00

```
SET_REG:PF5020:OTP_MIRROR:OTP_PROG_IDH:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_PROG_IDL:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_DEBUG1:0x01
SET_REG:PF5020:OTP_MIRROR:OTP_SW_COMP1:0x24
SET_REG:PF5020:OTP_MIRROR:OTP_SW_COMP3:0x00
SET_REG:PF5020:OTP_MIRROR:OTP_SW_RAMP:0x00
//---------Configure Controller----------
SET_REG:PF5020:OTP_Misc:FCMD:0x80
//This is probably not necessary
//MREF_Read_1_MSB
SET_REG:PF5020:OTP_Misc:FADDR_START:0x00
SET_REG:PF5020:OTP_Misc:FDATA:0xAC
SET_REG:PF5020:OTP_Misc:FCMD:0xA9

//MREF_Read_2_MSB
SET_REG:PF5020:OTP_Misc:FADDR_START:0x02
SET_REG:PF5020:OTP_Misc:FDATA:0xDC
SET_REG:PF5020:OTP_Misc:FCMD:0xA9

//8'b0011_1000
SET_REG:PF5020:OTP_Misc:FADDR_START:0x08
SET_REG:PF5020:OTP_Misc:FDATA:0x38
SET_REG:PF5020:OTP_Misc:FCMD:0xA9

//MREF_Read_3_MSB
SET_REG:PF5020:OTP_Misc:FADDR_START:0x09
SET_REG:PF5020:OTP_Misc:FDATA:0xDC
SET_REG:PF5020:OTP_Misc:FCMD:0xA9

//set FUSE Address to write to 0Ch
//set DATA to write to fuse address 0C
//run single data write command on OTP controller
SET_REG:PF5020:OTP_Misc:FADDR_START:0x0C
SET_REG:PF5020:OTP_Misc:FDATA:0xD2
SET_REG:PF5020:OTP_Misc:FCMD:0xA9

SET_REG:PF5020:OTP_Misc:MAX_PGM_TRIES:0x08
SET_REG:PF5020:OTP_Misc:MRR_SVDR_IN:0x13

SET_REG:PF5020:OTP_Misc:MR_TEST_H:0x00
SET_REG:PF5020:OTP_Misc:MR_TEST_L:0x02

SET_REG:PF5020:OTP_Misc:MREF_TEST_H:0x00
SET_REG:PF5020:OTP_Misc:MREF_TEST_L:0x00
```

SET_REG:PF5020:OTP_Misc:PULSE_DUR_1:0xBB
SET_REG:PF5020:OTP_Misc:PULSE_DUR_2:0x08

SET_REG:PF5020:OTP_Misc:FADDR_START:0x00
SET_REG:PF5020:OTP_Misc:FADDR_STOP:0x36

//---------SET CRC Values----------
SET_REG:PF5020:OTP_Misc:FCMD:0xA5
//This command may take 100-200us to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
SET_REG:PF5020:OTP_Misc:FCMD:0xA4
//This command may take 100-200us to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
GET_REG:PF5020:OTP_MIRROR:OTP_S0_CRC_LSB
GET_REG:PF5020:OTP_MIRROR:OTP_S0_CRC_MSB
GET_REG:PF5020:OTP_Misc:SECT_STATUS

//Burn the OTP fuses
//39. Apply VDDOTP = 7.8 V
//40. Wait 100 µs for voltage at pin VDDOTP to stabilize

//---------Program OTP Fuses----------
SET_REG:PF5020:OTP_Misc:FCMD:0x96
//This command may take 50-100ms to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS

//---------BURN WP and BE bits----------
SET_REG:PF5020:OTP_Misc:FADDR_STOP:0xFF
SET_REG:PF5020:OTP_Misc:FADDR_START:0xFC
SET_REG:PF5020:OTP_Misc:FDATA:0xAA
SET_REG:PF5020:OTP_Misc:FCMD:0x87

//This command may take 1ms to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS

GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
SET_REG:PF5020:OTP_Misc:FADDR_START:0xFD
SET_REG:PF5020:OTP_Misc:FDATA:0x55
SET_REG:PF5020:OTP_Misc:FCMD:0x87

//This command may take 1ms to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
SET_REG:PF5020:OTP_Misc:FADDR_START:0xFE
SET_REG:PF5020:OTP_Misc:FDATA:0xAA
SET_REG:PF5020:OTP_Misc:FCMD:0x87

//This command may take 1ms to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
SET_REG:PF5020:OTP_Misc:FADDR_START:0xFF
SET_REG:PF5020:OTP_Misc:FDATA:0x55
SET_REG:PF5020:OTP_Misc:FCMD:0x87

//This command may take 1ms to complete. To calculate the time one could poll the FSTATUS register for the busy to go low
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:FSTATUS
GET_REG:PF5020:OTP_Misc:SECT_STATUS

//Verify Mirror Registers = Fuse Value
GET_REG:PF5020:Functional:DEVICE_ID
SET_REG:PF5020:OTP_Misc:FADDR_START:0x00
SET_REG:PF5020:OTP_Misc:FADDR_STOP:0x36
SET_REG:PF5020:OTP_Misc:FCMD:0xAB
SET_REG:PF5020:OTP_Misc:FCMD:0xA0
SET_REG:PF5020:OTP_Misc:FCMD:0xA1
SET_REG:PF5020:OTP_Misc:FCMD:0xA4
GET_REG:PF5020:OTP_MIRROR:OTP_S0_CRC_LSB
GET_REG:PF5020:OTP_MIRROR:OTP_S0_CRC_MSB

GET_REG:PF5020:OTP_Misc:SECT_STATUS
GET_REG:PF5020:OTP_Misc:FSTATUS

//If SECT_STATUS = 0x3F & FSTATUS = 0x00 part is programmed correctly.
//Verify CRC_LSB and CRC_MSB match the values in section "SET CRC VALUES"
//Alert the user to remove that the VDDOTP pin is set to 0V

SET_DPIN:PF5020:TBBEN:low

//Rev,A