

BOOSTXL-TPS650861 EVM User's Guide

This user's guide provides guidelines to the program the OTP memory of the TPS65086100 device using the BOOSTXL-TPS650861 BoosterPack™ plug-in module, an MSP430F5529 LaunchPad™ development kit, and the IPG-UI EVM GUI software. Programming the OTP memory allows for rapid prototyping which provides quick time to market as well as an option to support many smaller projects with different OTP programs in a cost effective manner.

Contents

1	Introduction	1
2	Hardware.....	2
3	Software.....	4
4	Programming Steps	5

List of Figures

1	MSP430F5529 LaunchPad™ Development Kit	2
2	BOOSTXL-TPS650861 BoosterPack™ Plug-in Module.....	3
3	BOOSTXL-TPS65086100 Block Diagram	4
4	Jumper to V _x Net.....	6

List of Tables

Trademarks

BoosterPack, LaunchPad are trademarks of Texas Instruments.
All other trademarks are the property of their respective owners.

1 Introduction

The steps in this document describe the basic procedure for programming the TPS65086100 device using the IPG-UI on the BOOSTXL-TPS650861 EVM board. See the [TPS65086100 Non-Volatile Memory Programming Guide](#) for detailed programming information. All programmed OTP settings should be validated during the prototyping phase to make sure that functions as desired because parts cannot be returned in case of incorrect programming. Report issues to <http://support.ti.com>.

This procedure requires:

- An An MSP430F5529 LaunchPad development kit, [MSP-EXP430F5529LP](#)
- A USB A to micro B cable (included with the LaunchPad development kit)
- A BOOSTXL-TPS650861 BoosterPack plug-in module
- TPS650861 devices (TPS65086100RSK)
- An internet connection

2 Hardware

This section briefly describes the hardware required to program TPS65086100 OTP memory using the BOOSTXL-TPS650861 BoosterPack plug-in module.

2.1 MSP430F5529 LaunchPad™ Development Kit

The MSP430F5529 LaunchPad development kit serves as a way to communicate between the IPG-UI software and the TPS65086100 device. The firmware on the MSP430F5529 LaunchPad development kit must be updated before it can communicate with the TPS65086100 device. If the firmware is not updated, do not plug the BOOSTXL-TPS650861 BoosterPack plug-in module into the MSP430F5529 LaunchPad development kit.

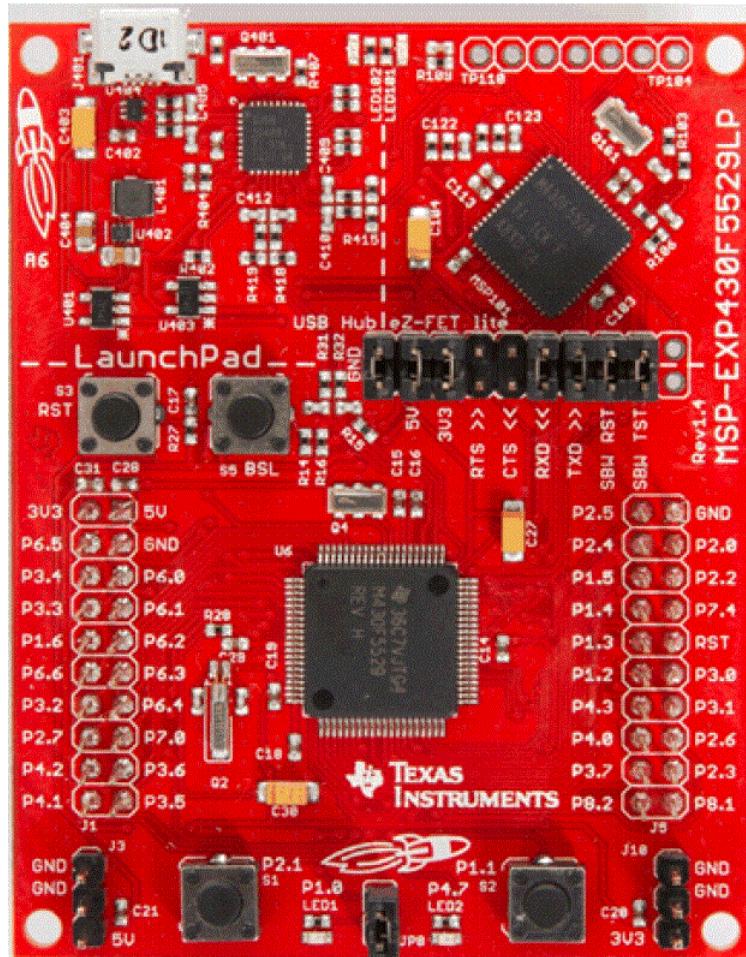


Figure 1. MSP430F5529 LaunchPad™ Development Kit

2.2 BOOSTXL-TPS650861 BoosterPack™ Plug-in Module

The BOOSTXL-TPS650861 BoosterPack plug-in module is a socketed board that can be used with an MSP430F5529 LaunchPad development kit to program TPS65086100 OTP memory. The socketed design allows devices to be easily swapped in and out so multiple devices can quickly be programmed. [Figure 3](#) shows the block diagram of the BOOSTXL-TPS65086100, which also shows how the board connects to the MSP430F5529 LaunchPad development kit and a PC.

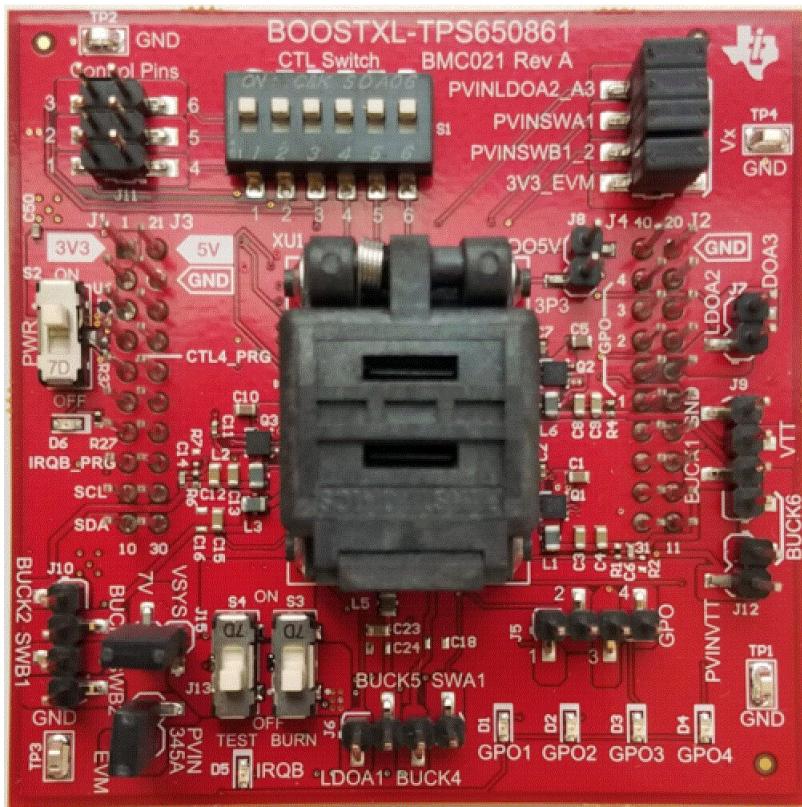


Figure 2. BOOSTXL-TPS650861 BoosterPack™ Plug-in Module

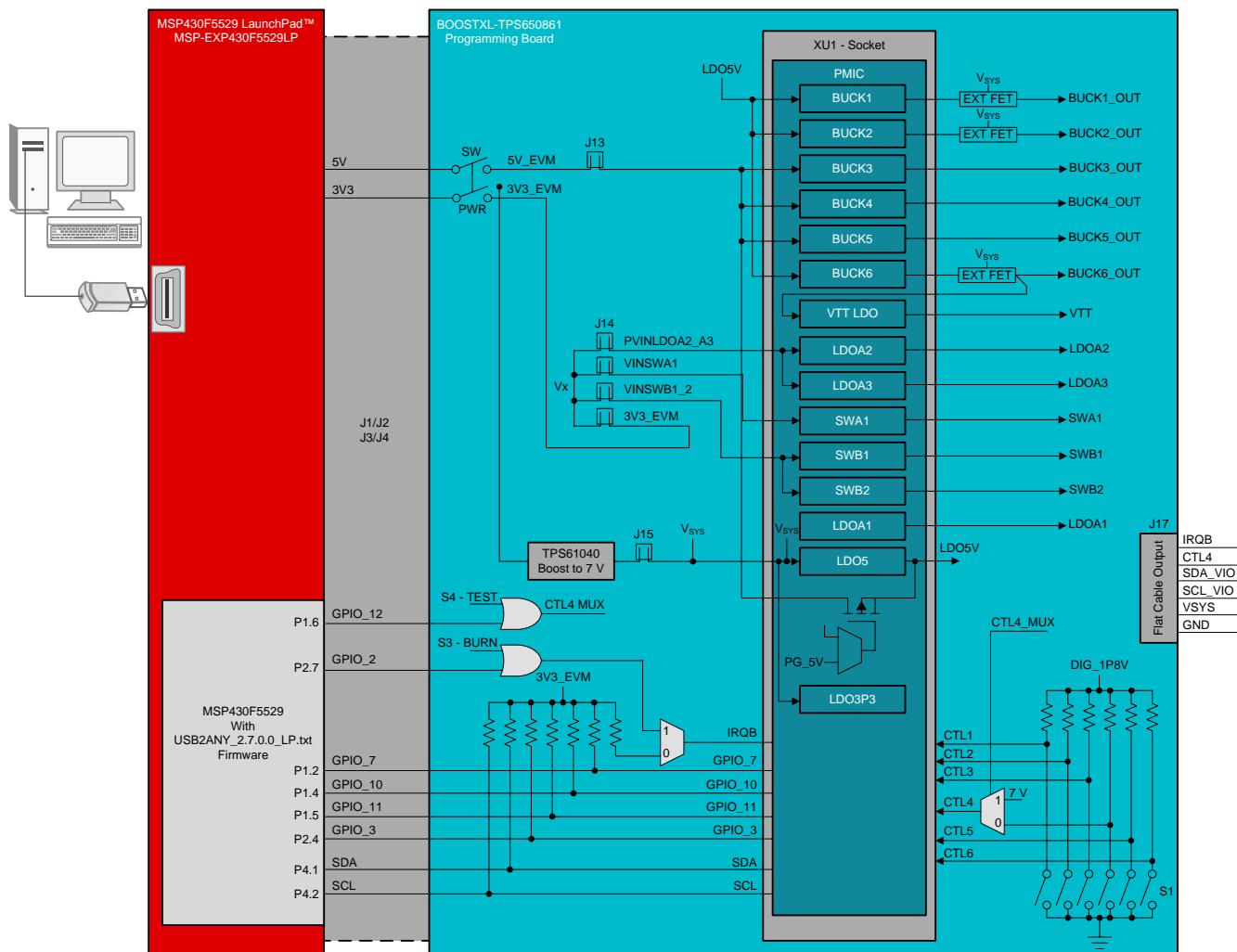


Figure 3. BOOSTXL-TPS65086100 Block Diagram

3 Software

Download these files to make sure that all of the required software is available:

1. The latest revision of the [IPG-UI EVM GUI](#).
2. The latest revision of the TPS65086100 IPG-UI device file (TPS650861-1.0.json) and script file (TPS650861-Script-1p0.js) from [here](#).
3. The [TPS65086100 OTP Generator](#) tool. Alternatively, any of the other OTP generator tools located in the technical documents tab of the [TPS650861 product folder](#) can be used.
4. The latest MSP430F5529 LaunchPad USB2ANY firmware (USB2ANY_2.7.0.0_LP.txt) from [here](#).
5. The [MSP430_USB_Firmware_Upgrade_Example-1.3.1.1-Setup.exe](#) from the [MSP430_USB_Developers_Package 5_20_06_02](#).

NOTE: If available, newer versions of any of these files can be used.

4 Programming Steps

The following sections give step-by-step instructions to program a TPS65086100 OTP memory.

4.1 *Update MSP430F5529 Firmware*

Update the MSP430F5529 LaunchPad development to the USB2ANY_2.7.0.0_LP.txt file before putting the BOOSTXL-TPS650861 on the LaunchPad development kit.

- Step 1. Press the S5 button while connecting the Micro USB cable.
- Step 2. Run the Firmware Upgrade Example.
- Step 3. Choose "Select Firmware".
- Step 4. Choose "Browse" and select the USB2ANY_2.7.0.0_LP.txt file downloaded previously.
- Step 5. Choose "Upgrade Firmware".
- Step 6. When complete, disconnect the USB cable.

4.2 *Update Programming Script File*

Use the [TPS65086100 OTP Generator](#) Excel-based tool to create a script file for IPG-UI EVM GUI.

1. Fill out or modify the Overview, Sequencing, and Additional Details tabs in the OTP Generator tool to match the desired settings. Refer to the [TPS65086100 Non-Volatile Memory Programming Guide](#) for more information.
2. Copy the B2 to B96 cells which are highlighted in purple in the "Script Generator" tab.
3. Open the TPS650861-Script-1p0.js file in a text editor such as Notepad. The script file is generally located in the following directory: C:\Users\[username]\Documents\Texas Instruments\IPG-UI\Devices\scripts
4. If the var custom_program_1[XX] does not already exist, paste the copied cells into the top of the TPS650861-Script-1p0.js file. For example, do not allow two var custom_program_100 to be in the same file. If necessary, Delete the obsolete program.
5. Save the updated TPS650861-Script-1p0.js file.

4.3 Set Up the BOOSTXL-TPS650861 BoosterPack™ Plug-in Module

The Programming Board Setup tab of the OTP Generator Tool has suggestions for different OTP settings. Set up the BOOSTXL-TPS650861 board for the desired OTP based on these suggestions:

- Step 1. Set the CTLx pins (S1) all to the "ON" position. This connects them all to GND and will prevent any rails from turning on unexpectedly during programming.
- Step 2. If an external supply is used for VSYS, remove J15 which typically connects VSYS to 7 V.
- Step 3. Typically VIN_BUCK345_ANA is connected to 5V_EVM using jumper J13. VIN_BUCK345_ANA can be disconnected from 5V_EVM and powered from another supply such as BUCK1 or BUCK2 (as long as they are providing 3.3V or greater) using a female-to-female wire jumper provided in the kit.
- Step 4. Typically PVINVTT is connected to BUCK6 using jumper J12. PVINVTT can be disconnected from BUCK6 and powered from another supply using a female-to-female wire jumper provided in the kit.
- Step 5. If LDOA2 or LDOA3 is used, PVINLDOA2_A3 should be connected to a 1.8V supply that is enabled prior to LDOA2 or LDOA3 being enabled. For programming purposes, 3.3 V input can also be used, though datasheet parametric data has not been tested at 3.3 V.
- Step 6. If SWA1 is used, PVINSWA1 should be connected to an appropriate voltage source based on PG target (if used) and sequencing.
- Step 7. If SWB1, SWB2, or SWB1_2 is used, PVINSWB1_2 should be connected to an appropriate voltage source based on PG target (if used) and sequencing.
- Step 8. The floating V_x net is provided on the BOOSTXL-TPS650861 board to supply PVINLDOA2_A3, PVINSWA1, and PVINSWB1_2 with power. This is done using jumper J14 as shown in [Figure 4](#). For example, for simple testing if the PG level for SWA1, SWB1, and SWB2 are all set to 3.3V, then the 3V3_EVM can be shorted to V_x using the provided jumpers, and then PVINLDOA2_A3, PVINSWA1, and PVINSWB1_2 can all be shorted to V_x as well to provide power for all five outputs. If SWA1 uses 1.8 V instead, its jumper to V_x can be removed and a female-to-female wire jumper provided in the kit can be connected to one of the BUCKs supplying 1.8V.

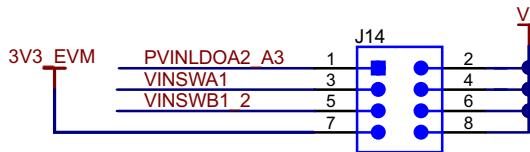


Figure 4. Jumper to V_x Net

- Step 9. If the external feedback (Ext FB) option is chosen for BUCK1 to supply 5 V then the expected output of BUCK1 on the BOOSTXL-TPS650861 EVM is 0.4 V. If the programmed device is being tested on the TPS650860EVM-116 EVM, R24 can be replaced with a 294 k Ω resistor and R25 populated with a 25.5 k Ω resistor. Adding ~5 pF capacitor on top of R24 can be used to improve transient response.

4.4 Program The Device Using IPG-UI EVM GUI

To plug the BOOSTXL-TPS650861 BoosterPack plug-in module into the MSP430F5529 LaunchPad development kit and program a TPS65086100 device, do these steps:

- Step 1. Plug the BOOSTXL-TPS650861 BoosterPack board into the MSP430F5529 LaunchPad development kit.
- Step 2. Put a TPS65086100 device into the socket with pin 1 (indicated with a dot) located in the upper left of the socket (indicated with a silkscreen arrow).
- Step 3. Start the IPG-UI EVM GUI.
- Step 4. Under "Create New Project", use the "Select Device" drop-down menu to select the latest revision of the TPS650861 device and create project.
- Step 5. Connect the MSP430F5529 LaunchPad development kit to the USB cable and confirm that the IPG-UI reports "Hardware Connected" in the bottom right corner.

- Step 6. Set the "PWR" switch (S2) to "ON" or use the "Power On" button in the Device Controls Power tab of the IPG-UI EVM GUI to provide power to the BOOSTXL-TPS650861 board.
- Step 7. Confirm the connection between the IPG-UI EVM GUI and the PMIC device by reading the SHUTDNSRC register (0x05) from the Register Map or Register Controls tabs in the IPG-UI EVM GUI. It should read back as 0x04, indicating that the last shutdown was because of undervoltage lockout.
- Step 8. In the Device Controls Programming tab:
 1. Click the button in section (1) to put the PMIC device in programming mode. This connects PMIC CTL4 pin to 7V, writes the PROGRAMMING_STATE bit in the OTP_CTRL1 register to 1b and then sets CTL4 back to controlled by the S1.
 2. In section (2), choose the OTP bank, generally Bank 0 unless it has already been programmed.
 3. In section (3), Select the OTP Program from the dropdown list.
 4. Click the button in section (4) to write the OTP program into the active (volatile) registers.
 5. Click the button in section (5) to confirm the values currently in the active (volatile) registers match the target values.

At this point, the device can be tested for the proper sequencing and voltage values. If there is any power fault, the part will reload the TPS65086100 blank OTP. At that time, all CTL pins should be set back to 0 V ("ON" position to connect to GND) and this sequence restarted from [Step 8.1](#).

6. When the settings are confirmed, or if testing is to be done on an alternate platform, click the button in section (6) to program the OTP memory with the current settings in the active registers.
7. To confirm the OTP memory has been programmed, remove power from the device and re-apply power. Click the button in section (1) to enter programming mode and click the button in section (5) to confirm the program still matches.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#), [TI's General Quality Guidelines](#), or other applicable terms available either on [ti.com](#) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025