

NVM Programming for VR13 Power Controllers

ABSTRACT

TPS53679, TPS53659, TPS53622 are Intel[™] VR13 Serial VID (SVID)-compliant power supply controllers that have programmable parameters. This guide also applies to proprietary footprint devices TPS53678, TPS53658 TPS53655, as well as TPS53681, even though it is not an Intel power controller. The PMBus interface configures the parameter values which are stored into non-volatile memory (NVM) as new bootup default values. This guide gives a tutorial on NVM programming, and the tools TI provides for the programming process. This guide applies to all the devices in the TI VR13 controller family.

C	or	ιte	nts	

1	Introduction	2
2	Technical Overview	4
3	Use of TI Programming Tools	8

List of Figures

Hardware Connections	. 5
Send Byte Protocol	. 6
Write Byte Protocol	. 6
Write Word Protocol	. 6
Read Word Protocol	. 6
Block Write Protocol	. 6
Block Read Protocol	. 6
Change Scanning Modes	. 9
Scanning Mode Selections	. 9
System View	10
Settings for Individual Device	11
Write to Hardware	11
Refresh All Parameters	12
Switch between Multiple Devices	12
Save a System File	13
Select System File	14
Select Offline System File	14
Offline System View	15
Import to Devcie	16
Import Project File	16
Select Project File	17
Select All and Write Checked Parameters	18
Successful Project File Import	19
Store to NVM	19
Import System File	20
Check All Parameters	21
System File Import Messages	21
Device File Export	22
	Hardware Connections



Introduction

29	Recommended CSV Export Settings	23
30	Device/Project Configuration Compare	23
31	Compare Project Files	24
32	Project File Compare	24
33	Build System	25
34	Modify Existing System	25
35	Existing Project File Devices	26
36	Add a <i>Project</i> File to a System File	27
37	Browse for System File	28
38	Load the Script	28
39	Scan and Start	29
40	Manufacturing Passed	30
41	Socketed Programming Board	31

List of Tables

1	Configuration File Formats	3
2	Hardware Connections for Offline NVM Programming	4
3	SMBus Transaction Types	5
4	Example NVM Data	7

Trademarks

Intel is a trademark of Intel Corporation.

1 Introduction

This guide gives an overview of NVM programming for the following TI VR13 controller devices: TPS53679, TPS53659, TPS53622, TPS53678, TPS53658, TPS53655 and TPS53681. Local TI sales and field applications representatives can provide relevant documentation to these devices. This guide gives a technical overview and step-by-step instructions.



1.1 Software Tools

1.1.1 Fusion Digital Power Designer

Fusion Digital Power Designer is a graphical user interface (GUI) that configures and monitors Texas Instruments digital power controllers, sequencer monitors, and health monitors. The GUI uses the PMBus protocol to communicate with the device over a serial bus using a proprietary USB adapter.

Use the *Fusion Digital Power Designer* GUI for engineering development. The GUI gives access to all available user-configurable settings, fault information, and telemetry readings. After you configure the device, you can export the settings to a configuration file. The export options are:

- System file (.tifsp) contains configuration data for all of the devices in a system
- Project file (.xml) contains configuration data for a single device
- PMBus programmer script (.csv) is a text file containing step-by-step programming instructions for non-TI software tools

Download the latest version of *Fusion Digital Power Designer* from this URL: http://www.ti.com/tool/fusion_digital_power_designer

1.1.2 Fusion Manufacturing Tool

The *Fusion Manufacturing Tool* is a graphical user interface (GUI) that programs TI digital power controllers in a production environment. Download the latest copy of the *Fusion Manufacturing Tool* from this URL: http://www.ti.com/tool/fusion_mfr_gui

1.1.3 Online and Offline Modes

The *Fusion Digital Power Designer* software operates in online mode when you have connected a device to the system that hosts the software. The software operates in offline mode when you have not connected a device to the system that hosts the software. The software installer provides a different shortcut for each mode.

1.1.4 Relevant File Formats

File	Description	Extension	Туре	Compatilble Software
Fusion project file	Contains configuration data for a single device. Intended for engineering development.	.xml	plain-text XML	Fusion Digital Power Designer
Fusion system file	Contains configuration data for multiple devices. Can be used in engineering development, and production programming.	.tifsp	plain-text XML	Fusion Digital Power Designer, Fusion Manufacturing Tool
PMBus programmer script	Simplified programming script (comma separated format) for production environment.	.CSV	plain-text comma separated value	Fusion Digital Power Designer and third-party tools

Table 1. Configuration File Formats

1.2 Hardware Tools

1.2.1 USB-to-GPIO Dongle

Both *Fusion Digital Power Designer* and *Fusion Manufacturing Tool* GUIs use the TI USB-GPIO adapter to interface between a host computer and controller devices. Use this link to get a TI USB-GPIO adapter: http://www.ti.com/tool/usb-to-gpio.

2 Technical Overview

This section shows how to program NVM with the TI software tools, or with non-TI tools.

2.1 Hardware Connections

You can program NVM on a fully-populated application board. However, successful programming requires only a few connections be made to these devices. In some cases offline gang-programming in socketed boards works better than a full In Circuit Test (ICT) solution.

Pin Name	Connection
V3P3	Connect to +3.3V supply, and bypass with a minimum of 1.0 μ F to ground.
VREF	Bypass to ground with a minimum of 1.0 μ F to ground.
ADDR	Connect a resistor divider from VREF to ADDR to Ground to set the PMBus address (required for programming), as described in the product datasheet.
SMB_DIO, SMB_CLK	Connect to programming host, pull-up to 3.3V with a 2.2 k Ω resistor. Note, the TI USB-to-GPIO interface adapter has internal pull-up resistors.
ATSEN, BTSEN, TSEN	Not used for NVM programming. Tie-off with a 1:1 resistor divider from VREF to TSEN to Ground, suggested resistor value is 49.9 k Ω . This sets the pin voltage to appx. 0.85 V, which the controller device interprets as 32°C.
AVR_EN, BVR_EN, VIN_CSNIN, CSPIN, AVSP, BVSP, AVSN, BVSN	Not used for NVM programming. Tie-off to ground.
ACSPx, BCSPx	Not used for NVM programming. Tie off to VREF.
APWMx, BPWMx, ASKIP, BSKIP, VRFAULT, SMBALERT, SALERT, PIN_ALT, SDIO, SCLK	Not used for NVM programming. Float these pins if not used.

Table 2. Hardware Connections for Offline NVM Programming





Figure 1. Hardware Connections

2.2 Communication Protocol and Interfacing

These devices comply with the PMBus version 1.3 specification. Use this link to find timing and electrical characteristics of the PMBus specification in the PMB Power Management Protocol Specification, Part 1, revision 1.3 available at http://pmbus.org.. The PMBus specification inherits its transport and network layer behavior from the SMBus specification. These devices comply with the SMBus 3.0 specification, which is available at this URL: http://smbus.org/specs/. Use the transaction prototypes to program the device NVM correctly. Refer to the Section 2.3 section for more information about prototype use in NVM programming.

The use of Packet Error Correction (PEC) is optional. If clock pulses are supplied for a PEC byte, these devices use PEC, otherwise they do not. For simplicity, Table 3 lists required transaction types without PEC bytes.

Transaction Type	Purpose
Send Byte	Used to issue NVM Store operations (STORE_DEFAULT_ALL).
Write Byte	Used to change the current PAGE
Write Word	Used to Write the value of VOUT_MAX.
Read Word	Used to Read the current value of VOUT_MAX.
Write Block	Used to Write the USER_DATA commands, which contain most of the NVM storable parameters these devcies support.
Read Block	Used to read the USER_DATA commands and checksum (MFR_SERIAL).

Table 3.	SMBus	Transaction	Types
----------	-------	-------------	-------

Technical Overview





2.3 Programming Process

6

To simplify the programming procedure and reduce programming time, TI VR13 controller devices combine NVM settings into a small number of registers. All settings in MFR_SPECIFIC commands map into registers USER_DATA_00 through USER_DATA_12. Only a few other registers are required. Program the USER_DATA command with the Block Read/Write command protocol described in the SMBus Specification. Table 4 lists the complete NVM configuration for a single device. All other settings map to USER_DATA commands.

The *Fusion Digital Power Designer* GUI lets power supply designers configure devices in a graphical environment. The GUI saves these settings in a configuration file or script that you can load on to other devices.



2.3.1 Programming Procedure

This section describes NVM programming through third-party tools. This procedure is completed automatically during import and export of *Project* files from *Fusion Digital Power Designer*.

Configure User-Programmable Parameters (one-time only)

- 1. Set all of the user-accessible parameters via the standard PMBus, and Manufacturer Specific commands, using *Fusion Digital Power Designer* or the *Technical Reference Manual* for the device.
- 2. Issue the STORE_DEFAULT_ALL command. This command commits these values to NVM, and updates the checksum value.
- 3. Wait approximately 100 ms.
- 4. Write PAGE to 00h.
- 5. Read-back and Record the value of IC_DEVICE_ID and IC_DEVICE_REV commands
- 6. Read-back and Record the value of the USER_DATA_00 through USER_DATA_12 commands
- 7. Read-back and Record the value of the MFR_SERIAL command
- 8. Read-back and Record the value of VOUT_MAX
- 9. Write PAGE to 01h
- 10. Read-back and Record the value of VOUT_MAX

Program and Verify NVM (repeat for each device)

- 1. Apply 3.3 V to the V3P3 pin to start the device. Make sure to disable power conversion for NVM programming.
- Read-back and verify that IC_DEVICE_ID and IC_DEVICE_REV values match those recorded previously. This verification ensures that user-parameters being programmed correspond to the same device/revision as previously configured.
- 3. Write PAGE to 00h.
- 4. Write the USER_DATA_00 through USER_DATA_12 commands, with the values recorded previously.
- 5. Write VOUT_MAX (Page 0) with the value recorded previously.
- 6. Write PAGE to 01h
- 7. Write VOUT_MAX (Page 1) with the value recorded previously.
- 8. Issue STORE_DEFAULT_ALL.
- 9. Wait approximately 100 ms.
- 10. Read-back the MFR_SERIAL command, and compare the value to that recorded previously. If the new MFR_SERIAL matches the value recorded previously, the software has successfully programmed the NVM.

2.3.2 Example NVM Data

Table 4 gives an example configuration that contains **all programmable parameters** in a VR13 controller device. Every application has different data. All other parameters stored in configuration files are either calculated, measured, or derived from these values. The *Fusion Digital Power Designer* GUI displays block commands in ascending order of significance (for example byte 0, byte 1, ...). The GUI displays word commands in descending significance (for example byte 1, byte 0).

PMBus Command	Transaction Type	CMD Code (hex)	Example Value (hex)
USER_DATA_00	Block Write, Block Read	B0h	1410020030F5
USER_DATA_01	Block Write, Block Read	B1h	00000000040
USER_DATA_02	Block Write, Block Read	B2h	890400000D0
USER_DATA_03	Block Write, Block Read	B3h	030010050080
USER_DATA_04	Block Write, Block Read	B4h	0906C325C777

Table 4. Example NVM Data

Technical Overview

PMBus Command	Transaction Type	CMD Code (hex)	Example Value (hex)
USER_DATA_05	Block Write, Block Read	B5h	E408C57983E1
USER_DATA_06	Block Write, Block Read	B6h	85DA1112E87F
USER_DATA_07	Block Write, Block Read	B7h	804B051053FA
USER_DATA_08	Block Write, Block Read	B8h	000000000A6
USER_DATA_09	Block Write, Block Read	B9h	000170848080
USER_DATA_10	Block Write, Block Read	BAh	00262EC0C185
USER_DATA_11	Block Write, Block Read	BBh	080220C08FE1
USER_DATA_12	Block Write, Block Read	BCh	40F08D20FF01
VOUT_MAX[PAGE 0]	Write Word, Read Word	24h	00FF
VOUT_MAX[PAGE 1]	Write Word, Read Word	24h	00FF
MFR_SERIAL	Block Write, Block Read	9Eh	05C60AD2

Table 4. Example NVM Data (continued)

3 Use of TI Programming Tools

This section gives short tutorials for common tasks using the Fusion Digital Power Designer software.

3.1 Fusion Digital Power Designer

WARNING

Create Fusion Digital Power Designer System files and Project files in online mode only. Do not update settings in offline mode. Files that you edit offline will not have the correct USER_DATA and checksum values stored in the System file, and can lead to unpredictable results.

3.1.1 Build and Export a System (.tifsp) File

The *Fusion Digital Power Designer* software has a convenient graphical interface to configure and test TI VR13 controllers.

An more detailed discussion of each parameter in the GUI can be found in the application report *Using the Fusion Digital Power Designer for TPS536xx VR13 Multiphase Solutions*. Contact vr@list.ti.com for more information.

- 1. Connect the USB-GPIO adapter to either a system board, or socketed programming board with a TI VR13 controller on-board.
- Apply 3.3 V to the V3P3 pins of the controller devices. It is not necessary to apply 5-V and 12-V power. Make sure to disable power conversion. You may need to pull the AVR_EN/BEN pins low to stop power conversion.
- 3. Start the Fusion Digital Power Designer software.
- 4. Click Change Scanning Modes, then select DEVICE_ID & DEVICE_CODE & IC_DEVICE_ID if the software does not recognize your device.



Fusion Di Version 7.0.	gital Power 9.1 [2017-04-24	Dəsig 4]	nər		
No Devices Fo	und! devices were found. Plea your device.	ase check tha	t the serial cable	e end of your USB adapter	is attached to your device a
Scanning Mode:	DeviceIDAndCod	eAndICDe	viceID		
USB Adapter Fin	mware Version: 1.0.	10			
Bus Sneed:	Packet Error Check	king:		ALERT Pullup:	2.2 kΩ 🗸
○ 100 kHz	Enabled		• Serial	CLOCK Pullup:	2.2 kΩ 🗸
 400 kHz 	O Disabled			DATA Pullup:	2.2 kΩ 🔍
Signals					
SMBALERT#:	ACK: High	Refrest	1		
Control Lines:	#1 #2	#3	#4	#5	
(dick to set)	🔿 High 🛛 🔿 High	🔘 High	🔘 High	🔘 High	Refresh All
farmer an and		0.000	(C) Law	Olm	

Figure 8. Change Scanning Modes

	. 🗆 🎽
	DEVICE_ID
DEVICE_COD	DEVICE_CODE
DEVICE_COD	IC_DEVICE_ID UCD3XXX Isolated
DEVICE_COD	DEVICE_ID & DEVICE_CODE
DEVICE_COD	DEVICE_CODE & IC_DEVICE_ID
DEVICE_COD	DEVICE_ID & DEVICE_CODE & IC_DEVICE_ID Skip
DEVICE_COD	É 🛛 112d 0x70 DEVICE_CODE 🗹
DEVICE_COD	E 🗹 113d 0x71 DEVICE_CODE 🗸

Figure 9. Scanning Mode Selections



Use of TI Programming Tools

 From the System View, click Click to Configure Device to view individual controller devices.
 The application report, Using the Fusion Digital Power Designer for TPS536xx VR13 Multiphase Solutions contains detailed information about NVM configuration for TI VR13 multiphase controllers.

🜵 Texas Instruments - Fusion Digital Powe	er Designer [Syster	n View]							• 🔀
File Tools Debug Help									
Q Scan for Device (Device_ID Device_Code IC	_Device_ID) 🛛 🖓 Bui	Id System Sy	stem Monif	tor 🛛 🔚 Save	e 🗸 🗸 Auto Write 🛛 🧶 Sto	p Pol	ling		
Power Rails Tree									
# △ Rail # △ Rail Name	Vout On De	ay Rise	Off Delay	Fall	Dependencies (Direct Only)				
Device: TP553622 (@ PMBus Address 8	38d				0	•	a de la comercia de l	Click to configure device	
1 1 Rail #1	0.00 0.48	N/A	N/A	N/A	CONTROL; Always Convertin	g			
2 2 Rail #2	0.00 0.48	N/A	N/A	N/A	CONTROL; Always Convertin	g			
🖻 Device: TP553622 @ PMBus Address 90	5d				Θ	-	æ	Click to configure device	
5 1 VCORE	0.00 0.48	N/A	N/A	N/A	CONTROL; Always Convertin	g			
6 2 VDDQ	0.00 0.48	N/A	N/A	N/A	CONTROL; Always Convertin	g			
🖻 Device: TP553622 @ PMBus Address 92	2d				Θ	•	4	Click to configure device	
3 1 Rail #1	0.00 0.48	N/A	N/A	N/A	CONTROL; Always Convertin	g			
4 2 Rail #2	0.00 0.90	N/A	N/A	N/A	CONTROL; Always Convertin	g			
K.		1	H G)	>
	strands of call lands								
Tips & Hints	PMBus Log							l	기만
								F	20
Fusion Digital Power Designer v7.0.9.1 [2017-04-24] USB Adapter v1.0.	10 [PEC; 400 kH	z] 🕹 Tex	AS INSTRUMEN	rs fusion digital power			4	

Figure 10. System View

Config General Advanced Debug	SMBALERT# Mask	All Config								
Command	Code	Value/Edit	Hex/Edit	Command	Code	Value/Edit	Hex/Ec			
▼ Calibration			1	On/Off Configuration						
IOUT_CAL_GAIN_TOTAL	0x38	5.000 🖂 mΩ	0xD140	ON_OFF_CONFIG	0×02	0x17 🖂	0x17			
IOUT_CAL_OFFSET_1	0x39	0.0000 🗸 A	0xE800	OPERATION	0×01	0x00 🖂	0x00			
IOUT_CAL_OFFSET_2	0x39	0.0000 🖂 A	0xE800	TON_DELAY	0x60	0.480 🖂 ms	0xB1E			
IOUT_CAL_OFFSET_3	0x39	0.0000 🗸 A	0xE800	▼ Status						
IOUT_CAL_OFFSET_4	0x39	0.0000 🖂 A	0xE800	MFR_MAX_TEMP_1	0xC0	-40 °C	0x07D8			
IOUT_CAL_OFFSET_5	0x39	0.0000 🗸 A	0xE800	MFR_SPECIFIC_03	0xD3	Phase6: 🗸	0x0000			
IOUT_CAL_OFFSET_6	0x39	0.0000 🗸 A	0xE800	MFR_SPECIFIC_05	0xD5	PMBR_V 🖂	0x00			
VOUT_SCALE_LOOP	0x29	1.000 🖂	0xE808	MFR_SPECIFIC_08	0xD8	CF_CPU 🖂	0x00			
VOUT_SCALE_MONITOR	0x2A	1.000 🖂	0xE808	READ_IIN	0x89	0.00 A	0x000			
▼ Configuration			Ĵ	READ_IOUT_1	0x8C	0.00 A	0x000			
FREQUENCY_SWITCH	0x33	600 🗸 kHz	0x0258	READ_IOUT_2	0x8C	0.00 A	0x0000			
IC_DEVICE_ID	0xAD	0x78 🗸	0x78 🗸	READ_IOUT_3	0x8C	0.00 A	0x000			
IC_DEVICE_REV	0×AE	0x01 🖂	0x01 🗸	READ_IOUT_4	0x8C	0.00 A	0x000			
MFR_SPECIFIC_00	0xD0	OCL:62 🖂	0x3C	READ_IOUT_5	0x8C	0.00 A	0x000			
MFR_SPECIFIC_01	0xD1	IOUT_G,	0x00	READ_IOUT_6	0x8C	0.00 A	0x000			
MFR_SPECIFIC_02	0xD2	SVID_P 🗸	0x02	READ_IOUT_ALL	0x8C	0.00 A	0x000			
MFR_SPECIFIC_04	0xD4	0.001 V	0x9014	READ_IOUT_ALL	0x8C	0.00 A	0x000			
MFR_SPECIFIC_06	0xD6	NVM_TS 🗸	0x05	READ_PIN	0x97	0.00 W	0x000			
MFR_SPECIFIC_07	0xD7	AC_LL:0 🖂	0x0B	READ_POUT	0×96	0.00 W	0x0000			
MFR_SPECIFIC_09	0xD9	USR2:7; 😒	0x7F66	READ_TEMPERATURE_1	0x8D	-74 °C	0xEDA			
MFR_SPECIFIC_10	0xDA	IIN_MAX 🖂	0xC8E4	READ_VIN	0x88	0.012 V	0x986-			
			U)				
Tips & Hints			PMBus Log							
READ_IOUT_3 [0x8C,Rail #1]	- Math		16:40:39.676: TPS53678 @ 96d: MFR_SPECIFIC_08 [0xD8]: wrote CF_CPU:No Fault [000b]; CF_SAR:OTF [101b] [0x							
Measured output current in amperes. Phase	se 3 Current.	0	to RAM 16:40:40.217:	TPS53678 @ 96d: MFR_SPECIFIC_08 [0xD	8]: wrote CF_CPU:N	o Fault [000b]; CF_SA	R:No Fau			
1			[0x00] to RAM							
-		\sim								

Use of TI Programming Tools

Figure 11. Settings for Individual Device

6. Click on Write to Hardware to set write the values to the device

Configuration TPS5	3622 @ PMB
File Device Tools	
🖞 Write to Hardware 🗌	C Discard Cha
Configure	General
	Mode 9
	SVID_A

Figure 12. Write to Hardware

- 7. Click Store Config. to NVM to store the NVM settings.
- 8. Click Device Menu
- 9. Click Refresh All Parameters to update USER_DATA commands in the GUI.

Texas

www.ti.com

RUMENTS



Use of TI Programming Tools

10. Select **Save Project As** from the **File** menu to export a *Project* file for a single device.

-\$ Co	nfigu	ation TP553622 @ PMBus Address 96d (60h) / 12				
File	Devio	Tools	_				
*⊇ w		Rail Dashboard	to NVM				
Conf		Store Configuration to Flash Memory	Transi				
		Restore Configuration from Flash Memory					
		Refresh All Parameters					
		Polling Status	3'h0E sho 3'h0F sho				
) Both	0Eh and (
		Respond	led with A				

Figure 13. Refresh All Parameters

- 11. Repeat steps 5 through 14 for each device.
- 12. To set NVM values for another device, return to the **System View** page and **Click to Configure Device**.

🛛 Texas Instruments - Fusion Digital Power Designer [System View]									
File Tools Debug Help									
Q Scan for Device (Device_ID Device_ID Device_ID	vice_Code IC_Device_ID) 🔒 🛞 Build Syste	m System Moni	tor 🛛 🔚 Save	🗸 Auto Write 🛛 🔴 Sto	op Polli	ing		
Power Rails Tree									0
# 🛆 Rail # 🛆 Rail Name	Vout	On Delay R	ise Off Delay	Fall D	ependencies (Direct Only))			
E Device: TP553622 @ PME	Bus Address 88d				. \varTheta	-	A	Click to configure device	
1 1 Rail #1	0.00	0.48 N	A N/A	N/A C	ONTROL; Always Convertin	ng			
2 2 Rail #2	0.00	0.48 N/	A N/A	N/A C	ONTROL; Always Convertin	ng			
🕀 Device: TP553622 @ PMBu	is Address 96d				Θ	-	4	Click to configure device	
5 1 VCORE	0.00	0.48 N/	A N/A	N/A C	ONTROL; Always Convertin	ng	1		60
6 2 VDDQ	0.00	0.48 N/	A N/A	N/A C	ONTROL; Always Convertin	ng		10	
🕒 Device: TP553622 @ PMBi	us Address 92d				Θ	-	4	Click to configure device	
3 1 Rail #1	0.00	0,48 N	A N/A	N/A G	ONTROL; Always Convertin	ng		()	
4 2 Rail #2	0.00	0.90 Nj	A N/A	N/A C	ONTROL; Always Convertin	ng			
<[(dill)						
, Constitution									70-121
Tips & Hints	PMBus Log	1							• •
	<u>^</u>								
	\sim								
	Final Action of the second sec								6 8
Fusion Digital Power Designer v7.0.1	9.1 [2017-04-24] USB Ad	lapter v1.0.10 [PEC	; 400 kHz] 👋 Tex	KAS INSTRUMENTS	fusion digital power				

Figure 14. Switch between Multiple Devices

13. To save a *System* file after all devices are configured, return to the **System View** click the **File** menu and select, **Save System File As**.





Figure 15. Save a System File

3.1.2 View a Project File or System File Offline

When the software operates in offline mode, you cannot update any NVM settings. Use offline mode only to view *System* files.

WARNING

Fusion Digital Power Designer offline mode is for viewing of files only. To edit NVM settings, use online mode, which requires a controller device to be connected. System files that users edit offline may not contain the correct USER_DATA and checksum values stored in the System file, and can lead to unpredictable results.

- 1. Open Fusion Digital Power Designer in offline mode.
- 2. Select Create offline system by opening existing system file.
- 3. Click Next



Use of TI Programming Tools

www.ti.com

🖗 Define System
Create new Offline system from start You will be able to design a new system with multiple devices selected from either a list of supported devices or previous created project files (.xml). The system can be saved to a file with ".th"p" extension Create new Offline system by opening existing system file You have previously defined the system (.th"sp) and would like to open it in Offline mode Modify existing system Modify your current system
Cancel Previous Next

Figure 16. Select System File

4. Browse to the file when prompted, and click Next.

🜵 Define System				_ 🗆 🔀						
Manually locate a system file above, or select a recently used system files below.										
Project File: C:\temp\inputf	Select File									
Part ID	Address	Note		Δ						
TPS53622	88d (0x58)									
TPS53622	92d (0x5C)									
TPS53622	96d (0x60)									
Last Used	⊽ File			Num Devices						
Cancel			Previous	Finish						

Figure 17. Select Offline System File



5. Click **Finish** to open the System file to review.

V Texas Instrume	ents - Fusion Digital	Power Designer	[System V	iew]					_ • ×
Q Scan for Device (D	vevice_ID Device_Cod	le IC_Device_ID)	🛞 Build S	ystem	System Monit	tor 🔛	Save 🗸 Auto Write	Start Polling	
Power Rails Tree									
# 🛆 Rail # 🛆	Rail Name	Vout	On Delay	Rise	Off Delay	Fall	Dependencies (Direct On	ly)	
Device: TP553	3622 @ PMBus Add	ress 88d [Offline]					0	-	Click to configure device
11	Rail #1	1.00	0.48	N/A	N/A	N/A	CONTROL; Always Conve	rting	
2 2	Rail #2	1.20	0.48	N/A	N/A	N/A	CONTROL; Always Conve	rting	
Device: TP553	622 @ PMBus Addr	ess 96d [Offline]					Θ	-	Click to configure device
5 1	VCORE	1.00	0.48	N/A	N/A	N/A	CONTROL; Always Conve	rting	
6 2	VDDQ	1.20	0.48	N/A	N/A	N/A	CONTROL; Always Conve	rting	
Device: TP553	622 @ PMBus Addr	ess 92d [Offline]					Θ	-	Click to configure device
3 1	Rail #1	1,70	0,48	N/A	N/A	N/A	CONTROL; Always Conve	rting	
4 2	Rail #2	0.90	0.90	N/A	N/A	N/A	CONTROL; Always Conve	rting	
<u>(<((</u>		III							(4(
L	1								
Tips & Hints	PI	MBus Log							
									~
-	<u>18</u>								E
Fusion Digital Power D	Designer v7.0.9.1 [2017	-04-24] inputfile.til	sp 🐶 Tex	AS INSTRU	MENTS fusion	digital po	wer		

Figure 18. Offline System View

3.1.3 Import a *Project* File to a Single Online Device

- 1. Connect the USB-GPIO to a board.
- 2. Apply 3.3 V to the V3P3 pin of the controller device.
- 3. Start the Fusion Digital Power Designer software.
- 4. From the System View window, Click to configure device
- 5. From the File menu, select Import to device
- 6. Select Project File
- 7. Click Next.



Use of TI Programming Tools

👆 Configuration TP553622 @ PMBus Address 96d (60h) / I2C Address 192d											
File Device Tools											
Save Project As	scard Changes Store Config to NVM 🙆 Restore N										
Import to device	Seneral Static Telemetry Transients Protection										
Export	Mode Settings										
	SVID_ALL_CALL: Only 8h0E should be responded Only 8h0F should be responded Both 0Eh and 0Fh should be resp Responded with ACK for SVID's all ca										
	PS_APH: One phase active in ps1 mode Two phase active is ps1 mode										

Figure 19. Import to Devcie

🕴 Fusion Digital Power Designer Device Configuration Import
Select Device Import Type
Select the type of file to import:
Project File A project file contains a device's PMBus configuration in XML format. It is the primary data file of the Fusion GUI, and allows you to edit a device's configuration and design in 'offline' mode: without a live connection to your device. A project file also contains information about your design and sequencing that is not stored on the device. For example, a definition of the voltage divider network on your system board.
In offline mode, project import is different than project open. Project import allows you to extract information from another project file into the project you current have open and are editing. This differs from when you open a project file in offline mode, which simply closes the project file you are editing and opens a new one.
O Data Flash
This is only available for devices that support direct programming of a device configuration to data flash. UCD92XX and UCD90XX devices support this through JTAG and SMBus in ROM mode. ROM mode is used in the Fusion GUI, since this can be done using the standard USB adapter and SMBus ROM commands. The checksums present in the EEPROM export files supported Motorola S-Record and Intel Hex will be validated during this import. Note that data flash exports are generally only good for a single firmware release. The GUI will warn you try to import a data flash that was exported for a different firmware release than the one currently loaded on your current device.
O PMBus, SMBus, or I2C Script
A full device configuration will be imported in terms of a series of SMBus or I2C writes to the device. You would normally not use this format to import/export your configuration through the Fusion GUI. This format is useful when you need to configure devices in a manufacturing environment or to assist in the development of host controller software.
SMBus is a higher level command protocol that sits on top of I2C. SMBus scripts can be easily read by a human. I2C scripts are more difficult to read because of how the device address and optional PEC byte gets encoded into the I2C write.
Note: I2C import via GUI ignores address and PEC bytes in import file: current device address (88d) and SMBus adapter PEC setting (PEC) will be used.
< Prev Next > Cancel

Figure 20. Import Project File



ł	Fusion Digital Power Designer Device Configuration Import												
	Select Project File												
		C .) +	-)	-									
	Project File:	C:\temp	o projectnie.xn		and which Maria has a section of	Select File							
	Manually loca	ite a proj	ject file above,	or select a recently used project file below. When select	ted, click Next to continue.								
	Last Used	V	Device	File	Directory								
			_										
	Clear Rece	nt File Lis	st		< Prev Next >	Cancel							

Figure 21. Select Project File

- 8. Click Select All to import all parameters of the device.
- 9. Click Write Checked.

18

Use of TI Programming Tools

🖗 Fusion Digital Power Designer Device Configuration Import

Review Parameters to Import

The following table summarizes any differences between the parameters in your project file and the values currently in memory on the device. By default, only modified parameters will be written out. You can chose, however, to skip certain parameters by unchecking it's checkbox. When ready, click the "Write" button to write checked parameters to the device.

Import	Parameter	Δ	Updated \triangledown	Device Value	Device Hex	New Value	New Hex	ŀ
	ON_OFF_CONFIG [0x02,Rail #1]		Yes	Mode: CONTROL	0x17	Mode: CONTROL	0x17	
	USER_DATA_00 [0xB0,Rail #1]		Yes	3329020030F3	0x3329020030F3	3329020030F3	0x3329020030F3	1
~	USER_DATA_01 [0xB1,Rail #1]		Yes	00000000040	0x00000000040	00000000040	0x00000000040	1
1	USER_DATA_02 [0xB2,Rail #1]		Yes	890000000D0	0x8900000000D0	890000000D0	0x8900000000D0	1
~	USER_DATA_03 [0xB3,Rail #1]		Yes	0400100A0080	0x0400100A0080	0400100A0080	0x0400100A0080	1
1	USER_DATA_04 [0xB4,Rail #1]		Yes	A45EC7F7C777	0xA45EC7F7C777	A45EC7F7C777	0xA45EC7F7C777	ŀ.
~	USER_DATA_05 [0xB5,Rail #1]		Yes	1E32C5978F3C	0x1E32C5978F3C	1E32C5978F3C	0x1E32C597BF3C	
~	USER_DATA_06 [0x86,Rail #1]		Yes	E5DB13104801	0xE5DB13104801	E5DB13104801	0xE5DB13104801	
~	USER_DATA_07 [0xB7,Rail #1]		Yes	80FF09205FFF	0x80FF09205FFF	80FF09205FFF	0x80FF09205FFF	1
1	USER_DATA_08 [0xB8,Rail #1]		Yes	0002000080BB	0x0002000080BB	0002000080BB	0x0002000080BB	1
~	USER_DATA_09 [0x89,Rail #1]		Yes	0025E0838080	0x0025E0838080	0025E0838080	0x0025E0838080	1
1	USER_DATA_10 [0xBA,Rail #1]		Yes	0A6028C0410D	0x0A6028C0410D	0A6028C0410D	0x0A6028C0410D	
-	USER_DATA_11 [0xBB,Rail #1]		Yes	00220000A021	0x00220000A021	00220000A021	0x00220000A021	
	USER_DATA_12 [0xBC,Rail #1]		Yes	80901200F801	0x80901200F801	80901200F801	0x80901200F801	1
~	VOUT_COMMAND [0x21,Rail #1]		Yes	1.000 V	0x0097	1.000 V	0x0097	
1	VOUT_COMMAND [0x21,Rail #2]		Yes	1,200 V	0x00BF	1.200 V	0x00BF	

Figure 22. Select All and Write Checked Parameters

Copyright © 2018, Texas Instruments Incorporated



www.ti.com

_ 🗆 🔀



		 -
inished		
rite Log:	Error Log:	
0x0400100A0080] to RAM Wrote USER_DATA_04 [0xB4,Rail #1] A45EC7F7C777 [0xA45EC7F7C777] to RAM Wrote USER_DATA_05 [0xB5,Rail #1] 1E32C597BF3C [0x1E32C597BF3C] to RAM Wrote USER_DATA_06 [0x86,Rail #1] E5DB13104801 [0xE5DB13104801] to RAM Wrote USER_DATA_07 [0xB7,Rail #1] 80FF09205FFF [0x80FF09205FFF] to RAM Wrote USER_DATA_08 [0x88,Rail #1] 0002000080BB [0x0002000080BB] to RAM Wrote USER_DATA_09 [0x89,Rail #1] 0025E0838080 [0x0025E0838080] to RAM Wrote USER_DATA_09 [0x89,Rail #1] 0025E0838080 [0x0025E0838080] to RAM Wrote USER_DATA_10 [0xBA,Rail #1] 00220000A021 [0x0A6028C0410D] to RAM Wrote USER_DATA_11 [0xB8,Rail #1] 00220000A021 [0x002000004021] to RAM Wrote USER_DATA_12 [0xBC,Rail #1] 80901200F801 [0x80901200F801] to RAM Wrote VOUT_MARGIN_HIGH [0x25,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_LOW [0x26,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_HIGH [0x25,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_HIGH [0x25,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_LOW [0x26,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_HIGH [0x25,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_HIGH [0x25,Rail #1] 0.000 V [0x0000] to RAM Wrote VOUT_MARGIN_HIGH [0x25,Rail #2] 0.000 V [0x0000] to RAM		

Figure 23. Successful Project File Import

10. When the *Project* file import finishes, the software **does not automatically issue an NVM store operation** command. You must click **Store Config to NVM** to update NVM settings after you import a *Project* file.

File Device Tools			
₩ Write to Hardware	Discard Changes	Store Config to NVM	🔓 Restore NVM Conf
Configure	General Stati	c Telemetry Transie	ents Protection SMBA
	Mode Settin	gs LL: Only 8'h0E shou Only 8'h0F shou Both 0Eh and 0	uld be responded uld be responded Fh should be responded



3.1.4 Import a System File to an Online System

- 1. Connect the USB-GPIO interface to the board
- 2. Apply 3.3 V to the V3P3 pins of the controller devices.

- 3. Start the Fusion Digital Power Designer software.
- 4. From File menu on the System View page, select Import System File...

фт	Texas Instruments - Fusion Digital Power Designer [System View]								
File	Tools Debug Help								
	Import System File	evice_Code IC_Dev	/ice_ID)	🛞 Build Sy	stem \$	System Monito			
	Save System File								
	Save System File As	-	Vout	On Delay	Rise	Off Delay			
	Preferences	Bus Address 88d							
	Exit		0.00	0.48	N/A	N/A			
	2 2 Rail #2	-2	0.00	0.48	N/A	N/A			
	Device: TP553622 @ PM	Bus Address 96d							
	r + VCOPE		0.00	0.40	N1/A	51/A			

Figure 25. Import System File

- 5. The default setting is that all of the devices are selected but none of the parameters are selected. Check the box in the **Import** column for each device.
- 6. Click **Check All** to review all stored parameters.



👆 Imp	ort System File						
Syster	n File: C:\temp\systemfile.tifsp						Browse
	2 <u>1</u>						
		- P		P.	1	1	Providence
mport	Devices Online		gurations	Import Order	Need GUI restart	Clear Log when done	Store to Flash when done
	TPS53622 @ PMBus Address 96d Yes	TPS53	622 @ Address 96d		0 No		M
1	TPS53622 @ PMBus Address 92d Yes	TPS53	678 @ Address 92d		0 No	M	
	TPS53622 @ PMBus Address 88d Yes	TPS53	658 @ Address 88d		0 No	M	
	10 M.						
Sho	w All Devices O Show Devices having	ng configuratio	on in file Only				
Review	w Parameters to Import for TPS53622	@ PMBus A	ddress 88d				
Import	Parameter	Updated ⊽	Device Value	Device Hex	New	Value	New Hex
	IOUT_CAL_OFFSET_TOTAL [0x39,Rail	No	0.000 A	0xE800	0.000	D A	0xE800
	IOUT_CAL_GAIN_TOTAL [0x38,Rail #2]	No	5.000 mΩ	0xD140	5.000	DmΩ	0xD140
	IOUT_CAL_OFFSET_1 [0x39,Rail #2]	No	0.000 A	0xE800	0.000	DA	0xE800
	VOUT_TRANSITION_RATE [0x27,Rail #2]	No	20.0000 mV/µs	0xE140	20.00	000 mV/µs	0xE140
	VOUT_SCALE_MONITOR [0x2A,Rail #2]	No	1.000	0xE808	1,000	0	0xE808
V	VOUT_SCALE_LOOP [0x29,Rail #2]	No	1.000	0xE808	1.000	0	0xE808
Sho	w All Parameters O Show Update Paran	neters Only					Check All Uncheck All
2000							
L							
1							
Timesta	mn Message						
Timeste	mp messege						
C					(
Cop	y Log Clear Log					Close] [Import]

Figure 26. Check All Parameters

- 7. Click Import.
- 8. Review the messages to make sure each device program is correctly. Imported *System* files contain an NVM store operation for each device by default. They need no separate NVM store command.

Timestamp	Message	4
11:52:51.321	Wrote VOUT_MAX 3.040 V [0x00FF] to RAM	
11:52:51.321	Wrote VOUT_COMMAND 2.560 V [0x00CF] to RAM	
11:52:51.321	Wrote VOUT_MARGIN_HIGH 0.000 V [0x0000] to RAM	
11:52:51.336	Wrote VOUT_MARGIN_LOW 0.000 V [0x0000] to RAM	
11:52:51.336	Wrote VOLT_MIN 0.000 V [0x0000] to RAM	
11:52:51.336	Saving device configuration (RAM) to flash memory	
11:52:52.116	Sucessfully importing configuration to TPS53622 @ PMBus Address 96d!	9
		2
Copy Log	Clear Log Close Import)

Figure 27. System File Import Messages



3.1.5 Export CSV Script for Third Party Tools

The *Fusion Digital Power Designer* software also exports configuration data to a comma-separated text file (.csv). Third-party software or programming vendors usually require the .csv format because it is easy to adapt to existing software.

- 1. To export data from a single device, click **Click to Configure Device** from the **System View**.
- 2. From the File menu, select Export...



Figure 28. Device File Export

- 3. From the Device Export page, click PMBus Programmer Script tab.
- 4. Use the settings shown in Figure 29 for most situations. Third-party software sometimes has other requirements.



Description The aller to a strike second star by the your current configuration to a device. Write and does in terms of standard SMBus commands (WriteByde, WriteWord, and WriteBidd). This can be easily the all provide the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the profession of the star between the second star by the star	Project File PMBus Programme	rammer Script				
Configuration Validation Subjection validation Configuration validation Configuration and verify it matches what was programmed. The script is oriented at fur for party programmers who can apport resetting the device after the configuration is written to non-volatile memory. Do not validate configuration Programming options Conf validate configuration Programming options Programming Programming options Programming options Programming options Programming options Programming Programming o	guration to a device. Writes are done in terms of standard SMBus commands (WriteByte, WriteWord, and Writ	nd WriteBlock). This can be easily				
Out not valuate commany uration Out on transact commany urates Programming options In ura dia of the fore programming. It selected, all rails on the target device will be turn off throsh ywriting ON OFF_CONFIG command (Code = 01) with value = 00h (OPERATION Only), followed by writing OPERATION command (Code = 01h) with value = 00h (OPERATION Only), followed by writing OPERATION command (Code = 01h) with value = 00h (OPERATION Code), followed by writing OPERATION command (Code = 01h) with value = 00h (OPERATION Code), followed by writing OPERATION command (Code = 01h) with value = 00h (OPERATION Code), followed by writing OPERATION command (Code = 01h) with value = 00h (OPERATION Code), followed by writing OPERATION command (Code = 01h) with value = 00h (OPERATION Code), followed by writing OPERATION code and added to the script. Your program can verify if the cleave in a faster the STORE_DEFAULT_ALL command is sent before continue to the next step. If the cleave in the script. Add IPEC byte Add IPEC byte Output Destination WRF_SERIAL stored in the script. Store Default Train to device and verify the value against the MRF_SERIAL stored in the script. Store Default Train to device and verify the value against the MRF_SERIAL stored in the script. Output Destination utput Folder: Chremp Stelect Elevanta I If sector, 322 4.0 Address (DA) (EF). (EXT) Reset to Default Fleeanta I review: TPSS3622 4.0 Address 88 PMBus Programmer Script.csv Elevanta I	the e script is the device Validation Validate command writes The script will read back commands after they are written to write will read back commands after they are written to write NACK on write.	n to verify writes. You should t have the capability to check for				
Programming options Other options □ Turn rais off before programming The delytice PMbus address Some devices require rais to be off before programming. If selected, all rais on the target device will be turn off first by writing OPCONFIG command (Code=Oth) with value = 00h (DerpetTion Only), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command (Code=Oth) with value = 00h (DerpetTion Donly), followed by writing OPREATION command is sent before continue to the next step □ Add PEC byte □ Add PEC byte □ Add PEC byte □ Add PEC byte □ Dutput Destination □ Dutput Destination utput Folder: C/temp Select (E ename: (Ph) (DV) Address (DA) (EF), (EXT) Reset to Default Filenama T eview: TPS53622 4.0 Address 88 PMBus Programmer Scri	O Do not validate command writes					
Code=0 Lh) with value = 00h (Immediate Off) to all rails ✓ Ad UC_DEVICE_DD (0xAD) to script If Selected, IC_DEVICE_DD (0xAD) matches the one in the script before continue with programming. ✓ Ad UPR_SERIAL valiation to script Iso ected, after programming, and configuration is stored to flash, your program will read back the MFR_SERIAL from the device and verify the value against the MFR_SERIAL stored in the script. Output Destination urger Volume urger Volume Volume Iso ected, after programming, and configuration is stored to flash, your program will read back the MFR_SERIAL from the device and verify the value against the MFR_SERIAL stored in the script. Output Destination urger urger Phylop (DV) Address (DA) (EF). (EXT) even wit TPS53622 4.0 Address 88 PMBus Programmer Script.csv	Other options Image: Construction option options Image: Construction option options Image: Construction option option option Image: Construction option option Image: Construction option option Image: Construction opting to thetttint	Other options Including Device PMbus address If select, script will include device address in the second column Store Default Timing 1,000 ms				
Output Destination utput Folder: C:\temp Select (E Ilename: {PN} {DV} Address {DA} {EF}.{EXT} Reset to Default Filename T review: TPS53622 4.0 Address 88 PMBus Programmer Script.csv Edag	sh, your program will against the					
utput Destination utput Folder: [C:\temp Select] [Ename: [PN] (DV) Address (DA) {EF}.{EXT} Reset to Default Filename T review: TPS53622 4.0 Address 88 PMBus Programmer Script.csv Log						
Import outer Select Ester E		Salart Proven				
eview: TPS53622 4.0 Address 88 PMBus Programmer Script.csv .og	1	Bernt to Default				
.og	SV	Neser to Detaur				

Figure 29. Recommended CSV Export Settings

- 5. After you complete the settings, click Export PMBus Programmer Script .
- 6. The system generates a text file, and saves it to the path you select as the output folder.

3.1.6 Compare Two Project Files

This procedure applies to *Project* files created in *Fusion Digital Power Designer 7.0* and later. This software compares *Project* files only. It does not compare *Fusion Manufacturing* scripts.

- 1. From the **System View**, click on a device name, and click **Configure Device**.
- 2. From the Tools menu, select Device/Project Configuration Compare

-v -comigar							
File Device	Tools						
1 Write to H	in [Device/Project Configuration Compare					
Configure	1	Numeric Encode/Decode Tester					

Figure 30. Device/Project Configuration Compare



Use of TI Programming Tools

3. Select two or more *Project* files to compare.

🜵 Project / Device Comparison Tool	_ 🗆 🔀
Devices / Project Files You have not specified any project files or devices to compare; click the Add link below. Add project file(s) · dd attached devices (offline or online) · ClearList · Force Report Refresh Options	
Group By: Category Rail Show: Category Cmd Code Show Hex Values: Inline w/ Decoded Separate Column Image: Column on the column on	
Command Code Rail	

Figure 31. Compare Project Files

4. The Results window shows the differences between the commands in a column for each device.

Devices / Project Files TPS53681@88 TPS53622@96 Add project file(s) - Add attact	Project: Ti Project: Ti hed devices (c	PS53681 @ PS53622 @ offline or on	PMBus Address 88d Project.xml PMBus Address 96d Project.xml line) • <u>ClearList</u> • <u>Force Report Refresh</u>	<u>Mor</u> Mor
Options roup By: Category Ra	il Show:	Gategory	Cmd Code Show Hex Values: O Inline w/	/ Decoded O Separate Column
Only show differences (diffe	erent values o	r missing co	mmands) 🔽 Auto size cells to fit text Show Only: 🔿 Read-Or	nly Commands 🔘 Writable Commands 💿 Both
ommand	Code	Rail	TPS53681@88	TPS53622@96
MFR_SPECIFIC_09	0xD9	Rail #1	USR2:7; OSR:50 mV [00b]; BLANK_TIME_RISING:54 ns [011b]; PH1_USR1:False; USR_OSR_HYS:5 mV [00b]; USR1:0 mV [111b];	USR2:15; OSR:50 mV [00b]; BLANK_TIME_RISING:54 ns [011b]; PH1_USR1:False; USR_OSR_HYS:5 mV [00b]; USR1:0 mV [111b];
MFR_SPECIFIC_10	0xDA	Rail #1	IIN_MAX:[0xC8]; IOUT_MAX:85 A	IIN_MAX:[0x32]; IOUT_MAX:30 A
MFR_SPECIFIC_11	0xDB	Rail #1	VBOOT:0xBF	VBOOT:0x97
MFR_SPECIFIC_12	0xDC	Rail #1	IIN_SHUNT:0.25 mΩ [01b]; IIN Option:Sensed IIN [1b]; Calulated IIN Option:Include 2nd rail [0b]; DYN_INT_LEVEL:0.000 mV [111b]; DYN_INT_LEVEL:1.000 us [0000b]; TMAX:90.000 oC [0x0]	IIN_SHUNT:0.25 mΩ [01b]; IIN Option:Sensed IIN [1b]; Calulated IIN Option:Indude 2nd rail [0b]; DYN_INT_LEVEL:0.000 mV [111b]; DYN_INT_LEVEL:1.000 us [0000b]; TMAX:90.000 oC [0x0]
MFR_SPECIFIC_13	0xDD	Rail #1	NVM_PIN_OP_WARN: True; NVM_BLK_TAOLOW: False; NVM_SA_2PH: True; NVM_OT_FAULT_LIMIT: False; NVM_TAOLOW_TH_SEL: True; VBOOT_SR: True; SVID_ALL_CALL: 0x1 [Only 8h0F should be responded with ACK for SVID's all call address]; PS_APH: 0xFalse [One phase active in ps1/2 mode]; IGNORE_PS: 0xTrue [Ignore SetPS commands from CPU]; VR_MODE: 0x7 [VR13.0 with SmV step];	NVM_PIN_OP_WARN: True; NVM_BLK_TAOLOW: False; NVM_SA_2PH: True; NVM_OT_FAULT_LIMIT: False; NVM_TAOLOW_TH_SEL: True; VBOOT_SR: True; SVID_ALL_CALL: 0x1 [Only 81:0F should be responded with ACK for SVID's all call address]; PS_APH: 0xFalse [One phase active in ps1/2 mode]; IGNORE_PS: 0xTrue [Ignore SetPS commands from CPU]; VR_MODE: 0x7 [VR13.0 with SmV step];
		and the second	DRS CTOP FINE FALL & DRS FTOA FINE FALL &	DDG CTOP FINE FALL & DDG FTOX FINE FALL &

Figure 32. Project File Compare

3.1.7 Build a System File from Existing Project Files

New projects often share components with previous projects. With the *Fusion Digital Power Designer* software, designers can assemble a *System* (.tifsp) file using controller devices from existing *Project* files or *System* files. Individual projects must not be modified offline. However you can safely assemble existing *Project* files into a *System* file when the software operates in offline mode.

- 1. Start Fusion Digital Power Designer in offline mode.
- 2. Open an existing System file.
- 3. From the System View, select Build System.



Figure 33. Build System

4. From the Define System window, select Modify an existing system.

🕹 Define System
O Create new Offline system from start
You will be able to design a new system with multiple devices selected from either a list of supported devices or previous created project files (.xml). The system can be saved to a file with ".tifsp" extension
O Create new Offline system by opening existing system file (.tifsp) or project file (.xml)
You have previously defined the system (.tifsp) or a single device configuration (.xml) and would like to open it in Offline mode
Modify existing system
Modify your current system
Cancel

Figure 34. Modify Existing System

- 5. Click Next.
- 6. The software displays the selected *Project* file names, the components and the addresses.
- 7. Click **Remove** to delete a device if needed.



Use of TI Programming Tools

www.ti.com

5	efine Syste Select devi	em ce from GUI support list:					Ċ	Select Device
(If you canno	t find the device you are looking for on th	e list, contact Texa	as Instruments sa	le representative)			
*	select devi	ce from GUI sample						Select Sample
F	From file:							Select File
	File Name		Part ID	Existing	Selected		Import	Note
			TPS53659	92		Remove	0	
	TPS53659	@ PMBus Address 105d Project.xml	TPS53659	105		Remove	0	
	TPS53622	@ PMBus Address 88d Project.xml	TPS53622	107		Remove	0	

Figure 35. Existing Project File Devices

- 8. To add a *Project* file, click **Select File**.
- 9. Browse to the *Project* file.
- 10. Update the PMBus address if necessary by clicking in the Selected Addresses cell.
- 11. Repeat steps 8 through 10 to add more *Project* files to the system.
- 12. Click Finish.

Texas

TRUMENTS

ж і	Jefine System							
	Select device from GUI support list: If you cannot find the device you are looking for on the lis	t, contact Texa	as Instruments sa	ile representat	tive)			Select Device
	Select device from GUI sample						C	Select Sample
	From file: C:\temp\TPS53679 @ PMBus Addre	ess 96d Projec	t.xml					Select File
~	File Name	Part ID	Existing	Selected	1	• • • • • • • •	Import Order	Note
5		TPS53659	92	Address		Remove	0	1
8	TPS53659 @ PMBus Address 105d Project xml	TPS53659	105		1	Remove	0	
9	TPS53622 @ PMBus Address 88d Project xml	TPS53622	107		1	Remove	0	
4	C:\temp\TPS53679 @ PMBus Address 96d Projec	TPS53679	96	96	•	Remove	0	offline device
				30 99 100 101 102 103 104 106 108 109 110 111 112 110 111 112 113 114 115 116	-			
(Cancel			118 119 120 121 122 123 124		Previous		Finish

Figure 36. Add a Project File to a System File

3.2 Fusion Manufacturing Tool

3.2.1 Import a System File

Use the *Fusion Manufacturing Tool* in mass-production environments to import a *System* file (.tifsp) to an online board.

To download the Fusion Manufacturing Tool, click here: http://www.ti.com/tool/FUSION_MFR_GUI..

- 1. Build a Fusion System File (.tifsp) using Fusion Digital Power Designer
- 2. Apply power to the target board
- 3. Connect the USB-to-GPIO interface adapter
- 4. Start the Fusion Manufacturing Tool
- 5. Click the Load Script tab.
- 6. Click **Browse** to select a .tifsp file.



Use of TI Programming Tools

www.ti.com

🜵 Digital Power Manufacturing App	_ 0 🔀					
File Advanced Help						
Edit Script 🗗 Load Script 🌛 Run Script						
Select Factory script (.xml file generated by MFR GUI) or system file (.tifsp generated by Fusion GUI) that you wish to use for manufacturing and click "Load".						
	Browse					
	beal Is					
	LUdu					

Figure 37. Browse for System File

- 7. After the software loads the System file, it shows a list of available devices.
- 8. Review this list to make sure the correct System files are selected.
- 9. Click Load Script.

🔋 Digital Power Manufacturing App								
File Advanced Help								
😹 Edit Script 💣 Lo	ad Script 🔊 Run Script							
Select Factory script (.xml file o	enerated by MFR GUI) or system file (.t/fsp generated by Fusion GUI) that you wish to use for manufacturing and click "Load".							
C:\temp\mysystemfile	tifsp	Browse						
System File File File								
Factory Script Version	1.0.3							
Description								
Number of devices	3							
Bus Mode	Serial							
Bus Speed	Speed400KHz							
Packet Error Checking	True							
		🕫 Load						

Figure 38. Load the Script

- 10. Click **Scan** to ensure that the USB-to-GPIO interface adapter recognizes each device in the *System* file.
- 11. Click Start to upload the file to the target board.

🖞 Digital Power Manufacturing Tool								
File Advanced Help								
🕞 Edit Script 📄 Load Script 🎐 Run Script								
🗉 🕼 System File	Execution Summa	ary Log (l	.ast Run)					
	Drag a column header here to group by that column							
	Time Severity Type ^V Extra				Message			
	04:45:30.727	GENERAL	CUSTOMER	none	Start Scanning for devices			
	04:45:31.116	GENERAL	CUSTOMER	none	Found TPS53622 @ address 88			
	04:45:31.117	GENERAL	CUSTOMER	none	Found TPS53681 @ address 92			
	04:45:31.117	GENERAL	CUSTOMER	none	Found TPS53679 @ address 100			
	04:45:31.117	SUCCESS	CUSTOMER	none	All devices in the script are found. I lick on "Start" button to run the script			
Batch option: Set the number of times to re-run script						-		
Runs Left # 0 (save) (edit)		_] Show PMB	us/USB Adapter/In:	strumentation	Activity Clear Log 🔯 Copy Log ++ 🔤 Email Log ++			
# Total Runs: 0						1		
# Total Pass: # I)evices tasks:							
# Total Fail: # 1	Tasks Pass:				Scan Start Cancel			
# Total Cancel: # 1	Fasks Fail:							
Session start: 16:42 Fri Nov 10								
Operator: ms								
(rave) edit								
Continuous Scan in background every 10 seconds								
Found all devices in the system								

Figure 39. Scan and Start

- 12. The Result Summary dialog box appears. Click OK.
- 13. Disconnect power from the target board.
- 14. Repeat from step 10 for all remaining boards.



File Advanced Help					Handle		
😹 Edit Script 📄 Load Script 🔗 Run Script							
	Execution Summa	ecution Summary Log (Last Run)					
	Time	ne Severity Type ^(*) Extra Message					
	04:46:19.880 04:46:19.881	GENERAL GENERAL	CUSTOMER INTERNAL	none	Factory version: TI Digital Power Manufacturing Tool Version: 7.0.19.0		
	04:46:19.881	GENERAL	INTERNAL	none	Operator: ms		
	04:46:19.898	GENERAL	INTERNAL	none	***************************************		
Batch option: Set the number of times to re-run script Runs Left # p \$\vec{save}\$ edit # Total Runs: 1 (2.794 s) 1st ra # Total Pass: 1 (100 % - Avg: 2.794 s) # Dei # Total Fail: 0 (0 % - Avg: 0.000 s) # Tai # Total Cancel:3 (0 % - Avg: 0.000 s) # Tai Session start: 16:42 Fri Nov 10 Operator: Save @dit	an vices ta sks Fail	un task for T un task for T un task for T	PS53622@88(1): R PS53681@92(2): R PS53679@100(3):	esult = Passed esult = Passed Result = Passe	A. A. A. A. A. A. A. A. A. A.		
Continuous Scan in background every 10 seconds		Copy message to cuppoard					
Manufacturing Passed							

Figure 40. Manufacturing Passed

15.

3.3 TI Programming Board

TI can provide a limited number of socketed programming boards for engineering and pre-production programming use. The programming board allows fast programming of prototyping samples or small production quantities. Boards are available for the unique package and pinout combinations that are supported by TI VR13 controllers. Request a board from your local field representative or sales representative.

This section describes the various connectors and components of the TI socketed programming board, and how to install devices properly to the socket.





Figure 41. Socketed Programming Board

3.3.1 Board and Connector Description

- 1. First 40-pin socket. Place DUT #1 in this socket, with pin 1 in the corner marked with a yellow star symbol.
- 2. 10-pin keyed connector for TI USB-to-GPIO adapter.
- 3. 2-pin connector (J2) for external 3.3V power supply
- 4. 3-position switch (S1). Determines whether the 3.3V power supply is sourced from external connector (J2):
 - Left position: sources 3.3-V power supply from PMBus adapter
 - Middle position: disconnected 3.3-V power
 - Right position: sources 3.3-V power supply from J2
- 5. 2-position switch (test mode selector). Switches between manual power-on and power-off (using S 2), or automatic control.
 - Left position. Automatic control. PMBus CNTL pin controls 3.3-V power applied to DUT #1 and DUT #2.
 - "Right position. Manual control. Switch S2 controls 3.3-V power applied to DUT #1 and DUT #2. a.. b.
- 6. Multiple board male connector (J1). Use J1 and J7 to mate multiple programming boards in series.
- 7. 2-position switch (S2). Toggles power applied to the socketed devices, when manual power control is selected using the test mode selector switch.
 - Up position: power is applied to DUTs (only in Manual mode).
 - Down position: Power is not applied to DUTs (only in Manual mode).
- 8. Second 40-pin socket. Place DUT #2 in this socket, with pin 1 in the corner marked with a yellow star

Use of TI Programming Tools

symbol.

- 9. VREF2_ON. LED illuminates when DUT#2 successfully powers up (VREF voltage increased to approximately 1.7 V)
- 10. Address selection for DUT#2. Changing the jumper position according to the table printed on the silkscreen of the board changes the PMBus address that DUT#2 responds to.
- 11. Multiple board female connector (J7). Use J1 and J7 to mate multiple programming boards in series.
- 12. VREF1_ON. LED illuminates when DUT#2 successfully powers up (VREF voltage increased to approximately 1.7 V)

3.3.2 Placing devices in the 40-pin socket

- 1. Use any of these methods to make sure that power is not applied to the sockets before placing or replacing devices into this socket.
 - Move S1 to the middle position, so that neither the PMBus adaptor nor the J2 connector supplies power
 - Remove the power supply to J2 when S1 is in the right position.
 - Remove the PMBus adaptor connection when S1 is in the left position.
 - Switch S2 to the OFF position when the board operates in Manual mode.
 - Pull the PMBus CNTL pin low when the board operated in Automatic control mode.
- 2. Properly insert the device to match the Pin 1 marking on the board.
- 3. Close and latch the socket lid firmly before applying power to the device.
- 4. Make sure that both VREF_ON LEDs illuminate before you continue NVM programming.

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your noncompliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/stdterms.htm), evaluation

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265 Copyright © 2018, Texas Instruments Incorporated