

#### ABSTRACT

This tutorial guides through the process of using Xilinx Vivado and Vitis development environments along with Texas Instruments supplied custom IP to bring up Serial Peripheral Interface (SPI) and non-timing critical General-Purpose Outputs (GPOs) for Texas Instruments AFE79xx EVM along with the companion LMK series clocking chip, thereby enabling an easier integration of the AFE79xx device into a system design. This guide will demonstrate how to use a Xilinx ZCU102 setup as an example.

## **Table of Contents**

1 Introduction	2
2 Prerequisites	2
3 Typical Bare-Metal Design Flow	
4 Background	4
5 AFE SPI IP Container Pinout	
6 TI AFE SPI IP Container	
7 Create Block Designs With TI AFE SPI IP	
8 Create New Platforms in Vitis	
9 Create New Application Projects in Vitis	
10 Build Application Projects	
11 Configure the AXI GPIO	
11.1 Initializing the GPIO	
11.2 Setting the Direction	
11.3 Setting High or Low for Corresponding Bits	
12 Configure the AXI SPI	20
13 Create Boot Images to Run on SD Card	21
14 Set up and Power on Hardware	
15 Set up ZCU102 Board Interface for VADJ_FMC	
16 Debug Application Projects and Set up Vitis Serial Terminal	
17 Execute the Application	

## **Trademarks**

All trademarks are the property of their respective owners.

1

This user guide is a walk-through of complete hardware and software flow to bring up SPI and GPO with TI supplied AFE SPI IP. The hardware in this case refers to the AFE SPI IP supplied by TI which uses a Microblaze processor along with AXI SPI, AXI GPIO, and other required peripherals. Shielding end-customer from the nuances of this setup is one of the core objectives in packaging these in a single custom IP container.

The specific step-wise objectives are as follows:

- Instantiate TI supplied AFE SPI IP in a Vivado project
- Map the supplied IP's required signals to FPGA IOs
- · Import hardware design and building a new Vitis application project for software development
- · Compile, link, and download C program to processor along with bit file for FPGA

## 2 Prerequisites

For effective use of this documentation, ensure to have the following prerequisites:

- Xilinx Vitis IDE v2020.1.0 (or higher)
- Xilinx Vivado v2020.1.0 (or higher)
- Xilinx FPGA board along with TI AFE EVM
- FPGA bit file download/debug programmer
- USB-UART cable for debug terminal
- TI supplied AFE SPI IP
- TI supplied C-APIs

## Table 2-1. Prerequisites

TI AFE	AFE79xx
Sample Configuration	2T-2R-1FB
Lanes	2 RX lanes (1RX, 1FB) and 2 TX lanes at 5 Gbps
AFE EVM	AFE79xx EVM
FPGA Board	Xilinx ZCU102 EVM



1 Introduction



## **3 Typical Bare-Metal Design Flow**

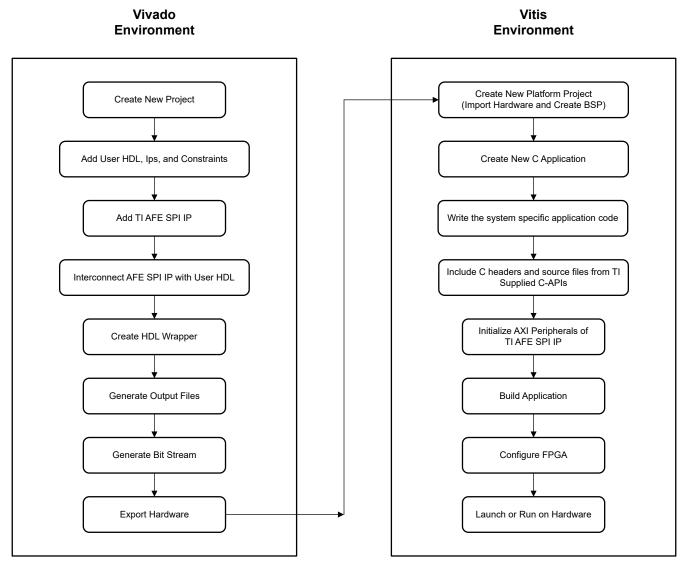


Figure 3-1. Bare-Metal Design Flow

# 4 Background

This example uses a soft core Microblaze because the Microblaze can be instantiated in most of the Xilinx FPGA families. The SPI, UART, and GPIO AXI blocks run on relatively lower frequency AXI clocks. As seen in Figure 4-1, the AXI peripherals are controlled by a Microblaze block through smart interconnects.

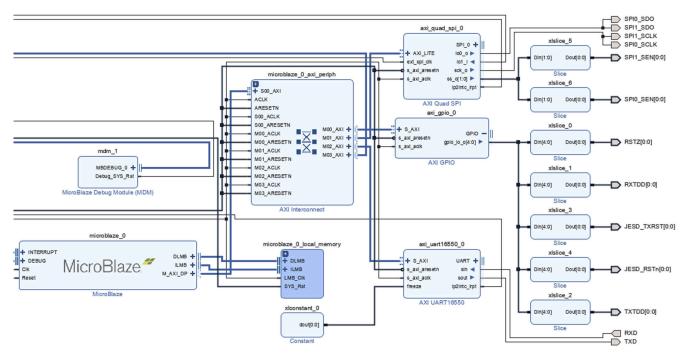
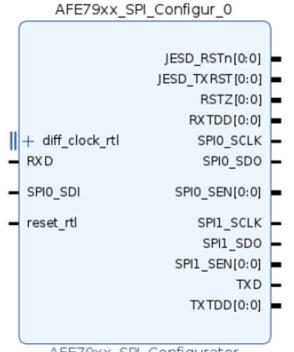


Figure 4-1. Internal Block Diagram of TI AFE SPI IP

The interconnection of various submodules within this IP block diagram is similar to typical Microblaze implementations. The HP port of the Microblaze drives AXI peripherals. The clocking of the entire IP is expected from a 100-MHz differential clock source. This example uses a 100-MHz differential clock source because this clock is typically available as 'user clock' in most FPGA EVMs. All other clock frequencies are derived internally through a clocking wizard.



## **5 AFE SPI IP Container Pinout**



## AFE79xx\_SPI\_Configurator

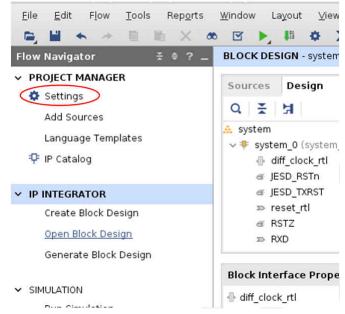
#### Table 5-1. Pinout Signals and Connections

SIGNALS	DIRECTION	EXTERNAL CONNECTIONS	
SPI0_SCLK, SPI0_SDO*, SPI0_SEN	Output	AFE SPI lines (_SDO* to SDI of AFE)	
SPI0_SDI	Input	AFE SDO	
SPI1_SCLK, SPI1_SDO, SPI1_SEN	Output	LMK SPI lines	
RSTZ	Output	RESETn of AFE	
JESD RSTn JESD TXRST	Output	JESED IP Cores RSTn and TX Rst	
RXD	Input	UART Terminal TX for debug	
TXD	Output	UART Terminal RX for debug	
diff_clock_rtl	Input	100-Mhz differential clocking	
Reset_rtl	Input	Reset (Active High) typically connected to FPGA board reset	

## 6 TI AFE SPI IP Container

By default, custom and user-added IPs are not visible in IP catalog of Vivado. The locations for these IPs must be configured manually. To configure the IP locations manually, follow these steps:

1. Open the Project Manager menu and click Settings (see Figure 6-1).



#### Figure 6-1. Project Manager Settings

 Click the + icon to add the location of TI provided IP (see Figure 6-2). Settings <@lincasa05.india.ti.com>

Q-	IP > Repository
Project Settings General	Add directories to the list of repositories. You may then add additiona then a tool-tip will alert you to the reason.
Simulation Elaboration Synthesis Implementation	IP Repositories
Bitstream V IP	
Repository Packager	No content
Tool Settings Project	Refresh All

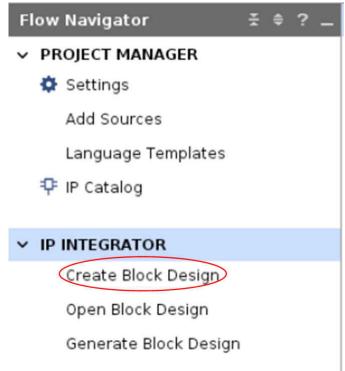
Figure 6-2. Adding IP Repository



## 7 Create Block Designs With TI AFE SPI IP

The IP is expected to be used only within a block design. Users can create a new block design or instantiate the IP within an existing one. To create a new block design with the IP, follow these steps:

1. Open the IP INTEGRATOR menu and click Create Block Design (see Figure 7-1).



## Figure 7-1. Creating Block Design

2. Enter the desired design name (see Figure 7-2) and keep the other two options as defaults. Click OK.

Create Block Design	<@lincasa06.india.ti	$\odot$ $\otimes$
Please specify name	of block design.	4
<u>D</u> esign name: ( D <u>i</u> rectory:	spi_bd	© ~
Specify source set:	🗅 Design Sources	~
?	ОК Са	ancel

## Figure 7-2. Naming of Block Design

3. Click the *IP Catalog* tab and notice the *User Repository* header (see Figure 7-3). The header appears only if the inclusion of the IP repository is done correctly, as explained in previous steps.

7

IP Catalog x Project Sur < ▶ = ? □ □
Cores   Interfaces
Q ≍ ≑ ≉ +; ≁ 2 ⊕ ■ ♦
Search: Q-
Name
🗸 🚞 User Repository (/proj/dsc7/hpasys/users 🛧
✓  ☐ UserIP
FE79xx_SPI_Configurator
🗸 🖆 Vivado Repository
> 🚍 Alliance Partners
> 🖹 Audio Connectivity & Processing
<
Details
Select an IP or Interface or Repository to see details

Figure 7-3. User Repository

- 4. Double-click AFE79xx SPI Configurator to launch the Add IP window.
- 5. Click Add IP to Block Design (see Figure 7-4).

Add IP	<@lincasa06.india.ti.com>			$\odot$	۲
?	Would you like to add 'AFE79 and add it as an RTL module		your block design, or custo	omize	it
	Add IP to Block Design	Customize IP	Cancel		

#### Figure 7-4. Adding IP to Block Design

6. The AFE SPI IP shows up in the block design (see Figure 7-5) if the previous steps were followed correctly.



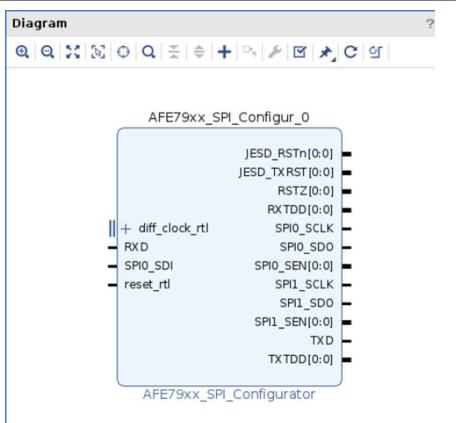
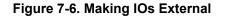


Figure 7-5. Final Diagram

The required IOs from the IP must be brought to the top-level file in the FPGA hdl design hierarchy. To do this, first make the required IOs as external. To make the required IOs as external, right-click the individual IOs of interest and select *Make External* (see Figure 7-6).

SPI0_SEN[0:0]		Paste	Ctrl+V
SPI1_SCLK	Q,	Search	Ctrl+F
SPI1_SD0	12	Select All	Ctrl+A
SPI1_SEN[0:0] TXD	+	Add IP	Ctrl+I
TXTDD[0:0]		Add Module	
	C	Make External	Ctrl+T
_SPI_Configurator		Pinning	Þ
		IP Settings	
	ľ	Validate Design	F6
		Start Connection Mode	Ctrl+H
		Make Connection	



In the above implementation example, the bare minimum required IOs are brought out as external.

The block design must then be validated for any errors and output products to generate. To validate the block diagram, right-click the block design under the *Design Sources* header.

9



Similarly, right-click the *View Instantiation Template* to use as a reference for wiring this block design on the top level hdl (see Figure 7-7).

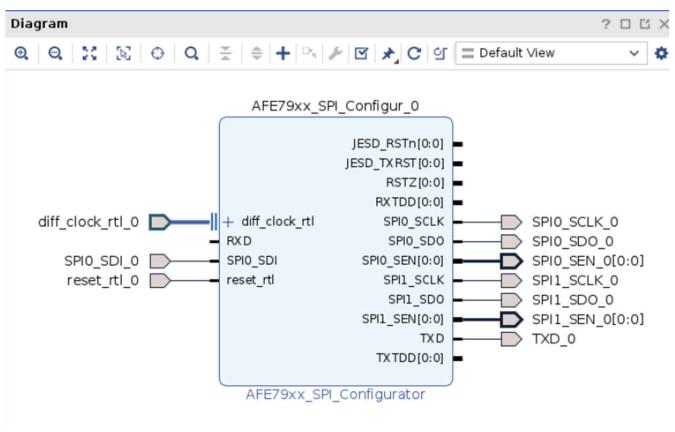


Figure 7-7. TI SPI Configurator Pinout

The IOs from the top level hdl which are connected to this IP must be declared in a *Constraints* file for them to be mapped to the hardware design, specific FPGA pins, and the correct IO levels.

The above steps complete the hardware design using TI AFE SPI IP.

## 8 Create New Platforms in Vitis

To create a new platform in Vitis, follow these steps below:

- 1. Open the *File* menu, go to *New*, and then click *Platform Project* (see Figure 8-1).
- Vorkspace1 Vitis IDE File Edit Search Run Xilinx Project Window Help New Alt+Shift+N > Application Project... 粽 ▼ 🔘 ▼ 🔗 ▼ 🏷 🤇 Library Project... Open File... ----Hw Kernel Project ... Ctrl+W Close Platform Project... Close All Ctrl+Shift+W Ctrl+N C) Other... Save Ctrl+S Save As... Ctrl+Shift+S Save All Move... Rename... F2 -9 ଛ୍ରୀ Refresh **F5** Import... Export... Alt+Enter Properties

#### Figure 8-1. New Platform Project

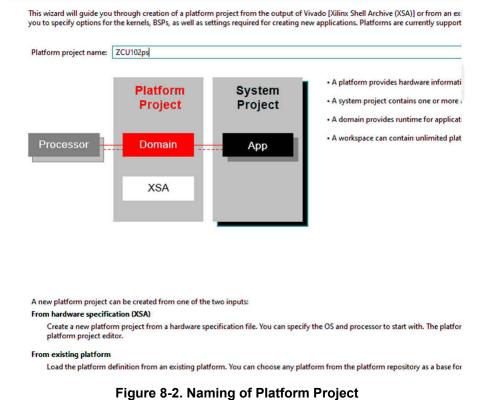
2. Enter the desired platform name. The name ZCU102ps was used as an example (see Figure 8-2).



😼 New Platform Project

#### Create new platform project

Enter a name for your platform project



3. After the new platform is named, a menu appears (see Figure 8-3). Select the XSA (Xilinx Support Archive) file.

-	Menar	Platform	Project
	140.44	1 IULI VIIII	1 TOJECE

#### Platform

Please select a platform to create the project

- article	e Specification				
	Provide your XSA file or use a	pre-built board descriptio			
KSA File:	vck190 zc702 zc706 zcu102 zed				Browse.
Specify th	Specification he details for the initial domain g system:	to be added to the platfor $\checkmark$	. More domains can be after the platforn	is created by double clicking the platfor	m.spr file

#### Figure 8-3. Hardware Specification

4. Browse and select the .XSA file from the FPGA folder shared along with this document (see Figure 8-4).

Х



P 1

.com					New Platforms
Platform					
Please select a platform to	create the project				
Create a new platform	from hardware (XSA) 🔄 Select a pla	tform from repository			
Hardware Specification					
Create Platform from XSA					×
← → ~ ↑			7	V Ö 🔎 Search ZCU	102_BEL
Organize 👻 New folder					• 🖬 🕐
🗟 Documents 🖈 ^	Name	Date modified	Туре	Size	
📰 Pictures 🛛 🖈	ZCU102_BEL.cache	4/22/2022 8:19 AM	File folder		
📖 This PC	ZCU102_BEL.hbs	4/22/2022 8:19 AM	Filefolder		
3D Objects	ZCU102_BEL.hw	4/22/2022 8:19 AM	File folder		
	ZCU102_BEL.ip_user_files	4/22/2022 8:19 AM	File folder		
Desktop	ZCU102_BEL.runs	4/22/2022 8:20 AM	File folder		
Documents	ZCU102_BEL.sim	4/21/2022 1:00 PM	File folder		
🕂 Downloads	ZCU102 BEL.srcs	4/22/2022 8:20 AM	File folder		
🁌 Music 🥚	TI_204c_IP_ref.xsa	4/21/2022 1:26 PM	XSA File	1,563 KB	
Fictures					
Videos					
L. Windows (C:)					
💣 Network					
File nan	ne:			*.xsa;*.dsa;	~
	- 0			Open	Cancel
				open	Contract.

#### Figure 8-4. Selecting the Platform

5. Right-click the new platform project to open the drop-down menu. Click Build Project to start the build (see Figure 8-5). This can take some time to complete the build.

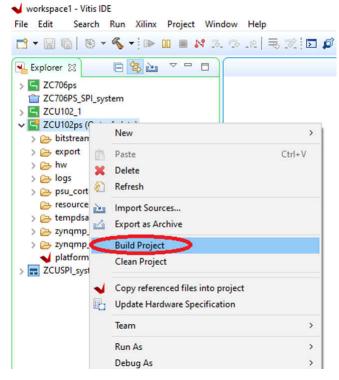


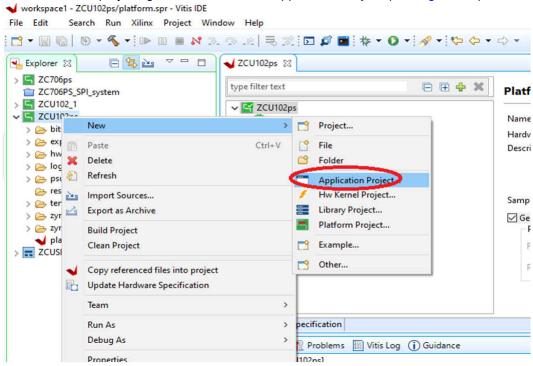
Figure 8-5. Building the New Project



## 9 Create New Application Projects in Vitis

After the build is complete, create a new application project in Vitis. To create a new project, follow these steps:

1. Right-click the Platform project, go to New, then click Application Project (see Figure 9-1).



#### Figure 9-1. Creating New Application Project

2. When the New Application Project window appears (see Figure 9-2), click Next.

#### Vew Application Project

ate a New Application Proj	ect				
This wizard will guide you th 1. Choose a <b>platform</b> or cre 2. Put application project in 3. Prepare the application ru 4. Choose a template for ap	ate a platform project from a system project, associate antime – domain	n Vivado exported XSA it with a processor			
	Platform Project	System Project			
Processor	Domain	Арр			
	XSA				
<ul> <li>A platform provides hardw</li> <li>A system project contains</li> <li>A domain provides runtim</li> <li>A workspace can contain to</li> </ul>	one or more applications the e for applications, such as o	at run at the same time. perating system or BSP.			
	Skip	o welcome page next time. (Can bi	reached with Back button)		
0			< Back	Next>	Finish Cancel

#### Figure 9-2. New Application Project

3. Select the newly created platform ZCU102ps and click Next (see Figure 9-3).

D X



		acute a nem p	latform from hardware (XSA)		
i:	Ø.				🕂 Add  🌺 Manage
ame ZC706ps [c ZCU102 1 ZCU102ps	[custom]	Board zc706 zcu102 zcu102	Flow Embedded SW Dev Embedded SW Dev Embedded SW Dev	xilinx C:\U	n Jsers\imhbdlab\workspace1\ZC706ps\export\ZC706ps\ZC Jsers\imhbdlab\workspace1\ZCU102_1\export\ZCU102_1\ Jsers\imhbdlab\workspace1\ZCU102ps\export\ZCU102ps\
atform Info			Acceleration Resources		Domain Details
	7CU102ps	~	The selected platform does not	t have application	
	xczu9eg-ffvb1156-2-e		acceleration capabilities		Domain name Details
Family:	zynquplus				standaione_domain CPU: psu_contexa55_00
Description:					
ZCU102ps					
Part: Family:	zynquplus		Acceleration Resources The selected platform does not acceleration capabilities	t have application	Domains

Figure 9-3. Selecting the Application Project

4. Type a new application name. ZCU102ps\_SPI was used as an example.

Vew Application Project			- 0 >				
Application Project Details			•••				
Specify the application project name and its sys	tem project properties						
Application project name ZCU102ps_SPI							
System Project							
Create a new system project for the applica	ation or select an existing one from the wo	kpsace 🕜					
Select a system project	System project details						
Create new	System project name: Z	U102ps_SPI_system					
	Target processor						
	Select target processor for	the Application project.					
	Processor	Associated applications					
	ps7_cortexa9_0	ZCU102ps_SPI					
		a ta a second a second to the second s					
② F	Show all processors in the	< Back Next >	Finish Cancel				
F standalone on microblaz √ New Application Project	igure 9-4. New Appl	<back next=""></back>	Finish Cancel				
F standalone on microblaz	<b>igure 9-4. New Appl</b> e_0 and click <i>Next</i> (s	<back next=""></back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new	<b>igure 9-4. New Appl</b> e_0 and click <i>Next</i> (s	<back next=""></back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin	igure 9-4. New Appl ie_0 and click Next (s domain	<back next=""> ication Project Name ee Figure 9-5).</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain for your project or create a new Select the domain for your project or create a new Select the domain for your project or create a new Select the domain for your project or create a new	igure 9-4. New Appl e_0 and click Next (s domain Ik to or create a new domain ave all the requirements of the application	<back next=""> ication Project Name ee Figure 9-5).</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will ha Select a domain	igure 9-4. New Appl ie_0 and click Next (s domain	<back next=""> ication Project Name ee Figure 9-5).</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl e_0 and click Next (s domain Ik to or create a new domain ave all the requirements of the application	<back next=""> ication Project Name ee Figure 9-5).</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will ha Select a domain	igure 9-4. New Appl re_0 and click <i>Next</i> (s domain ak to or create a new domain ave all the requirements of the application	<back next=""> ication Project Name ee Figure 9-5). emplate selected in the next step standalone_domain</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl re_0 and click Next (s domain ak to or create a new domain ave all the requirements of the application Domain details Name: Display Name:	<back next=""> Cation Project Name ee Figure 9-5). emplate selected in the next step standalone_domain standalone on microblaze_0</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl re_0 and click Next (s domain ak to or create a new domain ave all the requirements of the application Domain details Name: Display Name: Operating Syst	<  ication Project Name ee Figure 9-5). emplate selected in the next step           standalone_domain standalone on microblaze_0 emmit standalone	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl re_0 and click Next (s domain ak to or create a new domain ave all the requirements of the application Domain details Name: Display Name:	<back next=""> Cation Project Name ee Figure 9-5). emplate selected in the next step standalone_domain standalone on microblaze_0</back>	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl re_0 and click Next (s domain ak to or create a new domain ave all the requirements of the application Domain details Name: Display Name: Operating Syst	<  ication Project Name ee Figure 9-5). emplate selected in the next step           standalone_domain standalone on microblaze_0 emmit standalone	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl re_0 and click Next (s domain ak to or create a new domain ave all the requirements of the application Domain details Name: Display Name: Operating Syst	<  ication Project Name ee Figure 9-5). emplate selected in the next step           standalone_domain standalone on microblaze_0 emmit standalone	>				
F standalone on microblaz New Application Project Domain Select a domain for your project or create a new Select the domain that the application would lin Note: New domain created by this wizard will have Select a domain standal ne on microblaze_0	igure 9-4. New Appl re_0 and click Next (s domain ak to or create a new domain ave all the requirements of the application Domain details Name: Display Name: Operating Syst	<  ication Project Name ee Figure 9-5). emplate selected in the next step           standalone_domain standalone on microblaze_0 emmit standalone	>				

6. Select Hello World from the list of templates and click Finish (see Figure 9-6).



name and address of the second s		•••
Femplates Select a template to create your project.		
Available Templates:		
Find:	Hello World	
<ul> <li>SW development templates</li> <li>Dhrystone</li> <li>Empty Application (C++)</li> <li>Hello Wold</li> <li>M/P Echo Server</li> <li>M/P TCP Pert Client</li> <li>M/P TCP Pert Client</li> <li>M/P UDP Pert Client</li> <li>M/P UDP Pert Server</li> <li>Memory Tests</li> <li>OpenAMP echo-test</li> <li>OpenAMP Martix multiplication Demo</li> <li>OpenAMP RPC Demo</li> <li>Peripheral Tests</li> <li>RSA Authentication App</li> <li>Zynq DRAM tests</li> <li>Zynq FSBL</li> </ul>	Let's say 'Hello World' in C.	

Figure 9-6. Selecting Template7. A fresh C project appears on top where the actual application development can start.

## **10 Build Application Projects**

To build an application project, follow these steps:

- 1. Right-click the application name and select Build Project (see Figure 10-1).
  - workspace1 ZCU102ps\_SPI/src/main.c Vitis IDE File Edit Search Run Xilinx Project Window Help 🖳 Explorer 🖾 F 🔁 🖄 c main.c 🖾 \* Except as contained in this notice, the name of th > 🔄 ZC706ps 27 28 \* in advertising or otherwise to promote the sale, u > T ZC706PS\_SPI\_system \* this Software without prior written authorization 29 > 🔄 ZCU102\_1 30 > S ZCU102ps \*\*\* 31 ZCU102ps\_SPI\_system [ ZCU102ps ] 32 #include <stdio.h> V C ZCU102ps\_SPI 1 Sandalone on nsu cortev 33 #include "xparameters.h" /\* EDK generated paramet > 🐝 Binaries New > tiafe\_bringup.h> > 🔊 Includes Paste Ctrl+V > 🗁 Debug platform.h" Delete × > 🕞 Include xil\_printf.h" 🗸 🗁 src Refresh math.h" 8 > .c main. > h platfo 🕍 **Export as Archive** tiAfe79 interface.h" > c platfo Build Project > h platfo Clean Project > .c tiAfe7 latform(); intf("SPI Writing for AFE Registers!"); S Iscript Team > p\_platform(); > 🧭 ide X ZCU102p Run As > set(); //Reset of PS SPI > 🕞 Debug Debug As 5 ZCU102ps\_S Properties > 👕 ZCUSPI\_system Problems 📗 Vitis Log (i) Guidance TCF Debug Virtual Terminal - Cortex-A53 #3

Figure 10-1. Building Project

2. Ensure that the project builds without any errors.



## **11 Configure the AXI GPIO**

Sequence for AXI GPIO configuration should be:

- 1. Initialize GPIO module
- 2. Set Direction
- 3. Set High or Low for corresponding bits

The above sequence must be completed before initializing AFE and LMK devices.

#### 11.1 Initializing the GPIO

The C syntax for initializing the AXI GPIO is in two steps:

- 1. XGpio GPOs: Initialize a pointer (GPOs) to the GPIO configuration register.
- 2. XGpio\_Initialize(&GPOs, XPAR\_AXI\_GPIO\_0\_DEVICE\_ID): Refer to *Xparameters.h* to find the correct AXI GPIO DEVICE ID.

#### **11.2 Setting the Direction**

XGpio\_SetDataDirection(&GPOs, 1, 0);

First argument &GPOs point to the GPIO instance initialized in previous command.

Second argument 1 indicates the GPIO bank. In **TI SPI AFE IP**, only the first bank is used.

Third argument 0 indicates all GPIO bits are set for outputs.

#### **11.3 Setting High or Low for Corresponding Bits**

XGpio\_DiscreteWrite(&GPOs, 1, regval);

First argument &GPOs point to the GPIO instance initialized in previous command.

Second argument 1 indicates the GPIO bank. In **TI SPI AFE IP**, only the first bank is used.

#### Table 11-1. Bit Mapping of IP I/Os

BIT DESCRIPTION	BIT POSITION
JESD RSTn	4
JESD TXRST	3
RSTn	2
RXTDD	1
TXTDD	0

For Example:

XGpio\_DiscreteWrite(&GPOs, 1, 0x14);

This command sets JESD RSTn and RSTn to 1, sets all other bits to 0

# 12 Configure the AXI SPI

The AXI SPI instance in TI AFE SPI IP is used in Standard Mode.

Peripherals select 0 and select 1 are used as chip selects for AFE and LMK clocking device, respectively

SCL frequency is hard coded to 10 MHz within TI IP

Key commands for SPI initialization and usage in Vitis are as explained below:

1. XSpi\_Config \*ConfigPtr;

Initialize a pointer (ConfigPtr).

- 2. ConfigPtr = XSpi\_LookupConfig(XPAR\_AXI\_QUAD\_SPI\_0\_DEVICE\_ID);
- Refer 'Xparameters.h' to find the correct AXI QUAD SPI DEVICE ID. 3. XSpi\_CfgInitialize(&Spidev, ConfigPtr, ConfigPtr->BaseAddress);
- Initialize a new instance of SPI (Spidev).
- 4. XSpi\_SetOptions(&Spidev, XSP\_MASTER\_OPTION);

Set the Spidev instance to be in Controller mode.

- 5. XSpi\_Start(&Spidev);
- 6. XSpi\_SetSlaveSelect(&Spidev, 1);

Select Peripheral: AFE.

7. XSpi\_SetSlaveSelect(&Spidev, 2);

Select Peripheral: LMK.

8. XSpi\_Transfer(&Spidev, WrBufdev, RdBufdev, 3);

Second argument WrBufdev is an array with 3 bytes (24-bit data to be transmitted on SPI).

Third argument RdBufdev is an array with 3 bytes, the last byte has the SPI read value.

Fourth argument is number of bytes to be transmitted/received...3 in our case.

The D23 is the MSB bit of the 24-bit data because the D23 indicates whether it is a Read or a Write SPI operation:

- If D23 is set to 1, then it is a read operation and RdBufdev[2] stores the read back address contents
- If D23 is set to 0, then it is a write operation and RdBufdev[2] has no significance

**EXAS** 

STRUMENTS

www.ti.com

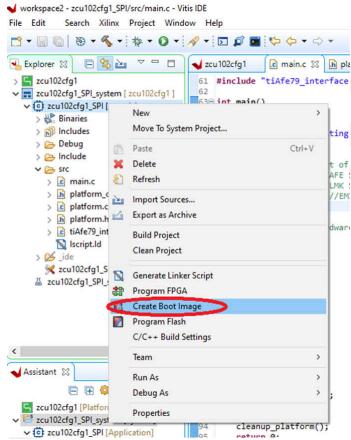


## 13 Create Boot Images to Run on SD Card

This section of steps is applicable only if there is a need to create a bootable SD Card. Skip these steps if not applicable.

To create a bootable SD Card with the active application, follow these steps:

1. Right-click the application name to open the drop-down menu and select *Create Boot Image* (see Figure 13-1).



#### Figure 13-1. Creating Boot Image

2. Select *Zynq MP* from the *Architecture* menu and select *Import from existing BIF file* option (see Figure 13-2). The *Output path* shows the file location of the *BOOT.bin* to be generated.

Create Boot Ima	ge	×
Create Boot Imag	2	
Creates Zynq MP B	oot Image in .bin format from given FSBL elf and partition files in specified output folder.	
Architecture	MP 🗸	
Create new BIF fil	Import from existing BIF file	
Import BIF file path:	_ide/bootimage/zcu102cfg1_SPI.bif	Browse
Basic Security		
Output BIF file path	: de/bootimage/zcu102cfg1_SPI.bif	Browse
		Browse
UDF data:		
UDF data:	Output format: BIN v	

#### Figure 13-2. Boot Image Input and Output Pathnames

3. After reviewing the locations of the boot elf file, FPGA bit file, and application elf, click *Create Image* (see Figure 13-3).



		التحقيق الم
ile path	Encrypted	Authenticated
bootloader) C:/Users: Control of the	none	none
ide/bitstream/TI_204c_IP_ref.bit	none	none
/Debug/zcu102cfg1_SPI.elf	none	none

Figure 13-3. Boot Image Partitions

The above step generates the BOOT.bin in the output path as selected in Figure 13-2.

- 4. Copy BOOT.bin to a FAT16 or FAT32 formatted SD Card.
- 5. To boot from SD Card, ensure the SW6 switch positions on the ZCU102 is as per Figure 13-4.



#### Figure 13-4. SW6 Switch Positions

6. With the above setting on SW6 and SD card inserted, ZCU102 can now directly boot with the application.

## 14 Set up and Power on Hardware

To set up and power on the hardware, follow these steps:

- 1. Dock the AFE EVM to J5 (HPC0) FMC on the ZCU102 board.
- 2. Connect a 1.5-GHz signal through a clock source to RFROM EVM at J14.
- 3. Connect J2 (JTAG) and J83 (UART) USB connectors from ZCU102 FPGA board to the computer.
- 4. Connect the 12-V Xilinx EVM Adapter for the ZCU102 at J52.
- 5. After all the above connections are made, power up the setup. Note that the AFE EVM in this example is completely powered by the ZCU102 FMC interface.



## 15 Set up ZCU102 Board Interface for VADJ\_FMC

To set the ZCU102 board interface for VADJ\_FMC, follow these steps:

- 1. Execute the ZCU102-Board User Interface software (available for download from Xilinx.com).
- Select the appropriate COM Port to enable communication between the onboard MSP430 of the ZCU102 and the PC. This software is required to turn on the FMC\_AUX supply of 1.8 V for the FMC bank of FPGA (see Figure 15-1).

(	locks	Voltag	ges	Power	FMC	GTR MUX	EEPROM Data	GPIO Commands	System Monitor	About
	Set	Read	Set	Boot Frequ	ency	Restore Dev	ice Defaults			
	Read	Si570 Us	er Fre	equency	Frequ	ency: 0.000 l	MHz			
	Read 1	5i570 MC	ST Fr	equency	Frequ	ency:				
	Rea	d Si5328	Freq	uency	Frequ	ency:				
							tem controller port	1 20200		
						COM58	int that system contr	v		
						Advanced	>>			
								ок		

#### Figure 15-1. ZCU102 Board User Interface

3. Select the Set VADJ to 1.8 V check box (see Figure 15-2).

System Controller

Clocks Set VAD	Voltages HPC0	Power HPC1	FMC	GTR MUX	EEPROM Data	GPIO Commands	System Monitor	About
Current	100	1						
Set VAI	OJ to 0.0 V							
Set VAI	DJ to 1.2 V							
Set VAI	DJ to 1.5 V							
Set VAI	OJ to 1.8 V	>						
No. of Concession, name								

# Figure 15-2. Setting VADJ

4. Confirm the same by reading the VADJ\_FMC voltage. The voltage value must be 1.80 V (see Figure 15-3).



	Clocks Voltages Powe	FMC	GTR MUX	EEPROM Data	GPIO Commands	System Monitor	About	_
	Get VCC1V2 Voltage	Voltage:						
	Get VCC3V3 Voltage	Voltage:						
	Get VADJ_FMC Voltage	Voltage:	1.80 Volts					
	Get MGTAVCC Voltage	Voltage:						
	Get MGTAVTT Voltage	Voltage:						
	Get UTIL3V3 Voltage	Voltage:						
]	Get UTIL5V0 Voltage	Voltage:						

Figure 15-3. VADJ\_FMC Voltage



## 16 Debug Application Projects and Set up Vitis Serial Terminal

To debug the application project and set up the Vitis serial terminal. follow these steps:

1. Right-click the project name and go to *Debug As* from the drop-down menu. Click *Launch on Hardware* (*Single Application Bug*) to run the debug (see Figure 16-1).

🔁 • 🔛 🕼   📎		€ 🔁 🚵 🗢 🗖 🔲 🖻 main.c Σ	1		☆ • O •
<ul> <li>ZC706ps</li> <li>ZC706PS_SPI_s</li> <li>ZCU102_1</li> <li>ZCU102ps</li> <li>ZCU102ps_SPI</li> <li>ZCU102ps</li> <li>Binaries</li> </ul>	_syst	m 46 x 47 c 48 em [ZCU102os1 49 S	:leanup_pl :PI_reset( lev spi se	("SPI atform ); // tup() up()	Writing for AFE Registers!");
> 🔊 Includes		Move To System Project			() ) ) ) ) () () () () () () () () () ()
> 🛃 Debug > 👝 Include > 👝 src > 🇭 _ide	2	Paste Delete Refresh	Ctrl+V	001)	
<ul> <li>ZCU102</li> <li>Debug</li> <li>ZCU102ps_</li> </ul>		Import Sources Export as Archive		001) ); "Done	; ;]");
> 👕 ZCUSPI_syster		Build Project Clean Project		tform	n();
		Generate Linker Script Program FPGA			
		Create Boot Image		ems	Vitis Log (i) Guidance
		Program Flash		PI, De	
		C/C++ Build Settings		enta.	Build of configuration Debug for project ZCU102ps_SPI ****
		Team	>	done	for 'all'.
		Run As	>	hed	(took 648ms)
<		Debug As		24	Launch on Hardware (Single Application Debug)
<		Properties		1000	2 Launch on criticiator (Jingre Application Debug)
< ✔Assistant 🛛	_	Properties		<b>Е</b> 608	2 Launch on Hardware (Single Application Debug (GDB)) Debug Configurations

#### Figure 16-1. Debugging Application Project

2. Connect the Vitis Serial terminal (see Figure 16-2) with baudrate 115200 (this can be used to see SPI write or read status).



#### Figure 16-2. Vitis Serial Terminal



## **17 Execute the Application**

To execute an application run, follow these steps:

1. Click the right arrow button as shown in Figure 17-1.

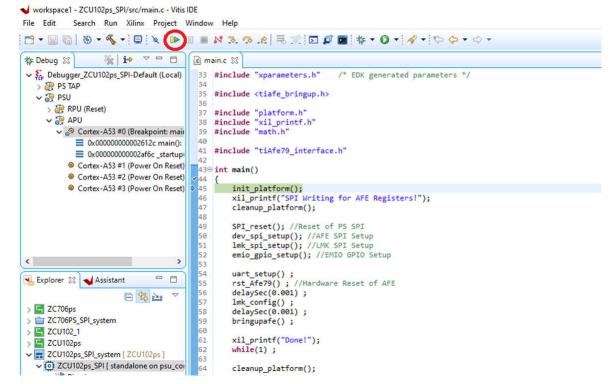


Figure 17-1. Executing Application

2. Notice the SPI logs being printed on the UART xil\_printf("Done!");

Console	Progress	Vitis Serial Terr	minal 🛛 🚺 Executab	les 📗 Vitis Log	Problems	Rebugger Console
Connected to:	Serial ( CON	157, 115200, 0, 8 )				
Connected to	COM57 at 11	5200				
SPI Writing fo		rs!				
UART Configu		II in line 385 0x171	.0			
		Il in line 501 0xf0 :				
		Il in line 514 0xf0 :				
		Il in line 570 0xf0 :				
		Il in line 583 0xf0 :				
Number of ite	erations of po	Il in line 719 0xf0 :	0			
		Il in line 732 0xf0 :				
Number of ite	erations of po	Il in line 2831 0xf0	:0			
	erations of po	Il in line 2848 0xf0	:0			

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), 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 will 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 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.

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

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265 Copyright © 2022, Texas Instruments Incorporated