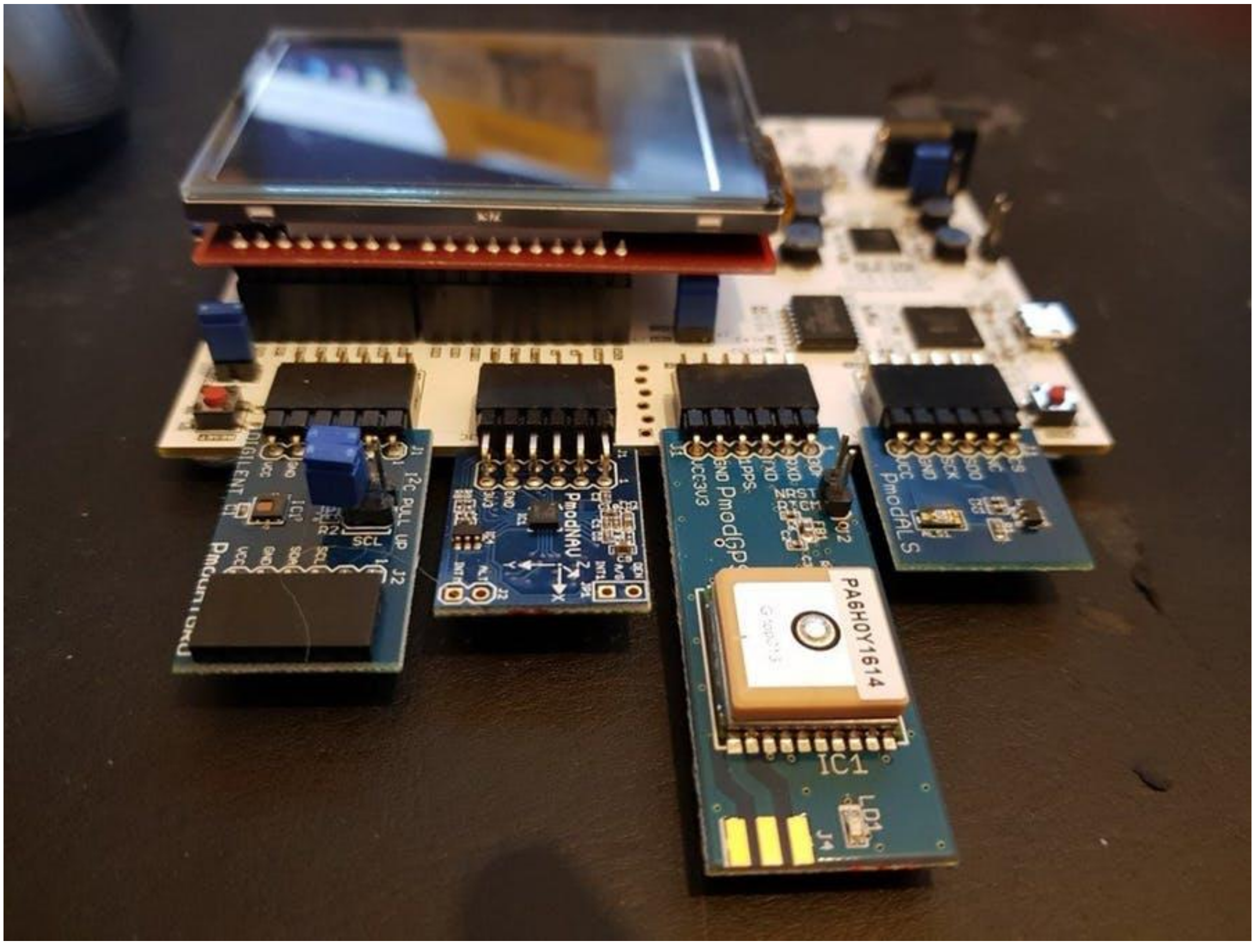
Measure Environmental Data Through Digilent Arty S7 & Pmods

Readers can learn how to collect and display the environmental data through Arty S7, 4 Pmods and a touchscreen display.



Things used in this project

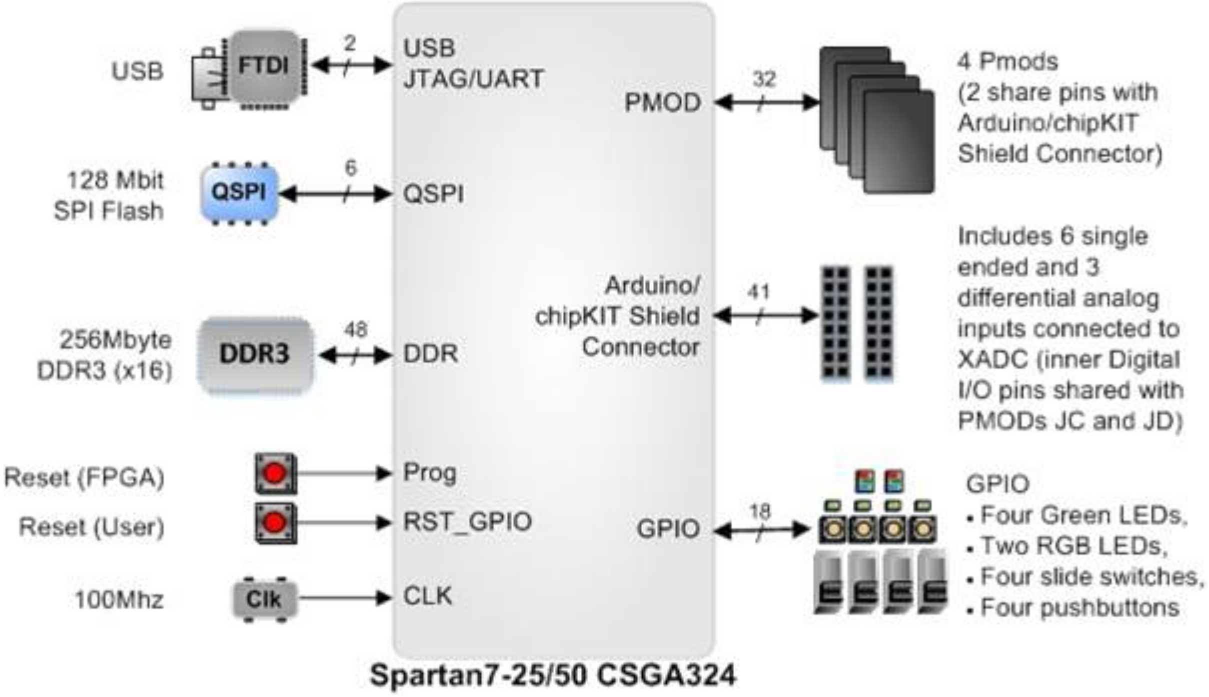
Hardware components

	Digilent Arty S7-50	×	1
	Digilent Pmod GPS	×	1
	Digilent Pmod NAV	×	1
	Digilent Pmod HYGRO	×	1
	Digilent Pmod ALS	×	1

Story

Introduction

The Digilent Arty S7 is a variant of the popular Digilent Arty series of boards, designed around the Xilinx Spartan 7 family of devices. When it comes to interfacing the Arty S7 provides a range of options from 4 Pmod ports to Arduino Shield connectors.



Arty S7 Block Diagram

In this tutorial, we use Arty S7, four Pmod's (see bleow list) and a multi-touch display shield (MTDS) to collect, process and display environmental data. To do this we will need the following

- Pmod NAV – 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer, plus digital barometer
- Pmod GPS – GPS receiver
- Pmod HYGRO – Relative humidity sensor with integrated temperature sensor
- Pmod ALS – Ambient light sensor

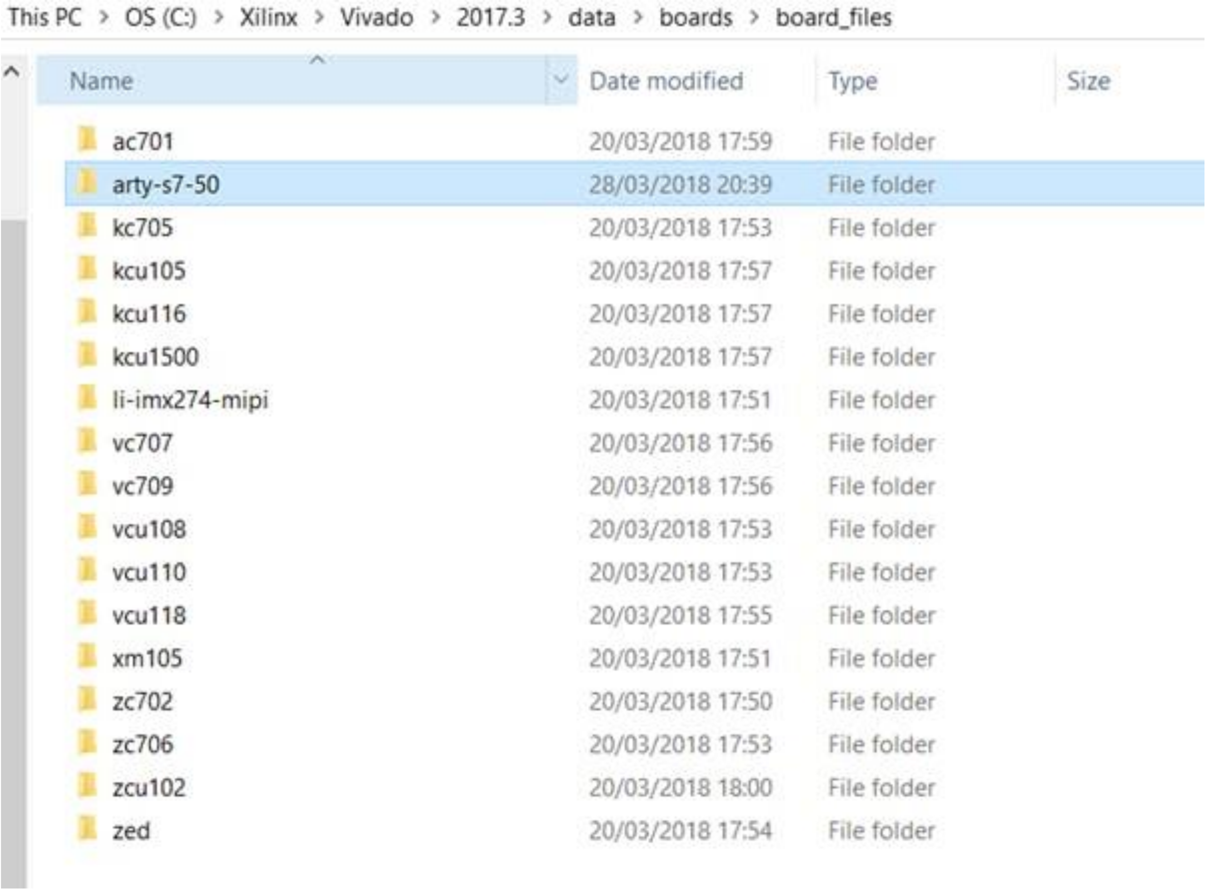
Step 1: Build the hardware platform

To create this example the first thing we need to do is to create a hardware platform using Xilinx Vivado. For this application, we instantiate a MicroBlaze processor within the Spartan 7 FPGA device. We then use Xilinx

SDK to create the software to drive the multi-touch display shield (MTDS) and Pmods.

Install Board Files

Firstly, we should install the Arty S7 board files which can be downloaded from GitHub. These files define for Arty S7 interfaces and memory configurations in Xilinx Vivado environment. Once the folder is extracted, they should be copied into your Vivado installation under the directory <vivado>/<vivado version>/data/boards/board_files as shown below (You can also refer to the tutorial provided by



This PC > OS (C:) > Xilinx > Vivado > 2017.3 > data > boards > board_files

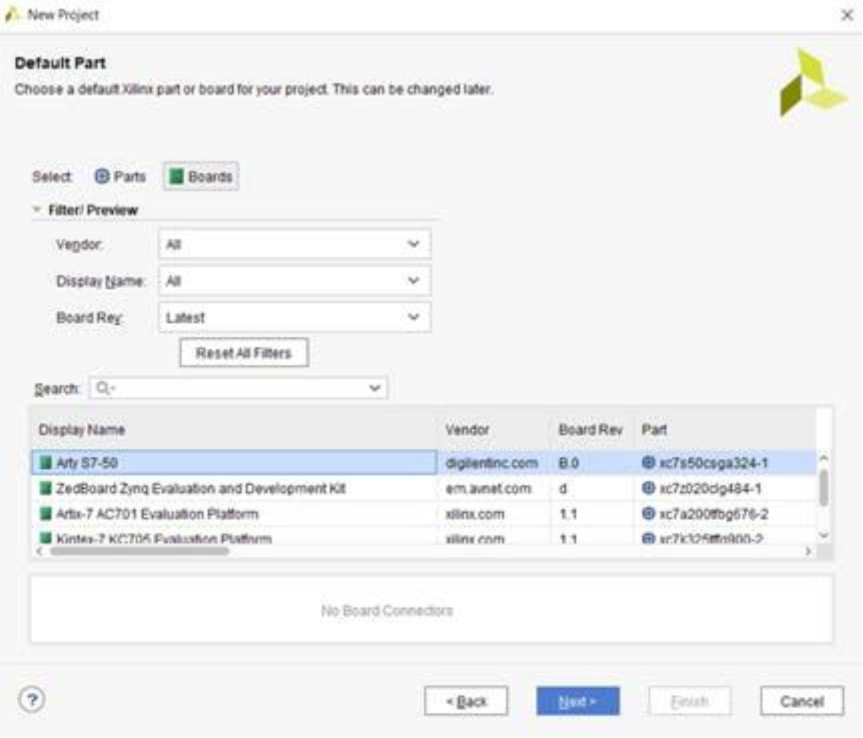
Name	Date modified	Type	Size
ac701	20/03/2018 17:59	File folder	
arty-s7-50	28/03/2018 20:39	File folder	
kc705	20/03/2018 17:53	File folder	
kcu105	20/03/2018 17:57	File folder	
kcu116	20/03/2018 17:57	File folder	
kcu1500	20/03/2018 17:57	File folder	
li-imx274-mipi	20/03/2018 17:51	File folder	
vc707	20/03/2018 17:56	File folder	
vc709	20/03/2018 17:56	File folder	
vcu108	20/03/2018 17:53	File folder	
vcu110	20/03/2018 17:53	File folder	
vcu118	20/03/2018 17:55	File folder	
xm105	20/03/2018 17:51	File folder	
zc702	20/03/2018 17:50	File folder	
zc706	20/03/2018 17:53	File folder	
zcu102	20/03/2018 18:00	File folder	
zed	20/03/2018 17:54	File folder	

Copying the Arty S7 board files in the Vivado installation

Create drivers for Pmods and MTDS

We can create drivers for Pmods and the MTDS by leveraging the existing Digilent Vivado library available from the Digilent GitHub. This library contains both the Xilinx Vivado and Xilinx SDK drivers for most Pmods. Once you are at the Digilent GitHub, you can either clone the directory locally or download a zip and extract it locally.

Next, we open Xilinx Vivado and create a new project targeting the Arty S7



Default Part

Choose a default Xilinx part or board for your project. This can be changed later.

Select: ○ Parts ● Boards

Filter/ Preview

Vendor: All

Display Name: All

Board Rev: Latest

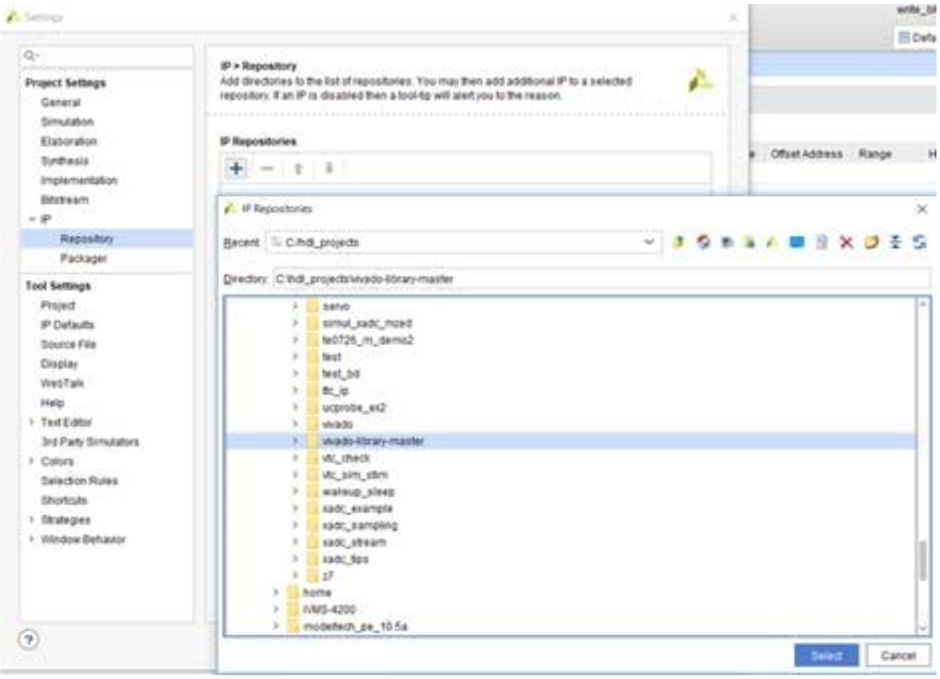
Search: Q-

Display Name	Vendor	Board Rev	Part
Arty S7-50	digilentinc.com	B.0	xc7s50csga324-1
ZedBoard Zynq Evaluation and Development Kit	em.avnet.com	d	xc7z020clg484-1
Artix-7 AC701 Evaluation Platform	xilinx.com	1.1	xc7a200tfbg676-2
Kintex-7 KC705 Evaluation Platform	xilinx.com	1.1	xc7k325tffg900-2

No Board Connectors

Project creation dialog selecting the Arty S7 Board

Once the project has been created, the next step is to map the Digilent Vivado Library as a repository such that we can add the Pmod and MTDS IP Cores into the design. This can be done by the project settings tab.



IP > Repository

Add directories to the list of repositories. You may then add additional IP to a selected repository. If an IP is disabled then a tool-tip will alert you to the reason.

IP Repositories

Recent: C:/hdl_projects

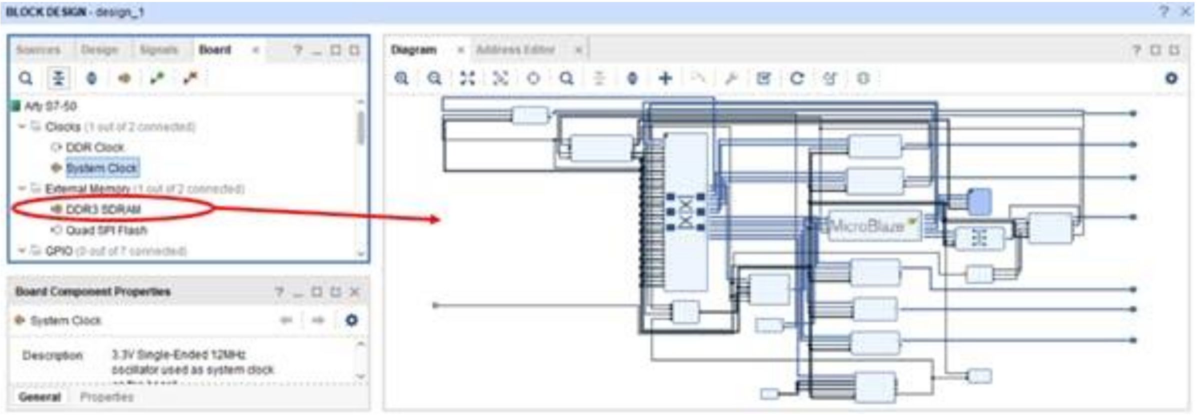
Directory: C:\hdl_projects\vivado-library-master

MicroBlaze

With the libraries all mapped, the next stage is to create a block diagram and instantiate the MicroBlaze. As we are using the Arty S7 board definition, this can be achieved very quickly. We need to create a block diagram and drag the following elements on to the block diagram from the board tab.

- System clock – This will add a clock wizard, connected to the system clock input, configure this to generate two outputs clk1 = 100 MHz, Clk2 = 200 MHz, also disabled the reset input.
- DDR3 SDRAM – this will add a Memory Interface Generator project configured for the Arty S7 and its fitted DDR memory.

Connect the 100 MHz clk to the DDR System clock input and the 200MHz clk to the DDR reference clock input.



Description: 3.3V Single-Ended 12MHz oscillator used as system clock

Adding board elements to the block diagram

The next step is to add the MicroBlaze. This can be done by selecting the *add IP option* and searching for MicroBlaze. Once it is added to the design, we can let Xilinx Vivado connect most of the system by using the *run connection automation* option. If never create MicroBlaze systems, this video provides a step by step example.

When you instantiate the MicroBlaze IP core, you need to enable the following elements before you run the connection automation

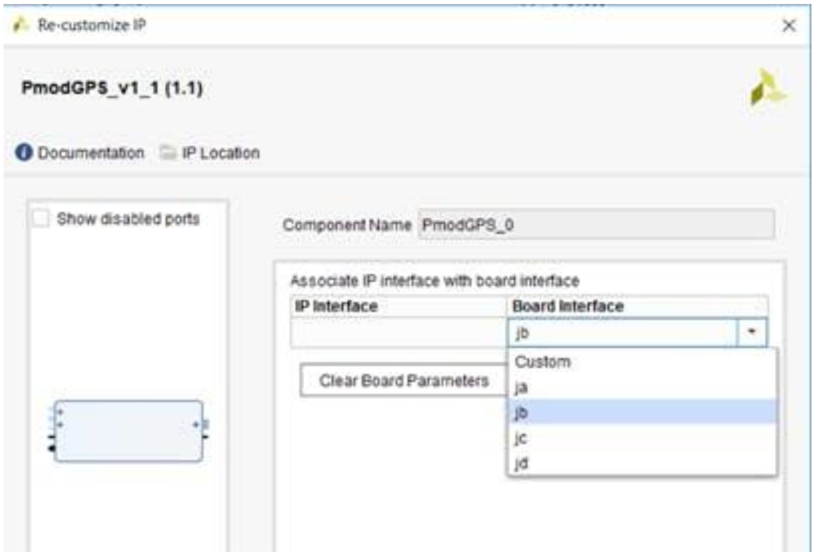
- AXI peripheral data and instruction interface
- Debug module

After we create the MicroBlaze system, we add required Pmods, MTDS and other IP cores (see below list) from Digilent Vivado library. These IP Cores require connecting to the correct output port (Pmod or Shield connector) and the AXI interconnect so that MicroBlaze system to control and interact with them.

- Pmod GPS
- Pmod NAV
- Pmod ALS
- Pmod HYRGO

- PmodMTDS
- AXI Timer – used to create a periodic interrupt timer

Once all required IP blocks have been added, we can associate its outputs with a Pmod or the shield interface by double clicking on the IP block and selecting the interface from the drop-down menu.



Re-customize IP

PmodGPS_v1_1 (1.1)

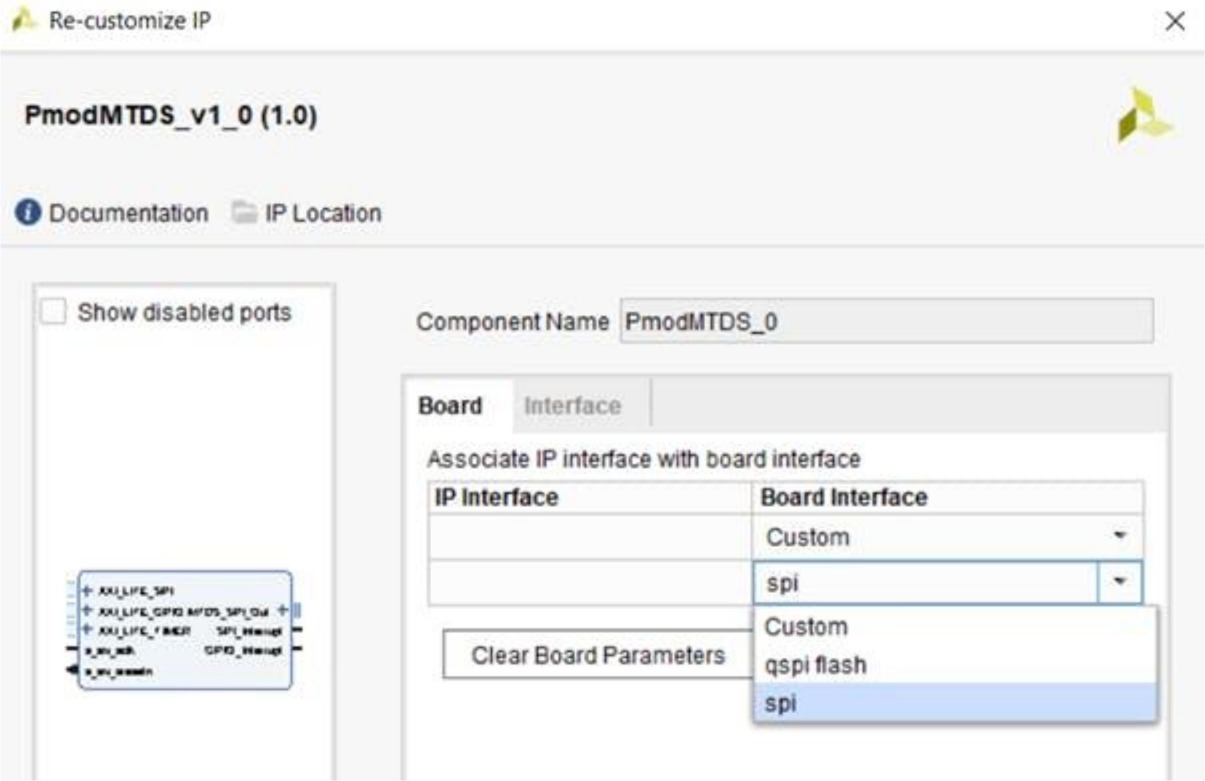
Component Name PmodGPS_0

Associate IP interface with board interface

IP Interface	Board Interface
	jb

Selecting the associated Pmod

For MTDS, we select the SPI interface



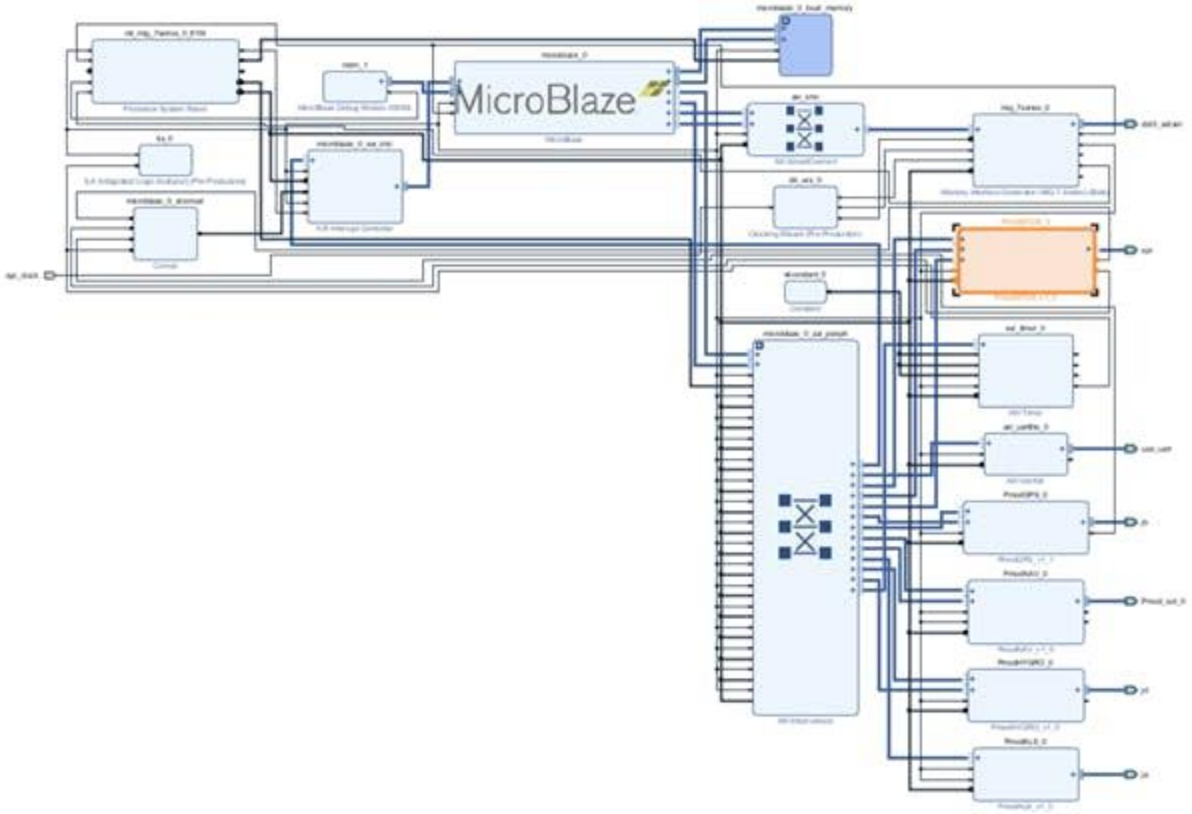
Component Name PmodMTDS_0

Associate IP interface with board interface

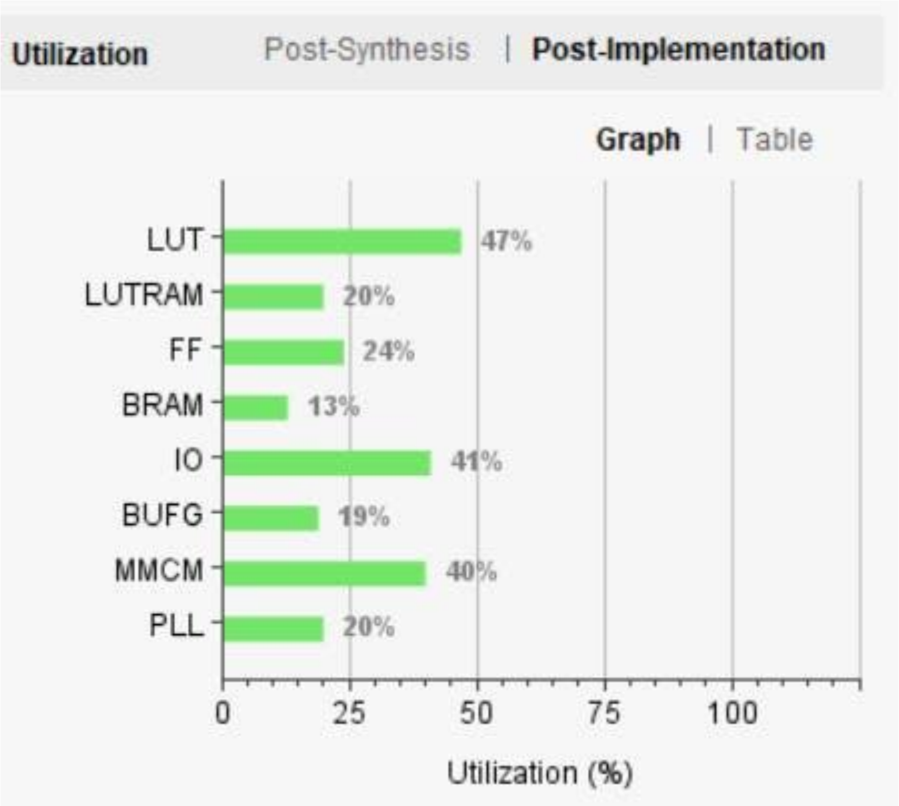
IP Interface	Board Interface
	Custom
	spi

Configuring the MTDS to connect to the Shield SPI

We have completed the top level design is completed and can export it to Xilinx SDK



Completed Vivado Design



Step 2: Create the software

We first need to prepare a MicroSD Card. This SD Card is inserted into the MTDS and holds a number bitmap images we might wish to use. The MTDS driver provides an images directory with several ready prepared bitmaps which be found at <install location>\ip\Pmods\PmodMTDS_v1_0\drivers\PmodMTDS_v1_0\examples\Resources

Copy these files to the SD Card, including any additional ones required for your application and insert the SD Card in to the MTDS.

Generate software application

There are three modes for the system. Modes will be changed via touch screen buttons on the MTDS

- Mode one – Will take information from the Pmod NAV and determine the heading and display on a compass on the display
- Mode Two - Will take information from the Pmod NAV and Pmod GPS and display navigation information
- Mode Three – Will take information from the Pmod HYGRO and Pmod ALS and display that information

Here is the outline for the overall software architecture

- Main function configures Pmods and other drivers, including configuring the interrupt controller for use with the AXI Timer. This function also draws the initial image on the touch display.
- A continual loop which will monitor the touch screen buttons and configure the display for the selected mode.

- An interrupt service routine which is dependent upon mode interfaces with the required Pmod(s) gathers the information and then updates elements of the display as required.

This ensures the system is efficient and reduce the possibility for missing any "touch screen event".

Now, we can make use of the software drivers provided by Digilent Vivado library. To interact with the MTDS, we can use the MYDISP API which provides several functions such as

- Initialize display.
- Draw a line, ellipses and arcs between two points.
- Display test.
- Load a bitmap image from the SDCard.
- Create buttons and monitor when they are pushed allowing action to be taken.

The MYDISP API is fully documented with auser manual under the PmodMTDS software driver at <install>ip\Pmods\PmodMTDS_v1_0\drivers\PmodMTDS_v1_0\examples\Documentation

This document will tell you everything you need to get the MTDS up and running. Similarly, you can find application examples for each of the Pmods used under the software driver directory.

Using these examples and the outline architecture above we can quickly and easily create the software application.