MicroZed™ Carrier Card for Arduino™
Getting Started Guide
Version 1.0
Document Control

Document Version: 1.0
Document Date: 7/31/2015
Document Author(s): Avnet

Prior Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>7/31/2015</td>
<td>Initial Release</td>
</tr>
</tbody>
</table>
Figures

Figure 1 – MicroZed Carrier Card for Arduino (shown with MicroZed SOM mounted) 5
Figure 2 – MicroZed Carrier Card for Arduino Block Diagram .................................. 7
Figure 3 – MicroZed Carrier Card for Arduino (shown with MicroZed SOM mounted) 9
Figure 4 – Arduino UNO R3 Compatible Shield Connectors .................................... 11
Figure 5 – Vivado IP Integrator View of Reference Design ..................................... 12
Figure 6 – Zynq I/O Peripherals used to interface to Shield and Pmod Examples ..... 12
Figure 7 – Detail of SEEED Studio Quad Relay Shield ........................................... 13
Figure 8 – Adafruit 2.8” Cap Touch TFT Display Shield ....................................... 15
Figure 9 – Adafruit 2.8” Cap Touch TFT Display Shield (rear view) .................... 16
Figure 10 – Wilink 8 Eval Module and Avnet Wilink 8 to Pmod Adaptor ............... 18
Figure 11 – FRDM-KL46Z Eval board controlling MicroZed’s Power Rails .......... 19
Figure 12 – Detail of FRDM-KL46Z I/O utilization ............................................... 20
1 Getting Started with MicroZed Carrier Card for Arduino

The MicroZed™ Carrier Card Kit for Arduino™ brings both Shield expansion and Peripheral Module (2x6 pin connectors) expansion to the MicroZed System-On-Module. The carrier card is ideal for building quick prototypes that leverage the large number of Arduino-compatible Shields and Pmod-compatible expansion boards.

This low-cost platform facilitates:

- Expansion of evaluation platform functionality via a wide range of 3rd party Arduino™-compatible "Shields"
- Expansion of evaluation platform functionality via a wide range of 3rd party “Pmod-compatible” boards, interfaced to MicroZed as well as the optional MCU sub-system, via the provided 2x6 Peripheral expansion connectors
- Power management and scheduling of the MicroZed board by an optional Microcontroller board. (Low power design-partitioning can be implemented with the optional MCU offloading some tasks from the Zynq SOC Apps Processor)

Figure 1 – MicroZed Carrier Card for Arduino (shown with MicroZed SOM mounted)

This Getting Started Guide will outline the steps to setup the MicroZed SOM and Arduino Carrier Card hardware. It documents the procedure to run a Linux design on the ARM® dual-core Cortex™-A9 MPCore™ Processing System (PS).
2 What’s Inside the Box?

2.1 Arduino Carrier Card Kit contents
– MicroZed Carrier Card for Arduino (AES-ARDUINO-CC-G)
– microUSB cable
– Documentation (not shown)
  – Quick Start Instruction card
  – Welcome Letter

3 What’s on the Web?
MZ Carrier Card for Arduino is a community-oriented kit, with all materials being made available through the MicroZed.org community website.

3.1 Official Documentation:
– Getting started guide
– Hardware user guide
– Schematics
– Bill of materials
– PCB Layout
– PCB net lengths
– Mechanical drawing
– 3D Model
– Board definition files for Vivado integration
– Programmable logic (PL) master user constraints

3.2 Tutorials and Reference Designs:
– Introduction to Zynq Design Tutorials
– PetaLinux BSP
– Booting Carrier Card using QSPI and eMMC
– Community projects

3.3 Trainings and Videos:
– Introduction to Zynq Software Design
– Introduction to Zynq Hardware Design
4 Arduino Carrier Card Key Features

4.1 Features

- Carrier Card support for MicroZed SOMs
- Full Arduino™-compatible Shield site with connection to MicroZed PL I/O
- Partial Arduino™-compatible Shield site for connection to optional low-power MCU Eval board sub-system
  - Supports MCU ON/OFF control of MicroZed power
  - Enables offloading of MicroZed tasks to the low-power MCU
  - Facilitates low-power sensor hub functionality, autonomous from MicroZed application processor
- Two 2x6 Pmod™-compatible peripheral interfaces for MicroZed
  - Supports interface for PS based SDIO/SPI
  - Supports interface for PL based I/O
- One 2x6 Pmod™-compatible peripheral interface to low-power MCU connector
  - Supports interface for I2C/UART

Figure 2 – MicroZed Carrier Card for Arduino Block Diagram
5 Arduino Carrier Card Basic Setup and Operation

The functionality of MicroZed, the Carrier Card and whichever shield or Pmod compatible expansion boards are attached, is determined by the application booted from MicroZed’s microSD card.

This *Getting Started Guide* provides system developers examples of how to achieve different solutions using MicroZed and this Carrier Card:

- a) MicroZed GPIO control of Relay outputs (using Seeed Studio Quad Relay Shield)
- b) MicroZed SPI QVGA TFT Display User Interface (plus I2C Cap Touch) (using Adafruit Cap Touch 2.8” QVGA TFT display Shield)
- c) MicroZed SDIO and UART based Wi-Fi and Bluetooth/BLE Interface (using Avnet Wilink 8 Pmod Adaptor)
- d) Microcontroller GPIO control of MicroZed power rails (using Freescale FRDM-KL46 Freedom board)

*Note:* Additional hardware is required in order to complete all of the listed examples:

- MicroZed SOM (Z7020 version)
- AC/DC Power Adaptor (microUSB connector, +5V 2A or better)
- Seeed Studio Quad Relay Shield (v2 or v3)
- Adafruit Capacitive Touch 2.8” QVGA TFT display Shield
- Avnet Wilink 8 Pmod Adaptor
- TI WL1835MODCOM8B or WL1837MODCOM8I Wi-Fi and Bluetooth board
- Freescale FRDM-KL46 Freedom MCU Eval board
- microSD card reader/adapter (for programming MicroZed’s uSD flash card)

A photo of the Arduino Carrier Card (plus the required MicroZed SOM) in it’s simplest configuration is shown below along with the locations of several key components.

5.1 Procedure

1. Navigate to [www.microzed.org/product/arduino-cc](http://www.microzed.org/product/arduino-cc)
   a) Select Products ▶ MicroZed Carrier Card Kit for Arduino
   b) Scroll down and click “View All” to see links to latest documentation
2. Return to [www.microzed.org/product/arduino-cc](http://www.microzed.org/product/arduino-cc)
   a) Scroll to bottom of page and click on Reference Designs and Tutorials
   b) Scroll down to MicroZed Carrier Card for Arduino
   c) Download the provided Getting Started Guide Examples Reference Design
3. Extract all files from this archive and follow the programming instructions provided
4. **Note:** If this your first use of a MicroZed SOM, it is recommended that you download and refer also to the Quick Start Instructions for MicroZed
5.2 Hardware Setup

Setup of typical development platform based on MZ Carrier Card for Arduino:

5. Attach Avnet MicroZed-7010 SOM to the Carrier Card
6. Attach target Shield board to JA1, JA2, JA3, JA4 connectors of the MicroZed Shield Expansion site and/or
7. Attach target Pmod board(s) to whichever connectors are designated
8. Attach +5V power using provided cable between the Carrier microUSB connector J7 and laptop computer
9. Attach USB Serial Console connection between MicroZed SOM connector J2 and laptop computer
10. Launch serial console app (Tera Term or equivalent), with the appropriate COM port set to 115200 bps.
5.3 Arduino Shield Interfaces Supported by Carrier Card

![Arduino UNO R3 Compatible Shield Connectors](image)

Figure 4 – Arduino UNO R3 Compatible Shield Connectors

5.4 Example Designs using Shields and Pmods

In the initial reference design, a single Zynq hardware platform definition is used to facilitate support of the four add-on board examples listed:

a) Seeed Studio Quad Relay Shield
b) Adafruit QVGA TFT Display Shield (plus Cap Touch input)
c) Avnet Wilink8 Wi-Fi and BT/BLE Pmod Adaptor
d) Freescale KL46 MCU controlling MicroZed power rails and Zynq power modes

The basic design (using Zynq GPIO) controls a Shield with multiple relay outputs and is readily tested as described below.

Avnet’s Wilink 8 Wi-Fi & Bluetooth Adaptor connected to Pmod-PS (J2) and Pmod-PL (J3) may be used concurrently with a shield connected to MicroZed’s shield connectors

More complex software examples are currently in development, including Linux supported low-cost graphical user interface using QVGA Touch Display (SPI) with Capacitive Touch overlay (I2C).
Figure 5 – Vivado IP Integrator View of Reference Design

Figure 6 – Zynq I/O Peripherals used to interface to Shield and Pmod Examples
Example #1: Quad Relay Shield (using GPIO)
The simplest expansion example is the addition of a Relay Shield. This allows MicroZed to switch multiple circuits of much higher voltages and currents, controlling devices (eg. motors, actuators, etc) that cannot be directly driven by Zynq's low voltage Digital I/Os.

A popular low-cost Relay Shield (sub $20) from Seed Studio is used in this example http://www.seeedstudio.com/wiki/Relay_Shield_V2.0
The control interface consists of 4 consecutive GPIO pins on the JA1 shield connector

<table>
<thead>
<tr>
<th>Microheader Pin #</th>
<th>MicroZed Pin Name</th>
<th>Zynq Pin #</th>
<th>Zynq GPIO #</th>
<th>Shield Pin #</th>
<th>Shield Signal Name</th>
<th>Relay #</th>
</tr>
</thead>
<tbody>
<tr>
<td>JX1, pin 47</td>
<td>JX1_LVDS_12_P</td>
<td>N18</td>
<td>55</td>
<td>JA1-8</td>
<td>ARD_D7</td>
<td>Relay 1</td>
</tr>
<tr>
<td>JX1, pin 49</td>
<td>JX1_LVDS_12_N</td>
<td>P19</td>
<td>56</td>
<td>JA1-7</td>
<td>ARD_D6</td>
<td>Relay 2</td>
</tr>
<tr>
<td>JX1, pin 48</td>
<td>JX1_LVDS_13_P</td>
<td>N20</td>
<td>57</td>
<td>JA1-6</td>
<td>ARD_D5</td>
<td>Relay 3</td>
</tr>
<tr>
<td>JX1, pin 50</td>
<td>JX1_LVDS_13_N</td>
<td>P20</td>
<td>58</td>
<td>JA1-5</td>
<td>ARD_D4</td>
<td>Relay 4</td>
</tr>
</tbody>
</table>

The **SysFs** Linux GPIO driver is used for control of these relays. This is the simplest approach as it does not require additional drivers or changes to the Linux device tree.

To implement this, only a minor addition was made to the default Zynq platform design: Four PS GPIO signals are routed via EMIO to the JX1 Microheader pins that connect to the applicable JA1 Shield connector pins.

Relay 1 can be turned-on directly from the serial console as follows:

**echo 55 > /sys/class/gpio/export**

**echo out > /sys/class/gpio/gpio55/direction**

**echo 1 > /sys/class/gpio/gpio55/value**

To turn-off Relay 1, write a zero to GPIO 55

**echo 0 > /sys/class/gpio/gpio55/value**

To control Relays 2, 3, 4, substitute their GPIO numbers (56, 57, 58) in place of the 55 The state of the Linux GPIO signals may be examined directly from the command line:

**cat /sys/kernel/debug/gpio**

Alternatively, this GPIO/relay control may be added to a user space C application (or suitable script or macro...)

Page 14
7 Example #2: TFT Touch Display Shield (SPI and I2C)

A more complex expansion example is the addition of a Shield with QVGA Display (SPI) and Cap Touch overlay (I2C).

This example is currently under development but once released, will facilitate a low-cost (sub-$45) User Interface for the MicroZed SOM.

From many low-cost TFT Touch Display Shields available from adafruit, their new Capacitive Touch 2.8” QVGA display (320x240, 18bit color) is targeted for this example.

7.1 Linux drivers used:
Frame Buffer TFT (fbtft_device) kernel driver

https://github.com/Xilinx/linux-xlnx/tree/master/drivers/staging/fbtft

Figure 8 – Adafruit 2.8” Cap Touch TFT Display Shield
TFT Display’s SPI interface uses 5 pins on the **JA2** shield connector:

<table>
<thead>
<tr>
<th>Microheader Pin #</th>
<th>MicroZed Pin Name</th>
<th>Zynq Pin #</th>
<th>Arduino Shield Pin #</th>
<th>Arduino Shield Signal Name</th>
<th>TFT SPI Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>JX1, pin 30</td>
<td>JB2-3 P</td>
<td>Bank 34, W14</td>
<td>JA2-6</td>
<td>D13</td>
<td>SPI_SCK</td>
</tr>
<tr>
<td>JX1, pin 32</td>
<td>JB2-3 N</td>
<td>Bank 34, Y14</td>
<td>JA2-5</td>
<td>D12</td>
<td>SPI_MISO</td>
</tr>
<tr>
<td>JX1, pin 35</td>
<td>JB4-5 P</td>
<td>Bank 34, T16</td>
<td>JA2-4</td>
<td>D11</td>
<td>SPI_MOSI</td>
</tr>
<tr>
<td>JX1, pin 37</td>
<td>JB4-5 N</td>
<td>Bank 34, U17</td>
<td>JA2-3</td>
<td>D10</td>
<td>SPI_SS</td>
</tr>
<tr>
<td>JX1, pin 36</td>
<td>JB6-7 P</td>
<td>Bank 34, V15</td>
<td>JA2-2</td>
<td>D9</td>
<td>SPI_DC</td>
</tr>
</tbody>
</table>

Touch Controller’s I2C interface uses 2 pins on the **JA4** shield connector:

<table>
<thead>
<tr>
<th>Microheader Pin #</th>
<th>MicroZed Pin Name</th>
<th>Zynq Pin #</th>
<th>Arduino Shield Pin #</th>
<th>Arduino Shield Signal Name</th>
<th>Touch I2C Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>JX2, pin 14</td>
<td>JX2_SE1</td>
<td>Bank 35, J15</td>
<td>JA4-5</td>
<td>A4</td>
<td>I2C_SDA</td>
</tr>
<tr>
<td>JX2, pin 18</td>
<td>JX2_LVDS_1_</td>
<td>Bank 35, B19</td>
<td>JA4-6</td>
<td>A5</td>
<td>I2C_SCL</td>
</tr>
</tbody>
</table>

Detail of the adafruit TFT Touch Shield hardware and software libraries is available at: [https://learn.adafruit.com/downloads/pdf/adafruit-2-8-tft-touch-shield-v2.pdf](https://learn.adafruit.com/downloads/pdf/adafruit-2-8-tft-touch-shield-v2.pdf)
7.2 Additional Reference Information (TFT Display):
https://communities.intel.com/message/266095
http://www.acmesystems.it/arietta_adafruit_lcd28
https://github.com/notro/fbtft/wiki

Further implementation detail of this solution will be made available shortly

8 Example #3: Wlink8 Pmod Adaptor (SDIO, UART)
The Arduino Carrier Card has been designed to accommodate Avnet's Wilink 8 Adaptor.

Pmod-PS connector J2 is used for the Wi-Fi (SDIO host interface)

Pmod-PL connector J3 is used for the Bluetooth/BLE (UART host interface)

A reference design with Linux driver support for WL1835MODCOM8B and WL1837MODCOM8I boards from Texas Instruments is available for download from: http://microzed.org/product/wilink-8-adaptor

8.1.1 For Wi-Fi operation:
Same pinout as the design for MicroZed IO Carrier Card
(as provided on Avnet's Wilink 8 Adaptor page).

8.1.2 For Bluetooth/BLE operation:
Pinout constraints in the Vivado design have been changed and the .bit file and BOOT.bin file regenerated, to accommodate this board’s different EMIO pinout for the connection between Bluetooth host UART interface and it's PL Pmod connector.

<table>
<thead>
<tr>
<th>Microheader Pin #</th>
<th>MZ Pin Name</th>
<th>Zynq Pin #</th>
<th>Pmod Pin#</th>
<th>PL I/O Name</th>
<th>Zynq Signal Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>JX2, pin 61</td>
<td>JX2_LVDS_14_P</td>
<td>G17</td>
<td>1</td>
<td>D0_P</td>
<td>UART0_RTSN</td>
</tr>
<tr>
<td>JX2, pin 63</td>
<td>JX2_LVDS_14_N</td>
<td>G18</td>
<td>2</td>
<td>D0_N</td>
<td>UART0_RX</td>
</tr>
<tr>
<td>JX2, pin 62</td>
<td>JX2_LVDS_15_P</td>
<td>F19</td>
<td>3</td>
<td>D1_P</td>
<td>UART0_TX</td>
</tr>
<tr>
<td>JX2, pin 64</td>
<td>JX2_LVDS_15_N</td>
<td>F20</td>
<td>4</td>
<td>D1_N</td>
<td>UART0_CTSN</td>
</tr>
<tr>
<td>JX2, pin 81</td>
<td>JX2_LVDS_20_P</td>
<td>N15</td>
<td>7</td>
<td>D2_P</td>
<td>BT_EN_SOC</td>
</tr>
</tbody>
</table>

# Wilink-8 Bluetooth Interface, implemented in the FPGA fabric
set_property PACKAGE_PIN G17 [get_ports UART0_RTSN]; # BT_HCI_CTS
set_property PACKAGE_PIN G18 [get_ports UART0_RX]; # BT_HCI_TX
set_property PACKAGE_PIN F19 [get_ports UART0_TX]; # BT_HCI_RX
set_property PACKAGE_PIN F20 [get_ports UART0_CTSN]; # BT_HCI_RTS
set_property PACKAGE_PIN N15 [get_ports BT_EN_SOC]; # BT_EN_SOC
Refer to the **WILINK 8 Getting Started Guide** document for more information. [http://microzed.org/product/wilink-8-adaptor](http://microzed.org/product/wilink-8-adaptor)

If used together with the Relay Shield (Example #1) plus the built-in Lighttp webserver, a system level solution can relatively easily be implemented, where the relay outputs can be controlled via Wi-Fi from a remote smartphone or tablet browser interface.

### 9 Example #4: Zynq Power Rail Control (using MCU GPIO)

The Low Power MCU Expansion site (JM1, JM2) on this Carrier Card facilitates the attachment of a companion (Arduino-compatible) microcontroller board to:

- Offload tasks from the Applications Processor to the Low-Power MCU
- Implement Low-Power Sensor Hub data collection and processing
- Power rail switching and Power mode control of the Zynq Apps Processor.

A basic example is available for download. This firmware provides pushbutton On/Off control of MicroZed's power rails, using a Freescale FRDM-KL46 Freedom board. Other more advanced power management schemes (Suspend/Resume, Wake-on-WLAN, etc) are also feasible with the hardware configuration provided.
Refer to the [www.microzed.org/product/arduino-cc](http://www.microzed.org/product/arduino-cc) website for more information.

Figure 11 – FRDM-KL46Z Eval board controlling MicroZed's Power Rails

**Note:** The signals connected between optional Freescale FRDM-K46F Freedom MCU board and this Carrier Card were carefully selected to also accommodate a specific use case where Freescale’s FRDM-FXS-MULTI sensor board is stacked below the MCU board.
10 Getting Help and Support

10.1 Avnet Support

The Arduino Carrier Card is a versatile development kit that extends the evaluation possibilities of the MicroZed SOM. All technical support is offered through the MicroZed.org website support forums. MicroZed users are encouraged to participate in the forums and offer help to others when possible.

http://MicroZed.org/forums/

For questions regarding MicroZed community website, please direct any questions to: MicroZed.org Web Master – webmaster@MicroZed.org

To access the most current collateral please visit the community support page at: www.MicroZed.org/content/support
Once on the MicroZed.org support page:

To access latest Arduino Carrier Card documentation and reference designs, click on the links provided at this board’s product webpage: www.microzed.org/product/arduino-cc

To access technical forums or online training and videos, click on the following links…

10.2 Xilinx Support
For questions regarding products within the Product Entitlement Account, send an e-mail message to the Customer Service Representative in your region:

– Canada, USA and South America – isscs_cases@xilinx.com
– Europe, Middle East, and Africa – eucases@xilinx.com
– Asia Pacific including Japan – apaccase@xilinx.com

For technical support including the installation and use of the product license file, contact Xilinx Online Technical Support at www.xilinx.com/support. The following assistance resources are also available on the website:

– Software, IP and documentation updates
– Access to technical support web tools
– Searchable answer database with over 4,000 solutions
– User forums