Zynq®-7000 AP SoC / Analog Devices Intelligent Drives Kit
Getting Started Guide
Version 1.0
Document Control

Document Version: 1.0
Document Date: 11/01/2104

Prior Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>11/01/2014</td>
<td>Initial Release \ MB</td>
</tr>
</tbody>
</table>
Contents

Avnet Design Kit Technical Support Files and Downloads Web Access Instructions ..... 4
  License Agreement.................................................................................................................. 4
Introduction ................................................................................................................................ 6
  What’s Inside the Box ........................................................................................................... 6
Getting Started With Zynq Intelligent Drives Kit ................................................................. 8
  Overview of the Motor Control Reference Design............................................................... 8
  Demo Preparations.................................................................................................................... 10
  Demo Requirements................................................................................................................. 10
  Hardware Setup....................................................................................................................... 11
    Updating the Reference Design [ CRITICAL STEP ] ....................................................... 13
Running the Demo ...................................................................................................................... 13
  MathWorks FOC controller...................................................................................................... 17
  Manual Mode......................................................................................................................... 18
  Dynamometer (optional).......................................................................................................... 19
  Restoring the SD Card image ................................................................................................. 21
Additional Documentation and Support .................................................................................... 25
Next Steps.................................................................................................................................. 25
Avnet Design Kit Technical Support Files and Downloads
Web Access Instructions

Thank you for purchasing an Avnet design kit. The technical support documents associated with this kit, including the User Guide, Bill of Materials, Schematics, Source Code and Application Notes, are available online. You, the Customer, can access these documents at any time by visiting Avnet's Design Resource Center ("DRC") at avnet.com/us/drc/support.

On your first visit to the DRC, you will be required to site register before you can download the documents. To get started, select the name of the manufacturer associated with your design kit from the drop down menu. A complete listing of available design kits will appear. Select the kit you purchased. Scroll to the bottom of the design kit page to access the support files. Before you download a file, you will be prompted to login. If you are an existing user, please login. If you are a new user, click on the “Need to sign-up?” text. Please complete the short registration form. Upon completion, be sure to retain your login and password information for future visits to Avnet’s DRC. Logging in once, gives you unlimited access to all technical support files and downloads. You will also have the chance to request e-mail notifications whenever there are updates to your design kit.

License Agreement
THE AVNET DESIGN KIT ("DESIGN KIT" OR "PRODUCT") AND ANY SUPPORTING DOCUMENTATION ("DOCUMENTATION" OR "PRODUCT DOCUMENTATION") IS SUBJECT TO THIS LICENSE AGREEMENT ("LICENSE"). USE OF THE PRODUCT OR DOCUMENTATION SIGNIFIES ACCEPTANCE OF THE TERMS AND CONDITIONS OF THIS LICENSE. THE TERMS OF THIS LICENSE AGREEMENT ARE IN ADDITION TO THE AVNET CUSTOMER TERMS AND CONDITIONS, WHICH CAN BE VIEWED AT www.avnet.com. THE TERMS OF THIS LICENSE AGREEMENT WILL CONTROL IN THE EVENT OF A CONFLICT.

1. Limited License. Avnet grants You, the Customer, ("You" "Your" or "Customer") a limited, non-exclusive, non-transferable, license to: (a) use the Product for Your own internal testing, evaluation and design efforts at a single Customer site; (b) create a single derivative work based on the Product using the same semiconductor supplier product or product family as used in the Product; and (c) make, use and sell the Product in a single production unit. No other rights are granted and Avnet and any other Product licensor reserves all rights not specifically granted in this License Agreement. Except as expressly permitted in this License, neither the Design Kit, Documentation, nor any portion may be reverse engineered, disassembled, decompiled, sold, donated, shared, leased, assigned, sublicensed or otherwise transferred by Customer. The term of this License is in effect until terminated. Customer may terminate this license at any time by destroying the Product and all copies of the Product Documentation.

2. Changes. Avnet may make changes to the Product or Product Documentation at any time without notice. Avnet makes no commitment to update or upgrade the Product or Product Documentation and Avnet reserves the right to discontinue the Product or Product Documentation at any time without notice.

3. Limited Warranty. ALL PRODUCTS AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. AVNET MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THE PRODUCTS AND DOCUMENTATION PROVIDED HEREBUNDER. AVNET SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY AGAINST INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF ANY THIRD PARTY WITH REGARD TO THE PRODUCTS AND DOCUMENTATION.

4. LIMITATIONS OF LIABILITY. CUSTOMER SHALL NOT BE ENTITLED TO AND AVNET WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND OR NATURE, INCLUDING, WITHOUT LIMITATION, BUSINESS INTERRUPTION COSTS,
LOSS OF PROFIT OR REVENUE, LOSS OF DATA, PROMOTIONAL OR MANUFACTURING EXPENSES, OVERHEAD, COSTS OR EXPENSES ASSOCIATED WITH WARRANTY OR INTELLECTUAL PROPERTY INFRINGEMENT CLAIMS, INJURY TO REPUTATION OR LOSS OF CUSTOMERS, EVEN IF AVNET HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE PRODUCTS AND DOCUMENTATION ARE NOT DESIGNED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN MEDICAL, MILITARY, AIR CRAFT, SPACE OR LIFE SUPPORT EQUIPMENT NOR IN APPLICATIONS WHERE FAILURE OR MALFUNCTION OF THE PRODUCTS CAN REASONABLY BE EXPECTED TO RESULT IN A PERSONAL INJURY, DEATH OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. INCLUSION OR USE OF PRODUCTS IN SUCH EQUIPMENT OR APPLICATIONS, WITHOUT PRIOR AUTHORIZATION IN WRITING OF AVNET, IS NOT PERMITTED AND IS AT CUSTOMER'S OWN RISK. CUSTOMER AGREES TO FULLY INDEMNIFY AVNET FOR ANY DAMAGES RESULTING FROM SUCH INCLUSION OR USE.

5. LIMITATION OF DAMAGES. CUSTOMER'S RECOVERY FROM AVNET FOR ANY CLAIM SHALL NOT EXCEED CUSTOMER'S PURCHASE PRICE FOR THE PRODUCT GIVING RISE TO SUCH CLAIM IRRESPECTIVE OF THE NATURE OF THE CLAIM, WHETHER IN CONTRACT, TORT, WARRANTY, OR OTHERWISE.

6. INDEMNIFICATION. AVNET SHALL NOT BE LIABLE FOR AND CUSTOMER SHALL INDEMNIFY, DEFEND AND HOLD AVNET HARMLESS FROM ANY CLAIMS BASED ON AVNET'S COMPLIANCE WITH CUSTOMER'S DESIGNS, SPECIFICATIONS OR INSTRUCTIONS, OR MODIFICATION OF ANY PRODUCT BY PARTIES OTHER THAN AVNET, OR USE IN COMBINATION WITH OTHER PRODUCTS.

7. U.S. Government Restricted Rights. The Product and Product Documentation are provided with “RESTRICTED RIGHTS.” If the Product and Product Documentation and related technology or documentation are provided to or made available to the United States Government, any use, duplication, or disclosure by the United States Government is subject to restrictions applicable to proprietary commercial computer software as set forth in FAR 52.227-14 and DFAR 252.227-7013, et seq., its successor and other applicable laws and regulations. Use of the Product by the United States Government constitutes acknowledgment of the proprietary rights of Avnet and any third parties. No other governments are authorized to use the Product without written agreement of Avnet and applicable third parties.

8. Ownership. Licensee acknowledges and agrees that Avnet or Avnet's licensors are the sole and exclusive owner of all Intellectual Property Rights in the Licensed Materials, and Licensee shall acquire no right, title, or interest in the Licensed Materials, other than any rights expressly granted in this Agreement.

9. Intellectual Property. All trademarks, service marks, logos, slogans, domain names and trade names (collectively “Marks”) are the properties of their respective owners. Avnet disclaims any proprietary interest in Marks other than its own. Avnet and AV design logos are registered trademarks and service marks of Avnet, Inc. Avnet's Marks may be used only with the prior written permission of Avnet, Inc.

10. General. The terms and conditions set forth in the License Agreement or at www.avnet.com will apply notwithstanding any conflicting, contrary or additional terms and conditions in any purchase order, sales acknowledgement confirmation or other document. If there is any conflict, the terms of this License Agreement will control. This License may not be assigned by Customer, by operation of law, merger or otherwise, without the prior written consent of Avnet and any attempted or purported assignment shall be void. Licensee understands that portions of the Licensed Materials may have been licensed to Avnet from third parties and that such third parties are intended beneficiaries of the provisions of this Agreement. In the event any of the provisions of this Agreement are for any reason determined to be void or unenforceable, the remaining provisions will remain in full effect. This constitutes the entire agreement between the parties with respect to the use of this Product, and supersedes all prior or contemporaneous understandings or agreements, written or oral, regarding such subject matter. No waiver or modification is effective unless agreed to in writing and signed by authorized representatives of both parties. The obligations, rights, terms and conditions shall be binding on the parties and their respective successors and assigns. The License Agreement is governed by and construed in accordance with the laws of the State of Arizona excluding any law or principle, which would apply the law of any other jurisdiction. The United Nations Convention for the International Sale of Goods shall not apply.
Introduction
The Zynq®-7000 AP SoC / Analog Devices Intelligent Drives Kit is designed as a flexible platform for motor control and industrial networking development. The kit is based on the Avnet ZedBoard™ using the Xilinx Zynq-7000 All Programmable SoC. The kit features the Analog Devices AD-FMCMOTCON1-EBZ FMC high performance servo drive module, providing motor drive up to 48 Volts at 18 Amps, precision motor phase current and voltage measurements, isolated power regulation, isolated signaling, and dual Gigabit Ethernet IEEE1588 PHYs for real-time Ethernet connectivity.

This Getting Started Guide will describe the steps to set up the Zynq-7000 AP SoC / Analog Devices Intelligent Drives Kit (Zynq IDK) and run the out-of-box demonstration.

For detailed information on ZedBoard, including tutorials, reference designs and board details please visit the website


For detailed information on the Analog Devices AD-FMCMOTCON1-EBZ FMC high performance servo drive module, including schematics, bill of materials, board file, and source code please visit their wiki page.

- http://wiki.analog.com/resources/eval/user-guides/ad-fmcmotcon1-ebz

What’s Inside the Box
- Avnet ZedBoard 7020 baseboard
  - Zynq-7000 SoC XC7Z020-CLG484-1
  - 512 MB DDR3
  - 256 Mb Spansion® Quad-SPI Flash
  - Onboard USB-JTAG Programming
  - 10/100/1000 Ethernet
  - USB OTG 2.0 and USB-UART
  - FMC expansion (low pin count)
  - 12V power supply
  - Multiple display options (Analog Devices ADV7511 1080p HDMI, 12-bit VGA, 128 x 32 OLED)
  - 4 GB SD card programmed with a simple Linux image that demonstrates basic capabilities of ZedBoard
- Analog Devices AD-FMCMOTCON1-EBZ high-performance servo FMC Module
  - Drive BLDC / PMSM / Brushed DC / Stepper motors up to 48V @ 18A
  - AD7401A Isolated 20MHz Sigma-Delta modulator
  - AD8251 Programmable Gain Amplifiers for full-scale current measurement
  - ADuM5000/7640 power and digital signal isolation
  - Dual Gigabit IEEE1588 Ethernet PHYs for real-time industrial communication
  - Xilinx XADC interface
  - Sensored or sensorless position measurement
  - 8 GB SD card programmed with ADI FMCMOTCON1 Base Reference Design
    - Zynq reference design for motor control featuring Analog Devices Ubuntu Linux framework
    - FAT32 partition: bootloader, devicetree blob and kernel image for system boot to UBUNTU Linux
    - EXT4 partition Linaro Ubuntu ARM root file system, Analog Devices IIO Oscilloscope
  - 24V power supply for FMC module motor drive stage
- Brushless DC motor
  - Hall Sensors and 1250 CPR indexed encoder
- Ethernet and USB cables
- USB dongle (male micro-B to female standard-A)
- Xilinx Vivado® Design Edition (device locked to XC7Z020)
- Optional Analog Devices Dynamometer
  - Included with AES-ZIDK-ADI-DYNO
  - Provides a compact, dynamic motor load test fixture
  - Includes one drive motor directly coupled to one load motor (generator)
  - Analog Discovery™ USB Oscilloscope for external control and monitor of Dyno signals
Getting Started With Zynq Intelligent Drives Kit

The Zynq-7000 AP SoC / Analog Devices Intelligent Drives Kit (Zynq IDK) comes with a ‘Getting Started’ motor control demonstration design flashed onto the 8GB SD card, which enables a single-board computer on the Avnet ZedBoard running UBUNTU desktop Linux. The system includes programmable logic-based interfaces and ARM-based Linux drivers for the Analog Devices High Performance Servo Solution on the FMC module.

Overview of the Motor Control Reference Design

The out-of-box reference design demonstrates Field Oriented Control (FOC) of a 3-phase brushless DC motor. The algorithm was first modelled and simulated in MathWorks Simulink software. Then code was generated using the MathWorks HDL Coder tool. Finally, the HDL code was integrated into a larger Vivado project. This IP core executes in the Programmable Logic (PL) of Zynq, while Linaro Ubuntu Desktop Linux runs user applications on the Zynq ARM processors, and communicates with the Zynq PL via AXI4-Lite and AXI4 DMA channels.

Digital PWM signals are driven from the Zynq SoC to the MOSFET driver stage on the FMCMOTCON1 module in order to deliver power to the motor. The AD7401A Sigma-Delta modulators sample two analog motor phase currents and the DC voltage bus, returning serial digital bitstreams to the Zynq SoC to be reconstructed with SINC3 low pass filters. The rotor position can be captured a variety of ways with the FMCMOTCON1 on-board circuitry, including rotary, Hall sensor, and resolver interfaces. In this reference design a rotary encoder communicates incremental position to the Zynq SoC for absolute position and speed estimation. All signals between the Zynq SoC and FMCMOTCON1 are digitally isolated. Finally, an HDMI monitor, mouse, and keyboard may be connected to ZedBoard to view system signals and enter control parameters using the ADI IIO Oscilloscope Linux application.

The design illustrates the following capabilities of the Zynq / Analog Devices Intelligent Drives Kit:

- Closed-loop velocity and torque control of a 3-phase brushless DC motor, implemented in Zynq programmable logic-based peripherals
- Field Oriented Controller (FOC) created by MathWorks using Simulink for simulation and HDL Coder to generate a programmable logic IP core
- SINC3 digital reconstruction filters for AD7401A sigma-delta modulator sampling of motor phase current and voltage
- High bandwidth DMA transfers of system data from programmable logic to DDR3 memory for consumption by Linux applications
- Linaro Ubuntu Desktop Linux executed on the Zynq Cortex-A9 processors to enable a single-board computer on ZedBoard
- Driving Linux-generated video content on the Analog Devices ADV7511 HDMI output interface
ADC Interface - Implements communication with the AD7401 sigma delta modulators present on the AD-FMCMOTCON1-EBZ and also the SINC3 filters for demodulating the 1-bit digital stream provided by these parts. This HDL block exposes a set of registers that can be accessed through the AXI4 Lite interface. An AXI4 Streaming interface connected to a DMA controller allows the block to stream real-time data to the application layer.

Controller Interface - Implements the interface to the IP control blocks in the system. An AXI4 Streaming interface connected to a DMA controller allows the block to stream real-time data to the application layer.

MathWorks FOC IP - The FOC controller model is provided by MathWorks and is integrated in the HDL design as a standalone IP core. The controller model is packaged into an IP core using the Simulink Workflow Advisor. It exposes a set of AXI4-Lite registers that can be used to control the IP operation as well as a set of interface signals for encoder input, current measurement data, inverter control, and internal operations monitoring. All monitoring signals connect to the Controller Interface IP which allows these signals to be monitored from the Linux IIO Scope application. The AXI4-Lite registers exposed by the controller IP core can be directly accessed through a UIO driver present in the ADI Linux distribution for Zynq.

Position & Speed Processing - Implements the algorithm for converting Hall, BEMF and Encoder signals into speed and position data. This HDL block exposes a set of registers that can be accessed through the AXI4 Lite interface. An AXI4 Streaming interface connected to a DMA controller allows the block to stream real-time data to the application layer.
Demo Preparations
The 8 GB SD card, labeled with Analog Devices logo, is factory-programmed with the Zynq firmware, Linux kernel image, user-space application software and file system for a stand-alone bootable system. There are several bootable Zynq designs on the SD card, so the first step is to copy the image for ZedBoard Motor Control into the boot directory.

1. Insert the SD Card labeled with Analog Devices logo into a laptop or PC.
2. Open the folder named zynq-zed-mc
3. Copy files BOOT.BIN and devicetree.dtb into the SD card root directory.
4. Eject the SD card and insert into ZedBoard (under FMC connector J1).

Note: the pre-formatted image on the SD card may be 6 or more months old. Therefore you will need to run an update script after the system is booted. This will be described in later steps.

Figure 2: Insert the SD Card

Demo Requirements
Accessories required for this demonstration, which are not included in the kit:

- HDMI monitor and cable
- USB mouse
- USB keyboard
- USB hub (3 or more ports)
Hardware Setup
Follow these steps closely to connect and configure the hardware to run the design.

1. **IMPORTANT**: Set the ZedBoard FMC IO voltage jumper to 2.5V to avoid any damage to the AD-FMCMOTCON1-EBZ.

![Figure 3: Setting ZedBoard FMC IO Voltage to 2.5V](image)

2. Set the ZedBoard boot-mode jumpers to boot from the SD card:

![Figure 4: Setting boot-mode jumpers to boot from the SD card](image)

**Assemble the kit as shown below:**
3. Connect the power sources as shown
   - 12V DC plugs into ZedBoard
   - 24V DC plugs into FMCMOTCON1 module (P4)
   - optional: 5V DC plugs into the Dynamometer (P3 under the metal bezel)

4. Connect an HDMI cable to ZedBoard and a monitor with at least 720p resolution.

5. Connect the motor encoder and phase leads as shown in Figure 5 (a) and (c) using the supplied cable. These connections are the same whether connecting a standalone motor or the motor housed in the AD- DYNO-EBZ fixture. The 5-wire cable permanently attached the motor is connected to the internal Hall Sensors which are not used in the Field Oriented Control demo.

6. The FMCMOTCON1 emergency stop switch (S2) shown in Figure 5 (b) must be toggled to the “UP” position to enable the power MOSFETs. Press the ‘Power Stage Reset’ once to clear the latch for the protection circuit.

7. Connect the included USB dongle to the USB OTG port, and then attach a USB mouse and keyboard through a USB hub (not included).

8. Provide internet connection by attaching the Ethernet cable to ZedBoard and a networked router. This is required to download and install the latest design updates.

9. Turn ZedBoard power switch (SW8) to ON.

10. Wait approximately 45 seconds for Linux to boot and display on your monitor. The Analog Devices IIO oscilloscope application will launch automatically.
Updating the Reference Design [ CRITICAL STEP ]

The pre-formatted image on the SD card may be 6 or more months old. Therefore you will need to run a simple update script to retrieve the latest files from the ADI wiki. This will be performed in-system using Linux, which was booted on ZedBoard in the previous steps. The process takes more than a few minutes to complete, so please be patient.

The SD card contains a FAT32 partition with multiple boot files for different Zynq target platforms. The SD card also contains an ext4 partition containing the persistent Linux file system. Both partitions will be updated in the following steps.

1. Press CTRL + ALT + T to open a Linux terminal window.
2. Enter the following command twice to update the Linux applications. The first time retrieves the latest update script. The second time updates all of the user space tools.
   
   analog@linaro-ubuntu-desktop:$ sudo adi_update_tools.sh

3. Enter the following command once to update the Zynq boot files from the ADI wiki.
   
   analog@linaro-ubuntu-desktop:$ sudo adi_update_boot.sh

4. Type the following command and wait for the system to reboot.
   
   analog@linaro-ubuntu-desktop:$ sudo reboot

Note: If these steps fail it may indicate that your office firewall has blocked access to the ADI wiki site. In that case see the instructions at the end of this document for re-imaging your SD card with an recent archive hosted by Avnet.

More details on this update process can be found on the Analog Devices wiki “Zynq Quick Start Guide”
http://wiki.analog.com/resources/tools-software/linux-software/zynq_images

Running the Demo

With the Zynq IDK hardware connected and firmware updated you are ready to run the Field Oriented Control demo. After the previous steps the system is booted on Zynq and a Linux desktop is presented on your HDMI monitor.
1. Press **CTRL + ALT + T** to open a Linux terminal window.
2. Go to the following directory:
   ```
   cd /usr/local/bin
   ```
3. Enter the following command to start the encoder calibration script:
   ```
   sh foc_script.sh
   ```

   ```ini
   Initializing the FOC controller ...
   w: reg[0x100] = 0x03
   w: reg[0x120] = 0x0
   w: reg[0x104] = 0x61a8
   w: reg[0x118] = 0x0
   w: reg[0x11c] = 0x51f
   FOC controller initialized. Select Run in the IIO Scope and hit Enter to continue.
   ```

4. In the **ADI IIO Oscilloscope** window select the **Motor Control** tab and click the **Controller Type** to select **Matlab Controller**
5. Check the ‘Run’ box in the IIO Scope to start the motor spinning in open loop.

   ![ADI IIO Oscilloscope](image)

   **NOTE:** If the motor does not start spinning, verify that step 6 in Hardware Setup was completed (Emergency stop is UP; press Power Stage Reset once)

6. In the **ADI IIO Oscilloscope** – Capture 1 window select **ad-mc-adc**: voltages 0-4 for display.
   - voltage0: phase current IA
   - voltage1: phase current IB
   - voltage2: total motor current IT
   - voltage3: motor DC voltage VBUS

7. In the **ADI IIO Oscilloscope** – Capture 1 window increase the Sample Count to 4000.
8. Press the Capture/Stop button to begin sample acquisition.
At this point the motor is spinning in open loop, drawing more current than is necessary. The controller is creating a rotating magnetic field in the stator windings that the permanent magnet rotor is following. However, we must now calibrate the motor encoder to the system in order to receive accurate position measurement required to closed the control loop and commutate the motor efficiently. The calibration routine is executed on the Zynq ARM processor under Linux. The algorithm estimates the amount of offset between the encoder mechanical position and the motor electrical position. The Field Oriented Control loop attempts to hold the Id current vector (rotor flux) at a DC value of zero. The calibration routine estimates and applies the encoder offset required in order to reach $I_d = 0$.

9. In the Linux terminal window hit **Enter** to complete the encoder calibration and begin closed loop operation.
10. Watch the Linux terminal output. You should observe the OFFSET ERR converge to an integer number below 100 within 5-10 seconds.

```
OFFSET / OFFSET ERR: 3021 / 883
w:  reg[0x120] = 0xffffffff
OFFSET / OFFSET ERR: 2971 / 464
w:  reg[0x120] = 0xffffffff
OFFSET / OFFSET ERR: 2931 / 55
w:  reg[0x120] = 0x36ff
OFFSET: 14079
Initializing the closed loop FOC mode... w:
  reg[0x108] = 0x5dc
w:  reg[0x10c] = 0xfa
w:  reg[0x110] = 0x5dc
w:  reg[0x114] = 0x96
w:  reg[0x104] = 0x9c40
Running in closed loop mode w:
  reg[0x100] = 0x02
```

11. If the OFFSET ERR increases indefinitely, press CTRL + C to stop the script, and then run `foc_script.sh` again and repeat steps 9 -11. The algorithm may require a few attempts to converge.

12. The Linux terminal output will indicate when the motor is running in closed loop.

13. In the IIO Scope notice the reduced current consumption now that the control loop is closed.
MathWorks FOC controller
You may also observe the signals of the MathWorks FOC controller, including the space vector modulation (SVM) voltage waveforms.

1. Press the Capture/Stop button to stop sample acquisition.
2. In the ADI IIIO Oscilloscope – Capture 1 window go to ad-mc-adc: and de-select voltages 0-4. Now select ad-mc-ctrl: voltages 0-4
   - voltage0: phase A voltage
   - voltage1: phase B voltage
   - voltage2: phase A current
   - voltage3: phase B current
3. Press the Capture/Stop button to begin sample acquisition.
This plot shows the relationship between the SVM waveforms and the two phase currents being driven to the motor.

**Manual Mode**
The ADI reference design includes a manual mode that implements 6-step commutation of the motor, allowing the user to specify rotation direction and PWM duty cycle. There is no velocity or torque control loop.

1. In the ADI IIO Oscilloscope control window stop the motor by de-selecting the Run box.
2. Remove the encoder cable from the FMCMOTCON1 connector.
3. Attach the Hall Sensor cable as shown below.

4. In the ADI IIO Oscilloscope control window click the Controller Type to select Manual PWM.
5. Click the Run box to start spinning the motor.

6. Change PWM field to adjust the velocity of the motor.
**Dynamometer (optional)**

A basic dynamometer ("dyno") may be purchased with the Zynq Intelligent Drives Kit. The dyno provides a test fixture to dynamically apply a load to the motor that is driven by the Zynq IDK. A 3-phase MOSFET bridge, under command of an integrated digital controller, adjusts the resistance seen at the phase leads of the load motor. An LCD and 3 push buttons provide a user interface to dynamically change load profiles and view measurement of the load motor speed and current. More details, including schematics, bill of materials, and board files can be found at the Analog Devices wiki.


Current and speed may also be measured using the USB-powered Analog Discovery™ module, included with the dyno. In addition, the 3-phase MOSFET bridge can be controlled with the digital I/O of the Analog Discovery module. This module is controlled from a PC GUI, or can be controlled directly from MATLAB using the Data Acquisition Toolbox. More information can be found at the MathWorks website.

In order interface the Dyno with the Analog Discovery™ USB Oscilloscope:

- Slide switch S1 to EXT_CTRL position
- Insert the Analog Discovery™ in P1 the connector(with the Analog Devices logo facing the LCD)

Two software packages are available for interfacing with the Analog Discovery™:

- Digilent® WaveForms™
  - [http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,66,849&Prod=WAVEFORMS](http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,66,849&Prod=WAVEFORMS)
- MathWorks Analog Discovery toolbox

The signals available to the Analog Discovery are:

<table>
<thead>
<tr>
<th>Dyno Signal</th>
<th>Analog Discovery channel</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_A</td>
<td>Scope Channel 1 Positive</td>
<td>Phase A motor current (185mv/A)</td>
</tr>
<tr>
<td>I_B</td>
<td>Scope Channel 2 Positive</td>
<td>Phase B motor current (185mv/A)</td>
</tr>
<tr>
<td>PWM1</td>
<td>Digital Channel 0</td>
<td>Phase A PWM (3.3V levels)</td>
</tr>
<tr>
<td>PWM2</td>
<td>Digital Channel 2</td>
<td>Phase B PWM (3.3V levels)</td>
</tr>
<tr>
<td>PWM3</td>
<td>Digital Channel 2</td>
<td>Phase C PWM (3.3V levels)</td>
</tr>
</tbody>
</table>

Congratulations!
You have now run the Getting Started demonstration with the Zynq-7000 AP SoC / Analog Devices Intelligent Drives Kit. Using this fully functional framework with embedded LINUX, you may now develop your motor control applications.

For more detailed information about the Analog Devices AD-FMCMOTCON1-EBZ module or this reference design, please visit their User's Guide wiki at:

Restoring the SD Card image
During the course of development, should the 8 GB SD card become corrupted or otherwise need to be restored to a known good state, the directions below will restore the factory image.

These steps will overwrite the contents of the SD card, so be certain that there is no existing data that needs to be retrieved from the SD card prior to following these steps.

The following directions were performed on a Windows 7 Professional machine, but the commands should work in a similar manner with later versions of Windows.

1. Download the image archive from ZedBoard.org. The instructions in this document assume that the archive is downloaded to a temporary folder which on this example system is under the C:\work\ directory.
   http://zedboard.org/product/zynq-motor-control

2. Power down ZedBoard and remove the 8 GB SD card. Insert the SD card into the Windows PC.  
   Note: The provided image file is a byte-for-byte copy of an 8 GB SD card. Ensure the size of your target SD card is an 8 GB device.

3. Open a Windows Command Prompt session with Administrator privileges by right clicking the Start → Accessories → Command Prompt menu item and selecting the Run as administrator option.

4. If multiple partitions exist on the SD card, each of the partitions should be removed using the diskpart utility. This will enable the entire SD card block range to be utilized for writing the image. Launch the diskpart tool from the Windows Command Prompt.

> diskpart

5. Discover the disk number of the SD card device by using the list disk DiskPart command. In this example, the SD card device is enumerated as the Disk 1 device.
6. Select the disk listing that matches the SD card device using the select disk DiskPart command. In this example, the SD card device is enumerated as the Disk 1 device. If the SD card is listed as a different disk on your system, be sure to substitute the appropriate value here.

**WARNING:** Selecting the incorrect disk at this point can corrupt critical data on your local machine, be sure to check that the selected disk value is the intended device.

```
DISKPART> select disk 1
```

7. List the partitions on the currently selected SD card using the list partition DiskPart command.

```
DISKPART> list partition
```

Figure 8 – Listing the Enumerated Disk Devices

Figure 9 –Listing the Partitions on the Selected SD Card
8. Remove the partitions from the currently selected SD card using the clean DiskPart command

```bash
DISKPART> clean
```


```bash
DISKPART> exit
```

10. Extract the image contents of the archive obtained from Step 1 above using an xz compatible utility such as 7-Zip.

11. Launch a disk imaging utility such as Win32 Disk Imager.

12. Select the source image file zed_fmcmtcon1_image_8gb_2014_10_30.img and choose the target drive device. In this example, the image is located in the C:\work\folder and the target SD card is assigned to the G:\ drive.

![Figure 10 – Source Image File Selected](image)

13. (Optional) Verify the integrity of the extracted image file using the Win32 Disk Imager MD5 Hash tool by clicking on the checkbox option. The checksum calculation may take a few minutes.

The result from the MD5 Hash tool should match the string **C5C8C6C122319A0F7B63BFBD29D63A8**. If the checksums do not match, the firmware image is corrupt or out of date.¹

![Figure 11 - Verifying Image File Integrity via MD5 Hash](image)

14. Click the Write button to begin writing the source image file contents to the SD card. The SD imaging process can take around 20 to 30 minutes to complete the 8GB transfer.

¹ *Note: checksum current at the time of writing (Nov 2014). For updates consult [http://zedboard.org/product/zy3q-motor-control](http://zedboard.org/product/zy3q-motor-control)*
15. Use Windows “safely remove” to eject the SD card.

![Image of safely remove window]  
Figure 12 - Windows “safely remove” to eject the SD card

16. Replace the re-imaged SD card into ZedBoard. The Zynq-7000 SoC / Analog Devices Intelligent Drives Kit is ready for operation with the latest SD card image.

![Image of ZedBoard with SD card]  
Figure 13 – Insert the re-imaged SD card into ZedBoard

This concludes the procedure of re-imaging the SD card.
**Additional Documentation and Support**

To access the most current collateral for the Zynq-7000 SoC / Analog Devices Intelligent Drives Kit please visit the product website:

http://zedboard.org/product/zynq-motor-control

http://www.avnet.com/zynq7000idk

To access the Avnet Technical Community Forums, please visit the following web page:

http://www.zedboard.org/support

Analog Devices AD-FMCMOTCON1-EBZ high-speed analog module user guide and wiki:

http://wiki.analog.com/resources/eval/user-guides/ad-fmcmotcon1-ebz

To search the Xilinx database of silicon and software questions and answers or to create a technical support case in WebCase, see the Xilinx website:

http://www.xilinx.com/support

**Next Steps**

Ensure Xilinx Vivado Design Edition is installed on your computer. A voucher for licensing the Xilinx software is included with this kit. For technical support, including the installation and use of a product license file, contact Xilinx Online Technical Support at http://www.support.xilinx.com

Download Zynq embedded system source files for this Getting Started demonstration design as a starting point for your system customization.²

http://wiki.analog.com/resources/eval/user-guides/ad-fmcmotcon1-ebz/reference_hdl/linux

Consult the Zynq-7000 SoC / Analog Devices Intelligent Drives Kit product website for notification of technical training.

http://zedboard.org/product/zynq-motor-control

http://www.avnet.com/zynq7000idk

² Zynq embedded system source files are available for subsequent customization of the reference design, but are not required to run this Getting Started demonstration. The system is stand-alone bootable from the SD card provided with the kit, which includes the bitstream within the BOOT.BIN file in the FAT32 partition.