ANALOG DEVICES

Quick Start Guide for EV-MCS-LVDRV-Z Motor Drive Evaluation Platform

Rev. 0.1

Created: Author: Last Modified: Modified by:

12/17/2014 12:17 PM Dara O'Sullivan 1/6/2015 12:06 PM Dara O'Sullivan

1 Contents

| 1 | 1 Contents | |
|---|--|----|
| 2 | 2 Revision History | |
| 3 | 3 Overview | |
| | 3.1 System requirements | |
| 4 | 4 Hardware Setup | |
| | 4.1 Low Voltage Board | |
| 5 | 5 Software Setup | |
| | 5.1 Programming with Serial Downloader | 8 |
| | 5.2 Programming with Segger J-Link | 10 |
| 6 | 6 GUI Configuration | |
| 7 | - | |
| 8 | - | |
| 9 | 9 Support | |

2 Revision History

| Version | Modified By | Date | Comments |
|---------|--------------------|----------|---------------------|
| 0.1 | Dara O'Sullivan | 1/6/2015 | Document finalized. |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

3 Overview

This document will give a high level introduction to the EV-MCS-LVDRV motor control development platform and will provide a step-by-step approach that will get a motor up running. Instructions are provided on hardware setup, executable download, and graphical user interface (GUI) operation.

3.1 System requirements

Before you start working on the motor control platform, please make sure you have the hardware and software listed below.

Required Hardware

- ADSP-CM408F EZ-KIT rev 0.2 (This is ordered separately from the EV-MCS-LVDRV-Z)
- EV-MCS-LVDRV-Z power board, including BLY171D-24V-6000 motor with encoder, power supply and USB to serial cable

Optional Hardware

- Segger J-Link Lite debugger (This comes with the ADSP-CM408F Ezkit and can be used for executable download, and code development, but is not necessary for basic setup)

Required Software

- ADSP-CM40x SW Enablement Package version 1.2.0 (available here : http://sdk.analog.com/dw/sdks.aspx?file=ADUSC03)
- ADIMonitor Graphical User Interface (available here: <u>https://ez.analog.com/docs/DOC-11971</u>)
- Motor Control demo program executable and linker map file (also available here https://ez.analog.com/docs/DOC-11971)

Optional Software

- IAR Embedded Workbench (for code development, version 6.6 or higher)
- Segger J-Link Lite driver software

4 Hardware Setup

This section will describe how to setup the hardware. This only has to be performed once, when bringing up a new platform.

4.1 Low Voltage Board

Connect the EZ-kit to the Power Board as shown in Figure 1. Make sure both Samtec connectors mate completely. Also, note the location of Encoder, Power and Motor connectors.

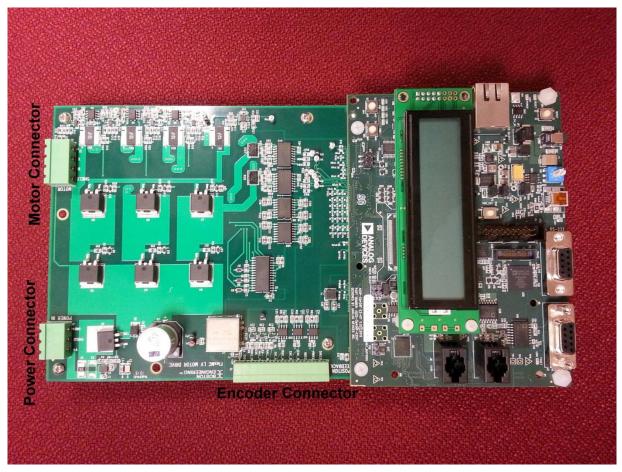


Figure 1 Low Voltage Power Board connected to CM408F EZ-kit.

Ensure that the Encoder connector is wired as shown in Figure 2 and Table 1.



Figure 2 Wiring of Encoder Connector.

| Pin | Color | Signal |
|-----|-----------------------|--------|
| 1 | Red | +5V |
| | (thick and thin wire) | |
| 2 | Black | GND |
| | (thick and thin wire) | |
| 3 | Green | HALL_U |
| 4 | Blue (thick) | HALL_V |
| 5 | White | HALL_W |
| 6 | Blue (thin) | ENC_A+ |
| 7 | Orange | ENC_A- |
| 8 | Yellow | ENC_B+ |

| 9 | Grey | ENC_B- |
|----|------|--------|
| 10 | NC | INDEX+ |
| 11 | NC | INDEX- |
| 12 | NC | Shield |

 Table 1 Encoder Connector

Ensure that the Motor and Power Connectors are wired as shown in *Figure 3*, *Table 2* and *Table 3*.

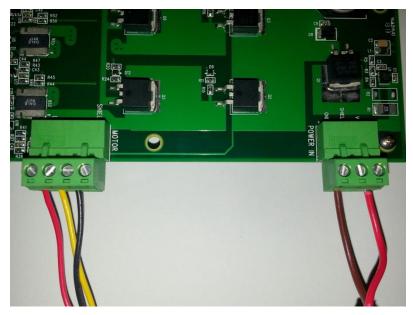


Figure 3 Wiring of Motor- and Power Connector.

| Pin | Color | Signal |
|-----|--------|---------------|
| 1 | Red | Motor phase U |
| 2 | Yellow | Motor phase V |
| 3 | Black | Motor phase W |
| 4 | NC | Shield |

Table 2 Motor Connector

| Pin | Color | Signal |
|-----|-------|--------|
| 1 | Brown | GND |
| 2 | NC | Shield |
| 3 | Red | +24V |

Table 3 Power Connector

5 Software Setup

The software setup steps are as follows:

- 1. Download and install the ADSP-CM40x SW Enablement Package version 1.2.0 from the link shown previously in "SW Requirements". This includes the serial boot-loader which is needed for download of the executable to the processor.
- 2. Download and install the GUI from the Engineer Zone link provided in "SW Requirements". This requires the .NET framework to be on the PC and it will prompt the user to download this if it is not detected.
- 3. Download the motor control demo executable program from the Engineer Zone link provided in "SW Requirements" and program this to the processor board (ADSP-CM408 EZkit).

The first two steps are fairly self-explanatory. The third step – programming of the executable to the processor – can be carried out in two alternate ways. These are detailed in this section.

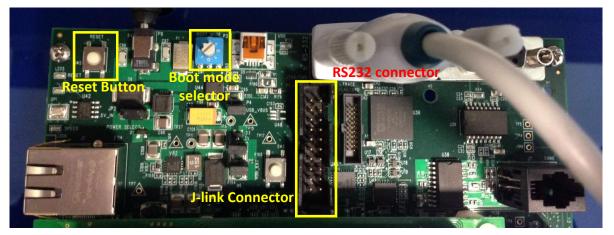
5.1 Programming with Serial Downloader

The serial downloader ("wsd.exe") is provided as part of the ADSP-CM408 SW Enablement Package and once this has been installed, assuming default installation directory structures, the downloader is found in C:\Analog Devices\ADSP-CM40x\CM403F_CM408F_EZ-KIT\tool\UARTFlashProgrammer. This method uses the **MC_Demo.hex** executable provided at the Engineer Zone link.

Steps:

1. Connect the EZ-kit to the PC using the USB-UART cable: UART (RS-232) connector on the EZkit, USB on the PC

2. Select Boot Mode 3 (UART Boot) on the EZkit using the selector switch P3 (adjacent to the power connector)



- 3. Power up the EZkit
- 4. Open the Windows Serial Downloader (wsd.exe) and select the appropriate options:

Target: ADSP-CM40x (flash)

Serial Port: Select the COM port to which the USB-serial cable is connected (if in doubt, check under 'Device Manager->Ports'

Baudrate: 115200

5. Click "Browse" under "File to download" and navigate to the provided executable "MC_demo.hex"

6. Select "Start"

| Windows Serial Downloader | | |
|--|-----------------------------------|-------------|
| Second stage kernel | | |
| C:\Analog Devices\ADSP-CM40x\CM40 | 3F_CM408F_EZ-KIT\tool\UARTFlash | Proc Browse |
| File to download | | |
| C:\Users\DOSulli2\Documents\SVN\Pr | aiaat DM Matay EOC\ SW/ Taga\u020 | LV E Browse |
| C. TOSERS (DOSUIIS (DOCUMERIS (SAM) Pr | UJEC PM MOULTPOC (SW Tags (V020_ | |
| Target | Serial Port | Baudrate |
| ADSP-CM40x (flash) | COM3 (USB Serial Port) 📃 💌 | 115200 💌 |
| Flash Action | | |
| Program | | |
| Frogram | | |
| Status | | |
| Sending second stage kernel. Read binary boot stream with 8756 byte | | |
| Verified boot stream. | 5. | |
| Autobaud succeeded. | | |
| Downloaded 8756/8756 bytes. Download completed. | | |
| Programming flash image. | | |
| Read Intel HEX flash image with 35124 b Autobaud succeeded. | oytes. | |
| Erased 9/9 pages. | | |
| Erase completed. Flashed 35124/35124 bytes. | | |
| Flash completed. | | |
| | | |
| | | |
| | Start Reset | Cancel |

7. Once the operation completes, select Boot Mode 1 (SPI Flash Boot) on the EZkit, using the selector switch P3 (adjacent to the power connector)

8. Reset the EZkit (SW2 adjacent to power connector)

At this point the application should now be executing or loaded from flash, and the GUI can be connected to run the motor.

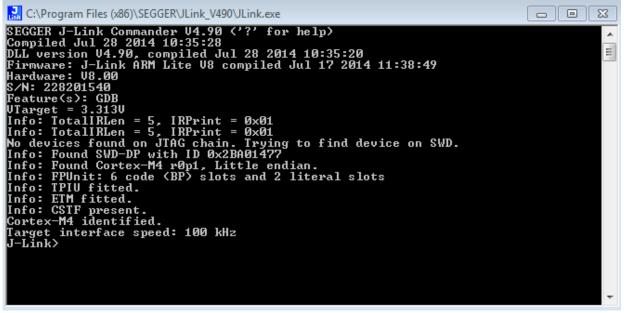
5.2 Programming with Segger J-Link

This method uses the JLink software and the debugger provided with the EZkit to program the processor flash memory.

- 1. Download and install Segger JLink Software (http://www.segger.com/jlink-software.html)
- 2. Click the blue download button under Windows. The version number may be different from the one shown below.

| | 3001 (J-Link User Guide), <u>UM08003</u> (J-Flash User Guide), <u>UM08004</u> (RDI User Guide) |
|------------|---|
| | for <u>J-Link DLL</u> , <u>J-Flash</u> , <u>J-Link RDI DLL</u> . |
| | ng sample projects for most popular eval boards. Support for ARM RDI standard. Makes J-Link compatible with RDI compliant debugger |
| ↓ Download | Software and documentation pack for Windows V4.90 [20,125 kb] md5 checksum: 8c3335d09848ef8fdc66cec27d60b1a3 |

- 3. Enter your Segger JLink Lite serial number in the next box. This number is printed on the microcontroller on the debugger board.
- 4. Click the confirmation box on the next screen, download the software and install (default location is C:\Program Files(x86)\Segger)
- 5. Connect the JLink to the PC USB port and the other end to the 20 pin connector on the CM40x board.
- 6. The PC will start automatically start installing drivers for your Jlink . Let it complete and turn on power to the EZkit.
- 7. Turn on power to the CM408x board.
- 8. Go to the location where Segger was installed and click on JLink .exe. In this case it is in: C:\Program Files (x86)\SEGGER\JLink_V490 (The version number and folder name may be different to the one in this example.)
- 9. The following should be displayed in the Jlink window. It should identify that a Cortex-M4 is connected to the Jlink .



- Enter the following at the prompt: exec device = ADSP-CM408BSWZ-BF The following will be displayed indicating that the chip has been identified correctly: J-Link>exec device = ADSP-CM408BSWZ-BF Info: Device "ADSP-CM408BSWZ-BF" selected (2048 KB flash, 64 KB RAM).
- 2. Copy the binary file you need to load into the CM40x in a known location. For this example, we will be placing the file **MC_demo.bin** in C:\temp

3. In the JLink window, type the following to load the binary code at address 0x1800 0000, which is the start of flash.

```
loadbin c:\temp\MC_demo.bin, 0x18000000
J-Link>exec device= ADSP-CM408BSWZ-BF
Info: Device "ADSP-CM408BSWZ-BF" selected (2048 KB flash, 64 KB RAM).
J-Link>loadbin c:\temp\MC_demo.bin, 0x18000000
```

A new window pops up, indicating progress:

| SEGGER J-Li | nk V4.90 - Flash download (2048 KB) | |
|-------------|---|--------|
| Compare | 90.6% | 0.944s |
| Erase | 0.0% | |
| Program | 0.0% | |
| Verify | 0.0% | |
| Cancel | Comparing range 0x181A0000 - 0x181BFFFF (2 Sectors, 128 KB) | 0.944s |

The JLink window will also show information about the flash procedure. J-Link>loadbin c:\temp\MC_demo.bin, 0x18000000 Halting CPU for downloading file. Downloading file... [c:\temp\MC_demo.bin] Info: J-Link: Flash download: Flash programming performed for 1 range (65536 by es) Info: J-Link: Flash download: Total time needed: 6.429s (Prepare: 0.665s, Compar e: 0.009s, Erase: 0.712s, Program: 4.971s, Verify: 0.015s, Restore: 0.055s) J-Link>

4. Reset the EZkit (SW2 adjacent to power connector)

At this point the application should now be executing or loaded from flash, and the GUI can be connected to run the motor.

6 GUI Configuration

Once the installation zip file is extracted, the GUI application is loaded by running the installer executable "ADIMonitorGUI.exe". This GUI requires the .NET framework and this will be downloaded if it is not already on the PC. The first step is to set up the serial port (having connected the USB-serial converter to the EZkit RS232 port). This is accomplished by right-clicking the bottom right-hand corner of the GUI and selecting *Connect*. The program should automatically select the correct COM port, but if several options are available, right-click *Configure* and select the correct one. Once this is selected press, *Save* and then *Connect*.

| TOURE CONTROL AND MONIT | TOR. | | | CONFIGURE CONTROL AND MONITO | R | | | | | |
|-------------------------|-------------------------------------|---------------------|---------|------------------------------|---------|---------------------|--------------|-------|-----------|-----|
| otor | Trigger | Downsampling | | Motor | Trigger | | | | Downsampl | ing |
| VF GAIN 7 | Auto O Single Delay 100 | 1 | 1 | VF GAIN 7 | Auto | Serial Port C | onfiguration | 8 | 1 | |
| F MAX RATE 10 | Address Address Prefix UID_ Filter | Butter | | VF BOOST 0 VF MAX RATE 10 | Address | Port | COM3 | - | Buffer | |
| MAX 2 MAX RPM 2000 | Plot # Name Address Type Plot | Count 0 Size 200 | \prec | I MAX 2 | Plot | Baud Rate | 57600 | | Count | 0 |
| F CTRL 1 | | | | MAX RPM 2000 | | Data Bits Parity | 8 None | | Size | 200 |
| CCW Rotation | | | | VF CTRL 1 | | Stop Bits | One | v | | |
| | Load Map Clear Update Configuration | | × | Configure | | Handshake | None | · | | |
| | | | | | Loa | Save | | ancel | | |
| | | | | | | | | | | |
| | | | | | | | | | | |

Figure 4: Serial Port Setup

The next step is to configure the motor control settings. These are listed in *Table 4*, along with an explanation of their relevance. The default settings are for open loop Volts/Hz control. Closed loop field-oriented control (FOC) can also be selected by changing the setting for VF_CTRL. (The executable provided will run in either mode; however, sample C code is only provided and supported for open loop mode).

| Parameter | Open Loop V/f control | Closed loop FOC |
|-----------------|-------------------------------------|--|
| VF_GAIN | Volts per Hz representation | Only used for open loop |
| VF_BOOST | Low speed | Only used for open loop |
| | boost voltage representation | |
| VF_MAX_RATE | RPM change per 0.1 ms sample | Only used for open loop |
| I_MAX | Not used | Not used |
| MAX_RPM | M | ax rpm |
| VF_CTRL | Set to 1 for open loop V/f | Set to 0 for closed loop FOC |
| CCW rotation | Check for CCW rotation. Uncheck for | CW rotation. When looking into shaft end |
| | of | motor. |
| Speed reference | Speed | l reference |

Table 4: Motor Control Settings

Once the configuration settings have been entered, press *Configure*. Typical settings for open loop control are depicted in *Figure 5*. Please note that the VF_GAIN and VF_BOOST settings do not directly refer to volts/Hz or volts quantities. Care should be taken when setting the VF_GAIN quantity. If it is set too high, the motor will draw large currents and an overcurrent trip may occur. Also if the VF_MAX_RATE is set too high, the applied motor voltage and frequency will ramp too quickly – especially if a large step change in speed reference occurs – potentially causing an overcurrent trip. If an overcurrent or PWM trip occurs, set the GUI start/stop command to *Stop* and reset the EZkit using the reset switch. Reduce VF_GAIN or VF_MAX_RATE to limit the steady state and transient current amplitudes.

| tor | | Trigger | Downsampling |
|------------|------------|-------------------------------------|--------------|
| 'F GAIN | 7 | Auto O Single Delay 100 | 1 |
| F BOOST | 0 | | Buffer |
| F MAX RATE | 10 | Address Prefix UID_ Filter | Buffer |
| MAX | 2 | Plot # Name Address Type Plot | Count 0 |
| IAX RPM | 2000 | | Size 200 |
| F CTRL | 1 | | |
| | / Rotation | Load Map Clear Update Configuration | n |

Figure 5: Motor Control Configuration – Typical open loop settings.

7 Running the Motor

In order to control the motor, navigate to the *Control and Monitor* tab. From here, the motor can be started by pressing the *Start* button, and the speed varied by moving the slider or entering a speed in the numeric box. This is illustrated in Figure 6.

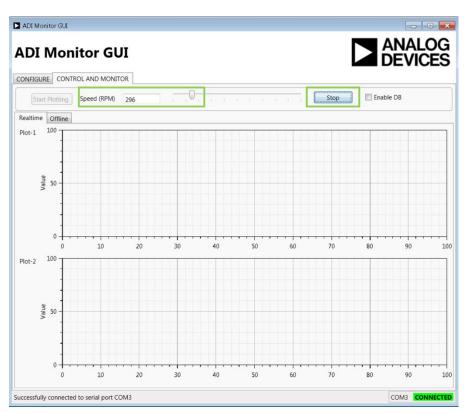


Figure 6: Motor Run Screen

8 Data Visualization

In order to import visualization data into the GUI, the linker map file (**MC_demo.map**) provided with the demo C code or executable must be loaded. This is selected from the *Configure* tab as illustrated in Figure 7 with the resultant variable list shown in Figure 8. The variables provided by default have a UID_ prefix in the map file and are listed in the Table below, along with their data types. It should be noted that the GUI cannot determine from the map file if a variable has signed or unsigned type, so if a variable is of signed type (e.g. duty_a) it is necessary to manually change the Type from UINT16 to SINT16 in the relevant column of Figure 8.

| Variable Name | Variable | Data Type |
|---------------|--|-----------|
| Idc_LV | DC bus current in Amps | SINGLE |
| Idc_LV_adc | DC bus current – ADC value | UINT16 |
| Vd | D-axis motor voltage per unit | SINGLE |
| Vdc_LV | DC bus voltage in Volts | SINGLE |
| Vdc_LV_adc | DC bus voltage – ADC value | UINT16 |
| Vq | Q-axis motor voltage per unit | SINGLE |
| Vu_LV | U-phase motor voltage (filtered) in Volts | SINGLE |
| Vu_LV_adc | U-phase motor voltage (filtered) – ADC value | UINT16 |

| | 1 | |
|--------------------|--|--------|
| Vv_LV | V-phase motor voltage (filtered) in Volts | SINGLE |
| Vv_LV_adc | V-phase motor voltage (filtered) – ADC value | UINT16 |
| Vw_LV | W-phase motor voltage (filtered) in Volts | SINGLE |
| Vw_LV_adc | W-phase motor voltage (filtered) – ADC value | UINT16 |
| duty_a | A phase duty cycle counter | SINT16 |
| duty_b | B phase duty cycle counter | SINT16 |
| duty_c | C phase duty cycle counter | SINT16 |
| hall_state_monitor | Hall signals code | UINT16 |
| ia_adc | U phase current – SAR ADC value | UINT16 |
| ia_sar | U phase current – SAR Amps value | SINGLE |
| ia_sinc | U phase current – SINC Amps value | SINGLE |
| ib_adc | V phase current – SAR ADC value | UINT16 |
| ib_sar | V phase current – SAR Amps value | SINGLE |
| ib_sinc | V phase current – SINC Amps value | SINGLE |
| ib_sinc_raw | V phase current – SINC ADC value | SINT16 |
| ic_adc | W phase current – SAR ADC value | UINT16 |
| ic_sar | W phase current – SAR Amps value | SINGLE |
| ic_sinc | W phase current – SINC Amps value | SINGLE |
| ic_sinc_raw | W phase current – SINC ADC value | SINT16 |
| qep_cnt_monitor | Quadrature encoder count | UINT16 |
| speed_fil | Speed in rpm (filtered) | SINGLE |
| speed_raw | Speed in rpm (unfiltered) | SINGLE |
| theta_enc | Electrical angle from encoder | SINGLE |
| theta_vf | Open loop electrical angle | SINGLE |

| .DI Mc | 😋 🔵 🗢 👢 « v020 | · · · · · · · · · · · · · · · · · · · | cts ▶ PMSM_FOC ▶ i | iai i Debug i E | xe 👻 🍾 | Search Exe | ۶ |
|-------------|------------------------|---------------------------------------|--------------------|-----------------|------------------|--------------------------|----------------------------|
| п | Organize 🔻 New | folder | | | | | 0 |
| DI | 涬 Favorites 📃 Desktop | Subversi Exe | ion library | | | Arrange by: | Folder 🔻 |
| NFIGU | 鷆 Downloads | = Name | * | | Date modified | Size | File count |
| Notor | laces 💱 Recent Places | MC_dem | no.map | | 1/5/2015 5:36 PM | 89 KI | В |
| VF G | 🧱 Libraries | | | | | | |
| VF B | la Cadence | | | | | | |
| VF N | 🧟 Documents 🌛 Music | | | | | | |
| IMA | Se Pictures | | | | | | |
| 1 IVIA | | | | | | | |
| | a Subversion | | | | | | |
| мах | ideos 💐 Subversion | ▼ | 111 | 1 | | | |
| MAX VF C | JUDE Videos | e name: MC_demo.m | | 1 | | p documents (' Open 🔶 | *.xls, *.xlsx, ▼ Cancel |
| | JUDE Videos | | nap | | | | |
| | JUDE Videos | e name: MC_demo.m | nap | | | | |
| | JUDE Videos | e name: MC_demo.m | nap | | | | |
| | JUDE Videos | e name: MC_demo.m | nap | | | | |
| | JUDE Videos | e name: MC_demo.m | nap | | | | |
| | JUDE Videos | e name: MC_demo.m | nap | | | | |
| | JUDE Videos | e name: MC_demo.m | nap | | | | |

Figure 7: Load Map file for Data Visualization

| Plot # | Name | Address | Туре | Min | Max |
|--------|------------|------------|--------|-----|------------|
| 1 | Idc_LV | 0x20032be8 | -71 | | 42949672 |
| 1 | Idc_LV_adc | 0x20032d7a | UINT16 | 0 | 65536 |
| 1 | Vd | 0x20032ce0 | SINGLE | 0 | 42949672 |
| 1 | Vdc_LV | 0x20032be4 | SINGLE | 0 | 42949672 |
| 1 | Vdc_LV_adc | 0x20032d78 | UINT16 | 0 | 65536 |
| 1 | Vq | 0x20032ce4 | SINGLE | 0 | 42949672 - |
| ٠ | 1 | | | | • |

Figure 8: Visualization Variables

In order to select a variable for plotting, selection is by means of the check-box in the Plot column. The

variable can be plotted in Plot 1 or Plot 2 (see Figure 6). There are certain constraints on the number of bytes and buffers that can be streamed using the GUI. The program will limit the total buffer size to 1.7k bytes. The number of variables plotted can be increased by reducing the buffer size, which is maximum 200, although a total maximum selected variable size of 8 bytes is allowed (e.g. 2xSINGLE or 4xINT16, or 1xSINGLE+2xINT16 etc).

The steps to prepare for plotting of variables are (refer to Figure 9):

- 1. Select all of the variables to be plotted and whether each one is to be on Plot 1 or Plot 2, bearing in mind the maximum buffer memory size (1.7k) and the maximum byte count per buffer (8).
- Down Sampling factor the buffer time slice length equals the PWM switching period x Down Sampling ratio, so for a down sampling ratio of 1, the sampling period will be 100µs, and with a buffer length of 200, the plot time slices will be of 20ms length. To look at longer time slices, increase the down sampling factor.
- 3. Press Update Configuration
- 4. On the Control and Monitor tab, press Start Plotting.
- 5. If the variables are to be changed, press *Stop Plotting* and return to the *Configure* tab to change the configured variables.

| Single Delay 100 Address Prefix UIDFilter # Name Address Type ic_sinc_hdr 0x20032494 SINGLE ic_sinc_raw_HDR 0x200325a0 UINT16 qep_cnt_monitor 0x200325ac UINT16 speed_fil 0x200324a4 SINGLE speed_ref 0x200324a8 SINGLE theta_enc 0x20032498 SINGLE | BOOST 0 MAX RATE 10 MAX RATE 10 MAX RATE 10 MAX RATE 10 MAX RATE 10 MAX RATE 10 MAX 2 Address Prefix UID_ Plot # Name Address Type Plot 1 ic_sinc_hdr 0x20032494 SINGLE 1 ic_sinc_raw_HDR 0x200325a0 UINT16 1 gep_cnt_monitor 0x200325aC UINT16 1 gep_cnt_monitor 0x200325aC UINT16 1 speed_fil 1 speed_raw 0x20032444 SINGLE |
|--|--|
| Address Prefix UID_ Filter Name Address Type Plot ic_sinc_hdr 0x20032494 SINGLE ic_sinc_raw_HDR 0x200325a0 UINT16 gep_cnt_monitor 0x200325ac UINT16 speed_fil 0x200324a0 SINGLE speed_raw 0x200324a4 SINGLE speed_ref 0x200324a8 SINGLE | MAX RATE 10 Address Filter Buffer MAX RATE 1 ic_sinc_hdr 0x20032494 SINGLE Count 8 MAX RPM 2000 1 ic_sinc_hdr 0x200325a0 UINT16 1 Size 150 CTRL 1 speed_fil 0x20032448 SINGLE Image: Count in the speed_file Image: Count i |
| Address Prefix UID_ Filter Name Address Type Plot ic_sinc_hdr 0x20032494 SINGLE ic_sinc_raw_HDR 0x200325a0 UINT16 gep_cnt_monitor 0x200325ac UINT16 speed_fil 0x200324a0 SINGLE speed_raw 0x200324a4 SINGLE speed_ref 0x200324a8 SINGLE | MAX RATE 10 Address Prefix UID_ Filter IAX 2 Plot # Name Address Type Plot 8 I ic_sinc_hdr 0x20032494 SINGLE 5 5 5 I ic_sinc_naw_HDR 0x200325a0 UINT16 5 |
| Name Address Type Piot ic_sinc_hdr 0x20032494 SINGLE | AX 2 Plot in Name Address Type Plot I ic_sinc_hdr 0x20032494 SINGLE Image: Size Size I ic_sinc_raw_HDR 0x200325a0 UINT16 Image: Size Size 150 CTRL I speed_fil 0x200324a0 SINGLE Image: Size Image: Size 150 CCW Rotation I speed_raw 0x200324a8 SINGLE Image: Size |
| ic_sinc_raw_HDR 0x200325a0 UINT16 Size 150 qep_cnt_monitor 0x200325ac UINT16 5000000000000000000000000000000000000 | AX RPM 2000 1 ic_sinc_raw_HDR 0x200325a0 UINT16 1 150 CTRL 1 qep_cnt_monitor 0x200325ac UINT16 1 150 CCW Rotation 1 speed_raw 0x200324a0 SINGLE 1 150 Configure 1 speed_ref 0x200324a8 SINGLE 1 1 |
| IC_SINC_TAW_HDR 0x200325a0 OINT16 Image: Constraint of the second | 1 IC_SINE_TAW_HDR 0x200325a0 OIN116 130 CTRL 1 qep_cnt_monitor 0x200325ac UIN116 130 CTRL 1 speed_fil 0x200324a0 SINGLE Image: Comparison of the speed_raw 0x200324a4 SINGLE Image: Comparison of the speed_raw 0x200324a8 SINGLE Image: Comparison of the speed_raw Image: Comparison of |
| speed_fil 0x200324a0 SINGLE Image: Comparison of the system speed_raw 0x200324a4 SINGLE Image: Comparison of the system Image: Comparison of the system speed_ref 0x200324a8 SINGLE Image: Comparison of the system Image: Comparison of the system | CTRL 1 speed_fil 0x200324a0 SINGLE Image: Configure CCW Rotation 2 speed_raw 0x200324a4 SINGLE Image: Configure Configure 1 theta_enc 0x200324a8 SINGLE Image: Configure |
| speed_raw 0x200324a4 SINGLE speed_ref 0x200324a8 SINGLE | CCW Rotation 1 speed_raw 0x200324a4 SINGLE |
| | 2 speed_ref 0x200324a8 SINGLE 1 theta_enc 0x20032498 SINGLE |
| theta_enc 0x20032498 SINGLE | |
| | Load Map Clear Update Configuration |
| Map Clear Update Configuration | |
| | |
| | |
| | |
| | |
| | |
| | |
| Map Clear Update Configuration | |

Figure 9: Setup of Variables for Plotting

The variables will start to plot in the Control and Monitor tab as illustrated in Figure 10.