AN1078 Demonstration ReadMe for MCLV-48V-300W Inverter Board with dsPIC33CK256MP508 Motor Control DIM (MPLAB® X IDE)

1. INTRODUCTION

This document describes the setup requirements for running the Sensor less FOC algorithm with a Sliding Mode Observer, which is referenced in AN1078 “Sensorless Field Oriented Control of a PMSM” and to implement three phase currents reconstruction algorithm using DC bus Current, which is referenced in AN1299“Single-Shunt Three-Phase Current Reconstruction Algorithm for Sensor less FOC of a PMSM” using a MCLV-48V-300W Inverter Board and dsPIC33CK256MP508 Motor Control Dual In-line Module (DIM).

The demonstration is configured to run on the MCLV-48V-300W Inverter Board and dsPIC33CK256MP508 Motor Control Dual In-line Module (DIM) in both Internal and External Op Amp configuration with the dsPIC33CK256MP508.

2. SUGGESTED DEMONSTRATION REQUIREMENTS

2.1. Motor Control Application Firmware Required for the Demonstration

To clone or download this application firmware from Github, go to the main page of this repository and then click Clone button to clone this repository or download as zip file.

Commented [JP1]: Maybe remove the bullet point.

2.2. Software Tools Used for Testing the firmware

> **Note:**
>
> In this document, hereinafter this firmware package is referred as firmware.

- MPLAB® X IDE v5.50
- DFP: dsPIC33CK-MP_DFP v1.6.176
- MPLAB® XC16 Compiler v1.70
- MPLAB® X IDE Plugin: X2C-Scope v1.3.0

> **Note:**
>
> The software used for testing the firmware prior to release is listed above. It is recommended to use the version listed above or later versions for building the firmware.

2.3. Hardware Tools Required for the Demonstration

- MCLV-48V-300W Inverter Board (EV18H47A)
- dsPIC33CK256MP508 Motor Control DIM (EV62P66A)
- 24V Power Supply (AC002013)
- 24V 3-Phase Brushless DC Motor (AC300020)

> **Note:**
>
> All items listed under this section Hardware Tools Required for the Demonstration are available at microchip DIRECT.

3. HARDWARE SETUP

This section describes the hardware setup needed for the demonstration.

1. Motor currents are amplified on the MCLV-48V-300W Inverter Board; it can also be amplified by the amplifiers internal to the dsPIC33CK256MP508 on the DIM. By default, the firmware and DIM are configured to sample and convert internal amplifier outputs ('internal op-amp configuration'), measuring the motor currents needed for implementing FOC.

   The Table-1 summarizes the resistors to be populated and removed to convert the DIM from internal op-amp configuration to external op-amp configuration or vice versa.

Table-1: SELECTION BETWEEN EXTERNAL AND INTERNAL AMPLIFIER OUTPUTS

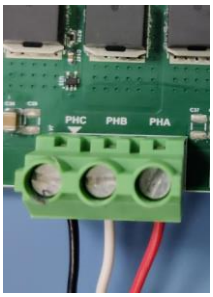
Current Signal	Internal Amplifier Output		External Amplifier Output		Firmware setting
	Jumper Resistor (0R) Settings on the DIM				
	Populate	Remove	Populate	Remove	
Phase Current **IA or IA_EXT**	R9	R6	R6	R9	• Configure and enable amplifiers in internal op-amp configuration. • Ensure to disable the internal amplifiers in external op-amp configuration.
Phase Current **IB or IB_EXT**	R29	R25	R25	R29	
Bus Current **IBUS or IBUS_EXT**	R14	R10, R7 and R5	R10	R14, R7 and R5	

Commented [JP2]: center align the text

Commented [JP3]: The Center aligns the text.

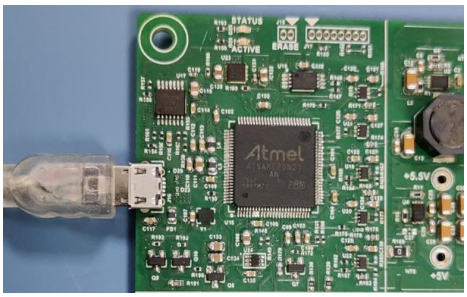
2. Insert the dsPIC33CK256MP508 Motor Control DIM into the DIM Interface Connector J8 provided on the MCLV-48V-300W Inverter Board. Make sure the DIM is placed correctly and oriented before going ahead.

3. Connect the 3-phase wires from the motor to PHC, PHB, and PHA of the J4 connector (no specific order), provided on the MCLV-48V-300W Inverter Board.



4. Plug in the 24V power supply to connector J1 provided on the MCLV-48V-300W Inverter Board. Alternatively, the Inverter Board can also be powered through Connector J3.

5. The board has an onboard programmer 'PICKIT™ On Board (PKOBv4)' which can be used for programming or debugging the dsPIC33CK256MP508. To use an on-board programmer, connect a micro-USB cable between Host PC and Connector J16 provided on the MCLV-48V-300W Inverter Board.



6. Alternatively, connect the Microchip programmer/debugger MPLAB PICkit 4 In-Circuit Debugger to the ICSP header J9 of the MCLV-48V-300W Inverter Board as shown below and to the Host PC used for programming the device. Ensure that PICkit 4 is oriented correctly before proceeding.

4. SOFTWARE SETUP AND RUN

4.1. Setup: MPLAB X IDE and MPLAB XC16 Compiler

Install MPLAB X IDE and MPLAB XC16 Compiler versions that support the device dsPIC33CK256MP508 and PKOBv4. The MPLAB X IDE, MPLAB XC16 Compiler, and X2C-Scope plug-in used for testing the firmware are mentioned in the Motor Control Application Firmware Required for the Demonstration section. To get help on

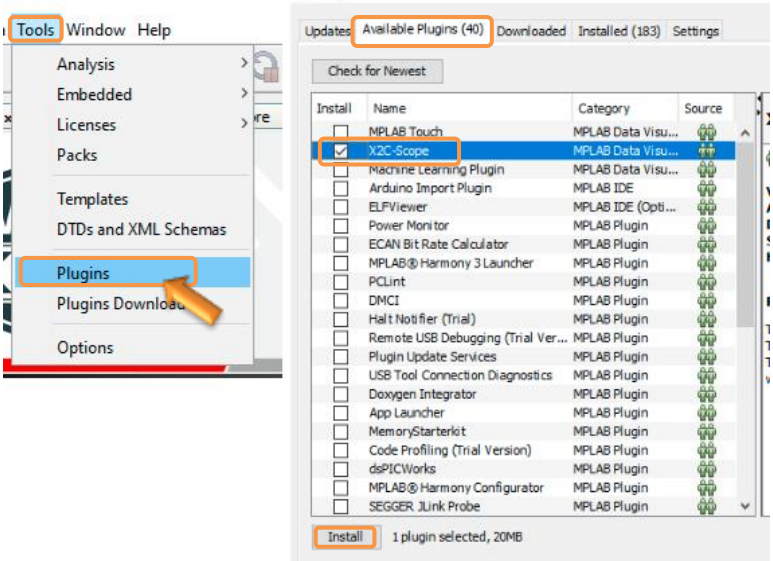
- MPLAB X IDE installation, refer to link
- MPLAB XC16 Compiler installation steps, refer to link

If MPLAB IDE v8 or earlier is already installed on your computer, then run the MPLAB driver switcher (It is installed when MPLAB®X IDE is installed) to switch from MPLAB IDE v8 drivers to MPLAB X IDE drivers. If you have Windows 8 or 10, you must run MPLAB driver switcher in 'Administrator Mode.' To run the Device Driver Switcher GUI application as administrator, right-click on the executable (or desktop icon) and select 'Run as Administrator. For more details, refer to MPLAB X IDE help topic *"Before You Begin: Install the USB Device Drivers (For Hardware Tools): USB Driver Installation for Windows Operating Systems."*

4.2. Setup: X2C-SCOPE

X2C-Scope is an MPLAB X IDE plugin that allows a developer to interact with an application while the application program is running. X2C-Scope enables you to read, write, and plot global variables in real-time. It communicates with the target using the UART. To use X2C-Scope, the plugin must be installed:

- In MPLAB X IDE, select *Tools->Plugins* and click on the Available Plugins tab.
- Select X2C-Scope plug-in by checking its check box and clicking Install.
- Look for tool X2C-Scope under *Tools->Embedded*.



5. BASIC DEMONSTRATION

5.1. Firmware Description

The firmware version needed for the demonstration is mentioned under the Motor Control Application Firmware Required for the Demonstration section.

This firmware is implemented to work on Microchip's 16-bit Digital signal controller (dsPIC® DSC) dsPIC33CK256MP508**.** For more information, see the *dsPIC33CK256MP508 Family datasheet (DS70005349)*.

The Motor Control Demo application uses a push button to start or stop the motor and a potentiometer to vary the speed of the motor. This Motor Control Demo Application configures and uses peripherals like PWM, ADC, UART, etc.

For more details refer Microchip Application note AN1078 "Sensorless Field Oriented Control of a PMSM" and AN1299 "Single-Shunt Three-Phase Current Reconstruction Algorithm for Sensorless FOC of a PMSM" available at Microchip website.

Note:

The project may not build correctly in Windows OS if the Maximum path length of any source file in the project is more than 260 characters. In case the absolute path is exceeding or nearing maximum length, do any (or both) of the following:

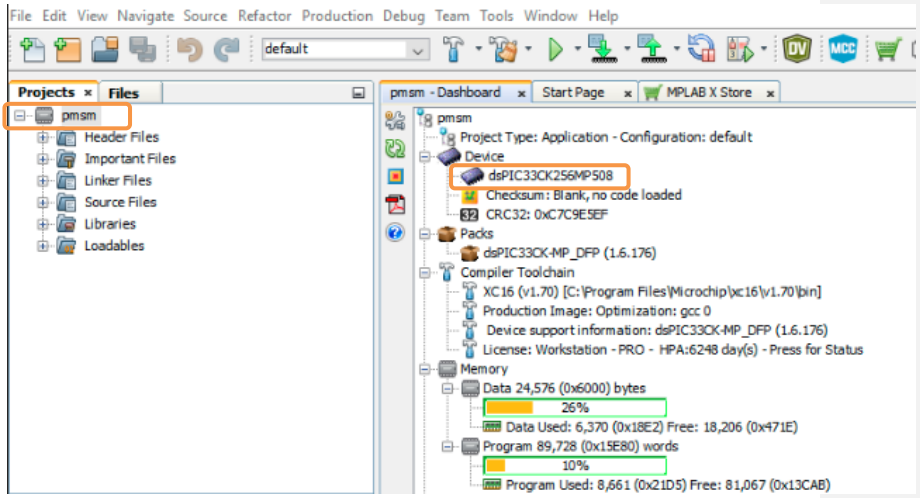
- Shorten the directory name containing the firmware used in this demonstration. If you renamed the directory, consider the new name while reading the instructions provided in the upcoming sections of the document.
- Place firmware in a location such that the total path length of each file included in the projects does not exceed the Maximum Path length specified.

Refer to MPLAB X IDE help topic *"Path, File, and Folder Name Restrictions"* for details.

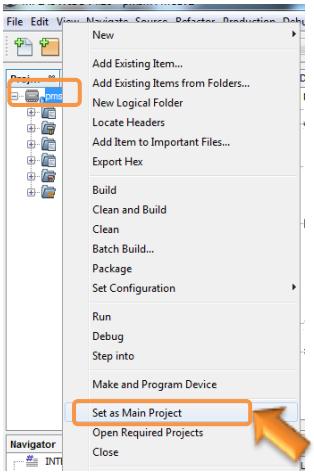
5.2. Basic Demonstration

Follow the below instructions step by step, to set up and run the motor control demo application:

1. Start MPLAB X IDE and open (`File>Open Project`) the project `pmsm.X` with device selection dsPIC33CK256MP508.



2. Set the project `pmsm.X` as the main project by right-clicking on the project name and selecting "Set as Main Project" as shown. The project **"pmsm"** will then appear in bold.



3. Open `userparams.h` (under `pmsm.X -> headerfiles`) in the project `pmsm.X`
   - Ensure that the macros `TUNING`, `OPEN_LOOP_FUNCTIONING`, `SINGLE_SHUNT`, and `TORQUE_MODE` are not defined.

     ```
     #undef OPEN_LOOP_FUNCTIONING

     #undef TORQUE_MODE

     #undef SINGLE_SHUNT
     ```

   - When internal amplifiers are used for current amplification (referred to as **'internal op-amp configuration'**), then **define** the macro **`'INTERNAL_OPAMP_CONFIG'`**.

     ```
     #define INTERNAL_OPAMP_CONFIG
     ```

     Otherwise, if external amplifiers are used for current amplification (referred to as **'external op-amp configuration'**), then **undefine** the macro **`'INTERNAL_OPAMP_CONFIG'`** in `userparams.h`

     ```
     #undef INTERNAL_OPAMP_CONFIG
     ```

   Note:

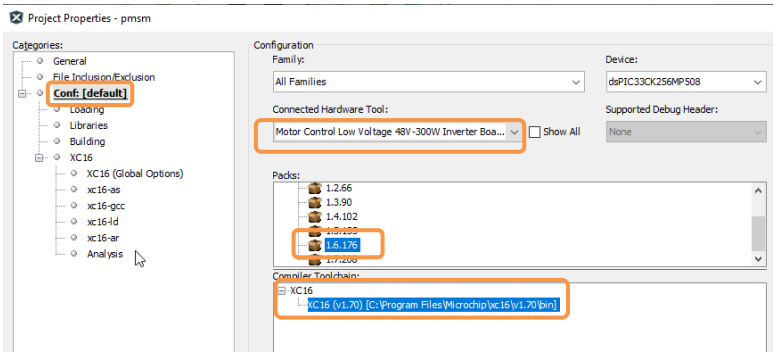
   The motor phase currents can be reconstructed from the DC Bus current by appropriately sampling it during the PWM switching period, called a single-shunt reconstruction algorithm. The firmware can be configured to demonstrate the single shunt reconstruction algorithm by defining the macro **`'SINGLE_SHUNT'`** in the header file `userparams.h`

   For additional information, refer to Microchip application note **AN1299, "Single-Shunt Three-Phase Current Reconstruction Algorithm for Sensorless FOC of a PMSM."**

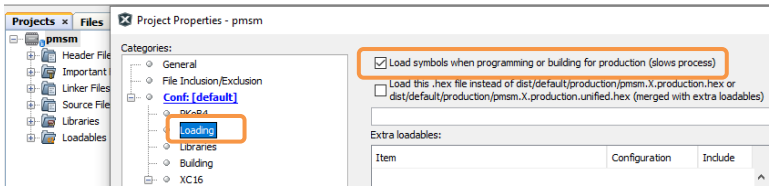
   By default, the firmware uses phase currents measured across the phase shunt resistors on two of the half-bridges of the three-phase inverter ('dual shunt configuration') to implement FOC.

4. Right-click on the project *pmsm.X* and select "Properties" to open its Project Properties Dialog. Click the "Conf: [default]" category to reveal the general project configuration information. The development tools used for testing the firmware are listed in the section 2.2 Software Tools Used for Testing the firmware.

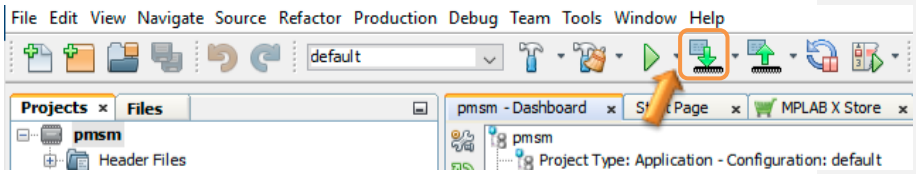
   In the *'**Conf-default**'* category window:
   - Select the specific Compiler Toolchain from the available list of compilers. Please ensure MPLAB® XC16 Compiler supports the device dsPIC33CK256MP508.In this case, "XC16(v1.70)" is selected.
   - Select the Hardware Tool to be used for programming and debugging.
   - Select the specific Device Family Pack (DFP) from the available list of Packs. In this case, "dsPIC33CK-MP_DFP 1.6.176" is selected.
   - After selecting Hardware Tool and Compiler Toolchain, click the button **Apply**



5. Ensure that the checkbox "**Load symbols when programming or building for production (slows process)**" is checked, which is under the "Loading" category of the Project Properties window.

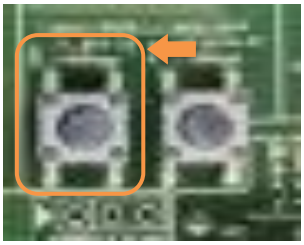


6. To build the project (in this case, *pmsm.X*) and program the device dsPIC33CK256MP508, click "**Make and Program Device Main project**" on the toolbar



7. If the device is successfully programmed, **LD2 ('LED1')** will be turned ON, indicating that the dsPIC® DSC is enabled.

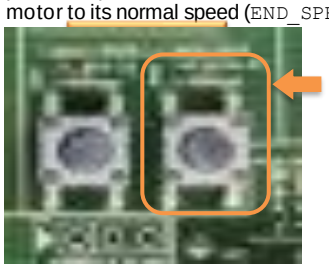
8. Run or stop the motor by pressing the push button **SW1**. The motor should start spinning smoothly in one direction in the 'Normal Speed Range.' Ensure that the motor is spinning smoothly without any vibration. The LED LD2 ('LED1') is turned ON to show the button is pressed to start the motor.



Commented [JP4]: The below figure looks skewed; kindly verify.

9. If desired, the motor speed can be varied using the potentiometer ('POT1').

10. To enter the extended speed range (`NOMINAL_SPEED_RPM` to `MAXIMUM_SPEED_RPM`), press the push button **SW2**. Press the push button **SW2** again to revert the speed of the motor to its normal speed (`END_SPEED_RPM` to `NOMINAL_SPEED_RPM`) range.



11. Press the push button **SW1** to stop the motor.

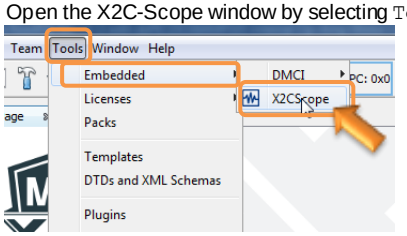
Note:

The macro definitions `END_SPEED_RPM`, `NOMINAL_SPEED_RPM`, and `MAXIMUM_SPEED_RPM` are specified in `userparms.h` file included in the project *pmsm.X*. The definitions `NOMINAL_SPEED_RPM`, and `MAXIMUM_SPEED_RPM` are defined as per the specification provided by the Motor manufacturer. *Exceeding manufacture specification may lead to damage of the motor or(and) the board.*

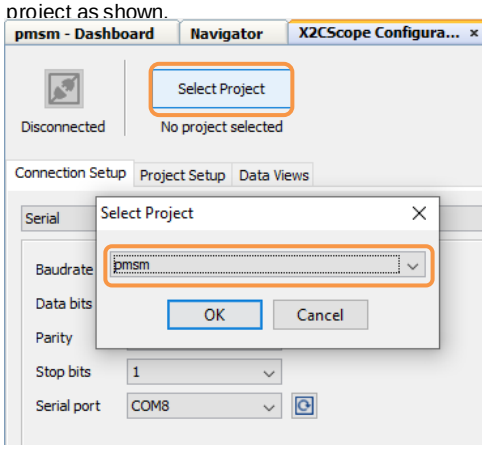
5.3. Data visualization through X2C-Scope plug-in of MPLABX

X2C-Scope is a third-party plug-in for MPLAB X, which helps in real-time diagnostics. The application firmware comes with the initialization needed to interface controller with the host PC to enable data visualization through X2C-Scope plug-in.

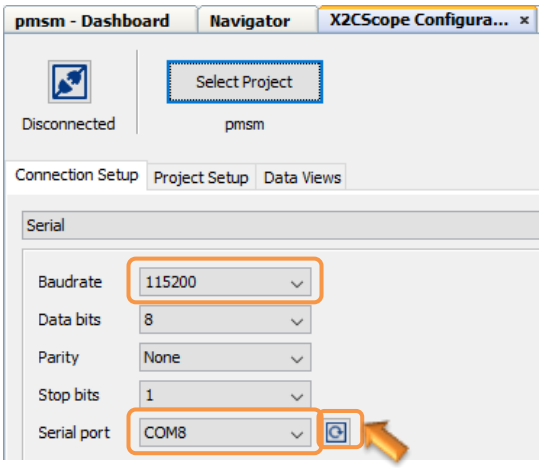
1. Ensure X2C-Scope plug-in is installed. For more information on how to set up a plug-in, refer to https://microchipdeveloper.com/mplabx:tools-plugins-available
2. To establish serial communication with the host PC, connect a micro-USB cable between the host PC and the MCLV-48V-300W Inverter Board (connector J16). This interface is also used for programming.
3. Ensure the application is configured and running as described under section 5.2 Basic Demonstration by following steps 1 through 11.
4. Open the X2C-Scope window by selecting `Tools>Embedded>X2CScope`.



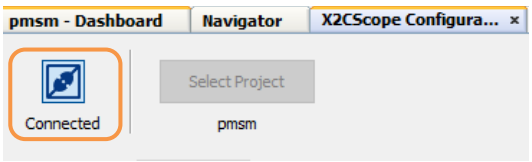
5. In the X2C-Scope Configuration window, using the "**Select Project**" menu, select **'pmsm'** project as shown.



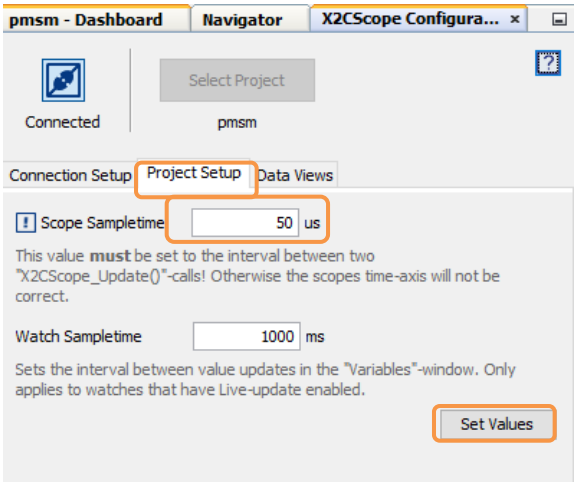
6. Serial communication needs to be set up, as shown in the following figure. Ensure the communication baud rate is set to 115200 as configured in the application firmware. The COM port used depends on the system settings. `The refresh button` lists the available COM Ports. Select the COM Port as per the connection.



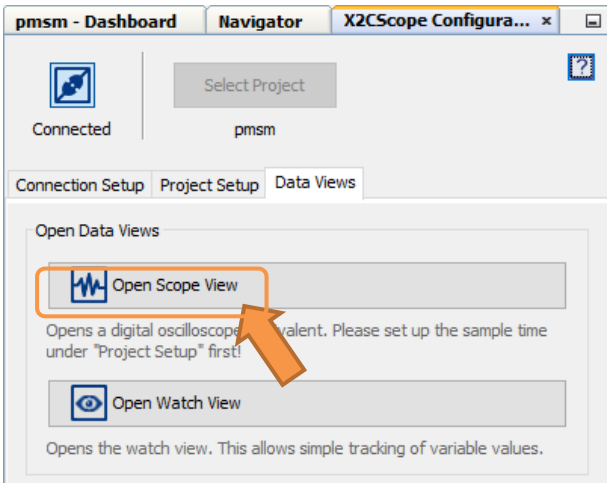
7. Once the COM port is detected, click on “Disconnected” and turn to “Connected”, to establish a serial communication between Host PC and the board.



8. Set the “Project Setup” as shown below and click “Set Values.” Set Scope sample time as the interval at which `X2CScopeUpdate()` is called. In this application, it is every 50µs.

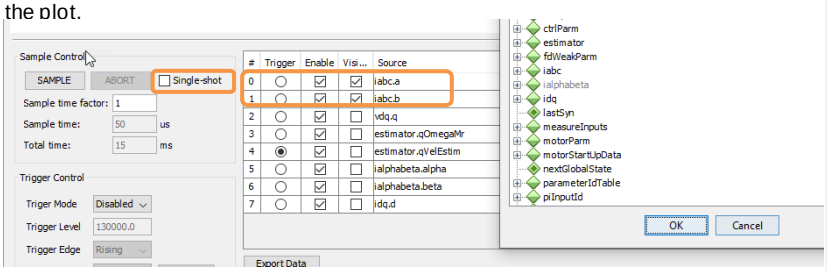


9. Click on 'Open Scope View' (under sub-window "Data Views"); this opens 'Scope Window'.

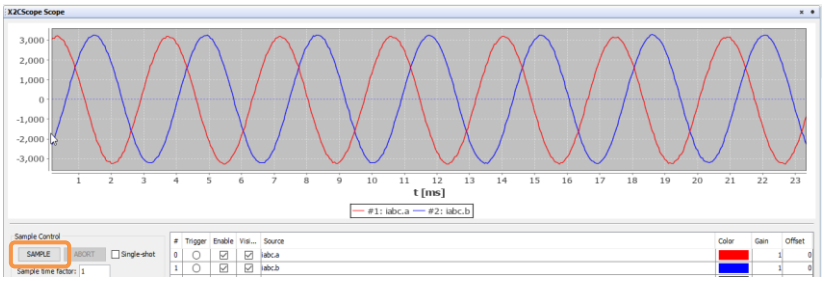


10. In this window, select the variables that need to be watched. To do this, click on the source against each channel, a window "Select Variables" opens on the screen. From the available list, the required variable can be chosen. Ensure checkboxes "Enable" & "Visible" are checked for the variables to be plotted.

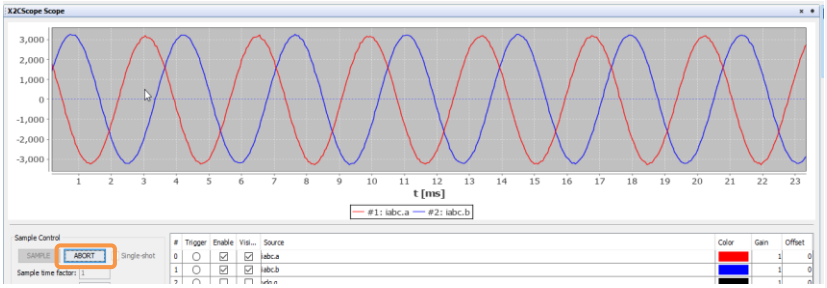
    To view data plots continuously, uncheck `Single-shot`. When `Single-shot` is checked, it captures the data once and stops. The `Sample time factor` value multiplied with Sample time decides the time difference between any two consecutive data points on the plot.



11. Click on SAMPLE, then the X2C-Scope window plots variables in real-time, which updates automatically.



12. Click on ABORT to stop.



6. REFERENCES:

For more information, refer to the following documents or links.

1. AN1292 Application Note “Sensorless Field Oriented Control (FOC) for a Permanent Magnet Synchronous Motor (PMSM) Using a PLL Estimator and Field Weakening (FW).”
2. AN1299 Application Note “Single-Shunt Three-Phase Current Reconstruction Algorithm for Sensorless FOC of a PMSM.”
3. MCLV-48V-300W Inverter Board User’s Guide
4. dsPIC33CK256MP508 Family datasheet (DS70005349).
5. Family Reference manuals (FRM) of dsPIC33CK256MP508 family
6. MPLAB® X IDE User’s Guide (DS50002027) or MPLAB® X IDE help
7. MPLAB® X IDE installation
8. MPLAB® XC16 Compiler installation

Commented [JP5]: Shall we add a reference to AN1299 as well?