AN1299 Tuning Guide

This document describes the procedure and setup necessary for tuning the algorithm described in AN1299 *"Single-Shunt Three-Phase Current Reconstruction Algorithm for Sensorless FOC of a PMSM"* (DS01299).

1.1 CONFIGURING DUAL-SHUNT MODE

1. As the first step, the user needs to follow the tuning procedure for AN1078, which explains all of the steps required to get the Slide Mode Controller (SMC) running on a specific motor and hardware.
2. Once the application is running, import the following parameters from the AN1078 `UserParms.h` file, to the AN1299 `UserParms.h`. file.

TABLE 1-1: PARAMETERS TO BE IMPORTED FROM AN1078 TO THE AN1299 `UserParms.h` FILE

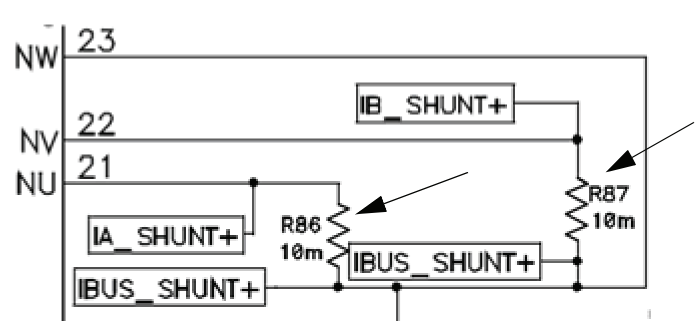
Start-up Parameters	`#define LOCKTIMEINSEC`
	`#define OPENLOOPTIMEINSEC`
	`#define INITIALTORQUE`
	`#define ENDSPEEDOPENLOOP`
Motor Parameters	`#define POLEPAIRS`
	`#define PHASERES`
	`#define PHASEIND`
	`#define NOMINALSPEEDINRPM`
	`#define MINSPEEDINRPM`
	`#define FIELDWEAKSPEEDRPM`
Oscillator Parameters	`#define PLLIN`
	`#define DESIREDMIPS`
PWM and Control Timing Parameters	`#define PWMFREQUENCY`
	`#define DEADTIMESEC`
	`#define BUTPOLLOOPTIME`
	`#define SPEEDLOOPFREQ`
Slide Mode Controller Parameters	`#define SMCGAIN`
	`#define MAXLINEARSMC`
	`#define FILTERDELAY`
Hardware Parameters	`#define VDD`

TABLE 1-1: PARAMETERS TO BE IMPORTED FROM AN1078 TO THE AN1299 `UserParms.h` FILE (CONTINUED)

PI Coefficients	#define DKP
	#define DKI
	#define DKC
	#define DOUTMAX
	#define QKP
	#define QKI
	#define QKC
	#define QOUTMAX
	#define WKP
	#define WKI
	#define WKC
	#define WOUTMAX
ADC Scaling	#define DQK
	#define DQKA
	#define DQKB
Field Weakening	#define dqKFw0
	#define dqKFw1
	#define dqKFw2
	#define dqKFw3
	#define dqKFw4
	#define dqKFw5
	#define dqKFw6
	#define dqKFw7
	#define dqKFw8
	#define dqKFw9
	#define dqKFw10
	#define dqKFw11
	#define dqKFw12
	#define dqKFw13
	#define dqKFw14
	#define dqKFw15

3. First, we will get AN1299 running in dual-shunt mode. Add the value of the shunt resistors into the `UserParms.h` file. This resistor value is connected from the three-phase inverter to Power Ground. As an example, consider using the dsPICDEM™ MCHV Development Board. These resistors are as shown in Figure 1-1.

FIGURE 1-1:



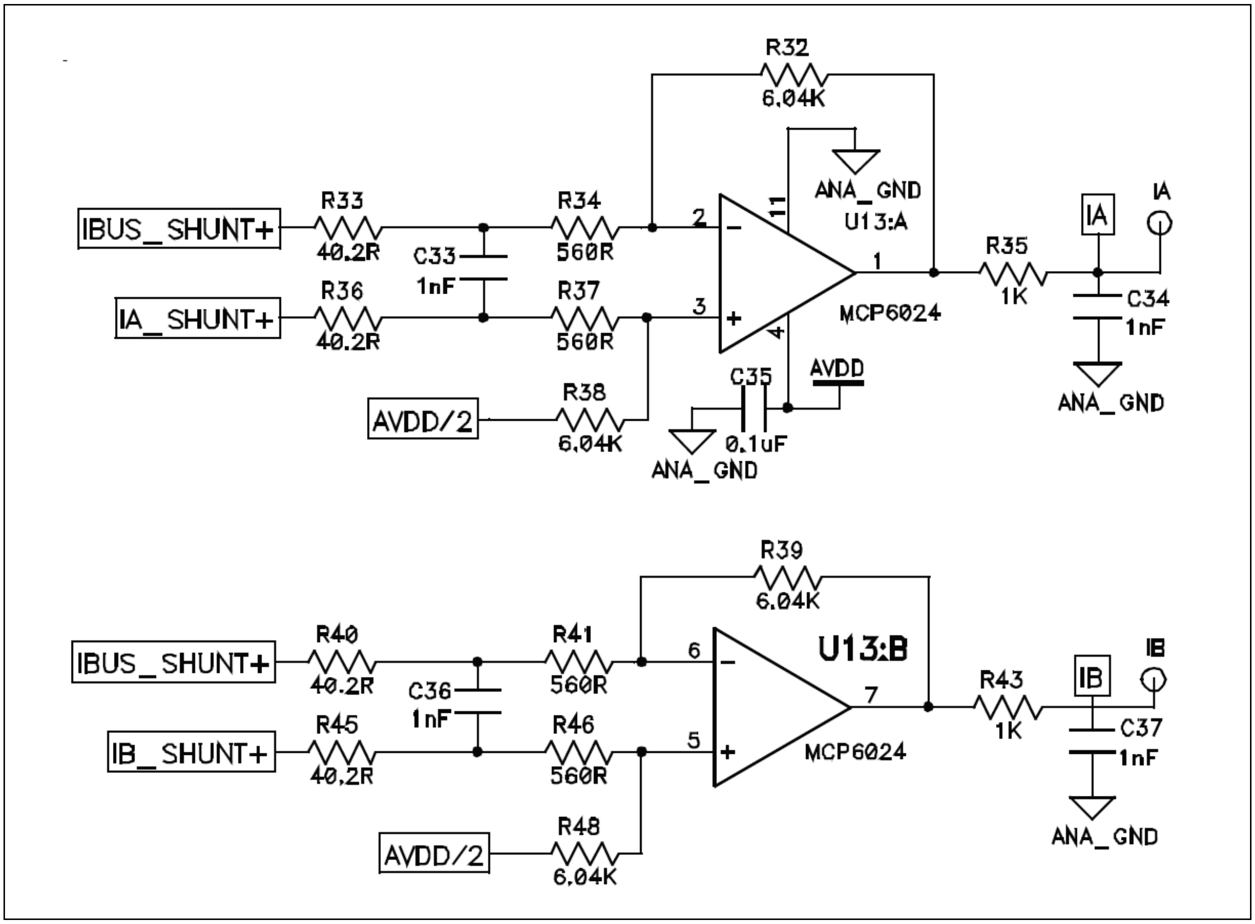
These are the shunt resistor values for dual shunt, and assuming they are both of the same value, this value of 10 mΩ is entered in the `UserParms.h` file as follows:

```
//*************** Hardware Parameters ****************

#define RSHUNT          0.010    // Value in Ohms of shunt resistors used.
```

4. After setting the values of the shunt resistors, enter the differential amplifier gain used to amplify the two shunt resistors. It is assumed that both amplifiers have the same gain, as shown in Figure 1-2.

FIGURE 1-2:



The value for `UserParms.h` is: 6.04K / (560 + 40.2) = 10.0633:

```
#define DIFFAMPGAIN     10.0633 // Gain of differential amplifier.
```

5. Once all parameters have been entered in the AN1299 source code, run the application code in dual-shunt mode by commenting out the following line:

```
//#define SINGLESHUNT    // define that enables single shunt three phase
                         // reconstruction. If this is not defined, control will run
                         // in dual shunt resistor mode.
```

Make sure that the code runs as indicated in the AN1078 Tuning Procedure.

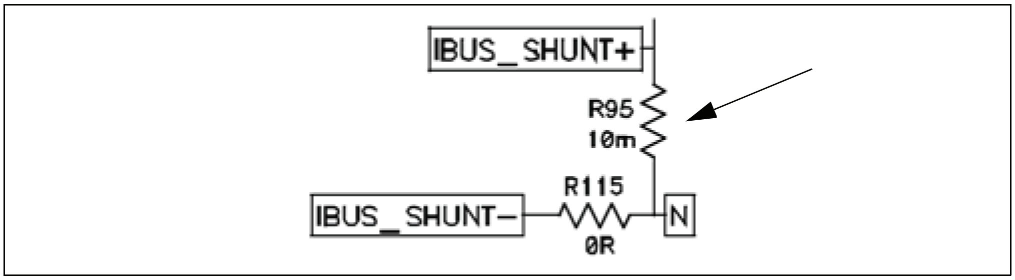
1.2 CONFIGURING SINGLE-SHUNT MODE

1. Once the application code is running in dual-shunt mode, enable the single-shunt algorithm by defining the following in the `UserParms.h` file:

```
#define SINGLESHUNT // define that enables single shunt three phase
                    // reconstruction. If this is not defined, control will run
                    // in dual shunt resistor mode.
```

2. Enter the value of the IBUS shunt resistor in the `UserParms.h` file. This resistor value is connected from the three-phase inverter to Power Ground. As the example of using the dsPICDEM™ MCHV Development Board. This resistor is found as shown in Figure 1-3.

FIGURE 1-3:



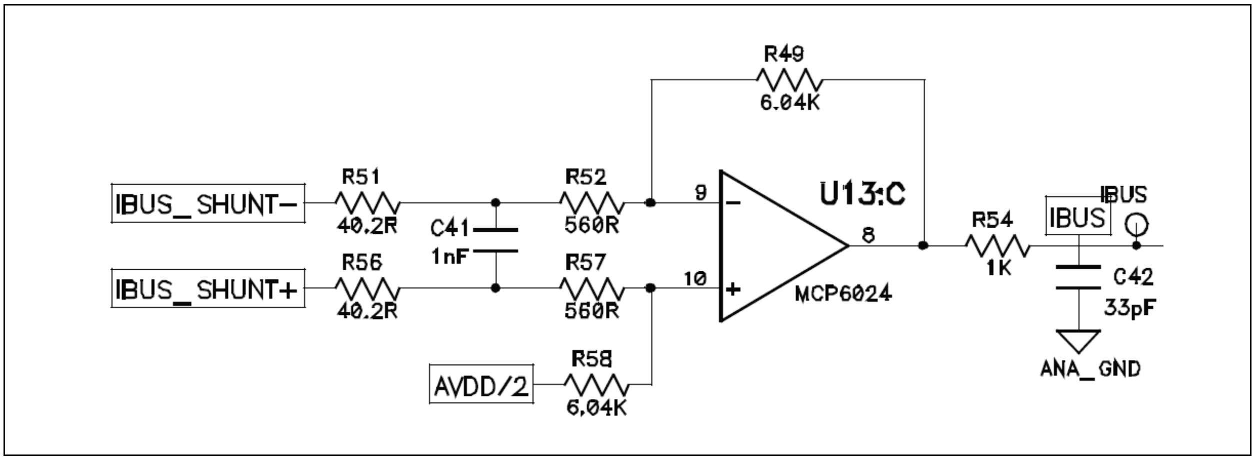
This is the shunt resistor value for single-shunt. Enter the value of 10 mΩ to the `UserParms.h` file as follows:

```
//*************** Hardware Parameters ****************

#define RSHUNT          0.010    // Value in Ohms of shunt resistors used.
```

3. After setting the value of the single-shunt resistor, enter the differential amplifier gain used to amplify the shunt resistor, as shown in Figure 1-4.

FIGURE 1-4:



The value for `UserParms.h` is: 6.04K / (560 + 40.2) = 10.0633:

```
#define DIFFAMPGAIN     10.0633 // Gain of differential amplifier.
```

4. Now that we are done with hardware changes, we start tuning with MPLAB IDE. Enable RTDM and DMCI in MPLAB IDE so that variables can be monitored in real-time to ensure that the measured single-shunt current is not too noisy, and that the minimum window for single-shunt measurement is wide enough to get a good reading. This is enabled by default, as shown below.

```
#define RTDM         // This definition enabled Real Time Data Monitor, UART interrupts
                     // to handle RTDM protocol, and array declarations for buffering
                     // information in real time
```

5. We will monitor the three currents and I_{BUS} as follows:

```
#define SNAP1      Ia
#define SNAP2      Ib
#define SNAP3      Ic
#define SNAP4      Ibus
```

   This is already set up in the `UserParms.h` file.

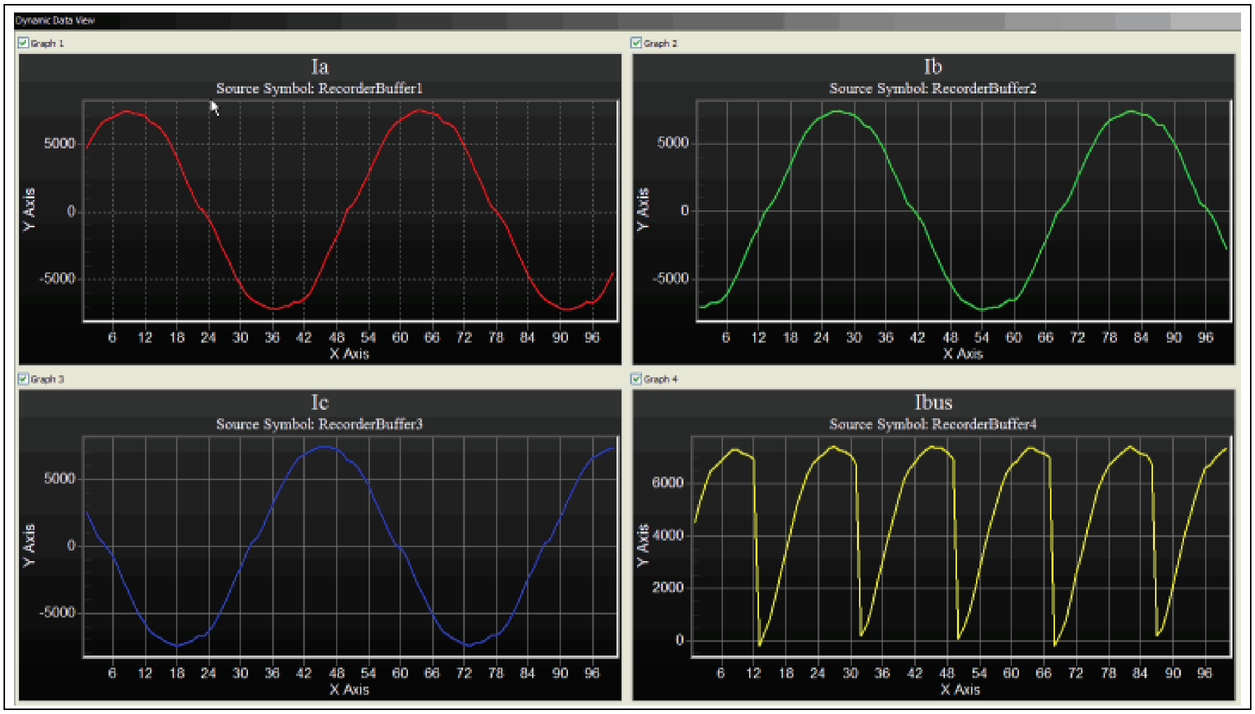
6. Enable running the motor only in open loop, so that the variables can be monitored, and avoid stalling if currents are not measured correctly. To run the algorithm in open loop, make sure the following is defined:

```
#define OPENLOOPONLY// When this is defined, no transition to sensorless
                    // control is done, so dsPIC keeps running the motor
                    // with forced commutation, and no use of theta and speed
                    // estimator output
```

7. Download the code and run it with RTDM enabled. Refer to the `AN1299 Demo ReadMe (MCHV).pdf` or `AN1299 Demo ReadMe (MCLV).pdf` file in the source code archive file for the step-by-step instructions on how to run an enhanced demo with RTDM communications.

   After the start-up ramp, the software will remain in open loop; therefore, estimated position and speed will not be used to control the motor. Reconstructed currents I_A, I_B, and I_C should look sinusoidal, as shown in Figure 1-5. Some distortion may be present during zero crossing, but that is normal.

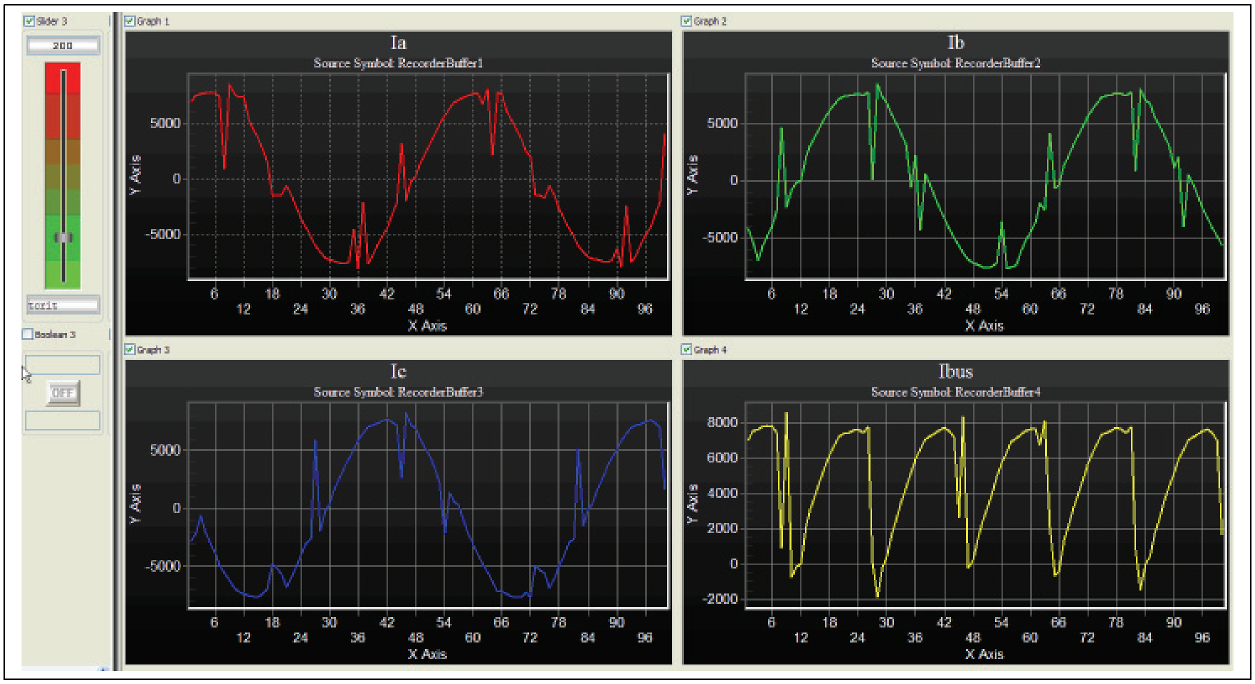
FIGURE 1-5:



The currents shown in Figure 1-5 are for a well-tuned system.

However, currents may be noisy, as shown in the following example plots in Figure 1-6.

FIGURE 1-6:



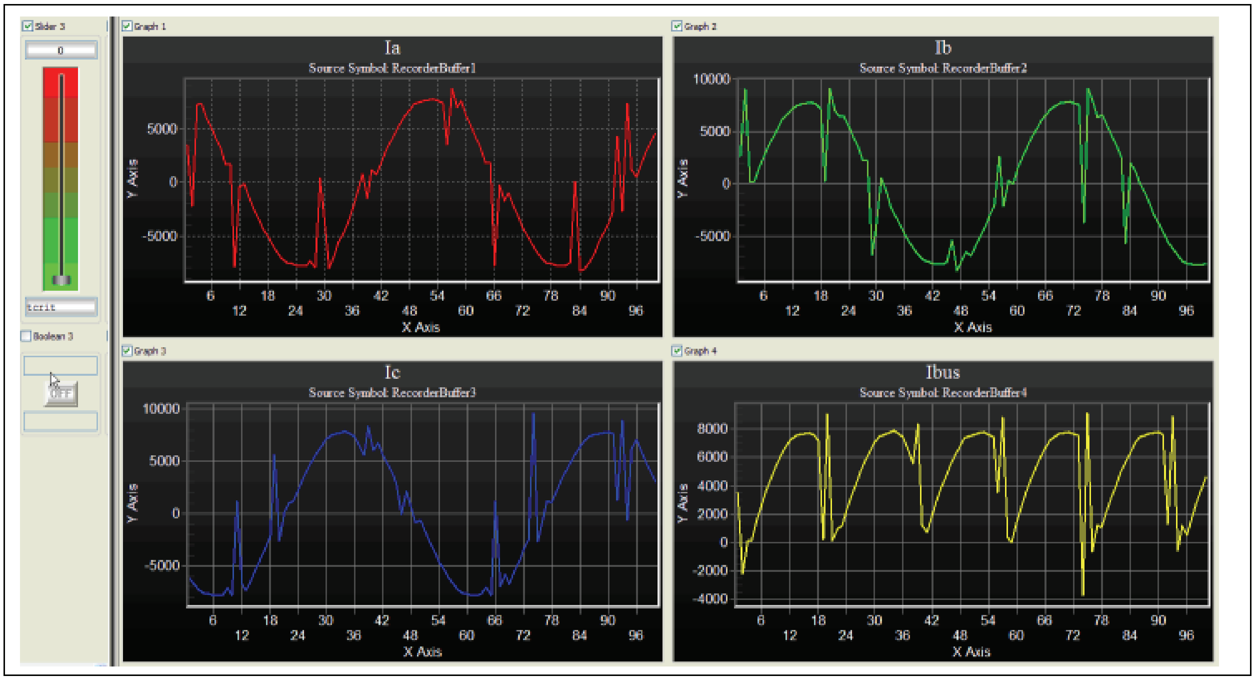
8. Notice the slider named `tcrit` can be tuned by knowing exactly what the OPAMP slew rate and delay is, and then entering a value in seconds to the following define. The value of 3.5E-6 is a good starting point, but you can tune this through experimentation in order to remove the noise in Ia, Ib, and Ic.

```
//************** Single Shunt Algorithm Parameters **********

#define SSTCRITINSEC    3.5E-6  // Critical Minimum window in seconds to measure
                                // current through single shunt
```

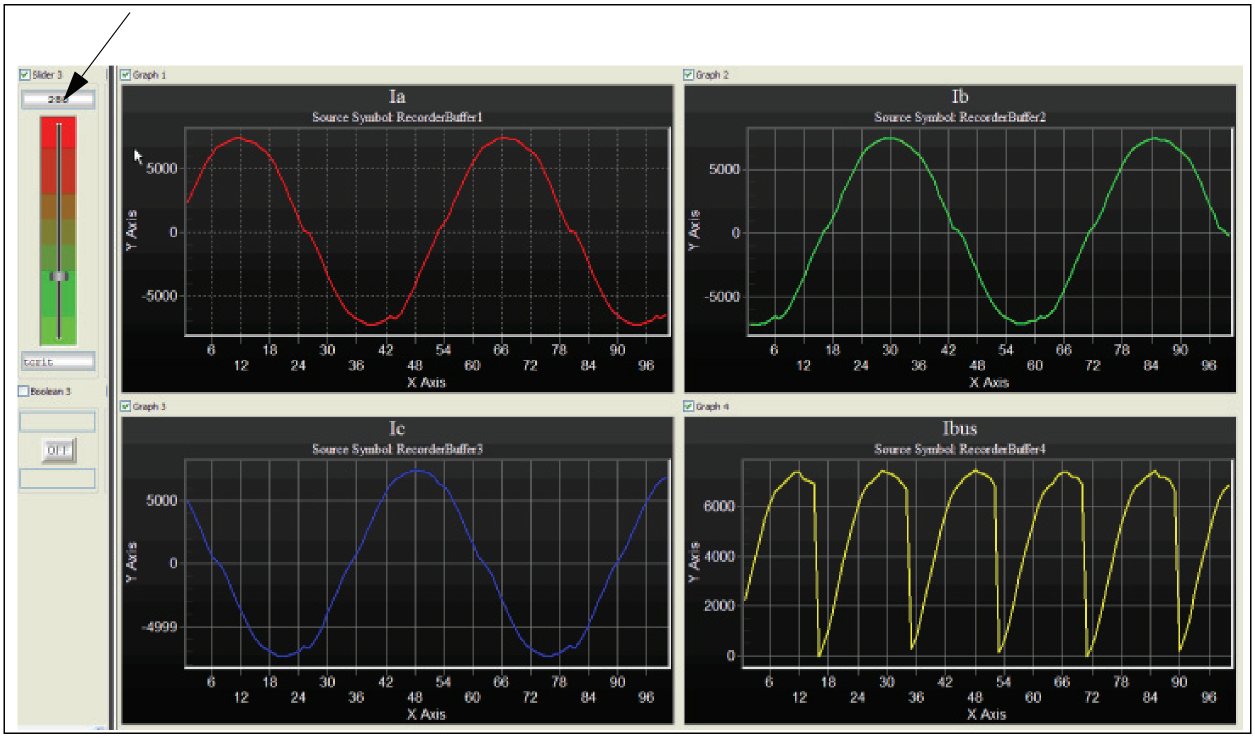
9. Another method is to start `tcrit` with a value of zero, meaning there is no pattern modification to measure single-shunt current, which will generate something like Figure 1-7.

FIGURE 1-7:



10. Then, start increasing this value until noise is no longer present. In the case of the application note running with dsPICDEM™ MCLV and a Hurst motor, a value of 280 was needed to remove all noise from the reconstructed signals.

FIGURE 1-8:



Keep in mind that the larger the value of `tcrit` is, the more distortion will be present in the voltage waveform, even if the currents look sinusoidal. This voltage distortion leads to audible noise, which also needs to be minimized.

Once the optimal value of `tcrit` has been found, convert this value to seconds using the following formula:

`tcrit` in seconds = `tcrit` / (MIPS * 2)

Therefore, if a value of `tcrit` = 280 has the best behavior, the following value needs to be entered in `UserParms.h`:

`SSTCRITINSEC` = 280 / (40 MIPS * 2) = 3.5 µs
(assuming the CPU is running at 40 MIPS)

```
//************** Single Shunt Algorithm Parameters **********

#define SSTCRITINSEC    3.5E-6  // Critical Minimum window in seconds to measure
                                // current through single shunt
```

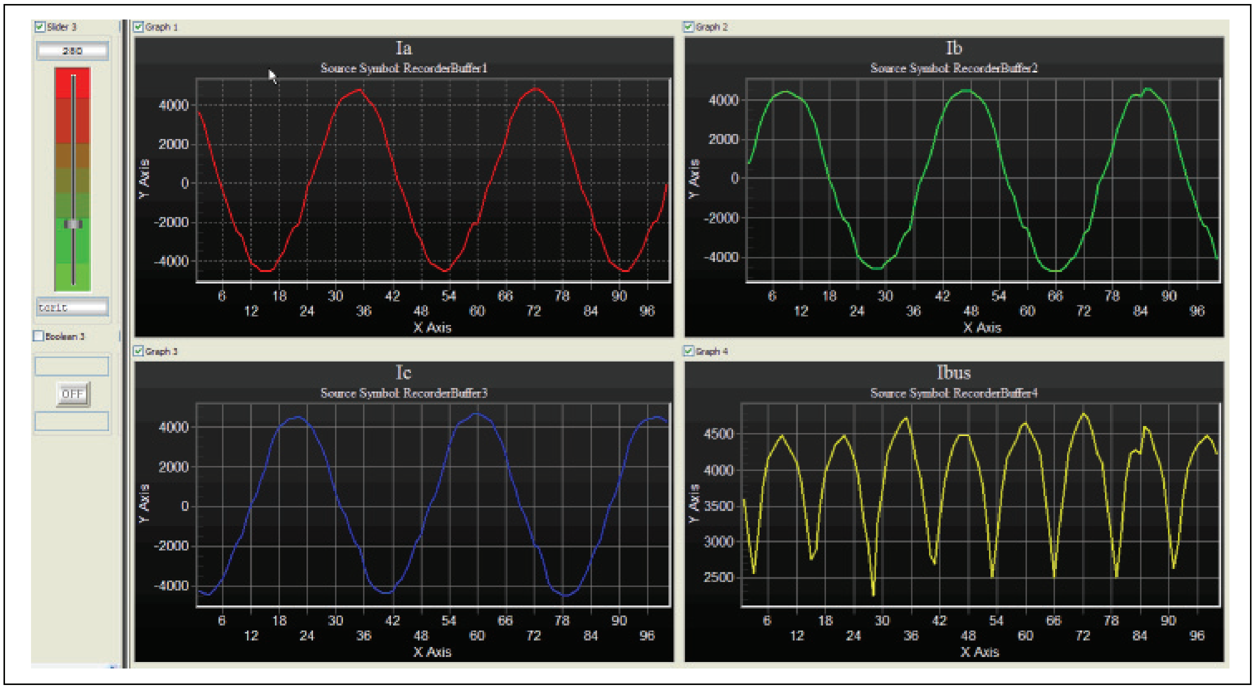
11. Once `SSTCRITINSEC` has been set, enable closed loop by commenting the code as follows:

```
//#define OPENLOOPONLY// When this is defined, no transition to sensorless
                    // control is done, so dsPIC keeps running the motor
                    // with forced commutation, and no use of theta and speed
                    // estimator output
```

Note:	If any user parameter defines are changed, the project needs to be rebuilt and the DSC reprogrammed.

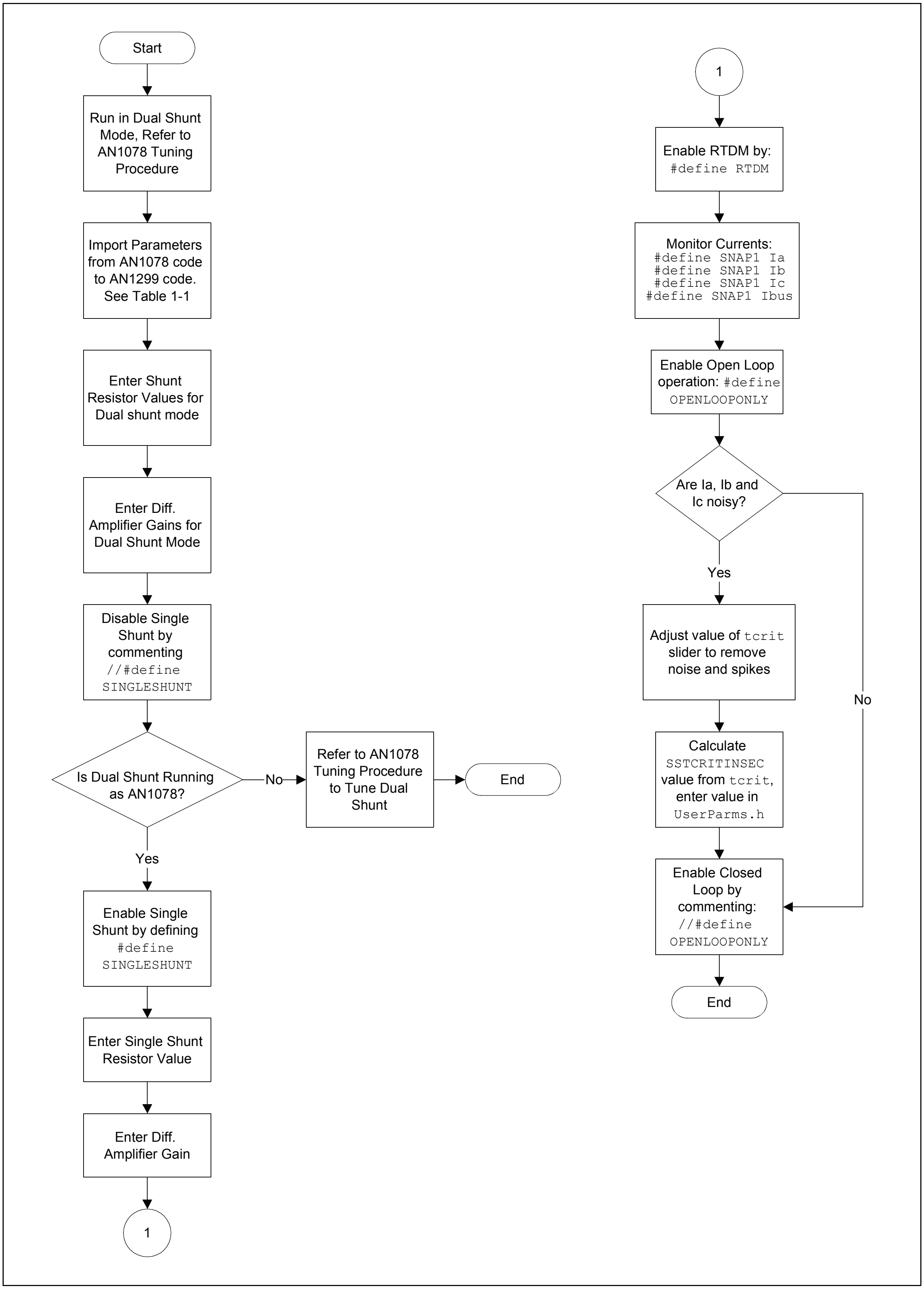
12. Run the motor, and measure the currents running in closed loop mode. Put some load on the motor to see the effect on the motor currents, which should appear as shown in Figure 1-9.

FIGURE 1-9:



If noise is still present, try fine tuning the `tcrit` value in order to have more time to measure the single-shunt current. This should allow the three-phase currents to be reconstructed with minimum noise.

1.3 TUNING PROCEDURE PROCESS FLOW CHART



Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

ISBN: 978-1-60932-344-8

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

01/05/10