Brushless DC motor drive board evaluation

Version: Thursday, April 03, 2014

Applies to: SAT0042 E4 brushless DC motor drive board

1 Initial Evaluation

1.1 Visual inspection

1.1.1 Verify the components are correctly installed

1.1.2 Verify the DNP components are not installed

R3	R14	R36	R71	L3
√	√	√	√	√

1.1.3 Verify the blue wires (if any) are correctly installed

not applicable

1.1.4 Verify pin 1 orientation and diode orientation

1.2 Apply Vbatt (12V) power to board

1.2.1 GND to J1-1 and J1-4, +12V to J1-2 and J1-3

1.2.2 Green LED D6 should be illuminated.

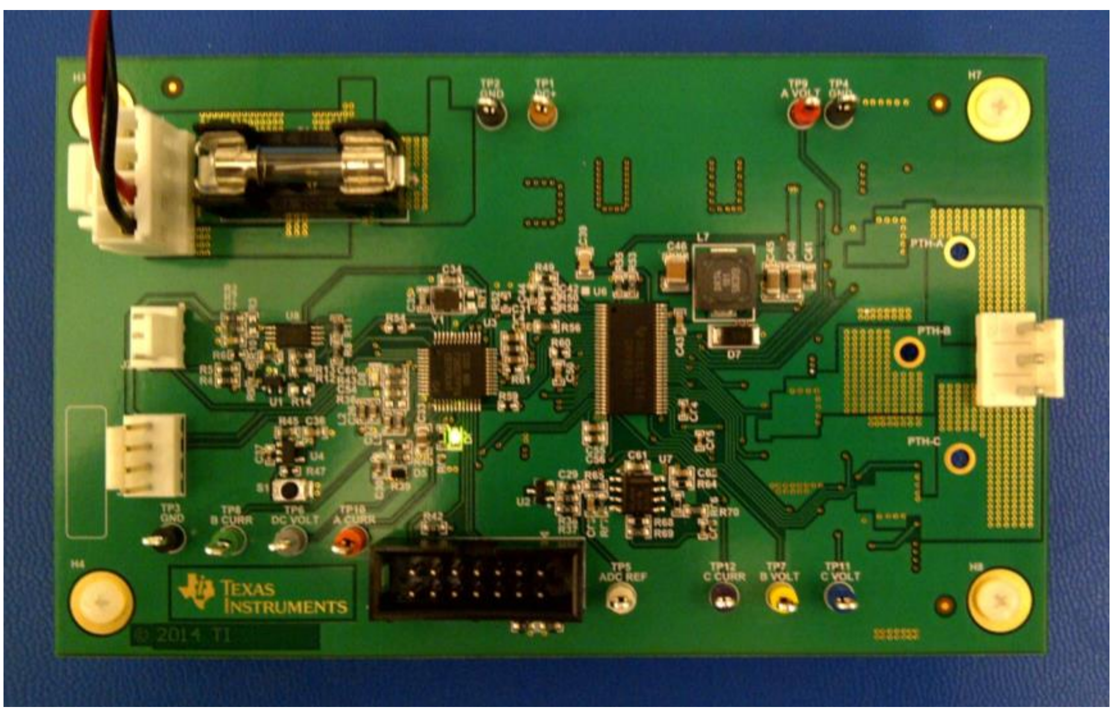


Figure 1 SAT0042 E4 board with 12V power applied and LED D6 illuminated (center)

1.2.3 Measure 12V current (no-load, idle conditions)

Measures approximately 20 mA at 12V before loading microcontroller code.

1.2.4 Verify DC+ (TP1) is 12V with respect to GND (TP2)

1.2.5 Verify +3.3V net is 3.3V; measure +3.3V at J5-4

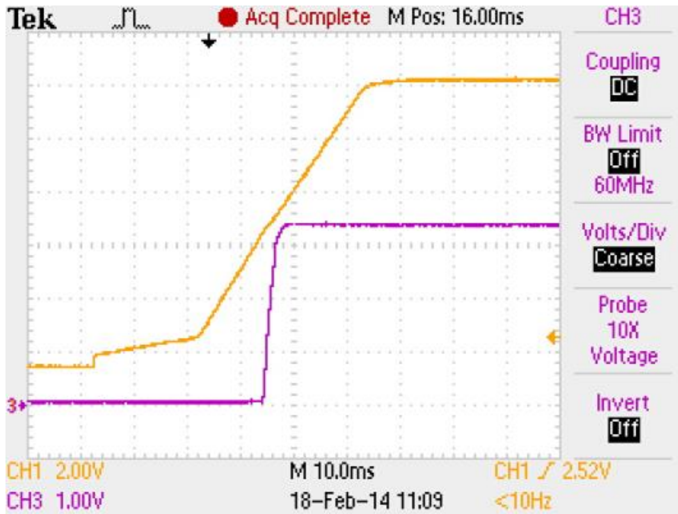


Figure 2 Top (orange) trace is 12V from bench supply, bottom (pink) trace is 3.3V on board

1.2.6 Measure ADC_REF at TP5, should be 3.3V if R36 is installed, should be 3V if R38 is installed.

R38 is installed (R36 is not installed) – measurement is 3.08V.

1.2.7 Verify the 20 MHz clock signal

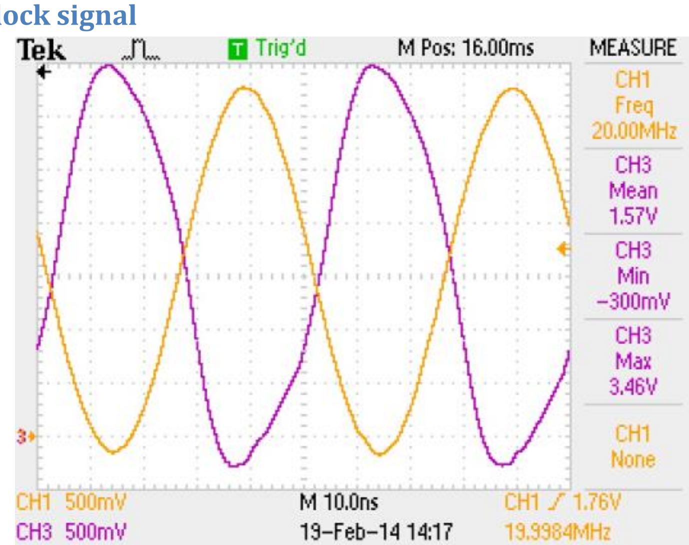


Figure 3 Scope traces of 20 MHz clock signal

1.3 Vary the Vbatt applied

1.3.1 Reduce the Vbatt (12V nominal) voltage until the Green LED is no longer illuminated. Record the voltage at which the Green LED transitions.

LED first illuminates at 6V, and turns off when supply is turned down to 4V.

1.3.2 Measure the input current (no-load, idle conditions) at 8V, 12V, 16V

19 mA @ 8V, 15 mA @ 12V, 13 mA @16V (approximately 180 mW)

1.4 Test point voltages

	Test Point Color	Signal Name	Typical measurement value
TP1		DC+	12V (set by external supply)
TP2	Black	GND	< 40 mV
TP3	Black	GND	< 40 mV
TP4	Black	GND	< 40 mV
TP5		ADC_REF	3.08 V
TP6		DC_Volt	1.04V
TP7		Phase B Voltage	Depends on motor conditions
TP8		Phase B Current	Depends on motor conditions
TP9		Phase A Voltage	Depends on motor conditions
TP10		Phase A Current	Depends on motor conditions
TP11		Phase C Voltage	Depends on motor conditions
TP12		Phase C Current	Depends on motor conditions

1.5 Motor voltage test point frequency response

Table 1 Frequency measurements of motor phase test points (TP7, TP9, TP11)

Input from signal generator, 2Vpp			
Measured at phase voltage test points			
No power on board			
Frequency (Hz)	A TP9	B TP7	C TP11
100	250		
200	250		
500	212		
1000	134	139	129
2000	78	84	78
5000	37	42	37
10000	24	31	21
20000	18	25	18
50000	17	23	13

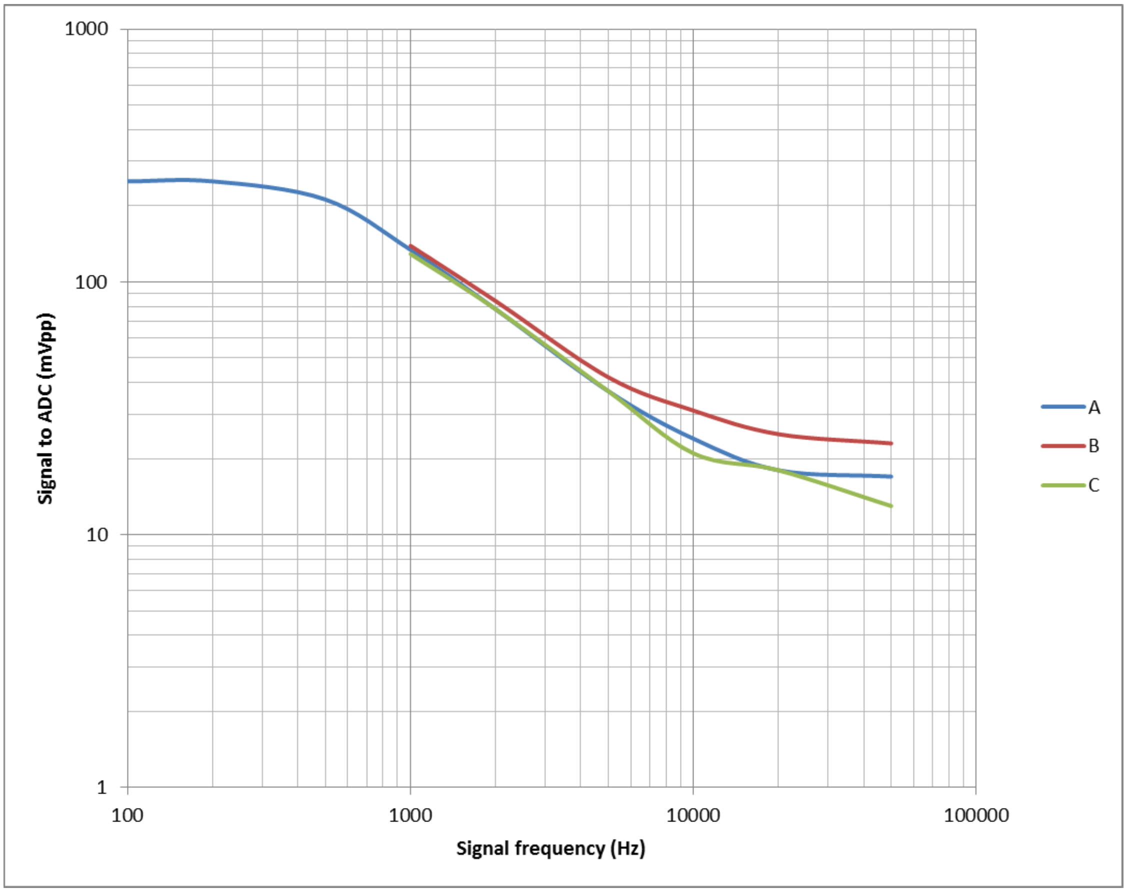


Figure 4 Frequency response of motor phase voltage filters

1.6 Reverse polarity protection

The board has components that should prevent damage during reverse polarity conditions on VBATT with respect to BATT_GND. When VBATT < VGS(th), then Q1 disconnects BATT_GND from the GND node.

No observable current from supply with reversed leads to J1.

2 Load C2000 software

2.1 connect to JTAG, load code with CCS, run in CCS debug environment,

Code Composer Studio™ (CCStudio) is an integrated development environment (IDE) for Texas Instruments (TI) embedded processor families. CCStudio comprises a suite of tools used to develop and debug embedded applications. It includes compilers for each of TI's device families, source code editor, project build environment, debugger, profiler, simulators, real-time operating system and many other features. The intuitive IDE provides a single user interface taking you through each step of the application development flow. Familiar tools and interfaces allow users to get started faster than ever before and add functionality to their application thanks to sophisticated productivity tools.

See the Code Composer Studio web page at http://www.ti.com/tool/ccstudio for information on downloading the integrated development environment for the C2000 code.

Step 1: Import the existing project, for example proj_lab02b, from the motorware directory. In this instance, there are two projects in the directory.

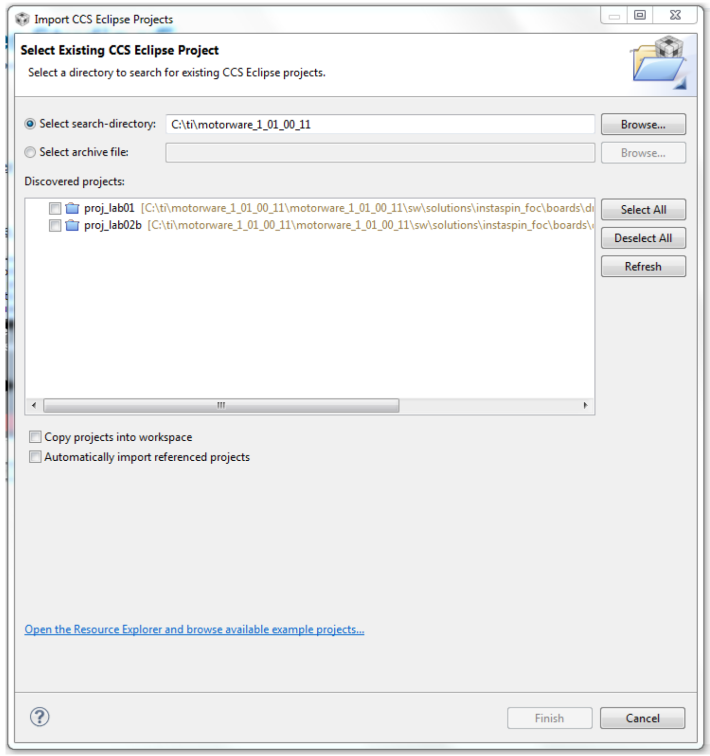


Figure 5 Import Existing CCS Eclipse Project screen

Step 2: Import the project 2b

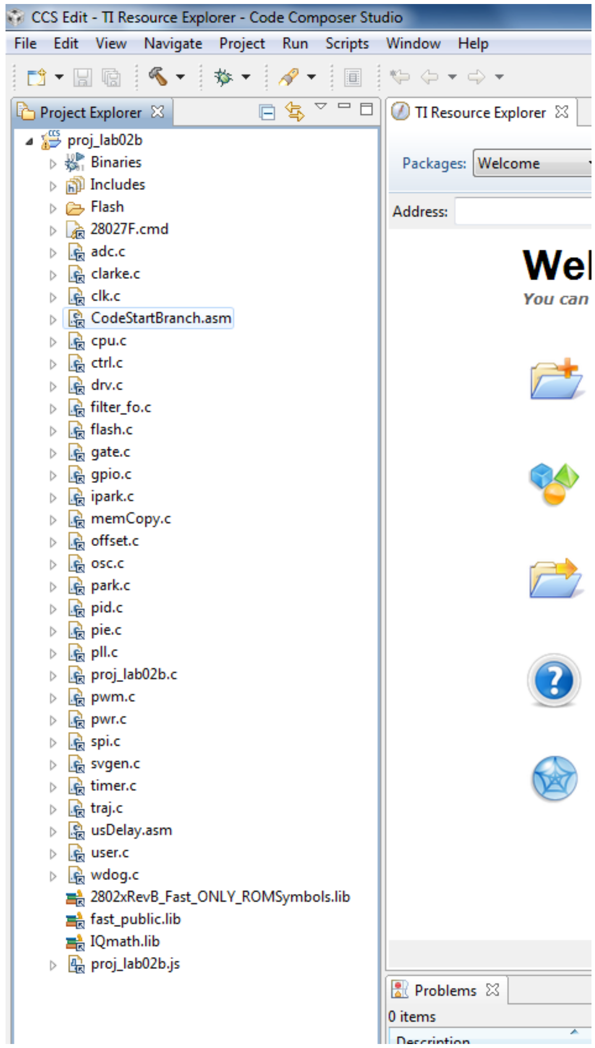


Figure 6 Imported project file and sub-files

Step 3: Set the target configuration:

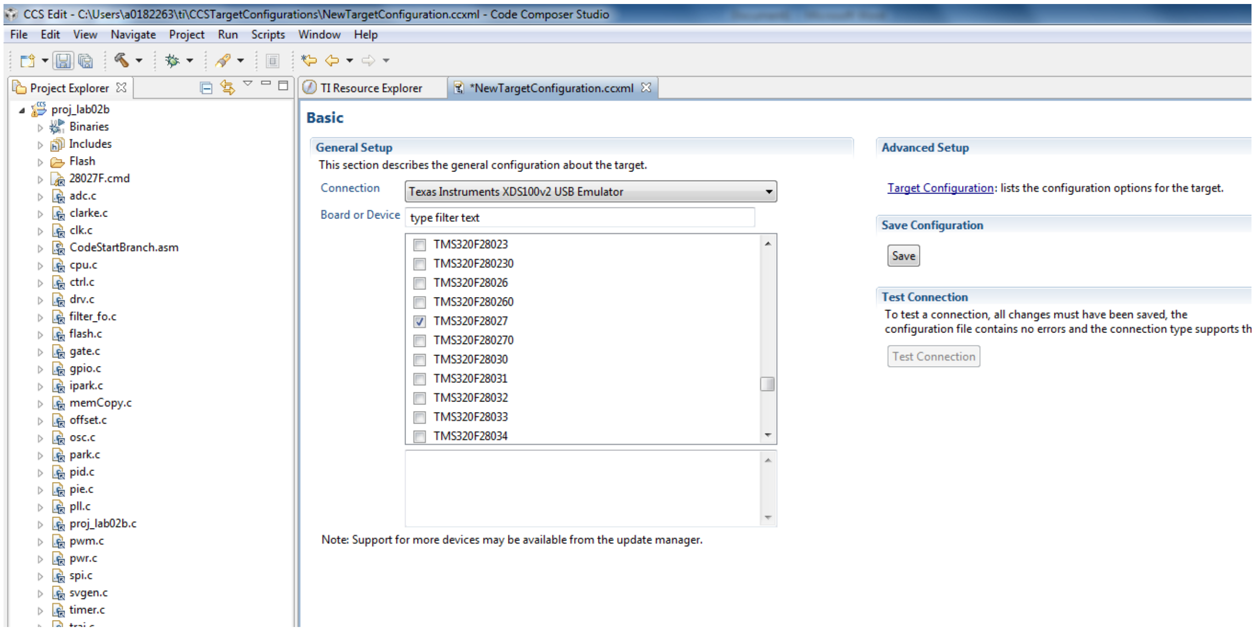


Figure 7 New target configuration screen

The connection will depend on the JTAG emulator you use. The target device on this board is the TMS320F28027 picollo microcontroller. After selecting the connection and target device, save the configuration set-up by clicking the "Save" button.

Step 4: Test the connection to the target.

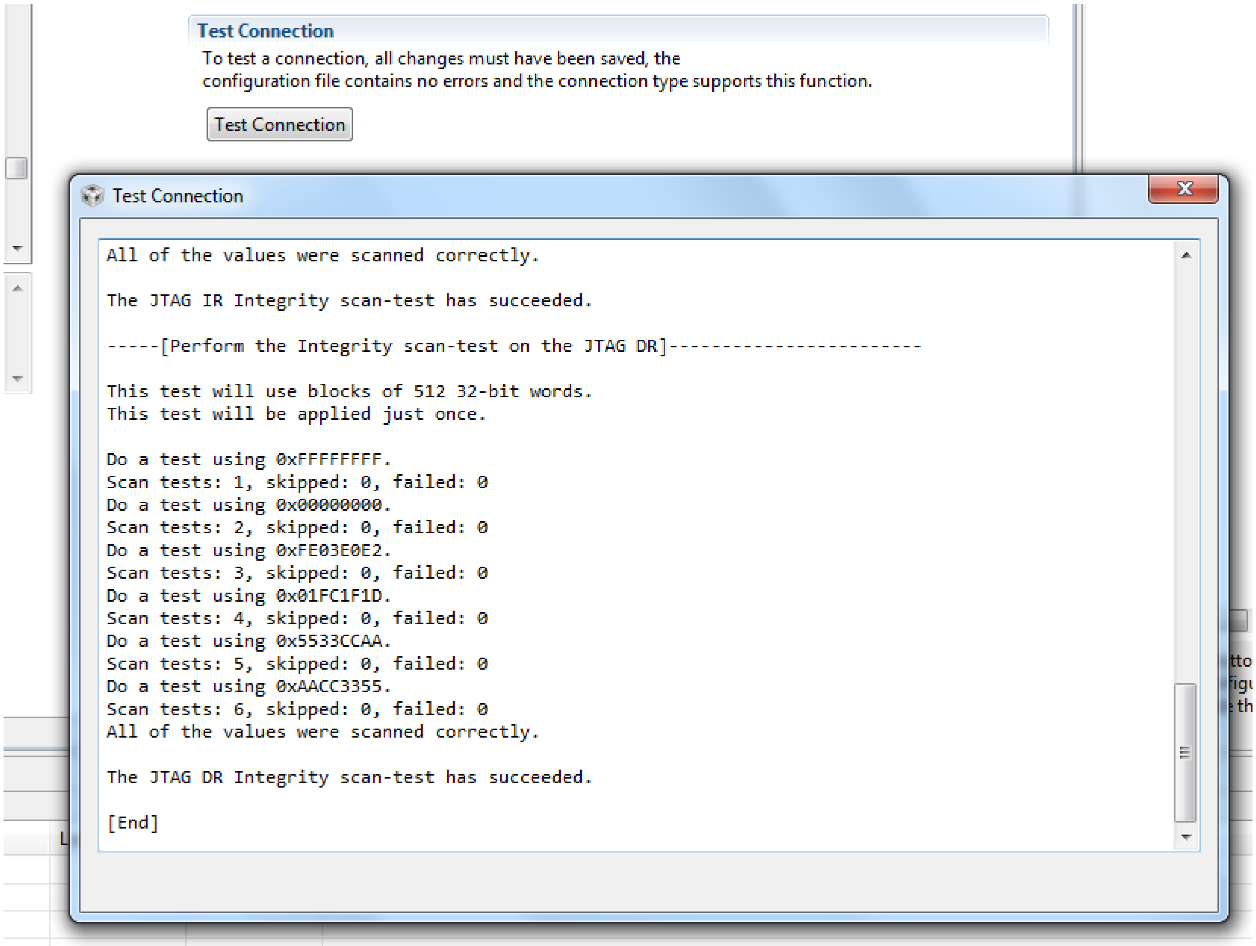


Figure 8 Test connection window after completion of connection test

Step 5: Build the project.

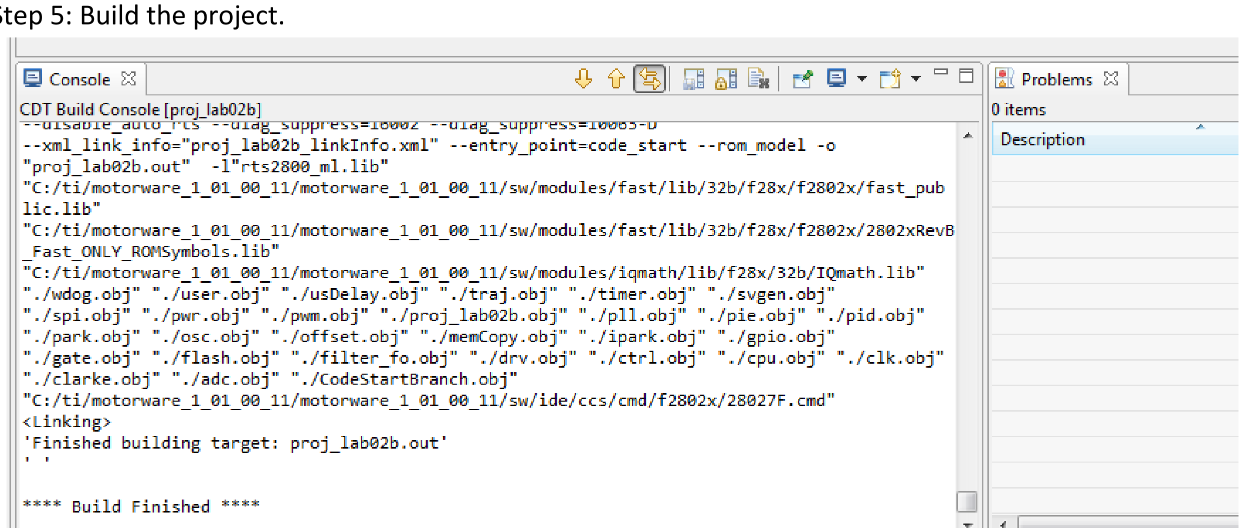
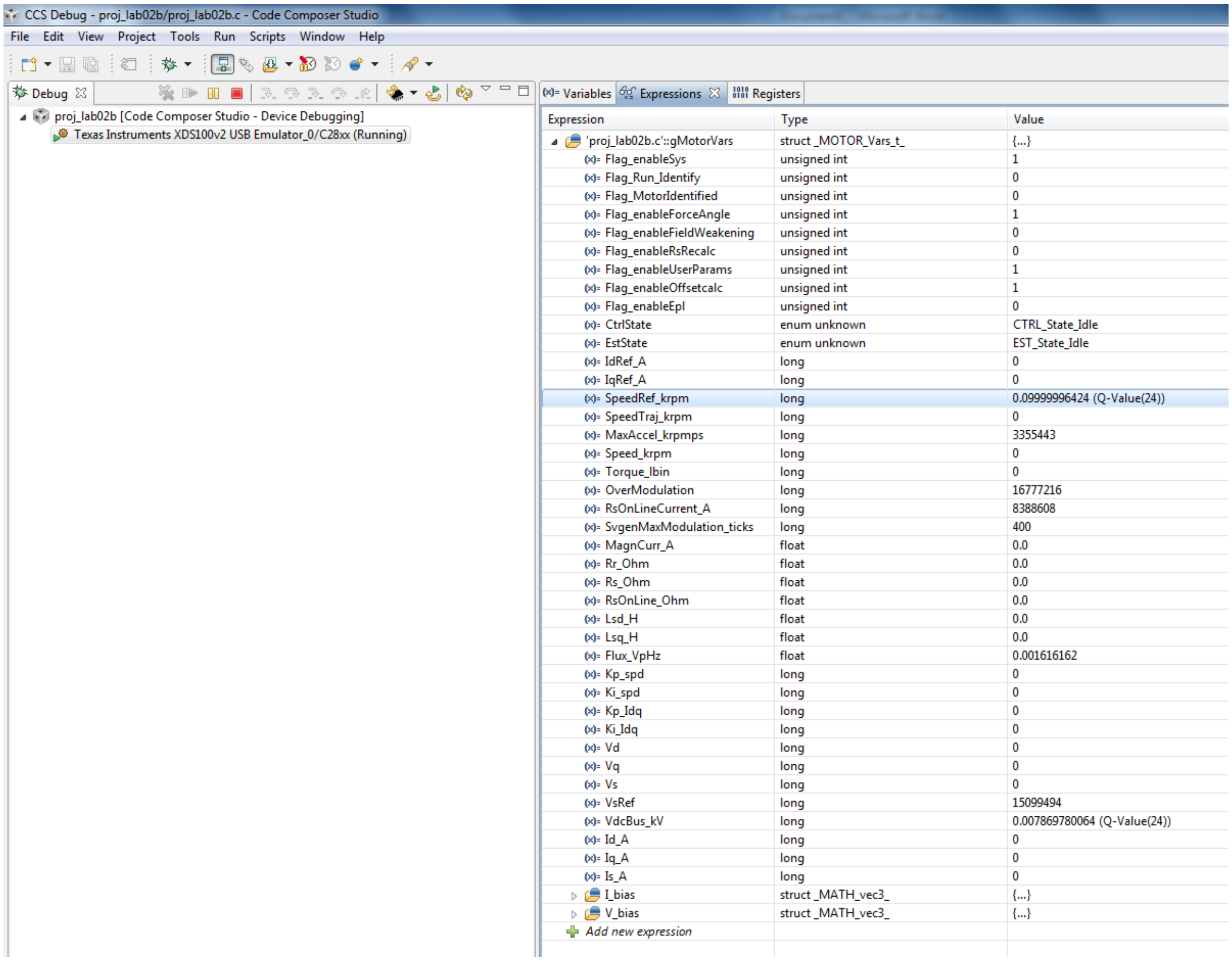


Figure 9 CCS console window after successful build of the project

Step 6: Start a Debug session with the project.



Expression	Type	Value
'proj_lab02b.c'::gMotorVars	struct _MOTOR_Vars_t_	{...}
Flag_enableSys	unsigned int	1
Flag_Run_Identify	unsigned int	0
Flag_MotorIdentified	unsigned int	0
Flag_enableForceAngle	unsigned int	1
Flag_enableFieldWeakening	unsigned int	0
Flag_enableRsRecalc	unsigned int	0
Flag_enableUserParams	unsigned int	1
Flag_enableOffsetcalc	unsigned int	1
Flag_enableEpl	unsigned int	0
CtrlState	enum unknown	CTRL_State_Idle
EstState	enum unknown	EST_State_Idle
IdRef_A	long	0
IqRef_A	long	0
SpeedRef_krpm	long	0.09999996424 (Q-Value(24))
SpeedTraj_krpm	long	0
MaxAccel_krpmps	long	3355443
Speed_krpm	long	0
Torque_lbin	long	0
OverModulation	long	16777216
RsOnLineCurrent_A	long	8388608
SvgenMaxModulation_ticks	long	400
MagnCurr_A	float	0.0
Rr_Ohm	float	0.0
Rs_Ohm	float	0.0
RsOnLine_Ohm	float	0.0
Lsd_H	float	0.0
Lsq_H	float	0.0
Flux_VpHz	float	0.001616162
Kp_spd	long	0
Ki_spd	long	0
Kp_Idq	long	0
Ki_Idq	long	0
Vd	long	0
Vq	long	0
Vs	long	0
VsRef	long	15099494
VdcBus_kV	long	0.007869780064 (Q-Value(24))
Id_A	long	0
Iq_A	long	0
Is_A	long	0
I_bias	struct _MATH_vec3_	{...}
V_bias	struct _MATH_vec3_	{...}
Add new expression		

Figure 10 Expressions window in the debug view of CCS, program running

Step 7: Run the project

Change the expression VdcBus_kV to Q-Value(24) by right-clicking in the Value field, then selecting Q-values and "24". Verify that the value for VdcBus_kV corresponds to the DC supply voltage. In the first case (below), the supply voltage is 8V.

VsRef	long	15099494	0x0000007C@Data
VdcBus_kV	long	0.007869780064 (Q-Value(24))	0x0000007E@Data
Id_A	long	0	0x00000080@Data

In the second case (below), the supply voltage is 7V.

Vs	long	0	0x0000007A@Data
VsRef	long	15099494	0x0000007C@Data
VdcBus_kV	long	0.006873965263 (Q-Value(24))	0x0000007E@Data
Id_A	long	0	0x00000080@Data

In the third case (below), the supply voltage is 12V.

Vs	long	0	0x0000007A@Data
VsRef	long	15099494	0x0000007C@Data
VdcBus_kV	long	0.01185280085 (Q-Value(24))	0x0000007E@Data
Id_A	long	0	0x00000080@Data

Step 8a: Set the Flag_enableSys to 1 by clicking in the value field and entering a 1.

Step 8b: Set the Flag_Run_Identify to 1 by clicking in the value field and entering a 1.

Motor will be driven with small and large motions, drawing up to 5+ Amps. After about a minute, the Flag_MotorIdentified is set to 1 by the controller. This indicates the motor has been successfully identified for sensorless operation.

Variables | Expressions | Registers

Expression	Type	Value	Address
'proj_lab02b.c'::gMotorVars	struct _MOTOR_Vars_t_	{...}	0x00000040@Data
Flag_enableSys	unsigned int	1	0x00000040@Data
Flag_Run_Identify	unsigned int	0	0x00000041@Data
Flag_MotorIdentified	unsigned int	1	0x00000042@Data
Flag_enableForceAngle	unsigned int	1	0x00000043@Data
Flag_enableFieldWeakening	unsigned int	0	0x00000044@Data
Flag_enableRsRecalc	unsigned int	0	0x00000045@Data
Flag_enableUserParams	unsigned int	1	0x00000046@Data
Flag_enableOffsetcalc	unsigned int	1	0x00000047@Data
Flag_enableEpl	unsigned int	0	0x00000048@Data
CtrlState	enum unknown	CTRL_State_Idle	0x00000049@Data
EstState	enum unknown	EST_State_Idle	0x0000004A@Data

Variables | Expressions | Registers

Expression	Type	Value	Address
'proj_lab02b.c'::gMotorVars	struct _MOTOR_Vars_t_	{...}	0x00000040@Data
Flag_enableSys	unsigned int	1	0x00000040@Data
Flag_Run_Identify	unsigned int	1	0x00000041@Data
Flag_MotorIdentified	unsigned int	1	0x00000042@Data
Flag_enableForceAngle	unsigned int	1	0x00000043@Data
Flag_enableFieldWeakening	unsigned int	0	0x00000044@Data
Flag_enableRsRecalc	unsigned int	0	0x00000045@Data
Flag_enableUserParams	unsigned int	1	0x00000046@Data
Flag_enableOffsetcalc	unsigned int	1	0x00000047@Data
Flag_enableEpl	unsigned int	0	0x00000048@Data
CtrlState	enum unknown	CTRL_State_OnLine	0x00000049@Data
EstState	enum unknown	EST_State_OnLine	0x0000004A@Data
IdRef_A	long	0	0x0000004C@Data
IqRef_A	long	0	0x0000004E@Data
SpeedRef_krpm	long	0.3000000119 (Q-Value(24))	0x00000050@Data
SpeedTraj_krpm	long	5033158	0x00000052@Data
MaxAccel_krpmps	long	3355443	0x00000054@Data
Speed_krpm	long	0.3000246882 (Q-Value(24))	0x00000056@Data
Torque_lbin	long	0.008493959904 (Q-Value(24))	0x00000058@Data
OverModulation	long	16777216	0x0000005A@Data
RsOnLineCurrent_A	long	0.5 (Q-Value(24))	0x0000005C@Data
SvgenMaxModulation_ticks	long	400	0x0000005E@Data
MagnCurr_A	float	0.0	0x00000060@Data
Rr_Ohm	float	0.0	0x00000062@Data
Rs_Ohm	float	1.101674	0x00000064@Data
RsOnLine_Ohm	float	0.0	0x00000066@Data
Lsd_H	float	1.182403e-09	0x00000068@Data
Lsq_H	float	1.182403e-09	0x0000006A@Data
Flux_VpHz	float	0.03356038	0x0000006C@Data
Kp_spd	long	0	0x0000006E@Data
Ki_spd	long	0	0x00000070@Data
Kp_Idq	long	0	0x00000072@Data
Ki_Idq	long	0	0x00000074@Data
Vd	long	0	0x00000076@Data
Vq	long	0	0x00000078@Data
Vs	long	0	0x0000007A@Data
VsRef	long	15099494	0x0000007C@Data
VdcBus_kV	long	0.01183354855 (Q-Value(24))	0x0000007E@Data
Id_A	long	0	0x00000080@Data
Iq_A	long	0	0x00000082@Data
Is_A	long	0	0x00000084@Data
I_bias	struct _MATH_vec3_	{...}	0x00000086@Data
V_bias	struct _MATH_vec3_	{...}	0x0000008C@Data
Add new expression			

Figure 11 Exressions window in CCS debug view during motor operation

Table 2 Analog-to-digital converter assignments on SAT0042 E4 board

From drv.c
//configure the SOCs for drv8301_027_ref
// sample the first sample twice due to errata sprz342f
//drv8301_027_ref
// ADC-A0 ADC_REF
// ADC-A1 IA-FB x
// ADC-A2 AIO2 mode (LED)
// ADC-A3 IC-FB x
// ADC-A4 AIO4 mode (U5 OUT)
// ADC-A5 internal temp sensor
// ADC-A6 IC-FB
// ADC-A7 ADC-Vhb2 (phase B) x
// ADC-B0 not available on 027
// ADC-B1 IB-FB x
// ADC-B2 VDCBUS x
// ADC-B3 IA-FB
// ADC-B4 ADC-Vhb3 (phase C) x
// ADC-B5 not available on 027
// ADC-B6 IB-FB
// ADC-B7 ADC-Vhb1 (phase A) x

2.2 read ADC measurements of, DC voltage at idle, currents should be zero

Supply voltage	VdcBus_kV (Q-Value(24))	Voltage
8V	0.00786	7.86 V
12V	0.01183	11.83 V
15V	0.01481	14.81 V

3 Spin a motor ("kit" motor)

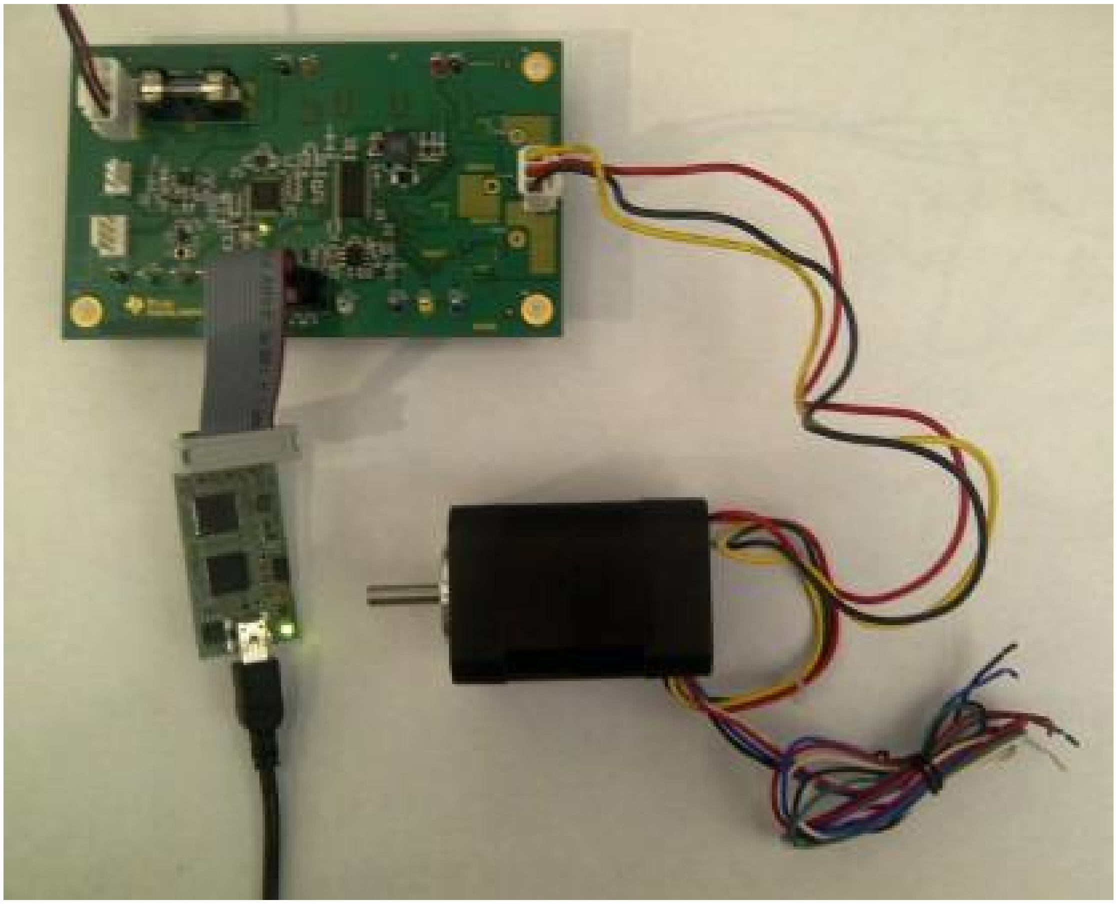


Figure 12 SAT0042 E4 board and Telco motor

3.1 Control the motor drive functions through CCS/JTAG

3.2 Use InstaSpin to identify motor parameters

3.3 Run at speed with nominal supply, no load, record currents, voltages

Kit motor	12VDC
Speed	Current
0 before loading	0.041
3000	0.35

4 Spin a pump motor (Cooper Standard 50W water pump)

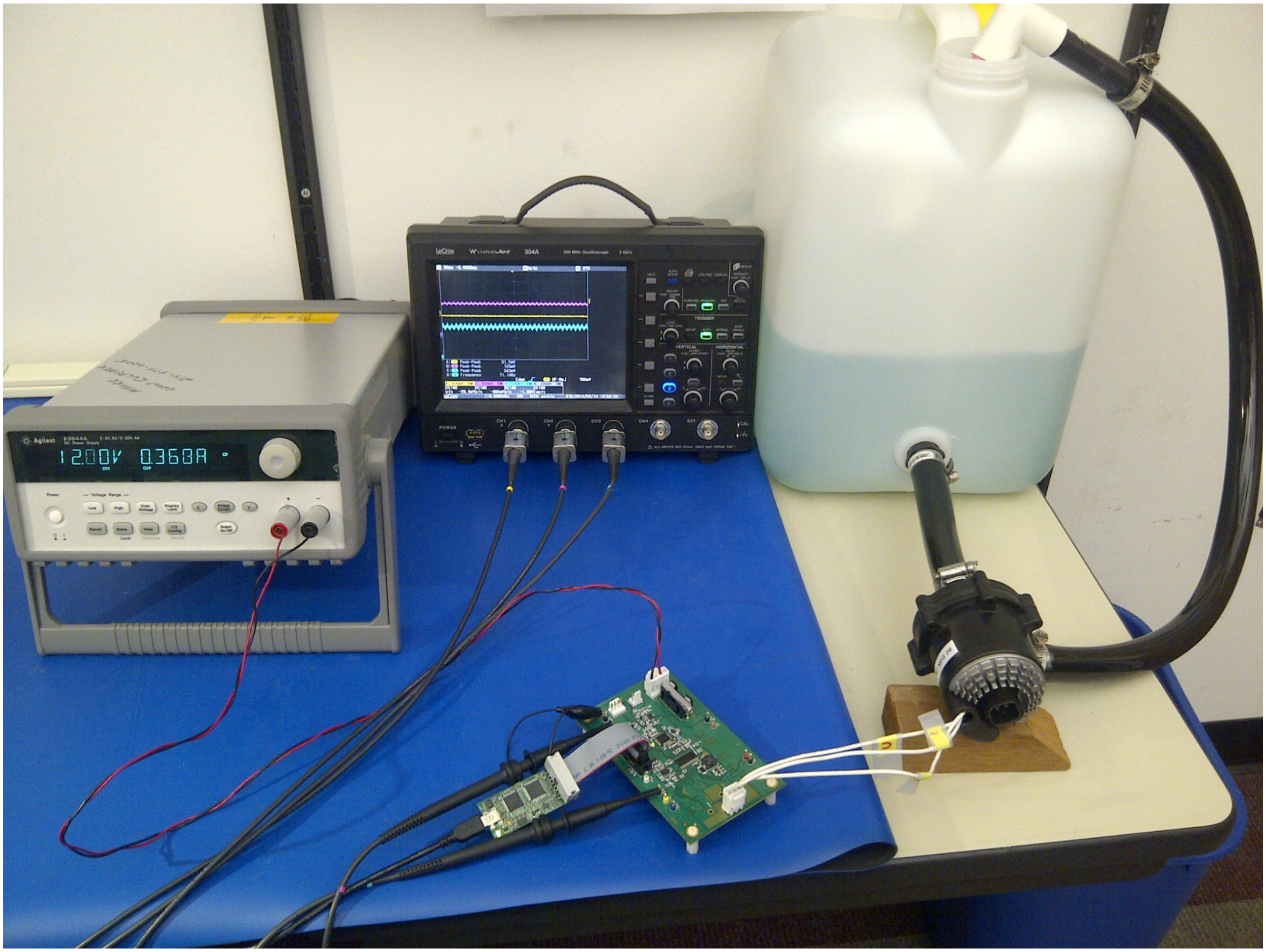


Figure 13 Test set-up with Cooper Standard 50W water pump

4.1 Instaspin to identify motor parameters

Cooper Standard Pump electrical parameters

Stator resistance:	0.26 Ohms
Stator Inductance	0.69 mH

4.2 Run at 1000 rpm with nominal supply, no load, record currents, voltages

Note: due to the construction of typical water pump motors, it is not recommended to run the motor for long periods of time without water or another fluid flowing through the pump.

4.3 Vary speed (positive only) and measure current

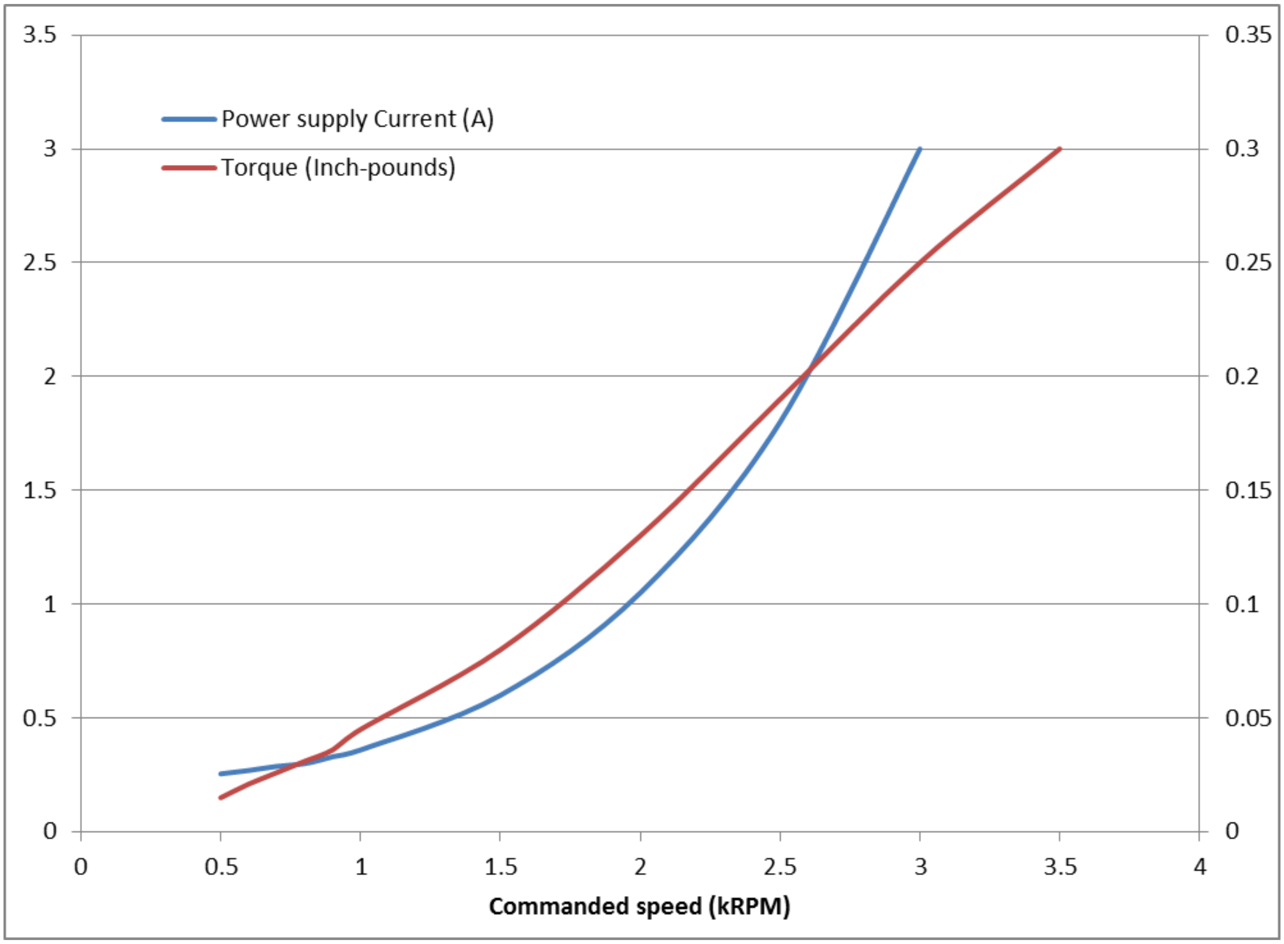


Figure 14 Power and torque as a function of motor speed

4.4 Vary supply voltage and observe changes in current and speed

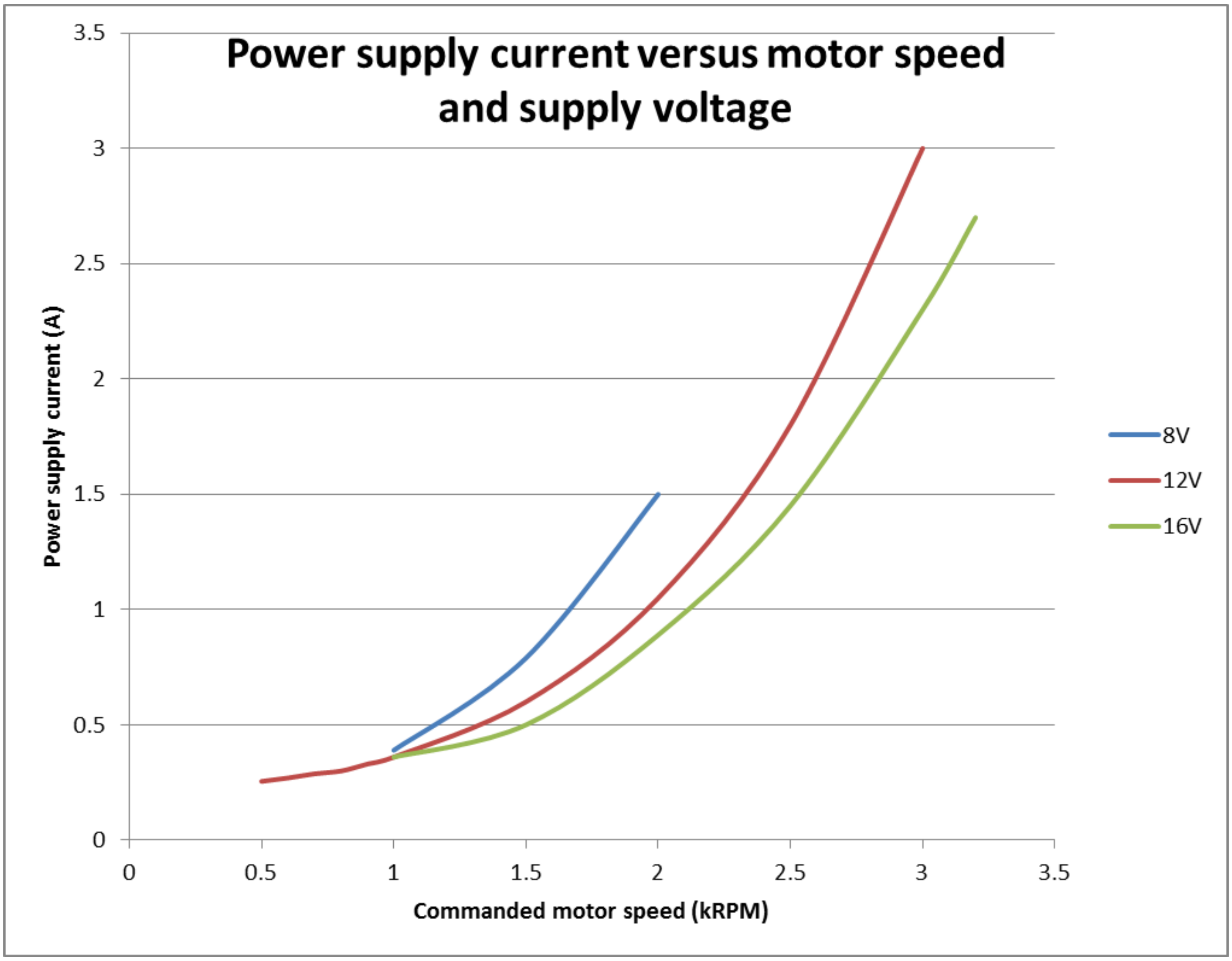


Figure 15 Power supply current versus supply voltage and motor speed

4.5 Measure flow rate (flow meter or bucket method)

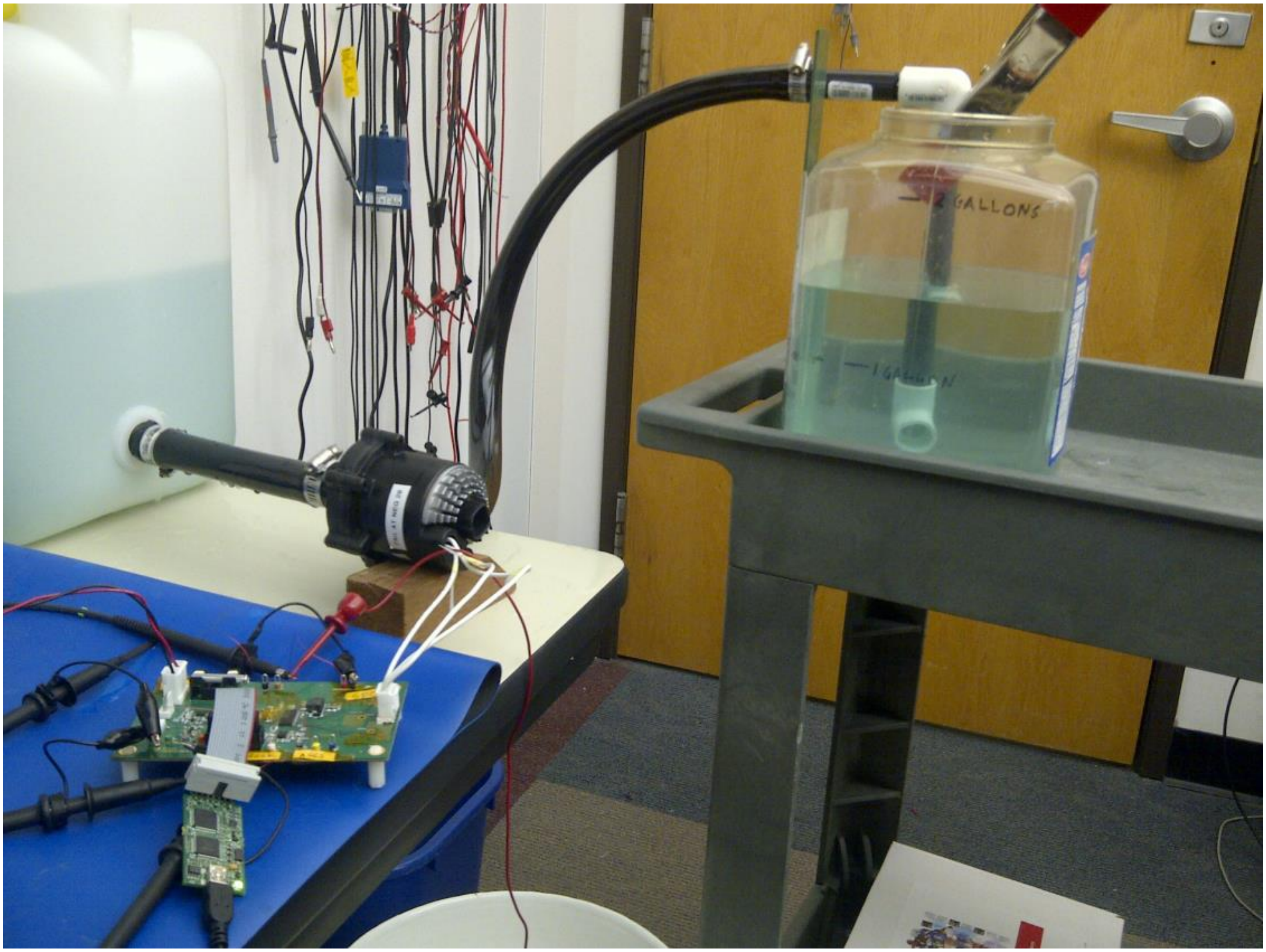


Figure 16 Flow rate measurement set-up

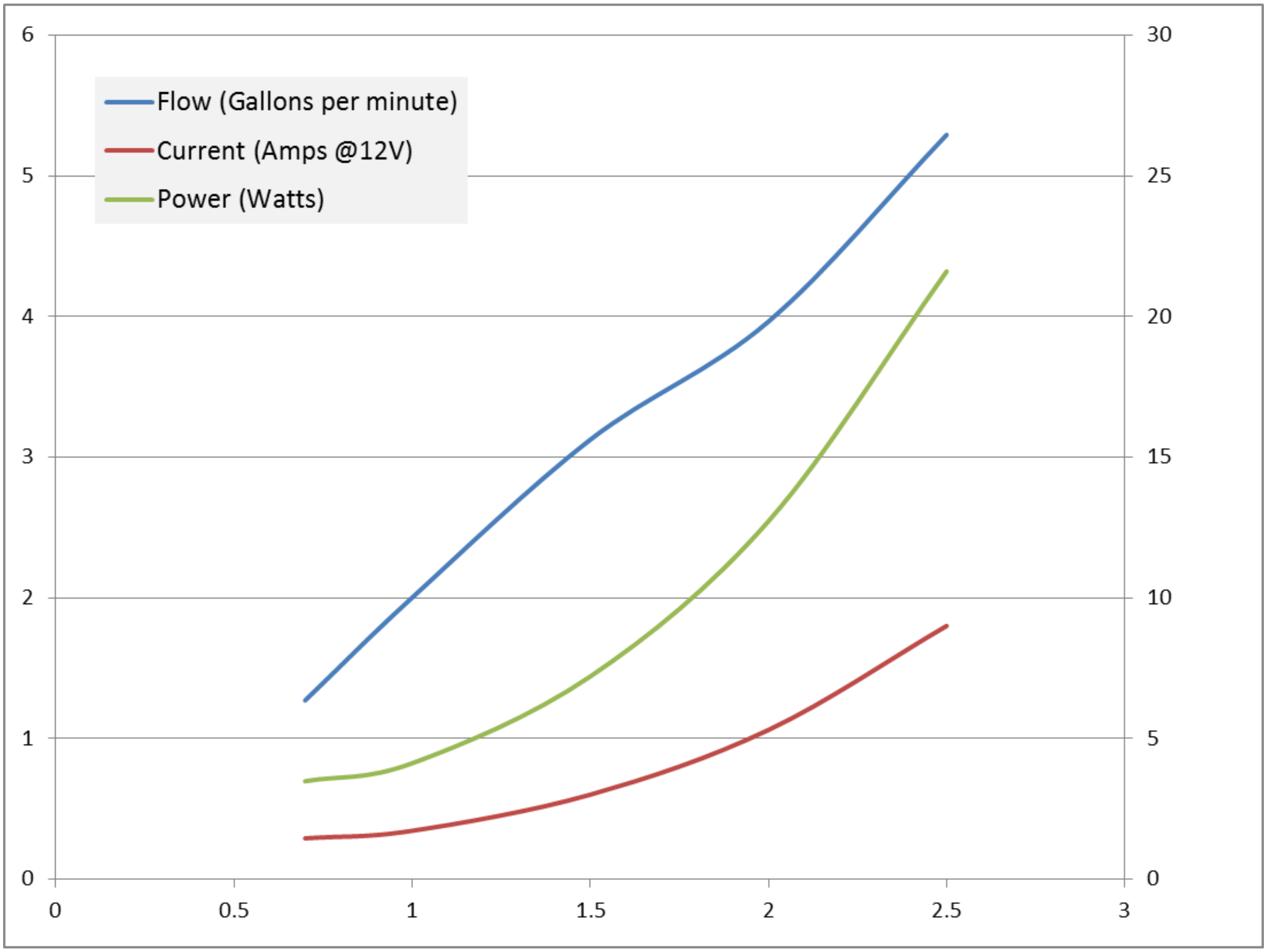


Figure 17 Flow rate measurements with Cooper Standard 50W water pump

5 Additional test data

5.1 Idle current with no dynamic motor load – table of current versus supply voltage

Conditions	Supply current with 12Vdc supply
Motor disabled, microcontroller not running	26 mA
Motor disabled, microcontroller program started, Flag_enableSys = 0	41 mA
Motor disabled, microcontroller program started, Flag_enableSys = 1	42 mA
Motor @ 0 RPM, microcontroller running, Flag_enableSys = 1	235 mA

5.2 Operational supply range

The DRV8301 buck converter correctly indicates POWER_GOOD when the input power on J1 is above 5.9V. When the input power on J1 is less than 4.7V, the DRV8301 will discontinue generating a 3.3V supply.

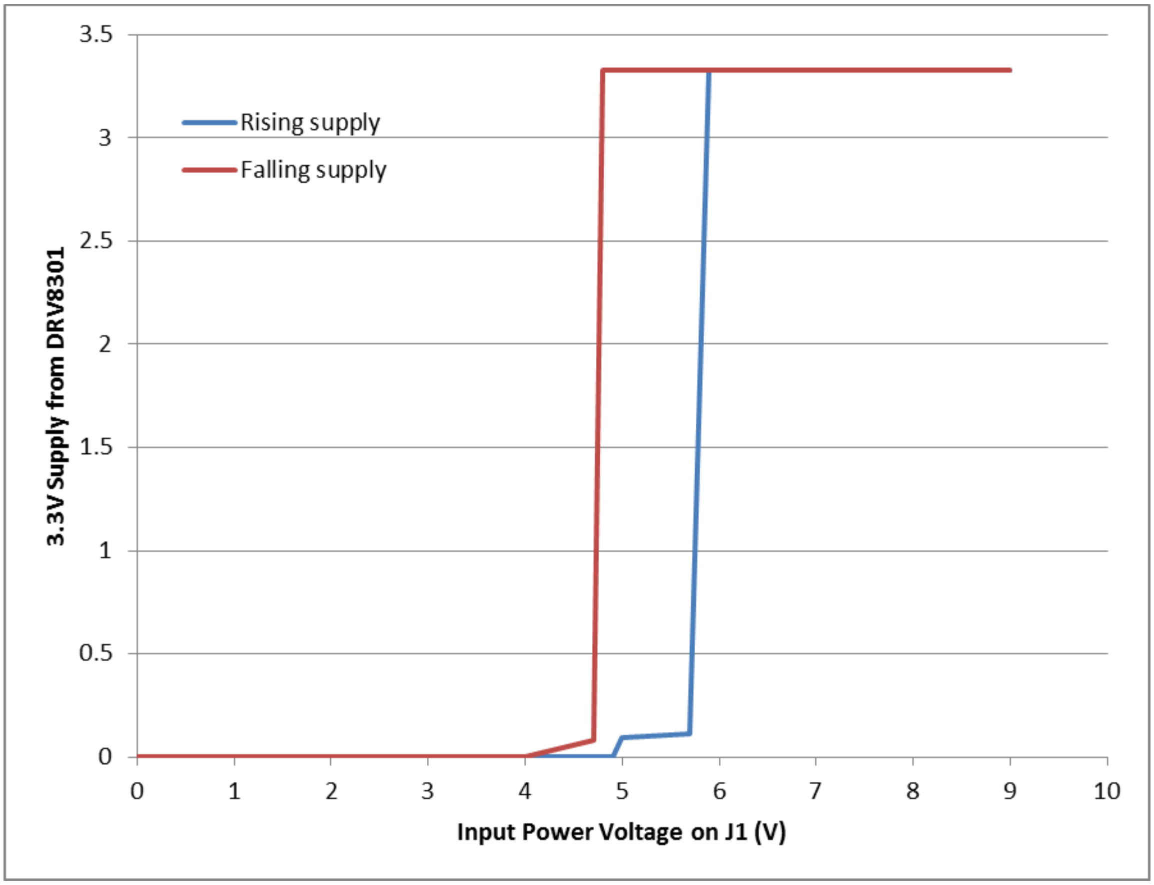


Figure 18 Operational range is indicated by 3.3V power supply versus input supply

5.3 Current sense voltages at load – oscilloscope plots

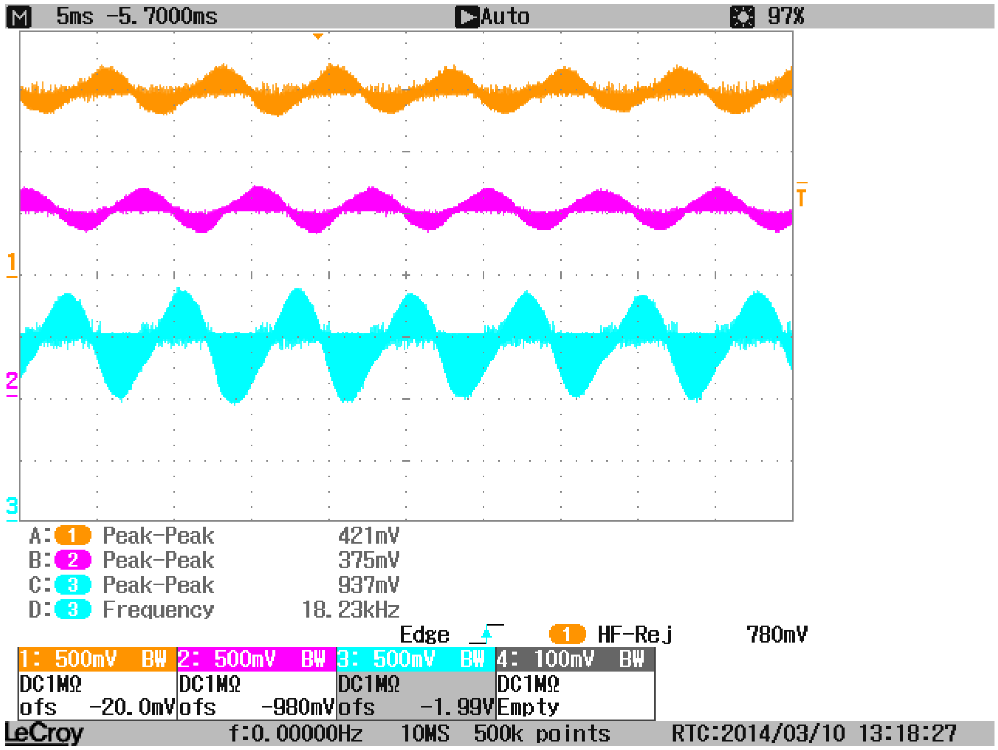


Figure 19 Motor current sense signals showing sinusoidal envelope

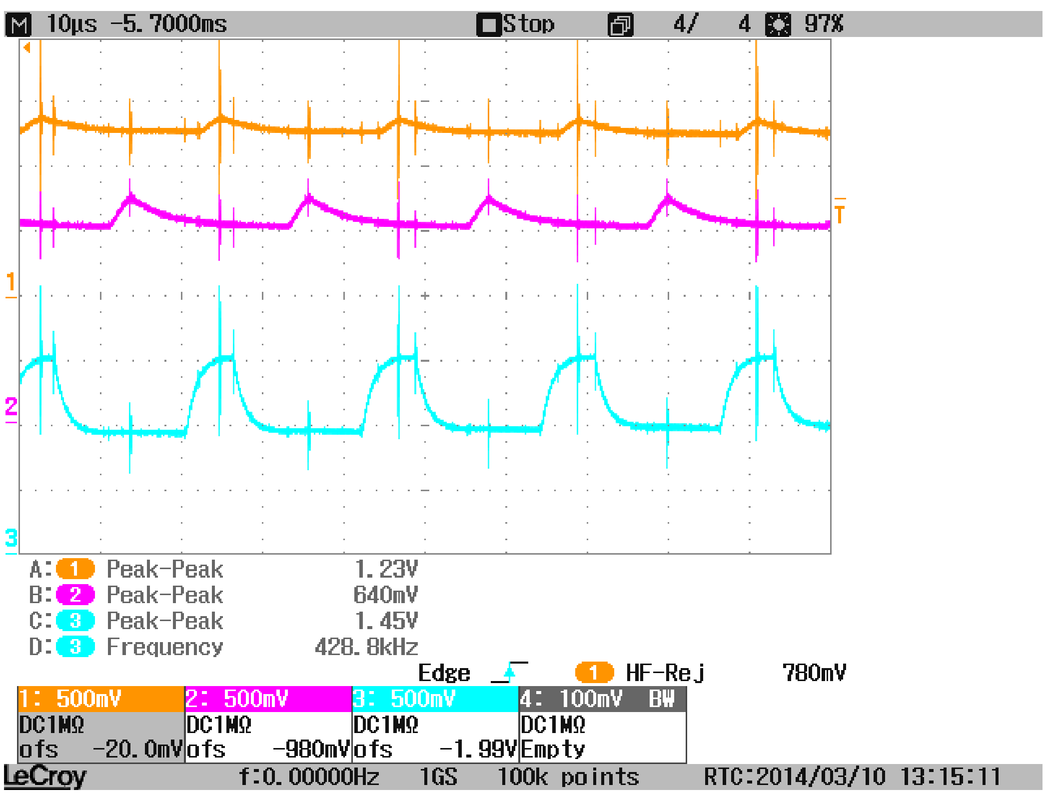


Figure 20 Detail of motor current sense signals

5.4 Phase voltages at load – oscilloscope plots

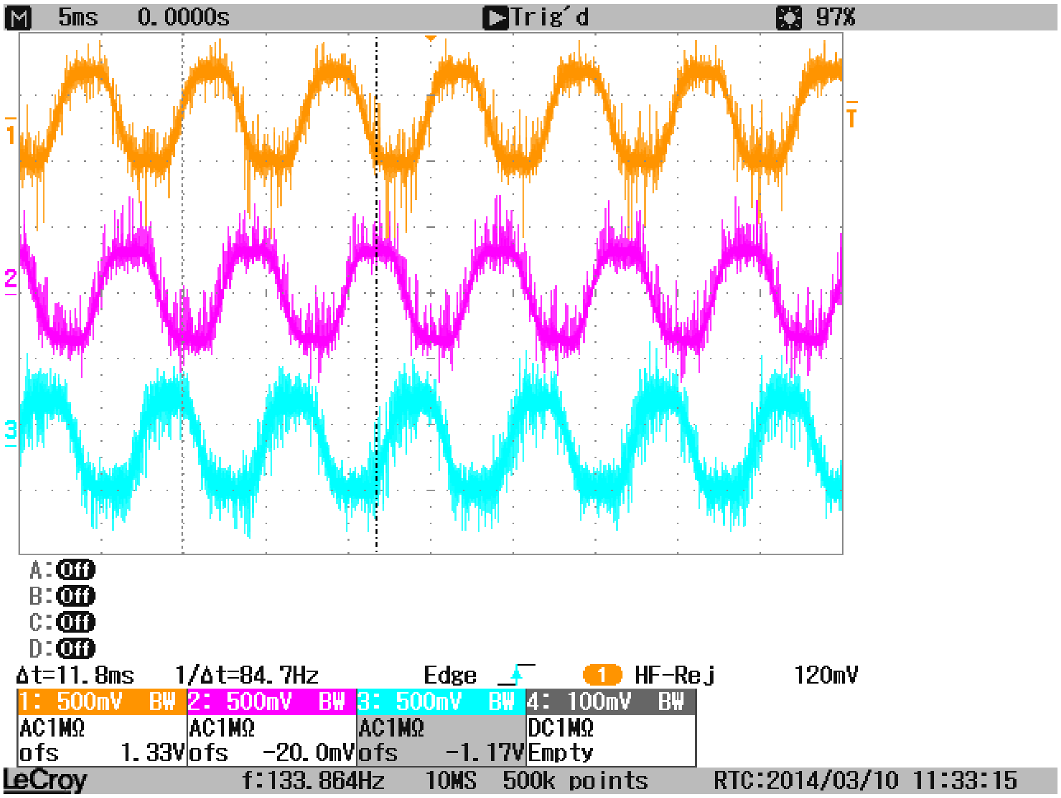


Figure 21 Three-phase motor voltages after filtering and scaling

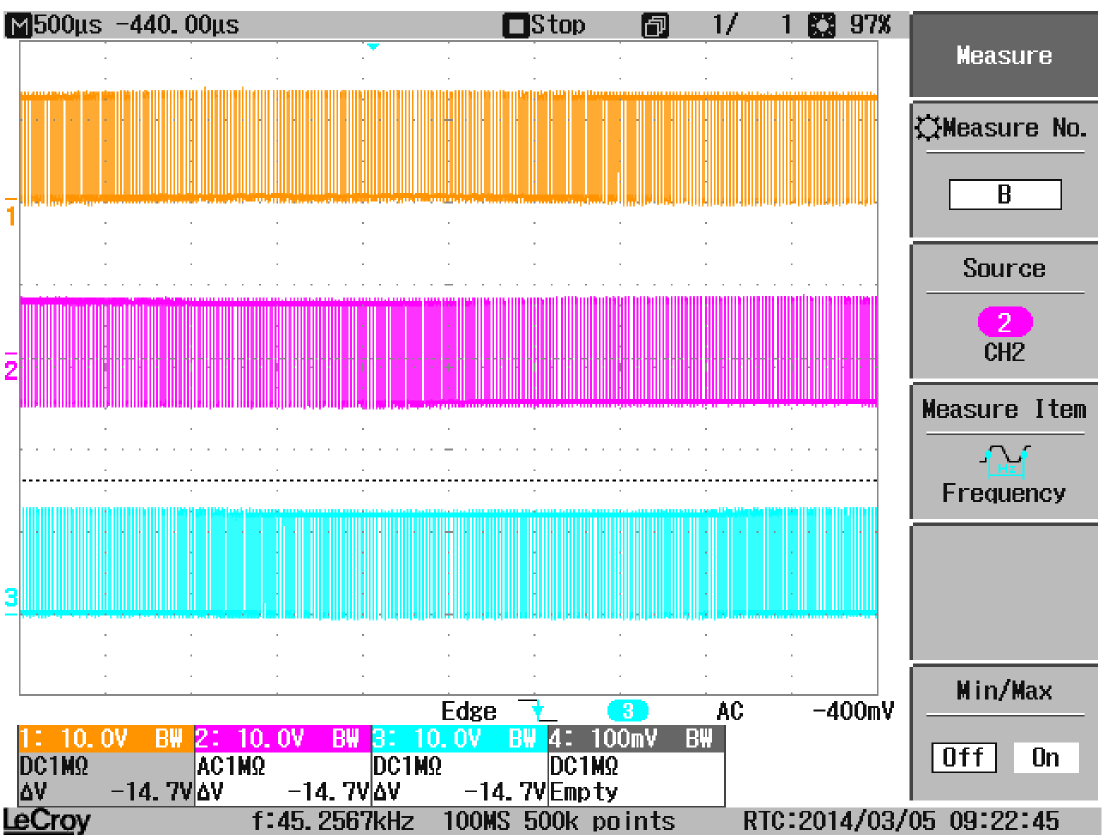


Figure 22 Three motor phase voltage signals - direct to motor windings

5.5 Temperature profile with load (top view) – infrared camera

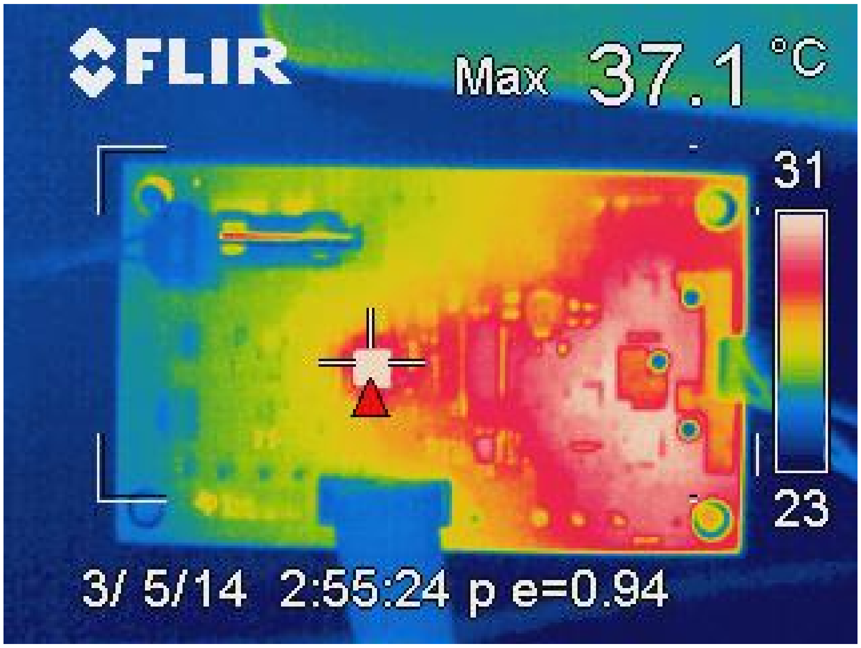


Figure 23 Infrared photo of SAT0042 E4 board in operation (top side)

IMPORTANT NOTICE AND DISCLAIMER

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265