MICROCHIP

AN889

VF Control of 3-Phase Induction Motors Using PIC16F7X7 Microcontrollers

Author: Rakesh Parekh
Microchip Technology Inc.

INTRODUCTION

An induction motor can run only at its rated speed when it is connected directly to the main supply. However, many applications need variable speed operations. This is felt the most in applications where input power is directly proportional to the cube of motor speed. In applications like the induction motor-based centrifugal pump, a speed reduction of 20% results in an energy savings of approximately 50%.

Driving and controlling the induction motor efficiently are prime concerns in today's energy conscious world. With the advancement in the semiconductor fabrication technology, both the size and the price of semiconductors have gone down drastically. This means that the motor user can replace an energy inefficient mechanical motor drive and control system with a *Variable Frequency Drive* (VFD). The VFD not only controls the motor speed, but can improve the motor's dynamic and steady state characteristics as well. In addition, the VFD can reduce the system's average energy consumption.

Although various induction motor control techniques are in practice today, the most popular control technique is by generating variable frequency supply, which has constant voltage to frequency ratio. This technique is popularly known as *VF control*. Generally used for open-loop systems, VF control caters to a large number of applications where the basic need is to vary the motor speed and control the motor efficiently. It is also simple to implement and cost effective.

The PIC16F7X7 series of microcontrollers have three on-chip hardware PWM modules, making them suitable for 3-phase motor control applications. This application note explains how these microcontrollers can be used for 3-phase AC induction motor control.

VF CONTROL

A discussion of induction motor control theory is beyond the scope of this document. We will mention here only the salient points of VF control.

The *base speed* of the induction motor is directly proportional to the supply frequency and the number of poles of the motor. Since the number of poles is fixed by design, the best way to vary the speed of the induction motor is by varying the supply frequency.

The torque developed by the induction motor is directly proportional to the ratio of the applied voltage and the frequency of supply. By varying the voltage and the frequency, but keeping their ratio constant, the torque developed can be kept constant throughout the speed range. This is exactly what VF control tries to achieve.

Figure 1 shows the typical torque-speed characteristics of the induction motor, supplied directly from the main supply. Figure 2 shows the torque-speed characteristics of the induction motor with VF control.

Other than the variation in speed, the torque-speed characteristics of the VF control reveal the following:

- The starting current requirement is lower.
- The stable operating region of the motor is increased. Instead of simply running at its base rated speed (N_B), the motor can be run typically from 5% of the synchronous speed (N_S) up to the base speed. The torque generated by the motor can be kept constant throughout this region.
- At base speed, the voltage and frequency reach the rated values. We can drive the motor beyond the base speed by increasing the frequency further. However, the applied voltage cannot be increased beyond the rated voltage. Therefore, only the frequency can be increased, which results in the reduction of torque. Above the base speed, the factors governing torque become complex.
- The acceleration and deceleration of the motor can be controlled by controlling the change of the supply frequency to the motor with respect to time.

FIGURE 1: TORQUE-SPEED CHARACTERISTICS OF INDUCTION MOTOR

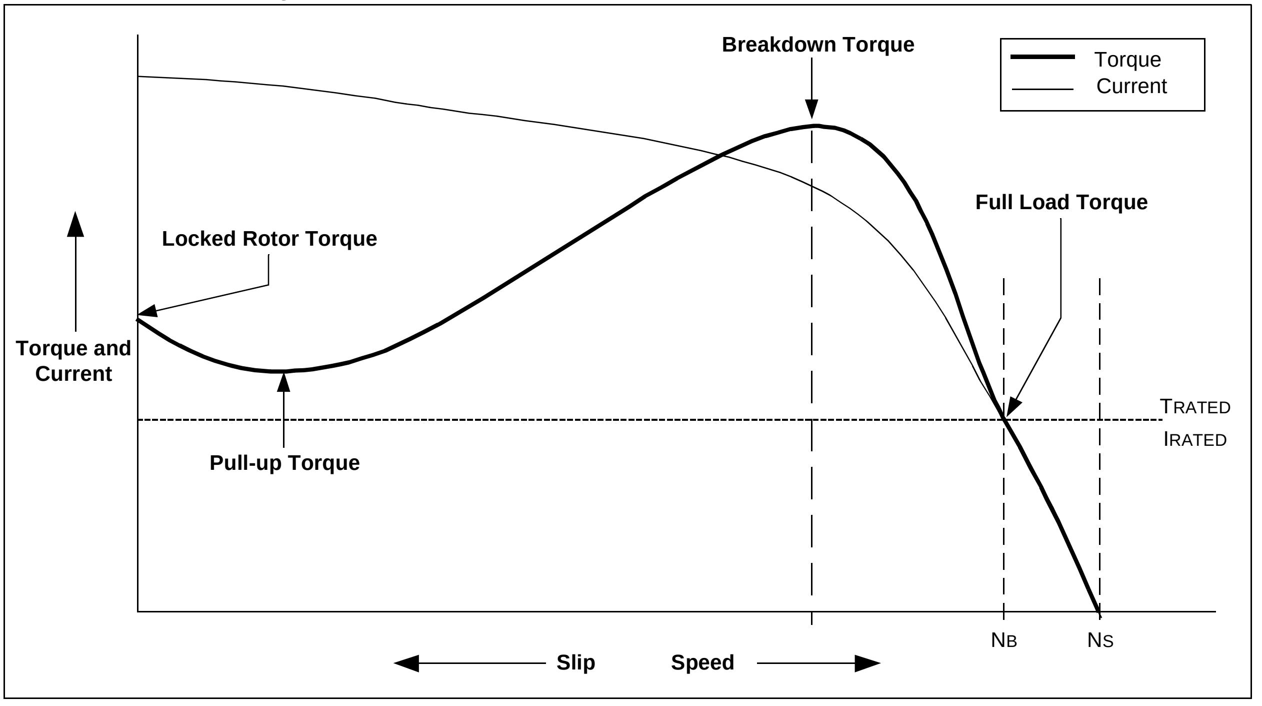
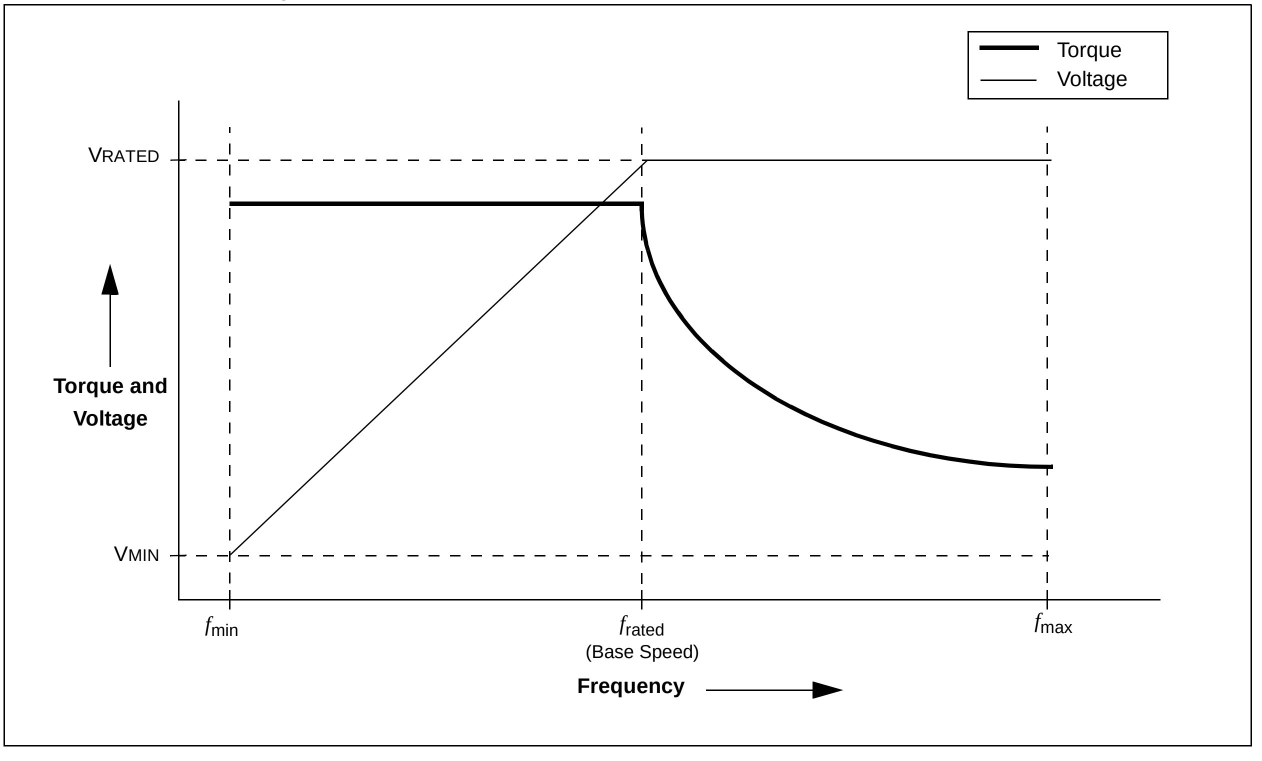


FIGURE 2: TORQUE-SPEED CHARACTERISTICS OF INDUCTION MOTOR WITH VF CONTROL



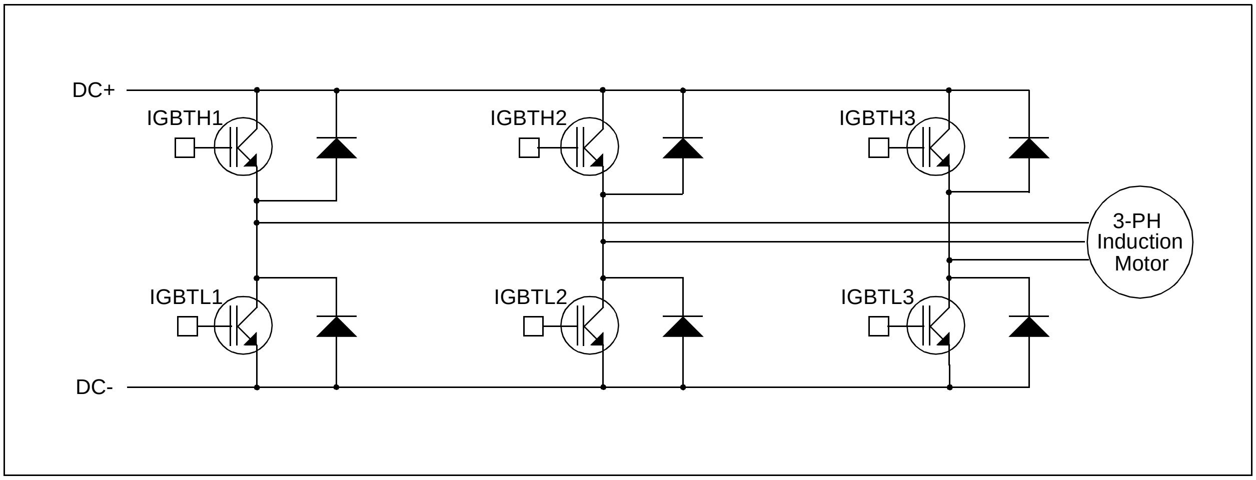
MOTOR DRIVE

The 3-phase induction motor is connected to a 3-phase inverter bridge as shown in Figure 3.The power inverter has 6 switches that are controlled in order to generate 3-phase AC output from the DC bus. PWM signals, generated from the microcontroller, control these 6 switches. Switches IGBTH1 through IGBTH3, which are connected to DC+, are called upper switches. Switches IGBTL1 through IGBTL3, connected to DC-, are called lower switches.

The amplitude of phase voltage is determined by the duty cycle of the PWM signals. While the motor is running, three out of six switches will be on at any given time; either one upper and two lower switches or one lower and two upper switches. The switching produces a rectangular shaped output waveform that is rich in harmonics. The inductive nature of the motor's stator windings filters this supplied current to produce a 3-phase sine wave with negligible harmonics. When switches are turned off, the inductive nature of the windings oppose any sudden change in direction of flow of the current until all of the energy stored in the windings is dissipated. To facilitate this, fast recovery diodes are provided across each switch. These diodes are known as *freewheeling diodes*.

To prevent the DC bus supply from being shorted, the upper and lower switches of the same half bridge should not be switched on at the same time. A dead time is given between switching off one switch and switching on the other. This ensures that both switches are not conductive at the same time as each one changes states.

FIGURE 3: 3-PHASE INVERTER BRIDGE



Control

Members of the PIC16F7X7 family of microcontrollers have three 10-bit PWMs implemented in hardware. The duty cycle of each PWM can be varied individually to generate a 3-phase AC waveform as shown in Figure 4. The upper eight bits of the PWM's duty cycle is set using the register CCPRxL, while the lower two bits are set in bits 4 and 5 of the CCPxCON register. The PWM frequency is set using the Timer2 Period register (PR2). Because all of the PWMs use Timer2 as their time base for setting the switching frequency and duty cycle, all will have the same switching frequency.

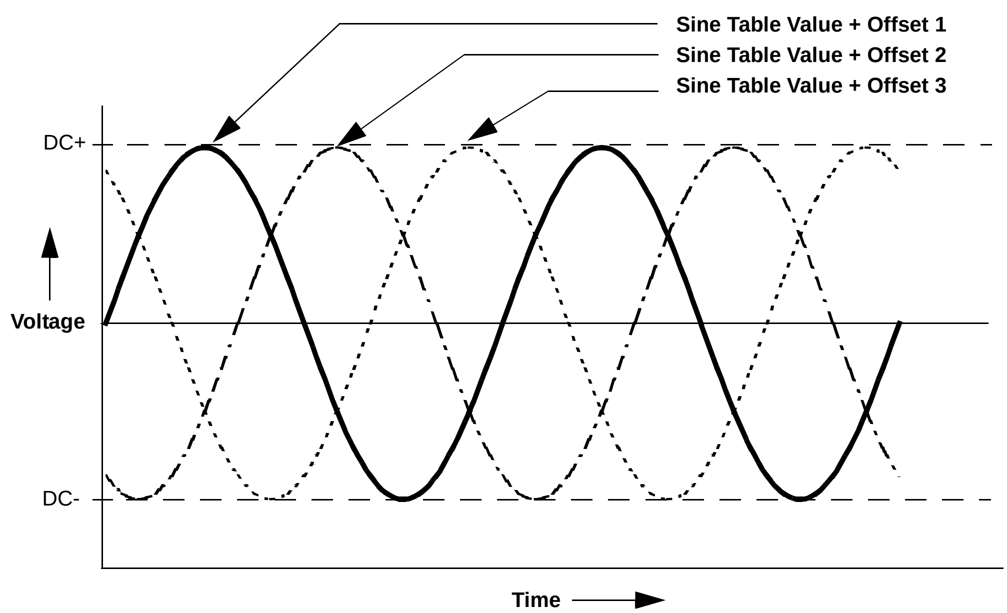
To derive a varying 3-phase AC voltage from the DC bus, the PWM outputs are required to control the six switches of the power inverter. This is done by connecting the PWM outputs to three IGBT drivers (IR2109). Each driver takes one PWM signal as input and produces two PWM outputs, one being complementary to the other. These two signals are used to drive one half bridge of the inverter: one to the upper switch, the other to the lower switch. The driver also adds a fixed dead time between the two PWM signals.

3-Phase Sine Waveform Synthesis

Along with the three PWM modules, the 16-bit Timer1 hardware module of PIC16F7X7 is used to generate the control signals to the 3-phase inverter.

This is done by using a sine table, stored in the program memory with the application code and transferred to the data memory upon initialization. Loading the table this way minimizes access time during the run time of the motor. Three registers are used as the offset to the table. Each of these registers will point to one of the values in the table, such that they will always have a 120-degree phase shift relative to each other (Figure 4). This forms three sine waves with 120 degrees phase shift to each other.

FIGURE 4: SYNTHESIS OF 3-PHASE SINE WAVEFORM FROM A SINE TABLE

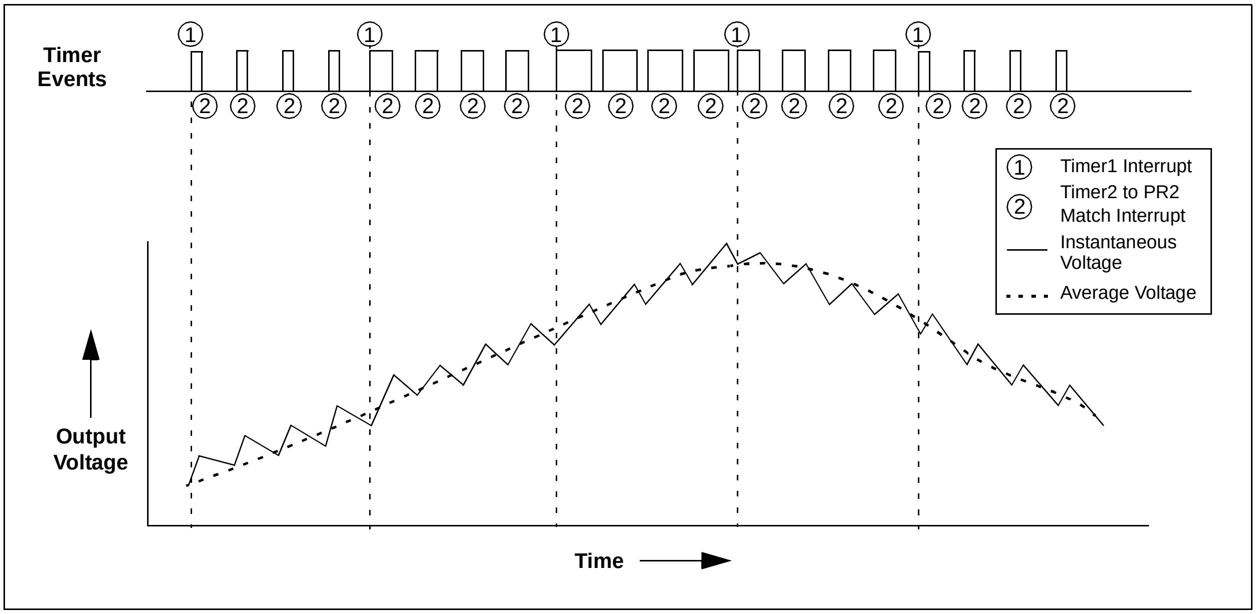


A potentiometer connected to a 10-bit ADC channel (AN1) determines the motor frequency. The microcontroller uses the ADC results to calculate the PWM duty cycle and thus, the frequency and the amplitude of the supply to the motor. For smooth frequency transitions, the channel AN1 is converted at every 4 ms.

The Timer1 reload value is based on the ADC result (AN1), the main clock frequency (FOSC) and the number of sine table entries (36 in the present application). After every Timer1 overflow, the value pointed to by the offset register on the sine table is read. The value read from the sine table is scaled based on the motor frequency input. The sine table value is multiplied with the frequency input to find the PWM duty cycle and is loaded to the corresponding PWM duty cycle register. Subsequently, the offset registers are updated for next access. If the motor direction key is pressed, then PWM1, PWM2 and PWM3 duty cycle values are loaded to PWM2, PWM1 and PWM3 duty cycle registers, respectively.

The new PWM duty cycle values will take effect at the next Timer2 overflow. Also, the duty cycle will remain the same until the next Timer1 overflow occurs, as shown in Figure 5. The frequency of the new PWM duty cycle update determines the motor frequency, while the value loaded in the duty cycle register determines the amplitude of the motor supply.

FIGURE 5: TIMER1 OVERFLOW, PWM DUTY CYCLE AND OUTPUT VOLTAGE



The equation used to calculate the Timer1 reload value is given in Equation 1. In the present application, the Timer1 prescaler is 1:8. PR2 is set to generate a 20 kHz PWM frequency with FOSC of 20 MHz.

The method of accessing and scaling of the PWM duty cycle is shown in an excerpt from the application code in Example 1.

EQUATION 1: TIMER1 RELOAD VALUE CALCULATION

$$\text{Timer1 Reload Value} = \text{FFFFh} - 2 \times \left(\frac{\frac{F_{OSC}}{4}}{\text{Sine Samples per Cycle} \times \text{Timer1 Prescaler} \times \text{Value of AN1}} \right)$$

EXAMPLE 1: SINE TABLE UPDATE

```
;*********************************************************************************************
;This routine will update the PWM duty cycle on CCPx according to the offset to the table with
;0-120-240 degrees.
;This routine scales the PWM value from the table based on the frequency to keep VF
;constant and loads them in appropriate CCPx register depending on setting of FWD/REV flag
;*********************************************************************************************
UPDATE_PWM_DUTYCYCLES
    MOVLW      LOW(SINE_TABLE_RAM)
    MOVWF      FSR                 ;Base address of sine table in RAM is loaded to FSR
    BANKSEL    TABLE_OFFSET1
    MOVF       TABLE_OFFSET1,W     ;Table_offset1 is copied To WREG
    ADDWF      FSR,F               ;Address to be read = sine table base adress + Table_offset1
    BANKISEL   SINE_TABLE_RAM
    MOVF       INDF,W              ;Copy sine table value pointed to by FSR to WREG
    BTFSC      STATUS,Z            ;Check is value read zero?
    GOTO       PWM1_IS_0           ;Yes, goto PWM1_IS_0
    MOVWF      NO_1_LSB            ;No, sine table value x Set_freq to scale table value
                                   ;based on frequency setting
    CALL       MUL_8X8             ;Call routine for unsigned 8x8 bit multiplication
    MOVF       RESULT_MSB,W        ;8 MSB of 16 bit result is stored at TEMP_LOC -
    MOVWF      TEMP_LOC            ;this represent PWM Duty Cycle value for phase 1
    GOTO       UPDATE_PWM2         ;Go for updating PWM Duty Cycle for 2nd phase
PWM1_IS_0
    CLRF       TEMP_LOC            ;Clear PWM Duty Cycle value for phase 1
UPDATE_PWM2
    MOVLW      LOW (SINE_TABLE_RAM)
    MOVWF      FSR                 ;Base address of sine table in RAM is loaded to FSR
    BANKSEL    TABLE_OFFSET2
    MOVF       TABLE_OFFSET2,W     ;Table_offset2 is copied to WREG
    ADDWF      FSR,F               ;Address to be read = Sine table base adress + Table_offset2
    BANKISEL   SINE_TABLE_RAM
    MOVF       INDF,W              ;Copy sine table value pointed to by FSR to WREG
    BTFSC      STATUS,Z            ;Check is value read zero?
    GOTO       PWM2_IS_0           ;Yes, go to PWM2_IS_0
    MOVWF      NO_1_LSB            ;No, sine table value x set_freq to scale table value
                                   ;based on frequency setting
    CALL       MUL_8X8             ;Call routine for unsigned 8x8 bit multiplication
    MOVF       RESULT_MSB,W        ;8 MSB of 16 bit result is stored at TEMP_LOC_1 -
    MOVWF      TEMP_LOC_1          ;this represent PWM Duty Cycle value for phase 2
    GOTO       UPDATE_PWM3         ;Go for updating PWM Duty Cycle for 3rd phase
PWM2_IS_0
    CLRF       TEMP_LOC_1          ;Clear PWM Duty Cycle value for phase 2
UPDATE_PWM3
    MOVLW      LOW(SINE_TABLE_RAM)
    MOVWF      FSR                 ;Base address of sine table in RAM is loaded to FSR
    BANKSEL    TABLE_OFFSET3
    MOVF       TABLE_OFFSET3,W     ;Table_offset3 is copied to WREG
    ADDWF      FSR,F               ;Address to be read=Sine table base address + Table_offset3
    BANKISEL   SINE_TABLE_RAM
    MOVF       INDF,W              ;Copy sine table value pointed by FSR to WREG
    BTFSC      STATUS,Z            ;Check is value read zero?
    GOTO       PWM3_IS_0           ;Yes, goto PWM3_IS_0
    MOVWF      NO_1_LSB            ;No, sine table value x set_freq to scale table value
                                   ;based on frequency setting
    CALL       MUL_8X8             ;Call routine for unsigned 8x8 bit multiplication
    MOVF       RESULT_MSB,W        ;8 MSB of 16 bit result is stored at TEMP_LOC_2 -
    MOVWF      TEMP_LOC_2          ;this represents PWM duty cycle value for phase 3
    GOTO       SET_PWM12           ;Go for checking direction of motor rotation reequired
PWM3_IS_0
    CLRF       TEMP_LOC_2          ;Clear PWM duty cycle value for phase 3
```


EXAMPLE 1: SINE TABLE UPDATE (CONTINUED)

```
SET_PWM12
    BANKSEL     CCPR1L
    BTFSS       FLAGS,MOTOR_DIRECTION  ;Is MOTOR_DIRECTION flag set for forward rotation?
    GOTO        ROTATE_REVERSE         ;No - Go for reverse rotation
    MOVF        TEMP_LOC,W
    MOVWF       CCPR1L                 ;Copy TEMP_LOC and TEMP_LOC_1 values to
    MOVF        TEMP_LOC_1,W           ;CCPR1L and CCPR2L respectively for
    MOVWF       CCPR2L                 ;forward rotation of motor
    BSF         STATUS,RP0
    MOVF        TEMP_LOC_2,W
    MOVWF       CCPR3L                 ;Copy TEMP_LOC_2 to CCPR3L
    BCF         STATUS,RP0
    BSF         LED_PORT,FWD_REV_LED   ;Turn on FWD_REV_LED to indicate
                                       ;forward rotation of motor
    RETURN

ROTATE_REVERSE
    MOVF        TEMP_LOC_1,W           ;Copy TEMP_LOC_1 and TEMP_LOC values to
    MOVWF       CCPR1L                 ;CCPR1L and CCPR2L respectively for
    MOVF        TEMP_LOC,W             ;reverse rotation of motor
    MOVWF       CCPR2L
    BSF         STATUS,RP0
    MOVF        TEMP_LOC_2,W
    MOVWF       CCPR3L                 ;Copy TEMP_LOC_2 to CCPR3L
    BCF         STATUS,RP0
    BCF         LED_PORT,FWD_REV_LED   ;Turn off FWD_REV_LED to indicate
                                       ;reverse rotation of motor
    RETURN
```

OVERVIEW OF SYSTEM HARDWARE

Figure 6 shows the overall block diagram of the power and control circuit for the motor control demo board. The main single phase supply is rectified by using a diode bridge rectifier. The ripple on the DC bus is filtered by using an electrolytic capacitor. The filtered DC bus is connected to the IGBT-based 3-phase inverter, which is controlled by the PIC16F7X7. The inverter output is a 3-phase, variable frequency supply with a constant voltage-to-frequency ratio.

A potentiometer connected to AN1 sets the motor frequency. Push button keys are interfaced for issuing commands, like Run/Stop and Fwd/Rev, to the microcontroller. Acceleration and deceleration features are implemented to change the motor frequency smoothly. Time for both of these features are user selectable and can be set during compile time. LEDs are provided for Status/Fault indications like Run/Stop, Forward/Reverse, Undervoltage, Overvoltage, etc.

The PWM outputs are generated by on-chip hardware modules on the PIC16F7X7. These are used to drive the IGBT drivers through optoisolators. Each IGBT driver, in turn, generates complementary signals for driving the upper or lower halves of the 3-phase inverter. It also adds a dead time of 540 ns between the respective higher and lower switch driving signals.

The IGBT driver has a shutdown signal ($\overline{SD}$) which is controlled by an overcurrent protection circuit. The driver also has its own on-chip Fault monitoring circuit for driver power supply undervoltage conditions. Upon any overcurrent or undervoltage event, the outputs are driven low and remain low until the time the Fault condition is removed.

Note:	Refer to **Appendix B: “Motor Control Schematics”** for schematics of the motor control demo board.

Overcurrent Protection

A non-inductive resistor is connected between the common source point of the inverter and the power ground. Voltage drop across this resistor is linearly proportional to the current flowing through the motor. This voltage drop is compared against the reference voltage signal, through an optoisolator (linear optocoupler), which represents overcurrent limit. There are three possible ways to compare these voltage signals:

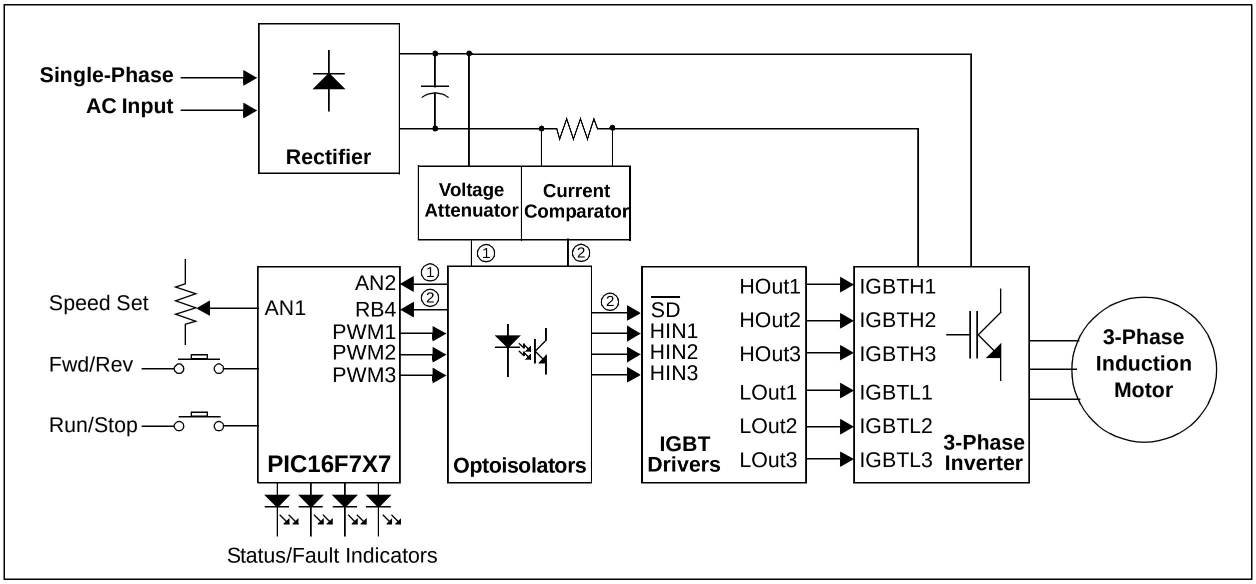
- Using an external comparator
- Using the PIC16F7X7 on-chip comparator
- In software, by reading the voltage drop across the resistor through one of the ADC channels

The design discussed in this application note implements an external comparator. It’s output drives the shutdown signal of the driver through an optoisolator (optocoupler). At the same time, this signal is provided to RB4. By using the PORTB interrupt-on-change feature, the microcontroller responds to Fault detection and stops the motor.

Overvoltage and Undervoltage Protection

To implement voltage protection, the DC bus voltage is attenuated by a potential divider. The resulting signal is fed to AN2 through an optoisolator (linear optocoupler). The application monitors the voltage via periodic A/D conversions of the value on RA2; if the voltage falls outside of a preset range, the motor is stopped.

FIGURE 6: BLOCK DIAGRAM OF THE MOTOR CONTROL DEMO BOARD



Isolation

The use of optoisolators ensures that power ground (P_GND) and control ground (D_GND) are separated. This means that development tools, such as MPLAB® ICD 2 and MPLAB® ICE can be safely connected to the system while it is connected to the AC supply. This simplifies the task of debugging a live system.

The isolation components are often removed when a design goes for production. To remove isolation:

- Remove the PWM drive optoisolators (U6 through U9).
- Remove the power isolation optoisolators (U17 and U18).
- Disconnect the voltage followers for U17 and U18 (U13B, U13C, U16A and U16B). DO NOT physically remove U13 and U16, since U13A and U16C are still used by the system.
- Remove all other components associated with the power isolation system (capacitors C41/42/43 and resistors R81/82/83/84/91/92/93/96).
- Make all grounds common by shorting P_GND to D_GND.

VF CONTROL FIRMWARE

While the PIC16F7X7 microcontroller makes 3-phase motor control possible, it is the firmware that makes VF control straightforward. In addition to maintaining the sine table and driving the PWM modules to produce the AC output (previously described in the **"3-Phase Sine Waveform Synthesis"** section), the firmware interprets control inputs and system status to sense and act on Fault conditions. It also manages other features of motor control, such as direction, acceleration and deceleration (as described below).

The VF control firmware uses a set of defined routines and parameters for operation. Users can change these parameters as needed for their applications. The firmware can also be incorporated as the motor control core of a larger application, using the parameters to pass information between sections of the code. An overview of the firmware's logic flow is provided in Figure 7 and Figure 8. A complete list of parameters and defined functions is provided in Tables 1 through 4.

Users are encouraged to download the complete source code of the firmware from the Microchip web site (www.microchip.com) and examine the application in more detail.

Acceleration and Deceleration

Acceleration and deceleration time can be specified during compile time. The actual motor frequency (`SET_FREQ`) and the required user frequency (`NEW_FREQ`), set through the potentiometer, is compared at 4 ms intervals. If the `SET_FREQ` and the `NEW_FREQ` are different, then the `SET_FREQ` is changed step by step (each step size is 0.25 Hz) until it reaches the `NEW_FREQ`.

The time to change the `SET_FREQ` by one step is calculated in software, depending upon the difference between the `SET_FREQ` and the `NEW_FREQ`, as well as the acceleration and deceleration parameters entered during compile time. If the `NEW_FREQ` is changed during the acceleration and deceleration process, then the time to change each step is recalculated.

FIGURE 7: MOTOR CONTROL FLOW CHART (MAIN AND ADC ROUTINES)

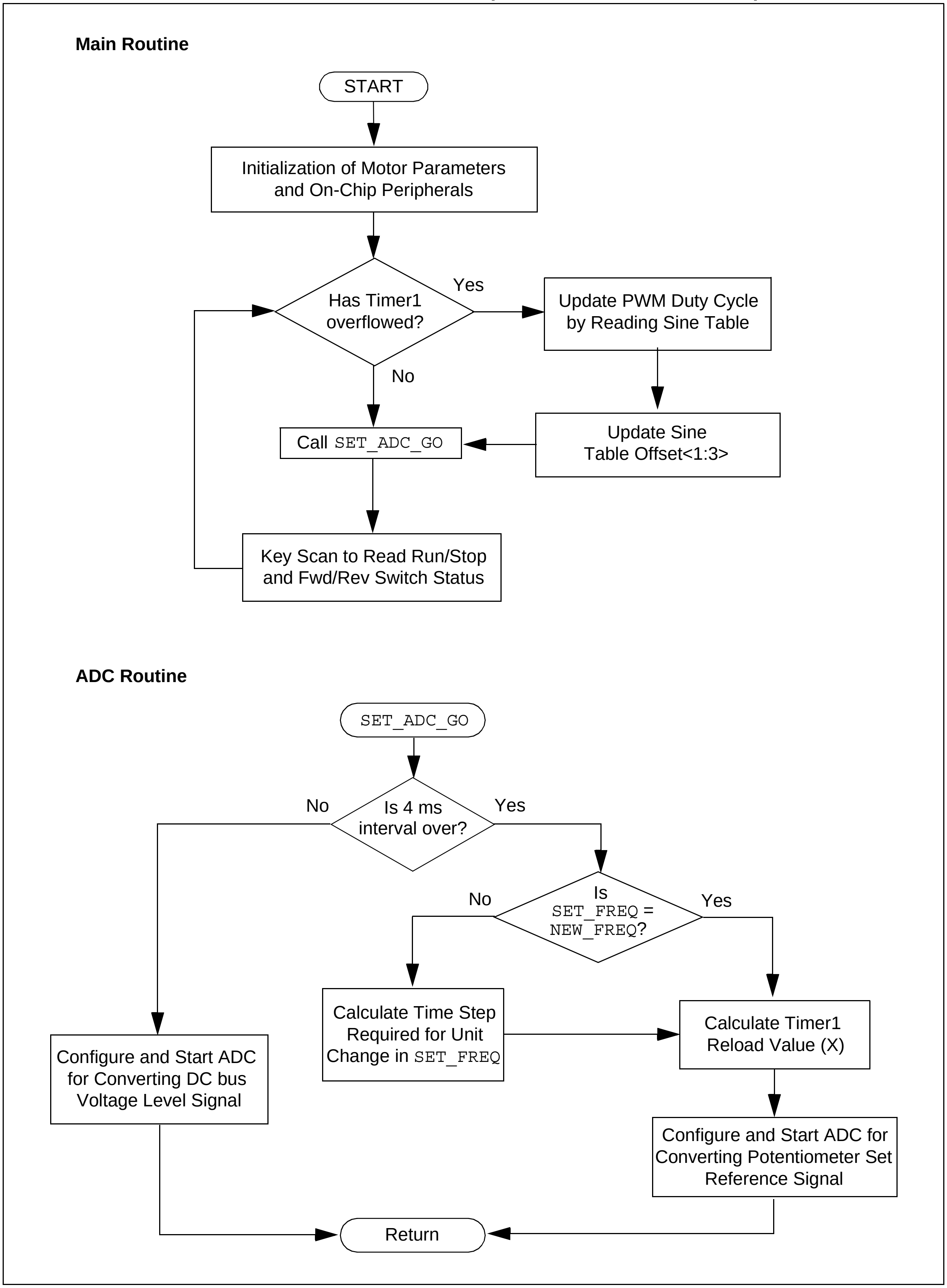


FIGURE 8: MOTOR CONTROL FLOW CHART (INTERRUPT SERVICE ROUTINE)

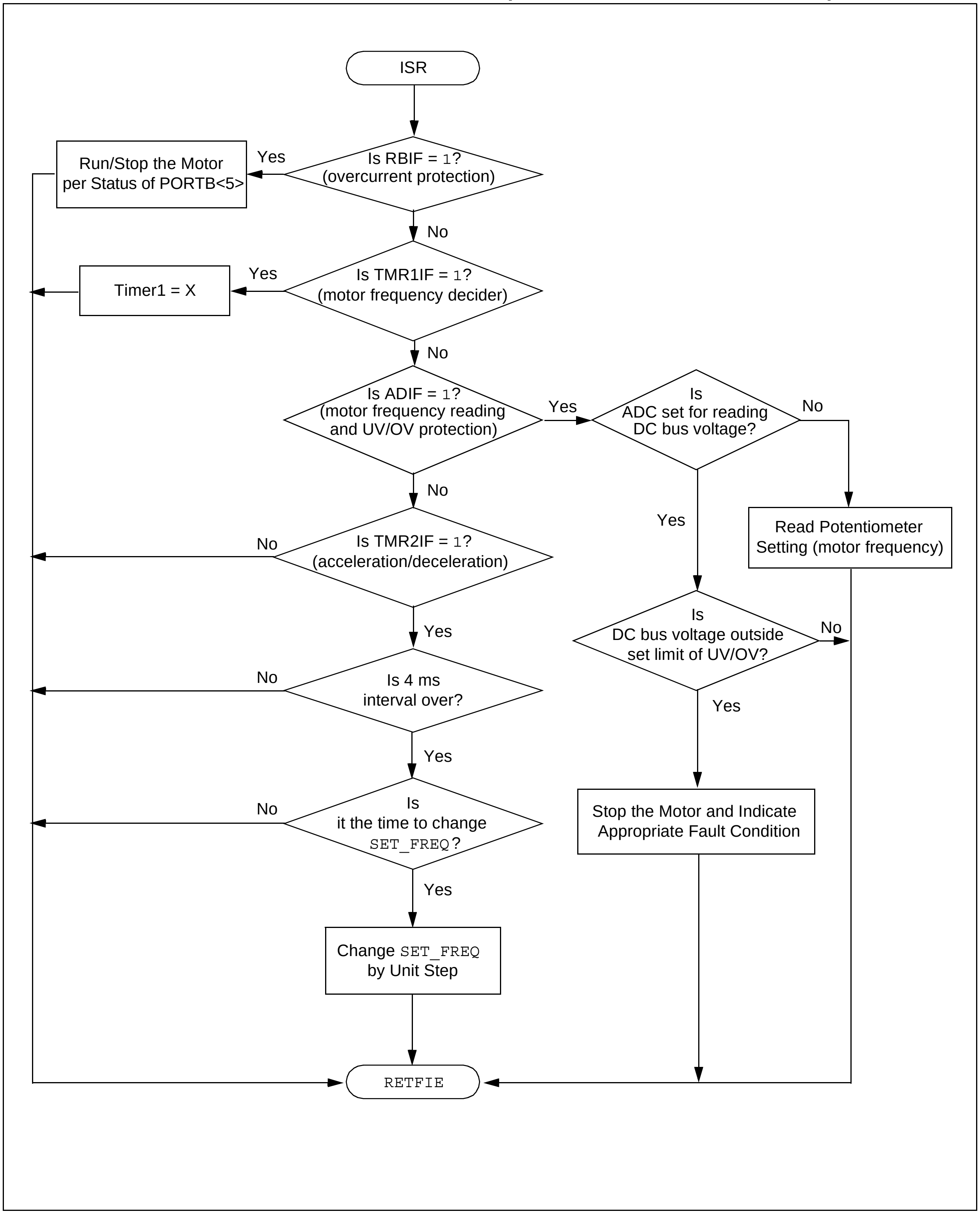


TABLE 1: USER DEFINED PARAMETERS IN SOFTWARE

Name	Description
OSC_FREQ	Defines the oscillator frequency. In the present application, this is set to 20 MHz.
TIMER1_PRESCALE	Defines Timer1 prescaler value. In the present application, it is set to 1:8.
TIMER2_PRESCALE	Defines the Timer2 prescaler value. In the present application, this is set to 1:1.
PWM_FREQUENCY	Defines the PWM switching frequency. In the present application, this is set to 20 kHz.
ACCELERATION_TIME	Defines the user set acceleration time for the motor speed. In the present application, this is set to 3 seconds.
DECELERATION_TIME	Defines the user set deceleration time for the motor speed. In the present application, this is set to 3 seconds.
SINE_TABLE_ENTRIES	Defines the length of the sine table. In the present application, this is set to 19.

TABLE 2: CONSTANTS IN SOFTWARE

Name	Description
FREQ_SCALE	Used to calculate Timer1 reload value. It’s value depends on FOSC, Timer1 prescaler and the number of sine table entries
PR2_VALUE	Defines the Timer2 overflow time period and thus, the PWM switching frequency. It’s value depends on FOSC, Timer2 prescaler and required PWM switching frequency.
DEC_CON	Used for calculating time required for unit step decrement in SET_FREQ. It’s value is: Deceleration Time x 250.
ACC_CON	Used for calculating time required for unit step increment in SET_FREQ. It’s value is: Acceleration Time x 250.
LIMIT_V_LOW	Defines the DC bus voltage limit for undervoltage protection to activate.
LIMIT_V_HIGH	Defines the DC bus voltage limit for overvoltage protection to activate.

TABLE 3: VARIABLES IN SOFTWARE

Name	Description
SET_FREQ	Actual motor frequency.
NEW_FREQ	Required motor frequency (set through the potentiometer).
TABLE_OFFSET1	Pointer to sine table for phase 1.
TABLE_OFFSET2	Pointer to sine table for phase 2.
TABLE_OFFSET3	Pointer to sine table for phase 3.

TABLE 4: FUNCTIONS IN SOFTWARE

Name	Description
UPDATE_PWM_DUTYCYCLES	Loads new duty cycle values to CCPRxL for generating the 3-phase sine wave. This routine also scales the sine table value depending on SET_FREQ.
UPDATE_TABLE_OFFSET	Changes the pointers to the sine table after every access to maintain 120-degree phase shift between generated sine waves.
SET_ADC_GO	Configures the ADC for reading DC bus voltage or the potentiometer setting for the required motor frequency. This routine also calculates time needed for unit change in SET_FREQ for acceleration and deceleration.
KEY_CHECK	Checks the status of Run/Stop and Fwd/Rev keys and acts accordingly.
CHECK_FAULT	Responds to setting of RBIF. This routine responds to the output status of the external current comparator.
TIMER1_OVERFLOW	Responds to setting of TMR1IF. This routine sets the user-defined flag indicating Timer1 overflow, which in turn is responsible for calling of UPDATE_PWM_DUTYCYCLES in the main routine.
AD_CONV_COMPLETE	Responds to setting of ADIF. This routine reads the frequency setting from the potentiometer through ADC. If the frequency is set below 5 Hz or above 60 Hz, it limits the frequency to 5 Hz or 60 Hz. This routine also reads the DC bus voltage level through ADC. If the level is outside the preset limits, it activates undervoltage or overvoltage protection as required.
TMR2_ISR	Responds to setting of TMR2IF. This routine is used for implementing acceleration and deceleration feature.

RESOURCE USAGE

The VF control application consumes both memory and hardware resources, as shown in Table 5. Substantial resources, particularly memory, are still available to users for development of their own applications.

TABLE 5: RESOURCES USED IN THE MOTOR CONTROL DEMO BOARD (USING PIC16F777)

Resource Type	Used	Available to User when PIC16F777 is Used
Program Memory	1041 words	7215 words
Data Memory	49 bytes	319 bytes
CCP Channels	3	0
ADC Channels	2	12
Timers	2	1
External Interrupts	1	4
I/O Lines	15	21

CONCLUSION

VF control provides a simple and cost efficient method for open-loop speed control of 3-phase induction motors. A low-cost VF solution can be implemented using the PIC16F7X7 family of devices. With three dedicated PWM modules implemented in hardware, it is ideal for controlling 3-phase induction motors. Additional on-chip resources, like multiple timers and ADC, allow users to easily implement safety and control features, such as current and voltage protection and configurable acceleration and deceleration time.

APPENDIX A: TEST RESULTS

TABLE A-1: TEST RESULTS

Test #	Set Frequency (Hz)	Set Speed (RPM)	Actual Speed (RPM)	Speed Regulation (%)
1	7.00	420	413	1.67
2	10.50	630	624	0.95
3	12.75	765	763	0.26
4	15.50	930	923	0.75
5	18.75	1125	1124	0.09
6	24.00	1440	1429	0.76
7	29.50	1770	1767	0.17
8	31.50	1890	1900	-0.53
9	36.25	2175	2184	-0.41
10	40.00	2400	2402	-0.08
11	44.50	2670	2670	0.00
12	46.50	2790	2805	-0.54
13	50.00	3000	3017	-0.57
14	54.50	3270	3275	-0.15
15	60	3600	3560	1.11

Note: Above tests are conducted on a motor with the following specifications: Terminal Voltage = 208-230V, Frequency = 60 Hz, Horsepower = 1/2 HP, Speed = 3450 RPM at full load, Rated Current = 1.8A, Test Condition = no load.

APPENDIX B: MOTOR CONTROL SCHEMATICS

FIGURE B-1: POWER SUPPLY

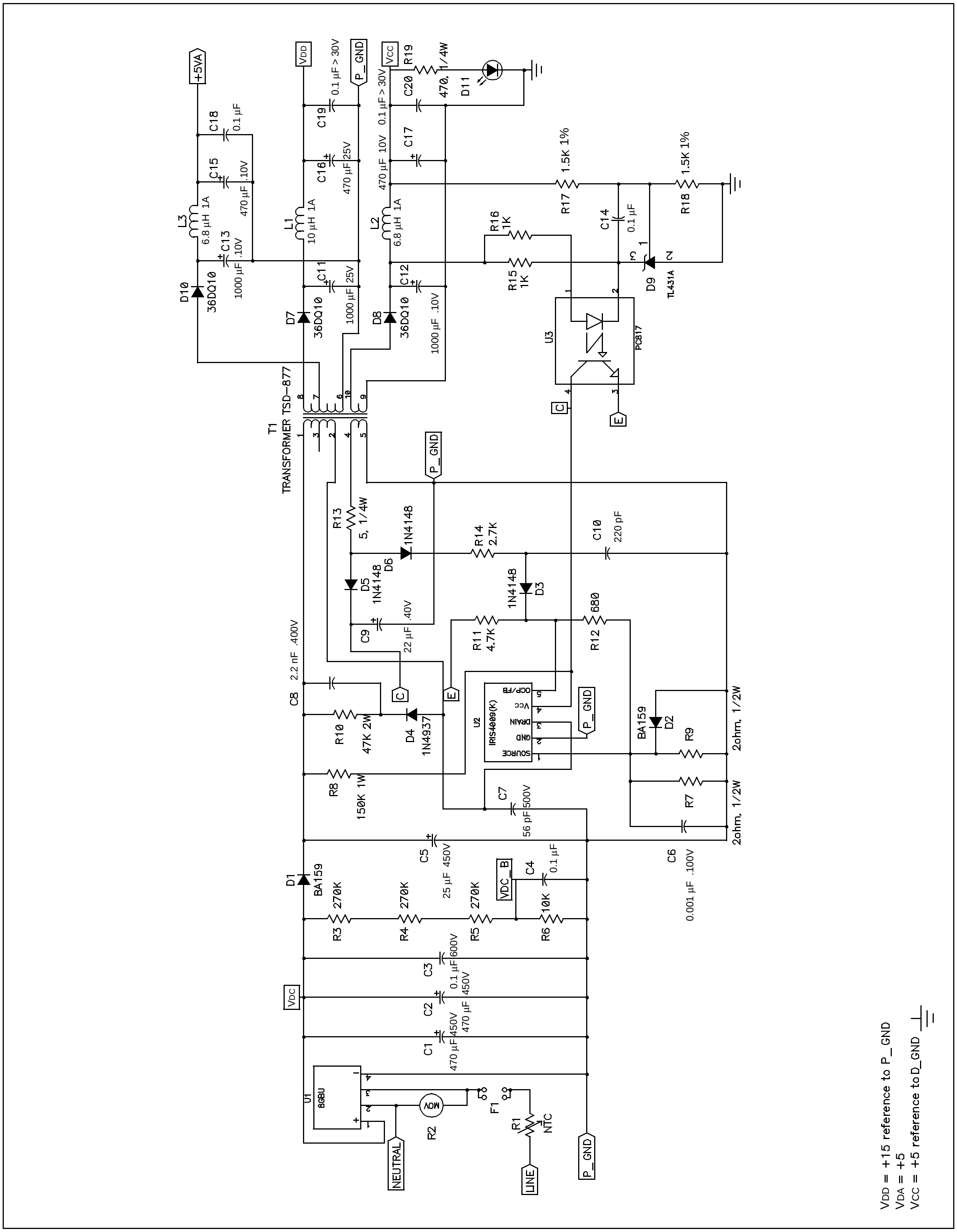


FIGURE B-2: SYSTEM CONTROL

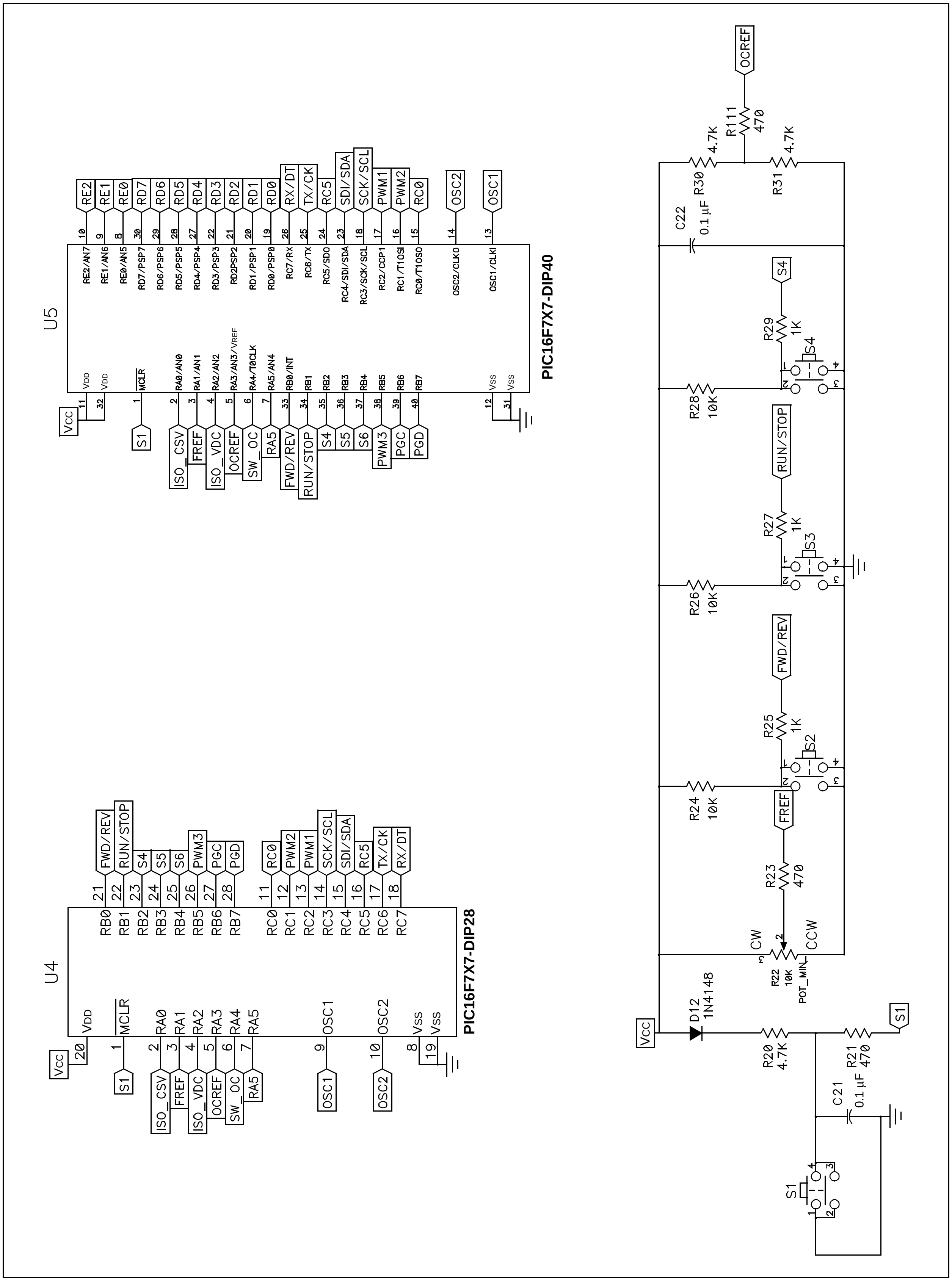


FIGURE B-3: INVERTER AND FEEDBACK

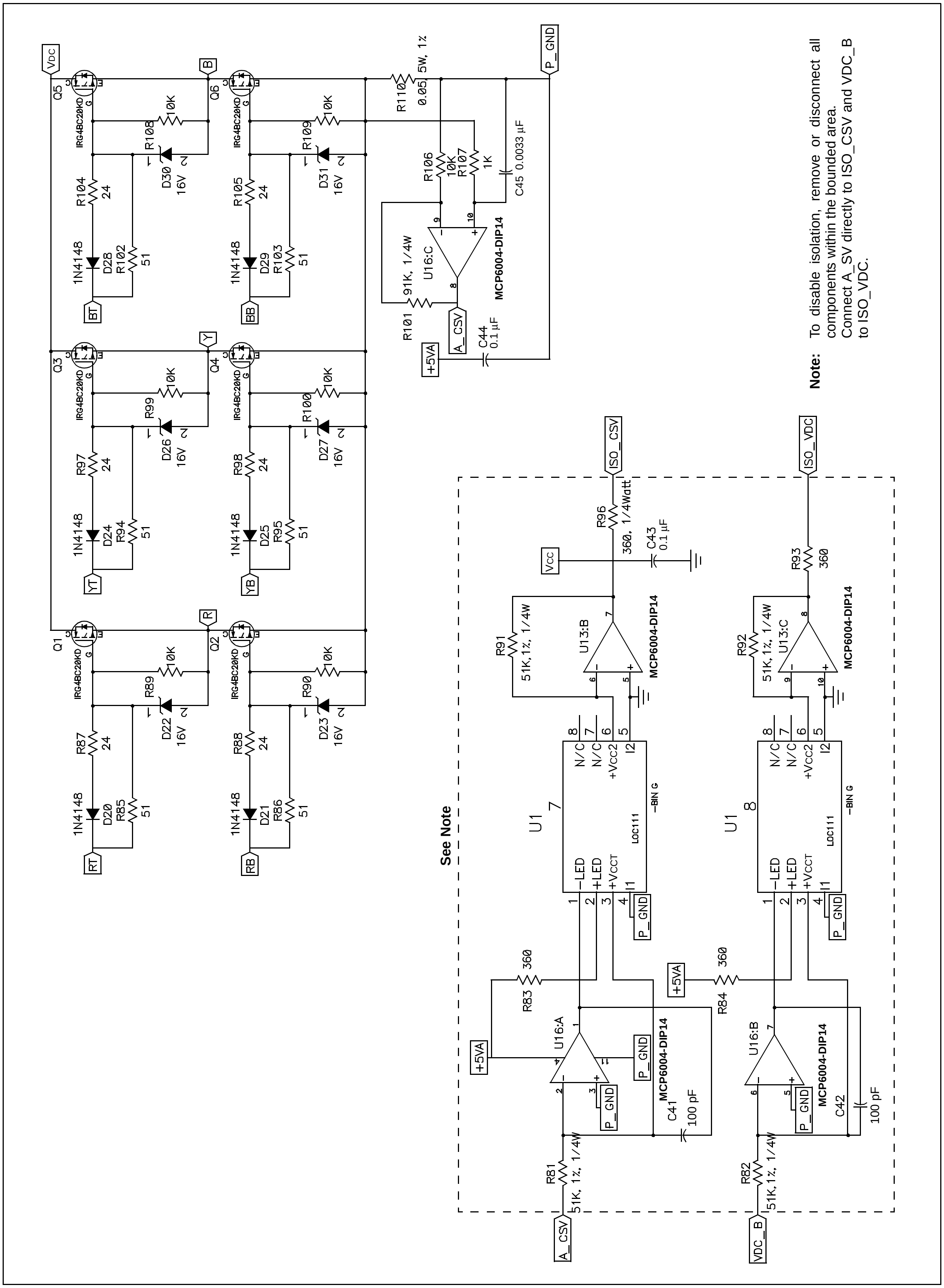


FIGURE B-4: INVERTER DRIVERS AND OPTOISOLATORS

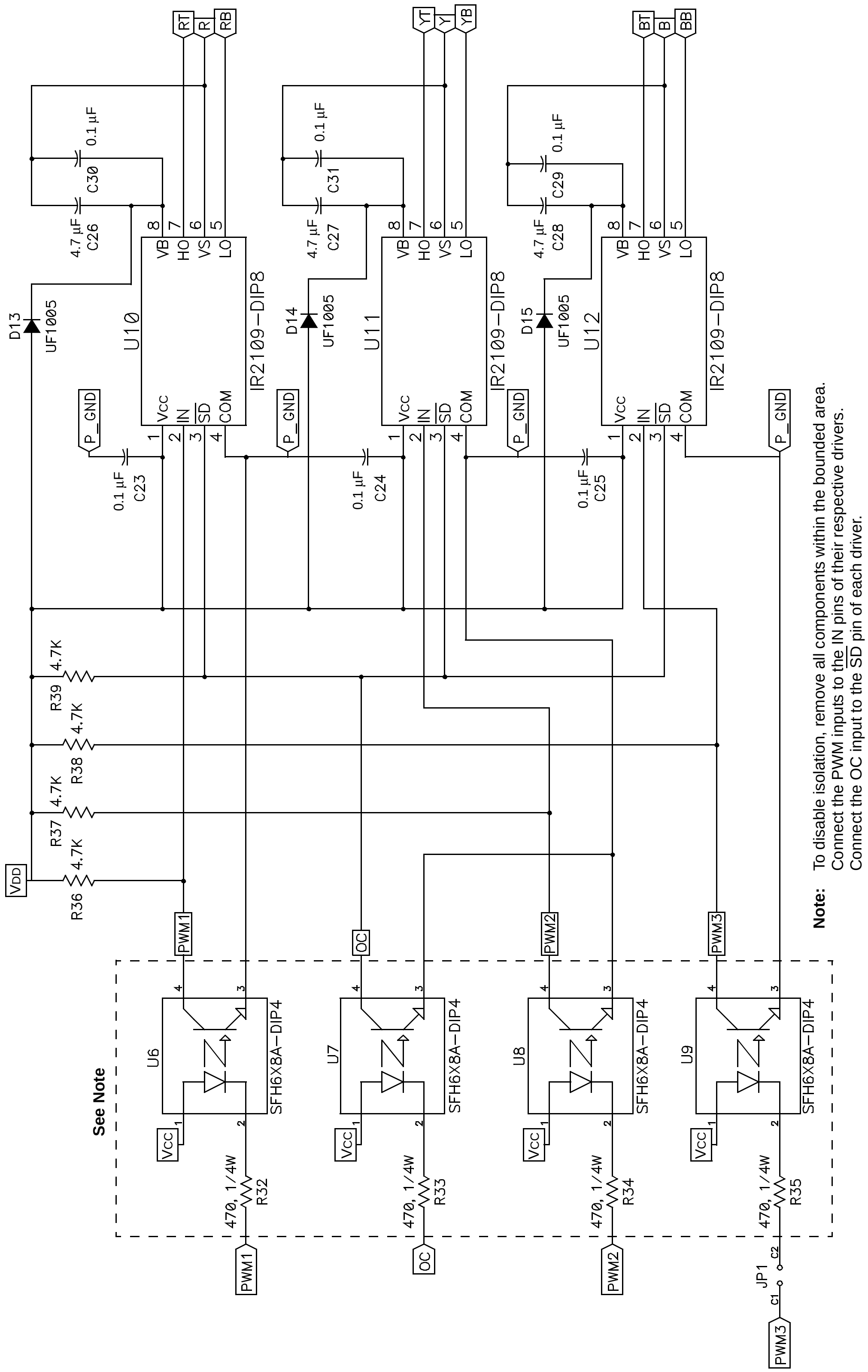
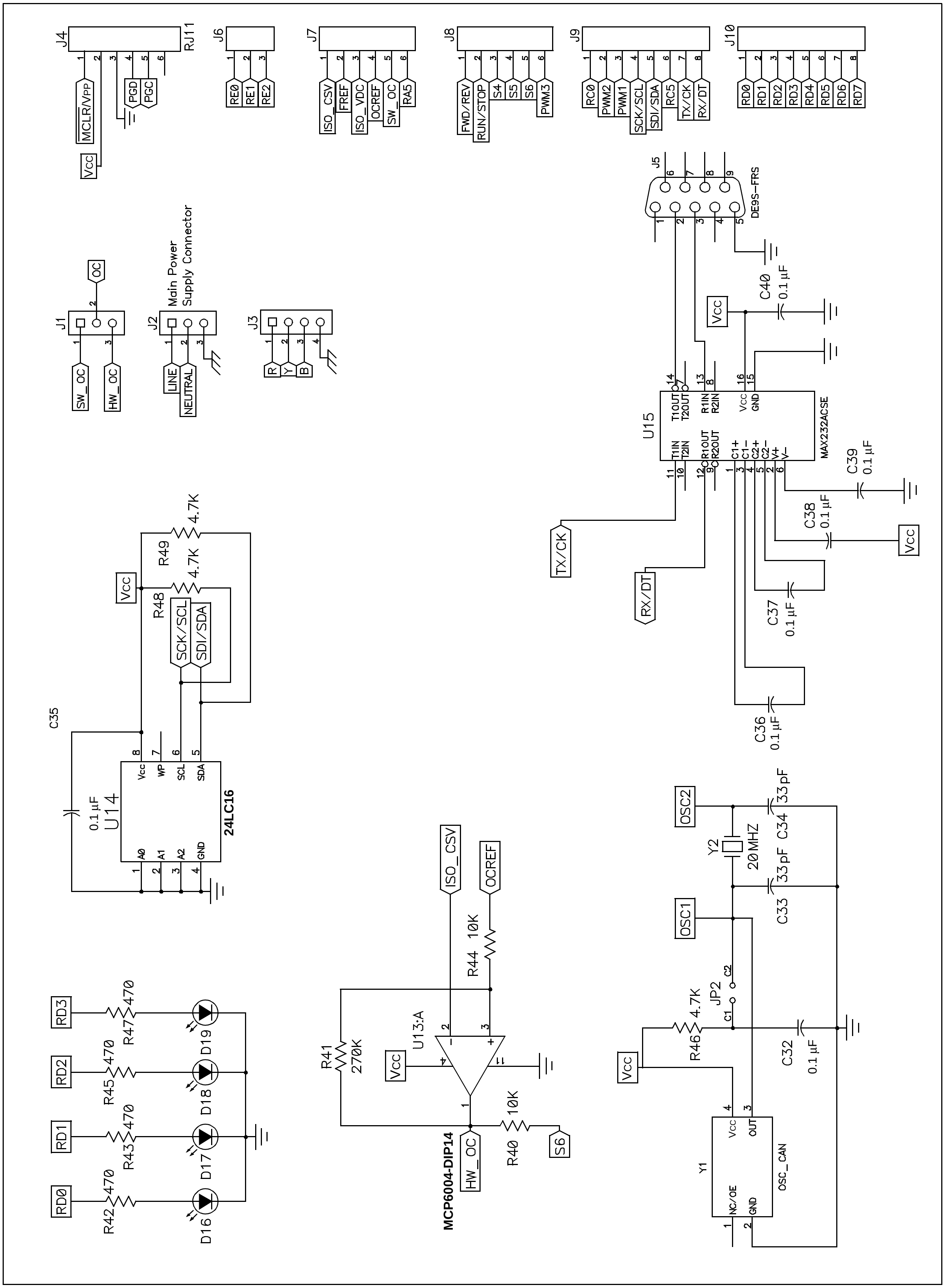


FIGURE B-5: DISPLAY AND COMMUNICATION SECTION



APPENDIX C: SOFTWARE DISCUSSED IN THIS APPLICATION NOTE

Because of its overall length, a complete source file listing of the PIC16F7X7 VF motor control firmware is not provided here. The complete source code is available as a single WinZip archive file, which may be downloaded from the Microchip corporate web site at:

www.microchip.com

NOTES:

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.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona and Mountain View, California in October 2003. The Company's quality system processes and procedures are for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

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: 480-792-7627
Web Address: http://www.microchip.com

Atlanta
3780 Mansell Road, Suite 130
Alpharetta, GA 30022
Tel: 770-640-0034
Fax: 770-640-0307

Boston
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848
Fax: 978-692-3821

Chicago
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
2767 S. Albright Road
Kokomo, IN 46902
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888
Fax: 949-263-1338

San Jose
1300 Terra Bella Avenue
Mountain View, CA 94043
Tel: 650-215-1444
Fax: 650-961-0286

Toronto
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Australia
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Unit 706B
Wan Tai Bei Hai Bldg.
No. 6 Chaoyangmen Bei Str.
Beijing, 100027, China
Tel: 86-10-85282100
Fax: 86-10-85282104

China - Chengdu
Rm. 2401-2402, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-86766200
Fax: 86-28-86766599

China - Fuzhou
Unit 28F, World Trade Plaza
No. 71 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7503506
Fax: 86-591-7503521

China - Hong Kong SAR
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Shanghai
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700
Fax: 86-21-6275-5060

China - Shenzhen
Rm. 1812, 18/F, Building A, United Plaza
No. 5022 Binhe Road, Futian District
Shenzhen 518033, China
Tel: 86-755-82901380
Fax: 86-755-8295-1393

China - Shunde
Room 401, Hongjian Building, No. 2
Fengxiangnan Road, Ronggui Town, Shunde District, Foshan City, Guangdong 528303, China
Tel: 86-757-28395507 Fax: 86-757-28395571

China - Qingdao
Rm. B505A, Fullhope Plaza,
No. 12 Hong Kong Central Rd.
Qingdao 266071, China
Tel: 86-532-5027355 Fax: 86-532-5027205

India
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-22290061 Fax: 91-80-22290062

Japan
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa, 222-0033, Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5932 or 82-2-558-5934

Singapore
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-6334-8870 Fax: 65-6334-8850

Taiwan
Kaohsiung Branch
30F - 1 No. 8
Min Chuan 2nd Road
Kaohsiung 806, Taiwan
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan
Taiwan Branch
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

Austria
Durisolstrasse 2
A-4600 Wels
Austria
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

Denmark
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45-4420-9895 Fax: 45-4420-9910

France
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany
Steinheilstrasse 10
D-85737 Ismaning, Germany
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy
Via Quasimodo, 12
20025 Legnano (MI)
Milan, Italy
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands
P. A. De Biesbosch 14
NL-5152 SC Drunen, Netherlands
Tel: 31-416-690399
Fax: 31-416-690340

United Kingdom
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44-118-921-5869
Fax: 44-118-921-5820

02/17/04