
Controlling 3-Phase AC Induction Motors Using the PIC18F4431

<i>Author: Jon Burroughs Microchip Technology Inc.</i>
--

INTRODUCTION

Previous Microchip authors (Parekh and Yedamale) have described the implementation of a 3-phase AC induction motor (ACIM) control with PICmicro® devices. The first application note (AN843) detailed ACIM control with the PIC18F452. More recently, ACIM control has been implemented with the PIC16F7X7 family of devices (AN889).

This application note describes how the PIC18F4431 may be used to control an ACIM using open and closed-loop V/f control strategies. The application code is built incrementally and demonstrates the following control methods:

1. Voltage-frequency (V/f) control
2. Voltage-frequency control with current feedback
3. Voltage-frequency control with velocity feedback and PID control

The PIC18F4431 incorporates a set of innovative peripherals, designed especially for motor control applications. The utility of these peripherals is demonstrated in both open and closed-loop three-phase ACIM motor applications.

It is assumed that the reader is already familiar with the theory and nomenclature of AC induction motors. For an excellent introduction to the basic concepts of induction motors control, please refer to Microchip's application note AN887, "AC Induction Motor Fundamentals" (DS00887).

USING THE PIC18F4431 FOR MOTOR CONTROL

Before getting into actual control applications, we should understand what distinguishes the PIC18F4431 from other Microchip devices used for motor control. The core is a set of unique peripherals that simplify external hardware requirements and also enable higher levels of motor control capability than the PIC18F452 or PIC16F7X7.

The three peripherals and their features are:

1. Power Control PWM (PCPWM) module:
 - Up to 8 output channels
 - Complimentary PWM outputs
 - Two hardware Fault protection inputs
 - PWM resolution up to 14 bits
 - Edge-aligned or center-aligned operation
 - Flexible dead time
 - Simultaneous update of duty cycle and period
2. Motion Feedback Module (MFM), comprised of a Quadrature Encoder Interface (QEI) and an Input Capture module (IC):
 - Standard quadrature encoder inputs (QEA, QEB and Index) (QEI)
 - High and Low Resolution Position Measurement modes (QEI)
 - Velocity Measurement mode using Timer5 (QEI)
 - Interrupt with configurable priority on event detection (QEI)
 - Pulse Width Measurement and Period Measurement modes (IC)
 - Edge and state change capture (IC)
3. High-Speed Analog-to-Digital Converter (HSADC):
 - Two independent sample-and-hold circuits
 - Single or multi-channel selection
 - Sequential or Simultaneous Conversion modes
 - Four-word FIFO result buffer with flexible interrupts

As we shall see, each of these features provides a distinct advantage in implementing more sophisticated motor control applications.

SYSTEM OVERVIEW

Motor Drive Requirements

Practically speaking, control of a 3-phase AC induction motor requires pulse-width modulated control of the six switches of a 3-phase inverter bridge connected to the motor's stator windings (Figure 1). The six switches form 3 pairs of "half-bridges", which can be used to connect the leg of a winding to the positive or the negative high-voltage DC bus.

As shown in the figure, two switches on the same "half-bridge" must never be on simultaneously, otherwise the positive and negative buses will be shorted together. This condition would result in a destructive event known as "shoot-through". If one switch is on, then the other must be off; thus, they are driven as complementary pairs. It should also be noted that the switching devices used in the half-bridge (in this case, IGBTs) often require more time to turn off than to turn on. For this reason, a minimum dead time must be inserted between the off and on time of complimentary channels.

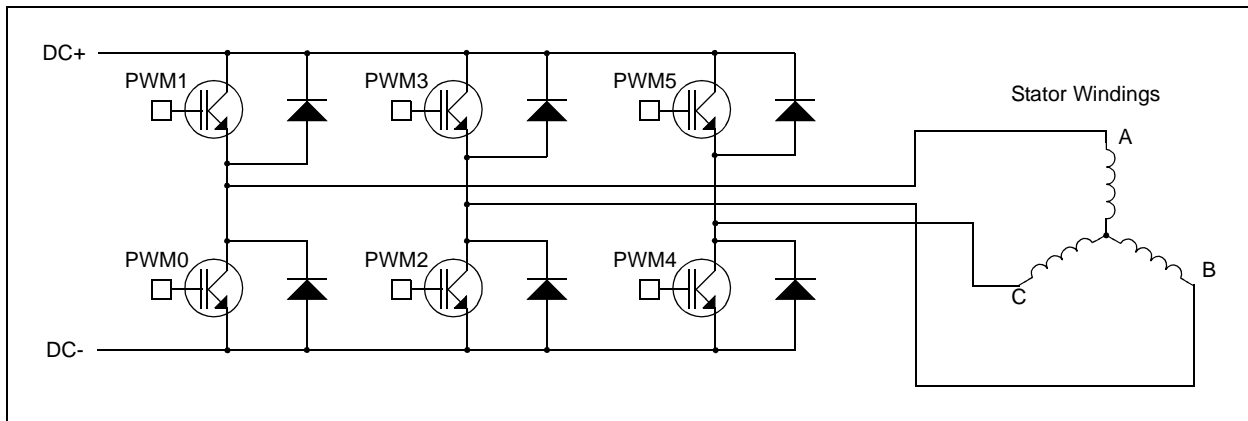
In AN843 and AN889, three PWM outputs were used to drive a 3-phase inverter bridge. In these cases, however, it was necessary to use external circuitry to generate the complimentary control signal for the lower leg and insert the proper dead time between them. For 3-phase control, what is ultimately needed to drive a 3-phase bridge is three *pairs* of complimentary PWM outputs, with dead time between the complimentary channels.

Hardware Overview

The motor control applications described in this document were developed and tested on a production model of Microchip's PICDEM™ MC Development Board. While the application software has been designed with this platform's control and communication requirements in mind, the control methods discussed are applicable to any 3-phase ACIM control application based on the PIC18F4431.

For a complete description of the board and its capabilities as a development tool, please refer to the User's Guide for the PICDEM MC Development Board (DS51453A). To give the reader a more clear idea of the hardware platform, a brief overview and schematics of the board are provided in **Appendix A: "PICDEM™ MC Board Overview"**.

FIGURE 1: A 3-PHASE INVERTER BRIDGE DRIVEN WITH 6 PWM INPUTS



OVERVIEW OF CONTROL STRATEGIES

Open Loop V/f Control

AC induction motors are often operated in open loop with no velocity or position feedback. The V/f ratio is maintained constant to provide a constant (maximum) torque over the operating range. This form of control is relatively inexpensive and easy to implement. Feedback from the rotor is not utilized and the rotor is assumed to follow the rotating flux generated in the stator, with a certain amount of slip present depending upon the load.

To drive an AC induction motor, the 3-phase inverter bridge is driven by a microcontroller's PWM outputs, as shown in Figure 2. By changing the PWM duty cycles in a regular manner, the PWM outputs are modulated to synthesize sinusoidal waveforms (three-phase AC) across the three motor windings (Figure 3).

AC is applied to the three stator windings as three sinusoidal currents, equal in amplitude and frequency, but offset from each other by 120 degrees. As a result, the current in the stator windings generates a rotating magnetic field. This rotating field induces electromotive force in the rotor, which in turn produces a magnetic field in the rotor that attempts to align with the rotating magnetic field in the stator. This causes the rotor to rotate. See AN887, "AC Induction Motor Fundamentals" (DS00887), for more details on induction motor construction and operating characteristics.

The operation of an ACIM is governed by two principles:

1. Base speed is directly proportional to the frequency of the alternating current applied to the stator and the number of poles of the motor.
2. Torque is directly proportional to the ratio of applied voltage and the frequency of the applied AC current.

Therefore, speed can be controlled by varying the input frequency of the applied alternating current and torque can be maintained constant by varying the amplitude in direct proportion to the frequency. These are the two basic aims of open-loop V/f control.

FIGURE 2: OPEN-LOOP V/f CONTROL BLOCK DIAGRAM

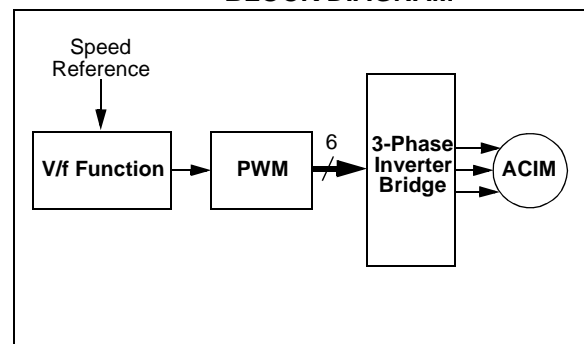
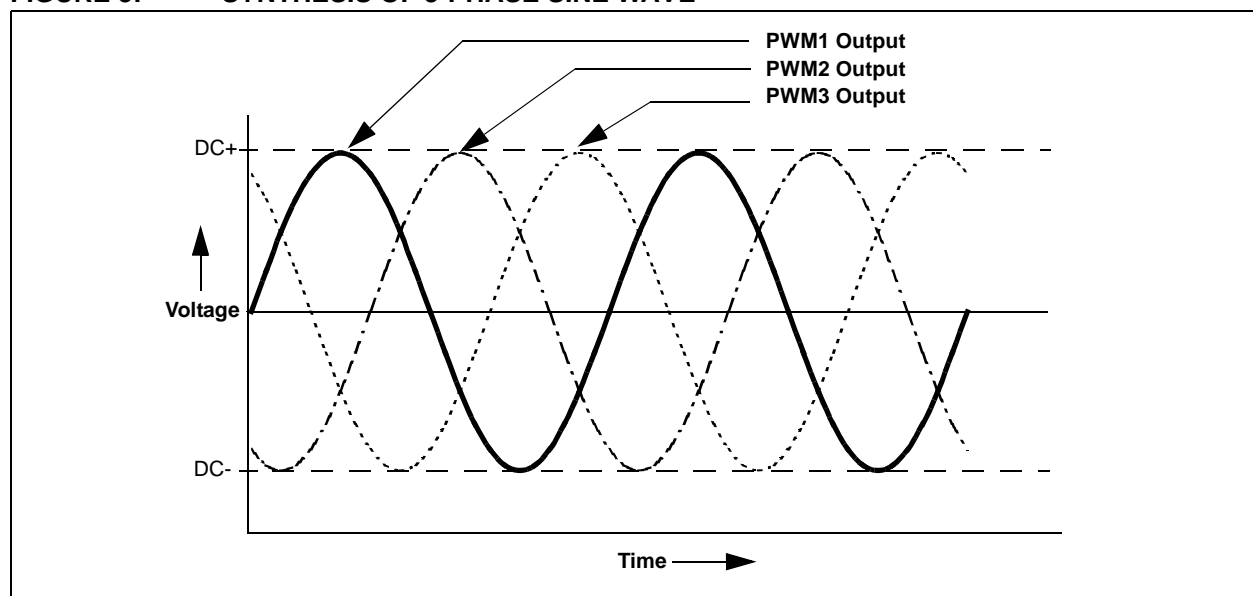


FIGURE 3: SYNTHESIS OF 3-PHASE SINE WAVE



V/f Control with Current Feedback

A disadvantage of open-loop V/f control is that the motor can stall if the speed is ramped up too quickly or the load otherwise changes rapidly. Without some form of feedback, it is impossible to detect whether the motor is turning as expected, or if it is stalled.

A stall causes high currents and the motor loses torque. By monitoring current, excessive slip can be detected, and the motor frequency can be adjusted downward accordingly. A high-current condition may also be caused by a malfunction of the inverter bridge. If a high-current condition persists, the drive should be shut down to prevent motor overheating or other damage.

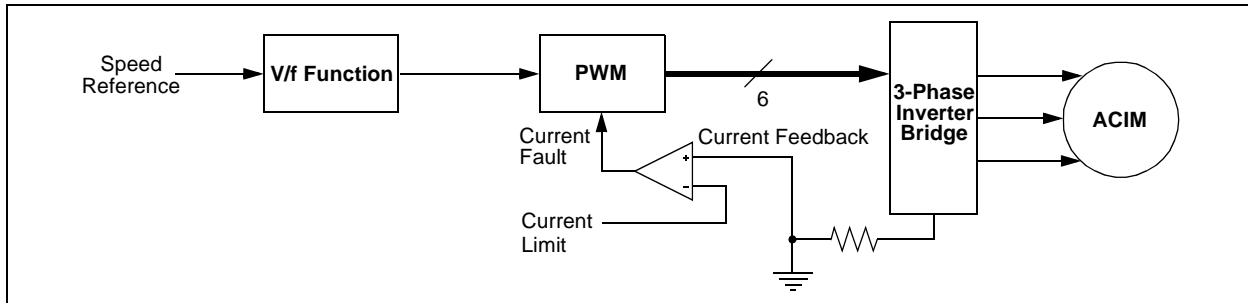
A conceptual diagram is illustrated in Figure 5. The speed reference is provided by the user, in this case via a potentiometer connected to an ADC channel. The V/f function in firmware calculates the maximum PWM duty cycle (amplitude) based upon the speed reference. The DC bus (bridge) current is measured using a

shunt resistor, which produces a voltage proportional to the current through it. This voltage is amplified and compared with an external comparator to a reference level that corresponds to the maximum allowable bus current. The comparator output drives the Fault A input of the PIC18F4431. If the Fault signal is asserted, the PWM output is inhibited for the following PWM period.

To detect a persistent overcurrent condition, the number of times the Fault signal is asserted is monitored in firmware. For example, if the Fault occurs more than 20 times within the last 256 PWM cycles, the motor is stopped and an overcurrent condition is indicated by blinking an LED. (The threshold number of events to trigger the overcurrent Fault can be changed in the firmware.)

The shunt voltage can also be monitored by using an ADC channel to detect increasing current. This way, corrective action can be taken by decreasing the drive frequency before the hardware Fault is activated.

FIGURE 4: CLOSED-LOOP V/f CONTROL WITH CURRENT FEEDBACK



V/f Control with Velocity Feedback

In open-loop V/f control, the rotor is assumed to follow the rotating flux generated in the stator, with a certain degree of slip present depending upon the load. In many applications, the load can vary widely and the resulting motor speed will vary accordingly. To improve speed control, a form of speed feedback can be added.

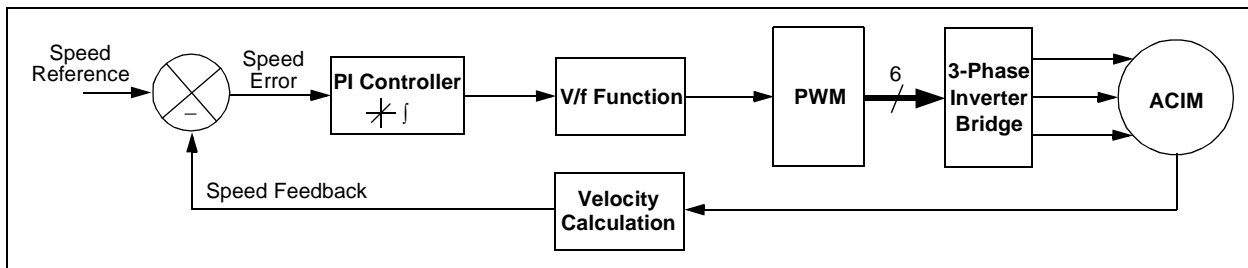
A simple implementation of closed-loop speed control is illustrated in Figure 5. The reference speed is still set by a potentiometer, as above. However, instead of directly using the reference speed to determine the drive frequency, it is compared to the actual motor speed to generate a speed error signal. Actual motor speed is established by a speed measurement with either the

Quadrature Encoder Interface (QEI) in Velocity mode, or input capture of a tachometer signal. In this particular application, the Quadrature Encoder Interface is used.

The speed error signal is then used as an input to a Proportional-Integral (PI) controller, which determines the desired drive frequency to the motor windings. The standard V/f process determines the amplitude of the drive waveform. The drive frequency and amplitude are then used to update the PWM duty cycles of the six PWM channels that drive the three-phase bridge.

Current feedback may also be used concurrently with velocity feedback. For clarity, it is not shown in this example.

FIGURE 5: CONCEPTUAL BLOCK DIAGRAM OF V/f CONTROL WITH VELOCITY FEEDBACK



FIRMWARE OVERVIEW

In this section, we will see in greater detail how the three control approaches discussed above are implemented in firmware. The firmware structure can be viewed in the flow diagrams shown in Figures 6 and 7. Keep in mind that these descriptions apply to the specific control applications written for Microchip's PICDEM MC demonstration board, particularly with regards to motor control and Fault display.

Initialization

The initialization routine sets the port pins to the desired states and initializes the peripherals. The High-Speed ADC, Power Control PWM, hardware Fault inputs and Quadrature Encoder Interface are initialized according to the control strategy that is being implemented (i.e., V/f, V/f with current feedback, V/f with current and position feedback). Initialization of these peripherals is discussed in detail below.

Main Loop

The main loop (see Figure 6) continuously checks Fault status and for key activity. Faults are handled by the Fault service routine; status is indicated by blinking LEDs 1 through 3.

Motor control is handled by the key service routine. Two push button switches toggle the motor between Run and Stop states and forward and reverse direction. When switching directions, the motor is first allowed to coast from its present angular velocity to zero and then accelerated to the reference speed in the opposite direction. This controlled manner of changing directions prevents high-current transients that could cause a Fault, provided that the acceleration rate is set appropriately for the motor. If PID control is being used, some of the PID functions may be calculated in this loop.

Key Activity Monitoring

SW1 and SW2 are monitored and debounced in firmware. SW1 is used to toggle between Run and Stop. SW2 is used to toggle between forward and reverse. (Figure 6). This is a subroutine within the main loop.

Fault Signals

Three Fault signals are monitored: overcurrent, overvoltage and overtemperature. The overcurrent and overvoltage Faults use the hardware Fault inputs to directly inhibit the PCPWM outputs on a cycle-by-cycle basis.

OVERCURRENT FAULT

A shunt resistor in the negative DC bus gives a voltage proportional to the current flowing through the three motor phases. This voltage is amplified and compared with a reference signal using an external comparator. On the PICDEM MC board, the reference signal may be adjusted for a current up to 6.3A. If the DC bus current signal exceeds the reference level, the Fault A pin is driven low, indicating an overcurrent Fault. Channel A is configured in Cycle-by-Cycle Fault mode. If the Fault occurs more than 20 times in 256 PWM cycles, then the motor is stopped and an overcurrent Fault is indicated by blinking LED1.

OVERVOLTAGE FAULT

The DC bus voltage is attenuated using a voltage divider and compared with a fixed reference signal using an external comparator. On the PICDEM MC board, when jumper JP5 is open, the overvoltage is set to 200V on the DC bus. If jumper JP5 is shorted, then the overvoltage limit is 400V. The Fault B pin is used to monitor the overvoltage condition. If the Fault occurs more than 20 times in 256 PWM cycles, the motor is stopped and an overvoltage Fault is indicated by blinking LED2.

OVERTEMPERATURE FAULT

The power module on the PICDEM MC board has a Negative Temperature Coefficient (NTC) thermal sensor that monitors the junction temperature of the IGBTs. It gives a 3.3V output for a 110°C junction temperature. The NTC is connected to AN8 through an analog optocoupler and is continuously measured. If it exceeds 80°C, the motor is stopped and an overtemperature Fault is indicated by blinking LED3.

ISR Loop

The ISR loop handles interrupts from the HSADC, Fault inputs, QE1 and time-base interrupts for three-phase waveform synthesis and PID control loop.

A/D Channel Conversion

AN0, AN1 and AN8 are converted sequentially using the high-speed ADC. AN0 is used to measure the DC bus current. AN1 is used as the input for the speed reference. AN8 is used to measure the power module junction temperature for detecting the overtemperature Fault condition.

FIGURE 6: MAIN LOOP, FAULT AND CONTROL SERVICE ROUTINES

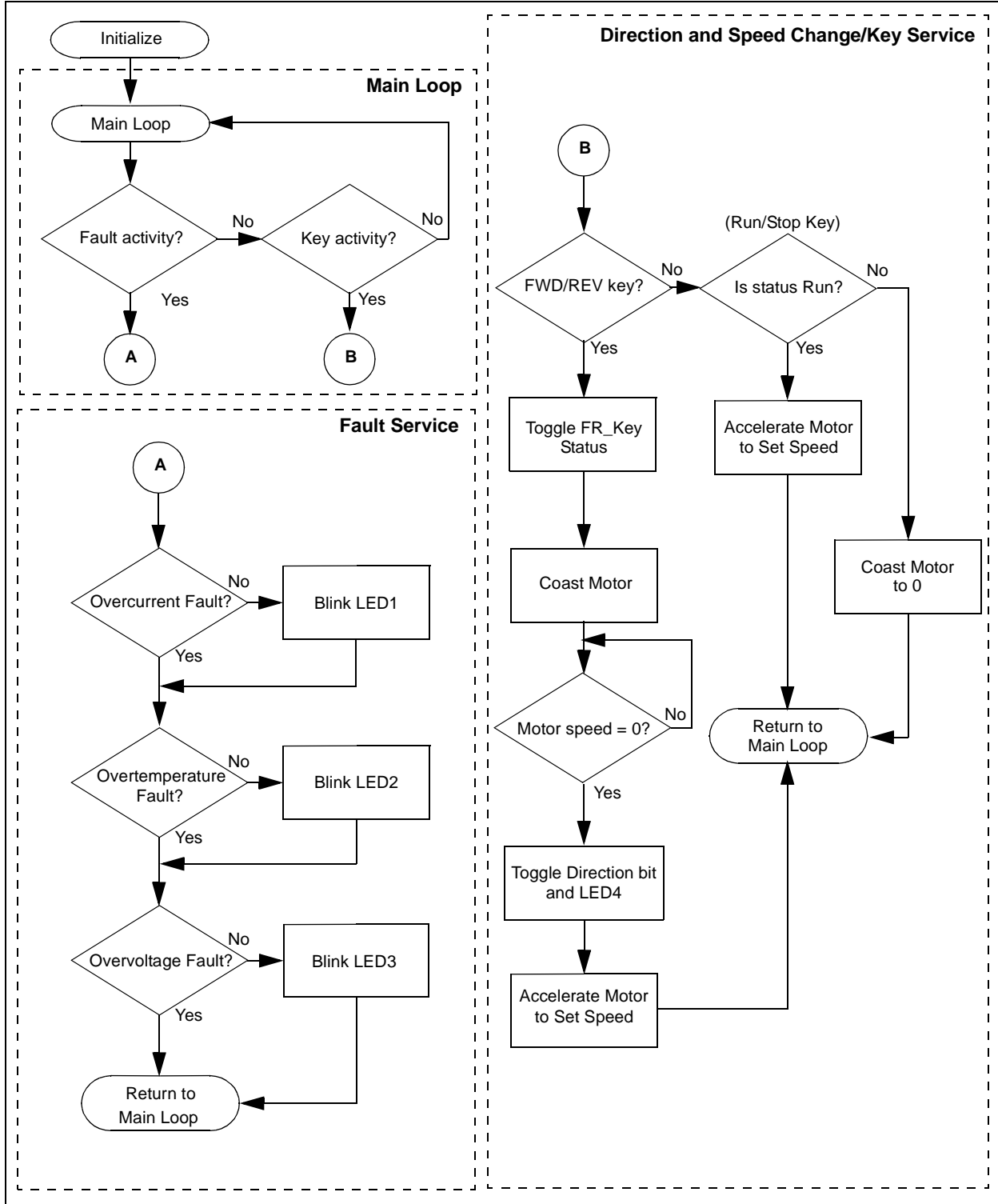
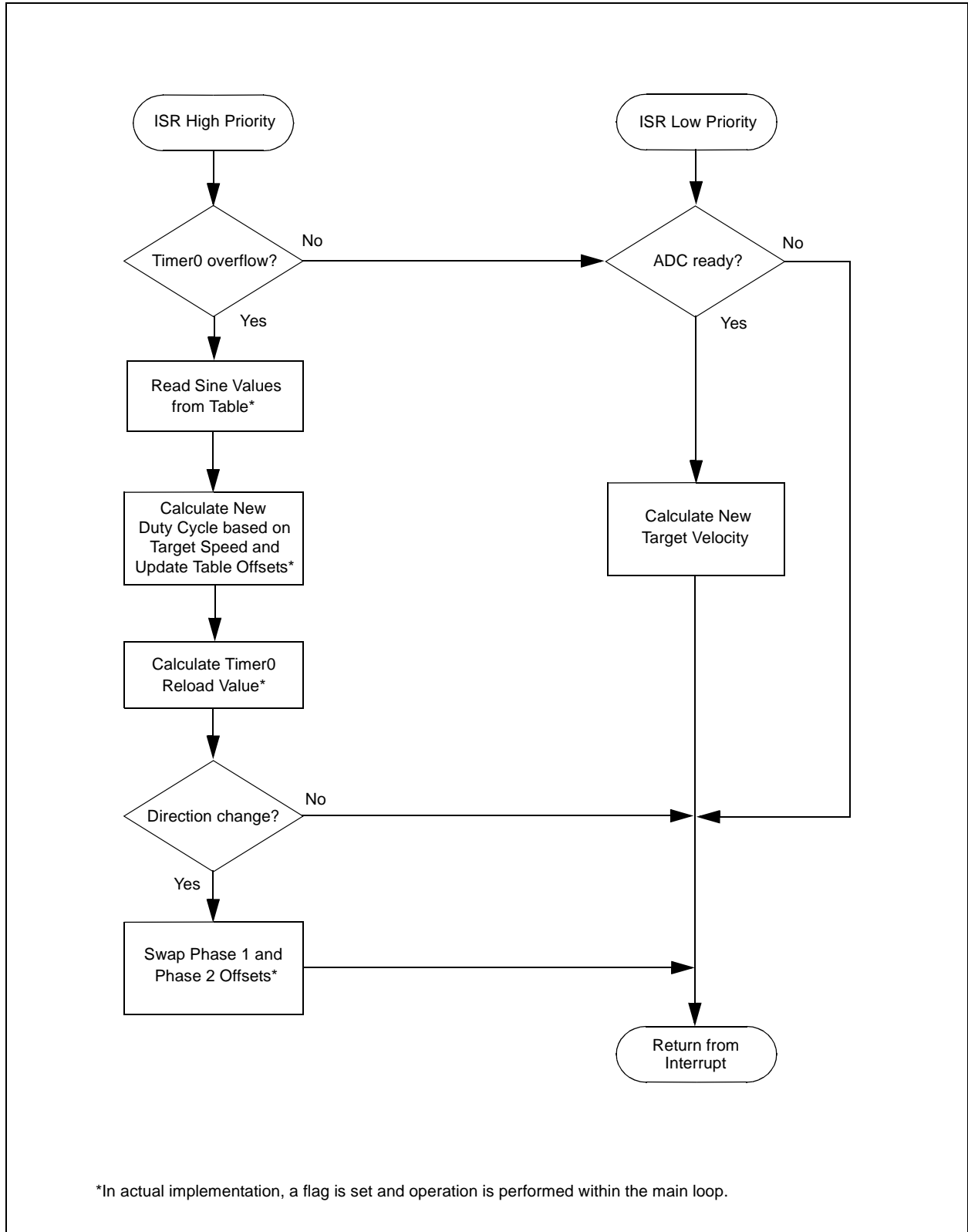


FIGURE 7: INTERRUPT SERVICE ROUTINE



INITIALIZING THE POWER CONTROL PWM MODULE AND HARDWARE FAULT INPUTS

The Power Control PWM module simplifies the task of driving a 3-phase inverter bridge by providing three pairs of complementary PWM outputs, with dead time inserted between complimentary channels. It also provides hardware-based Fault inputs that are capable of shutting down the PWM outputs completely in a Fault situation.

To initialize the PCPWM module:

1. Configure the PCPWM time base:
 - a) Select a PWM time base postscale value of 1:1.
 - b) Select a PWM time base prescale value input of 1:1 ($F_{OSC}/4$).
 - c) Configure the PWM time base for Free-Running mode (for edge-aligned operation).
2. Load the PTPERH:PTPERL register pair to obtain a PWM frequency of 20 kHz. The value to be used depends on the controller's clock frequency; refer to the data sheet to determine the proper value.
3. Configure the PCPWM output:
 - a) Enable PWM0 through PWM5 as outputs.
 - b) Set the PWM I/O pairs (PWM0/1, 2/3 and 4/5) as complementary pairs.
4. Configure the special event trigger:
 - a) Set the special event trigger postscaler to 1:1.
 - b) Configure the special event trigger to occur when the time base is counting upwards.
 - c) Enable updates from duty cycle and period buffer registers.
 - d) Configure for asynchronous overrides from the OVDCON register.
5. Configure the PCPWM dead time:
 - a) Select $F_{OSC}/2$ as the dead-time prescaler.
 - b) Load DTCON<5:0> with a dead-time value to achieve a 2 μ s dead time. The actual value depends on the controller's clock frequency; refer to the data sheet to determine the proper value.
6. Disable the output overrides on the PWM pins by setting bits POVD<5:0>.
7. Clear the special duty cycle register pair (SEVTCMPH:SEVTCMPL).
8. Clear all of the regular PWM duty cycle register pairs (PDCxH:PDCxL) to set the duty cycles to 0.
9. Enable the PWM time base.
10. Enable PWM Fault detection:
 - a) Enable both Fault A and Fault B.
 - b) Configure both Fault inputs to disable PWM channels 0 through 5.
 - c) Configure both Fault inputs to operate in Cycle-by-Cycle mode.

INITIALIZING THE HIGH-SPEED ADC MODULE

Three analog values are measured in this application:

- AN0 (DC bus current)
- AN1 (potentiometer input for the speed reference)
- AN8 (IGBT junction temperature in the inverter module)

The high-speed ADC incorporates several features, such as Auto-Conversion mode and a FIFO result buffer, that reduce the firmware overhead associated with monitoring multiple analog channels and enhance ADC throughput.

To initialize the HSADC module:

1. Configure ADC operation:
 - a) Enable Continuous Loop mode.
 - b) Enable Multi-Channel mode.
 - c) Configure auto-conversion sequence to sample sequentially from Group A and Group B.
 - d) Assign VREF+ and VREF- to AVDD and AVSS, respectively.
 - e) Enable the FIFO buffer.
 - f) Select the left-justified format for the A/D result.
 - g) Set the A/D acquisition time to 12 TAD (required for sequential conversion).
 - h) Set the A/D conversion clock to $F_{OSC}/32$
 - i) Turn on the ADC.
2. Configure interrupts and event triggers:
 - a) Set the A/D interrupt to be generated on every 2nd and 4th write to the FIFO buffer.
 - b) Disable external ADC triggers.
3. Configure input group assignments:
 - a) Assign AN0 to Group A. This will alternate with AN8 on every FIFO interrupt.
 - b) Assign AN8 to Group B.
4. Configure RA0, RA1 and RE2 as analog inputs:
 - a) Set the ANSEL0<1:0> and ANSEL<0> bits.
 - b) Set the TRISA<1:0> and TRISE<2> bits.

INITIALIZING THE QUADRATURE ENCODER INTERFACE

When implementing closed-loop V/f control, a form of velocity feedback is required. The Quadrature Encoder Interface can be used in conjunction with Timer5 to provide very accurate velocity feedback and direction information. In Velocity Measurement mode, velocity event pulses are generated on each edge of the QE A signal. Timer5 counts upward and its value is captured on each velocity pulse when it is reset to zero. The captured Timer5 value is stored in the Velocity registers (VELRH:VELRL), which is used by V/f control with Velocity Feedback mode.

To enable velocity measurement:

1. Configure the QEI module for Velocity mode measurement:
 - a) Enable Velocity mode (clear the $\overline{\text{VELM}}$ bit).
 - b) Set the QEI module mode to one of the 2x capture configurations.
 - c) Set the pulse reduction ratio to 1:1.
2. Configure Timer5 as the QEI time base:
 - a) Disable the Timer5 special event Reset.
 - b) Enable Continuous Count mode.
 - c) Set the input clock prescaler to 1:1.
 - d) Enable Synchronous Timer mode.
 - e) Enable the timer.
3. Enable the Timer5 interrupt and set as low priority

V/f CONTROL FIRMWARE

The heart of the ACIM control is accomplished with the PCPWM peripheral operated in Complimentary mode. The duty cycle of the three PWM channels are changed in a regular manner using a Timer0 interrupt to synthesize the three-phase waveforms that drive the motor.

A sine table is stored in program memory. It is transferred to data memory during initialization for faster access. Three registers are used as offsets to the table through indirect addressing. Each of the offset values

points to one of the values in the table, such that there is always a 120-degree phase shift between the phases. (Each of the waveforms in Figure 3 is created by an offset register associated with its PWM.) The code sample in Example 1 shows how the table is read using the indirect addressing registers.

In this application, the potentiometer determines the target motor speed reference signal. Waveform synthesis is identical for open-loop V/f and V/f with velocity feedback. The difference lies in how the motor drive frequency is generated from the target frequency. Depending upon the control strategy, the reference signal from AN1 is used in one of two ways:

DRIVE FREQUENCY CALCULATION FOR OPEN-LOOP V/f CONTROL

In this method, the motor drive frequency is directly calculated from the potentiometer input. Specifically, the upper byte of the A/D Result register is divided by four to give the target drive frequency f in Hz. In this application, the target speed has a lower boundary of 12 Hz and an upper boundary of 60 Hz. The V/f function determines the drive amplitude corresponding to that frequency.

Since the synchronous speed (in RPM) for an induction motor is $120fp$, where p is the number of stator poles, the target drive speed (in RPM) can be directly calculated as 30 times the value of ADRESH (120 divided by times ADRESH divided by 4), divided by p . For the current application, a motor with two stator poles is assumed. This reduces to a motor speed equal to 15 times ADRESH.

EQUATION 1: CALCULATING DRIVE FREQUENCY AND SPEED

Drive Frequency:

$$f_{\text{target}} = (\text{ADRESH}/4) \\ (5 \leq f_{\text{target}} \leq 60)$$

Calculated Speed:

$$N_s = (120/f \cdot p) \times (\text{ADRESH}/4) = (30 \cdot \text{ADRESH})/p$$

EXAMPLE 1: ACCESSING THE SINE TABLE THROUGH INDIRECT ADDRESSING

```

UPDATE_PWM_DUTYCYCLES      ;first update PWM1
    MOVF    TABLE_OFFSET1,W ;place offset for first sine value inWREG
    MOVF    PLUSW0,W         ;use the value in WREG as an offset to FSR0
                                ;FSR0 points to beginning of sine table
                                ;value at sine_table + table_offset1
                                ;is copied to WREG
    BZ      PWM1_IS_0        ;check to see if value is 0
    MULWF  FREQUENCY         ;multiply the table value times the frequency
    MOVFF  PRODH,PDC0H_TEMP  ;copy the result to the duty cycle register
    MOVFF  PRODL,PDC0L_TEMP
    BRA    UPDATE_PWM2      ;continue on to update PWM2
PWM1_IS_0
    MOVLW  0x02              ;If table value is zero,
                                ; make the duty-cycle a small non-zero value
    MOVWF  PDC0L_TEMP
                                ;continue on to update PWM2
    
```

AN900

DRIVE FREQUENCY CALCULATION FOR V/f CONTROL WITH VELOCITY FEEDBACK

The microcontroller uses the ADC measurement to calculate the target speed. The speed error is calculated by subtracting the actual speed (as measured by the

QEI) from the target speed. The velocity error is used as an input to the PID algorithm, which determines the motor drive frequency. The V/f function determines the corresponding drive amplitude.

EQUATION 2: CALCULATING DRIVE FREQUENCY FROM VELOCITY ERROR

1. Calculate Actual Speed from QEI Velocity Mode:

$$F_{\text{actual}} = \frac{F_{\text{OSC}}}{4} \times \frac{\text{QE edges per revolution}}{\text{Value of velocity register pair}} = f_{\text{actual}} \times 60$$

where: F_{actual} is the actual rotor speed (RPM) and f_{actual} is the actual rotor speed (Hz)

2. Calculate Speed Error:

$$f_{\text{error}} = (f_{\text{target}} \times (100 - \text{slip})) - f_{\text{actual}}$$

where: f_{target} is the target drive frequency (Hz), f_{error} is the frequency error and slip is the expected percent slip

3. Calculate Drive Frequency from PID Algorithm:

$$f_{\text{target}} = (Kp \times f_{\text{error}}) + (Ki \times \sum f_{\text{error}}) + (Kd \times (f_{\text{error}_t} - f_{\text{error}_{t-1}}))$$

where: Kp is the proportional gain
 Ki is the integral gain
 Kd is the differential gain

$\sum f_{\text{error}}$ is the cumulative sum of frequency errors
 $f_{\text{error}_t} - f_{\text{error}_{t-1}}$ is the difference in error between the current and immediately previous time period

PWM Waveform Synthesis

The sinusoidal waveform is created by constantly changing the PWM duty cycle for each output. The motor drive frequency determines how often the PWM duty cycle values are updated and thus, the frequency of the synthesized waveform. The peak-to-peak drive amplitude corresponds to the maximum PWM duty cycle, as this generates the maximum voltage output of each half-bridge of the inverter. The duty cycle determines the drive amplitude at any given point in the cycle.

The duty cycle update rate is set by modifying the Timer0 reload value. This determines the interval until next Timer0 overflow. The PWM Duty Cycle (PDC) registers of the three PWM units are modified as follows:

- When a Timer0 interrupt occurs, an updated target drive frequency is determined by either Equation 1 or Equation 2 (depending on the control method being used).
- The sine value for each phase is read from the sine table, pointed to by the offset value for that phase.
- The PWM duty cycle for a particular phase is calculated by multiplying the sine value from the table by the updated motor drive frequency. The 16-bit product is stored in the PDC register for that phase. Steps 2 and 3 are repeated for each phase.
- The offset values are updated for the next table access.
- If the direction of rotation is to be reversed, then the offsets of two phases are swapped. The offsets of Phase 1 and 2 are swapped for this purpose.
- The Timer0 reload value is calculated based on the updated motor drive frequency (Equation 3), where f is the drive frequency. In the current version of firmware, the number of sine table entries is set at 19. The reload value determines the value at which the PWM duty cycle is updated.
- The new PWM duty cycle values take effect at the beginning of the next PWM period. The duty cycle determines the drive amplitude at any given point in the cycle.

EQUATION 3: CALCULATING TIMER0 RELOAD VALUE

$$\text{Timer0 Reload Value} = \text{FFFFh} - \frac{\frac{F_{\text{OSC}}}{4}}{(2 \cdot (\text{sine table entries} - 1)) \times \text{Timer0 prescaler value} \times f}$$

PID CONTROL FIRMWARE

PID is a well-known, commonly used method of feedback control. As seen in the PID algorithm in Equation 2, PID generates a control signal by multiplying the error, the integral of the error and the derivative of the error by individual gains and then summing the results. The proportional term generates a corrective signal in proportion to the error. The integral term generates a corrective signal proportional to summation of the error over time. The derivative term generates a corrective signal in proportion to the rate of change of the error. In velocity control applications, the derivative gain is often set to zero, as PI control is usually sufficient for achieving well-tuned speed control.

To implement V/f control with velocity feedback, the target speed, actual speed and speed error are all calculated as shown in Equation 2. The speed error is passed to the PID algorithm. Integral error is calculated in the PID routine by accumulating the speed error over time. Derivative error is calculated by subtracting the last error value from the present error value. Since the routine is called at fixed time intervals, the difference in the two error values is proportional to the rate of change of error. In this application, the PWM period interrupt rate is used to determine the update rate of the PID calculation.

The PID functions used in this application note are described in AN937, "Implementing a PID Controller Using a PIC18 MCU".

CLOSED-LOOP SLIP CONTROL

In many applications, it is desirable to control slip in order to optimize for torque, efficiency or power factor depending upon changing requirements. Figure 8 shows how torque, power factor and efficiency may vary with the degree of slip for a typical motor. By varying the amount of desired slip, the motor performance can be optimized for any of these three attributes. For example, torque may be maximized by allowing a higher degree of slip; efficiency optimized by allowing a lesser degree of slip.

To control slip, the actual motor speed is compared against the drive frequency to determine the present slip frequency. The slip frequency is compared to the desired slip frequency to produce a slip frequency error. Drive frequency and amplitude are modified in order to minimize the slip frequency error.

Figure 9 shows how a slip control could be implemented. Identical hardware is used as in V/f control with velocity feedback. Only the algorithm is modified.

FIGURE 8: TORQUE, POWER FACTOR AND EFFICIENCY VERSUS SLIP

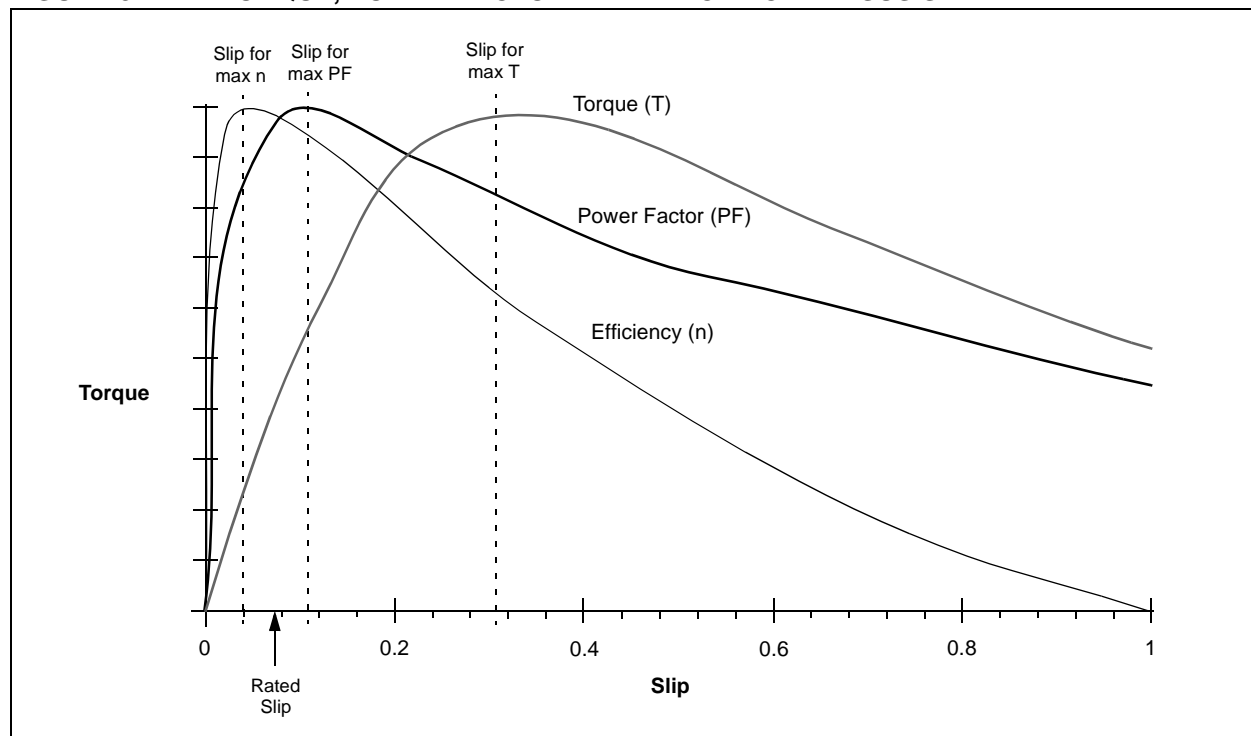
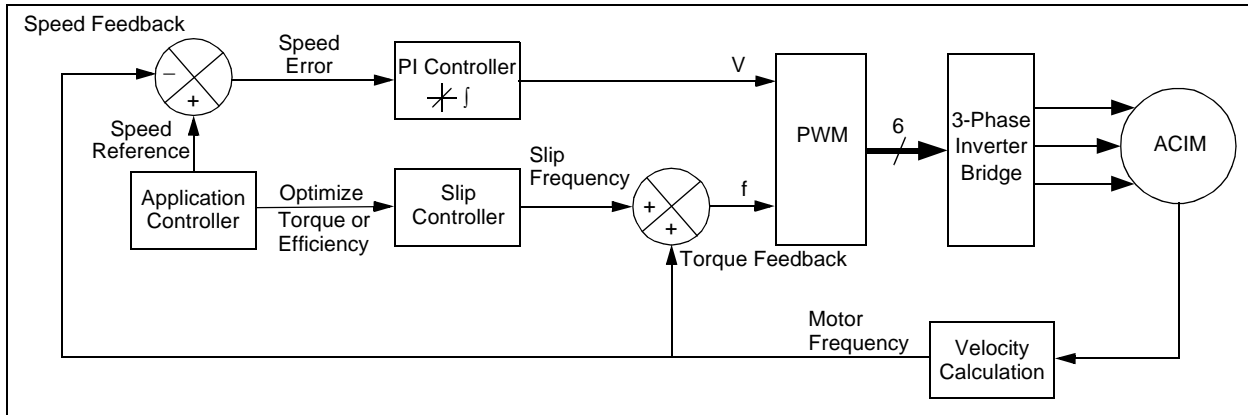


FIGURE 9: CONCEPTUAL BLOCK DIAGRAM FOR CLOSED-LOOP SPEED AND SLIP CONTROL



COMPARING MICROCONTROLLERS FOR AC INDUCTION MOTOR CONTROL APPLICATIONS

In developing ACIM applications, an important consideration is often the total part count required to implement the solution. For 3-phase control applications using the PIC18F452 and PIC16F7X7, this means a bridge driver that is capable of generating the complimentary outputs and inserting dead time.

In contrast, the PCPWM module of the PIC18F4431 can be configured to generate complimentary outputs with configurable dead time, resulting in a simpler bridge driver circuit. This can translate into a reduced part count for the application.

The PIC18F4431 also offers the MFM to measure both rotor speed and direction with little or no additional support from external components. With previous control solutions, external circuitry would be needed to provide pulse conditioning and direction indication from the encoder; an additional on-chip timer resource would also be needed to convert this into velocity data. The MFM can directly interface with a quadrature encoder or other Hall sensors and calculate direction and velocity with a minimum of additional hardware or firmware overhead.

The capabilities of the three different microcontrollers in 3-phase ACIM control are compared in Table 1. The range of control strategies for the microcontrollers is presented in Table 2.

TABLE 1: COMPARISON OF HARDWARE CAPABILITIES IN 3-PHASE ACIM CONTROL

Microcontroller	PWM Outputs	Dead Time	Complimentary Signal Generation	Velocity Feedback
PIC18F452	3 total: 2 CCP, 1 firmware generated	External hardware	External hardware	None
PIC16F7X7	3 (CCP)	External hardware	External hardware	None
PIC18F4431	8 (PCPWM)	Built into PCPWM	Built into PCPWM	QEI/IC

TABLE 2: COMPARISON OF PICmicro CONTROLLERS AND ACIM CONTROL STRATEGIES

Control Method	Microcontroller Family			
	PIC18FXX31	PIC18FXX39	PIC16F7X7	PIC18FXX2
V/f Control	Yes	No	Yes, with external dead-time provisions	Yes, with firmware PWM and external dead-time provisions
V/f Control with Current Feedback	Yes, using hardware Fault input and HSADC	No	Yes, with external dead-time provisions	Yes, with firmware PWM and external dead-time provisions
V/f Control with Velocity Feedback	Yes, using QEI or input capture	No	No	No
Single-Phase V/f Control	Yes	Yes	Yes	Yes

CONCLUSION

The combination of PIC18 architecture and thoughtfully designed peripherals make the PIC18F4431 an excellent choice for 3-phase AC induction motor control applications.

The PCPWM module provides sufficient PWM outputs and modes to directly drive a gate driver/inverter module, without the need of additional hardware to create complimentary channels or insert dead time. The MFM allows users to easily implement speed and direction monitoring with minimal hardware or firmware overhead. All of these features, along with the HSADC, make it possible to design a wide range of powerful motor-control solutions with a minimum of parts.

REFERENCES

- P. Yedamale, *AN843, "Speed Control of 3-Phase Induction Motor Using PIC18 Microcontrollers"* (DS00843). Microchip Technology Inc., 2002.
- R. Parekh, *AN887, "AC Induction Motor Fundamentals"* (DS00887). Microchip Technology Inc., 2003.
- R. Parekh, *AN889, "V/f Control of 3-Phase Induction Motors Using PIC16F7X7 Microcontrollers"* (DS00889). Microchip Technology, Inc., 2003.
- "PICDEM™ MC Development Board for PIC18FXX31 User's Guide"* (DS51453). Microchip Technology, Inc., 2004.

APPENDIX A: PICDEM™ MC BOARD OVERVIEW

A PICDEM MC demonstration board was used to develop, test and debug the ACIM control code discussed in this application note. The overall block diagram is shown in Figure A-1.

The board has a diode bridge rectifier that converts a single-phase AC input to DC, while a power capacitor bank provides a stable DC bus. A switching power supply generates several DC supply levels for digital, analog and power electronics. A 3-phase IGBT-based inverter bridge with integrated gate drivers is used to drive the motor from the DC bus.

The on-board user interface has two momentary push buttons, a potentiometer and four LEDs to indicate status. In this application, switch SW1 is used to toggle between motor Run and Stop. Switch SW2 is used to

toggle between forward and reverse rotation. The potentiometer is used for setting the desired speed. The LEDs are used for indications of different states of control and as Fault indicators. The board can also be controlled with a host PC over a serial port using Microchip's own Motor Control GUI.

The control circuit and power circuits are electrically isolated from each other by optoisolators. With the isolation between power and control circuits, programming and debugging tools can be plugged to the development board with power connected to the board. For debugging the code in this application note, an MPLAB® ICD 2 was connected directly to the PICDEM MC board during development.

For a complete description, please refer to the "PICDEM MC Development Board for PIC18FXX31 User's Guide" (DS51453).

FIGURE A-1: PICDEM MC EVALUATION BOARD FUNCTIONAL BLOCK DIAGRAM

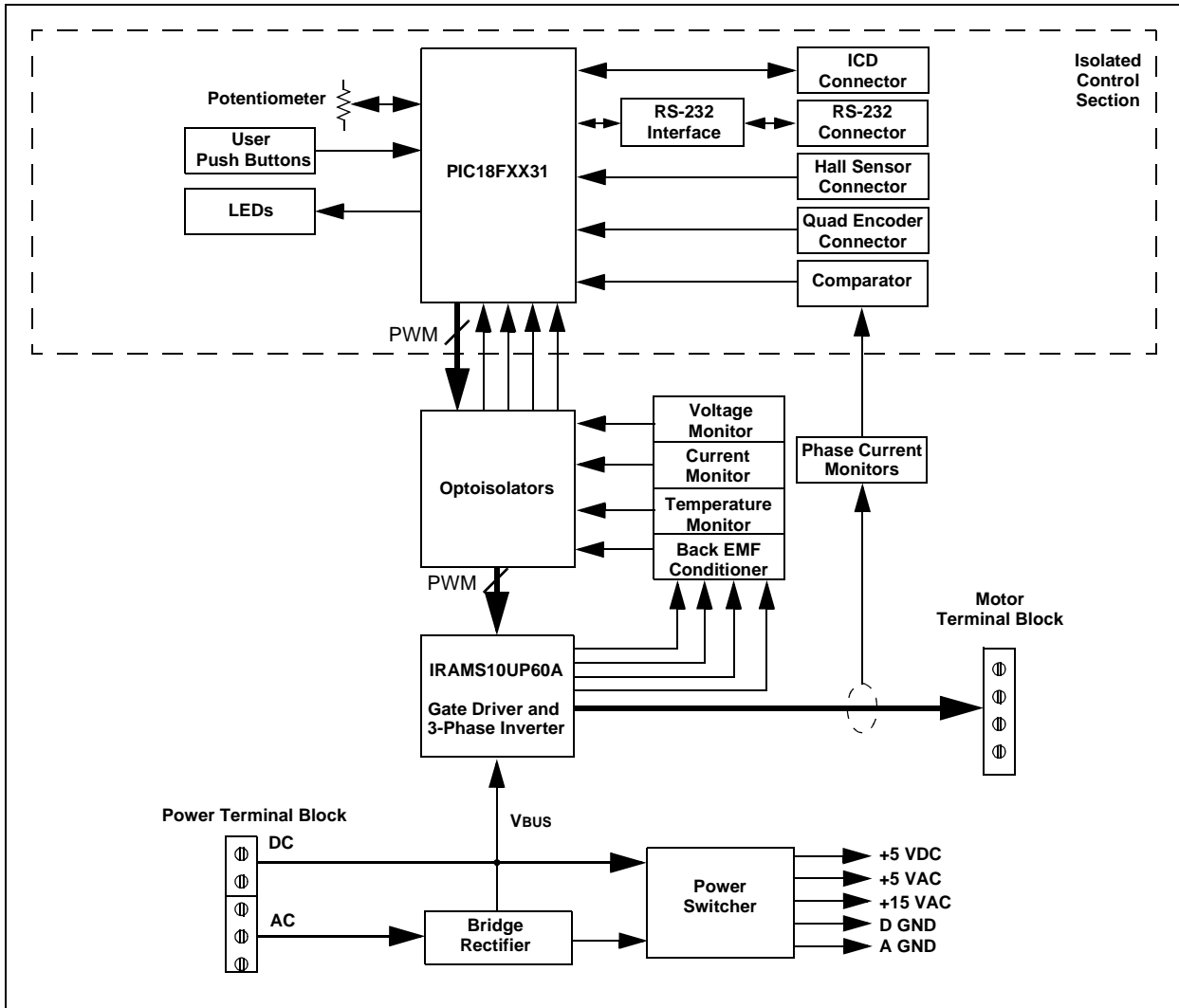
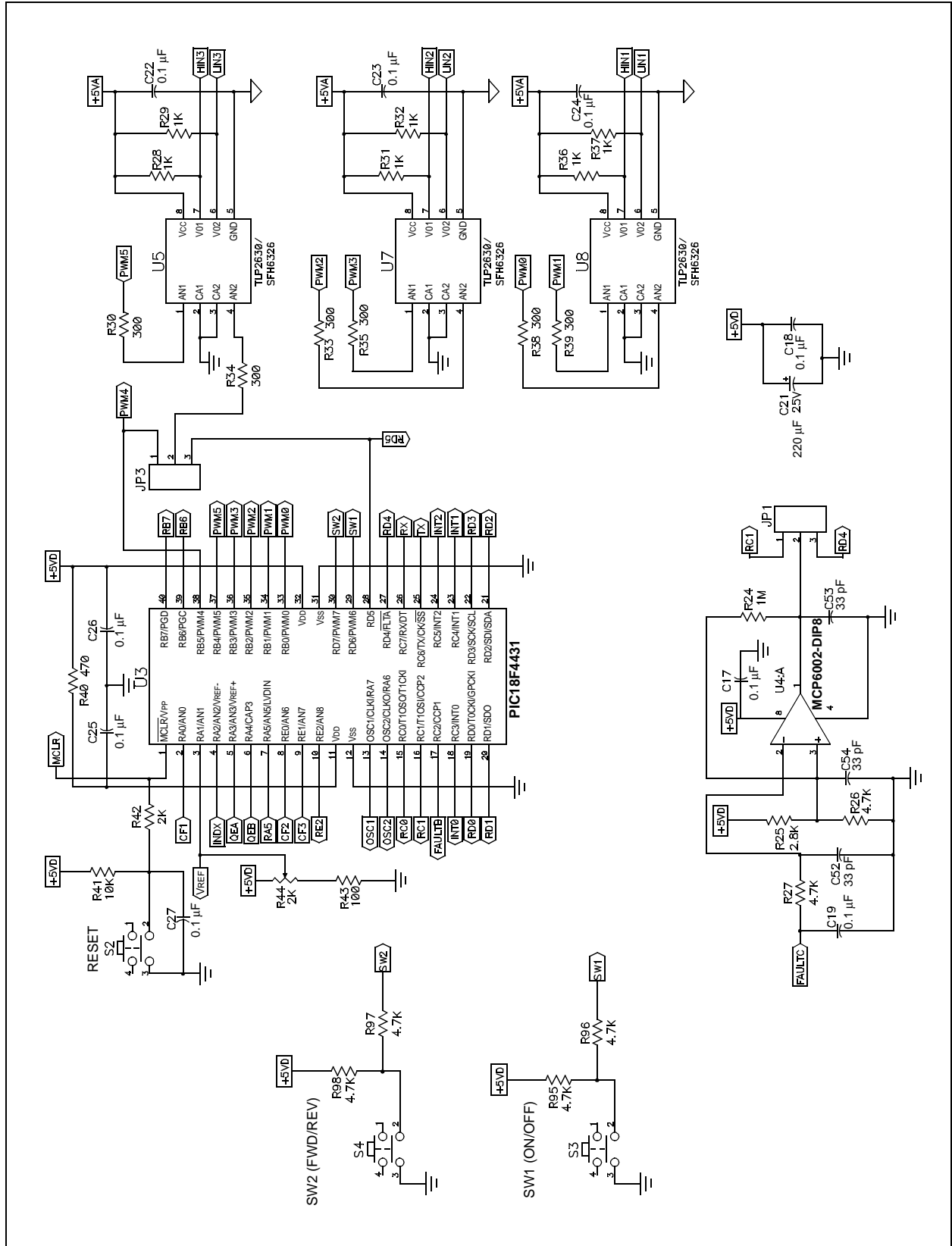


FIGURE A-2: BOARD SCHEMATIC, PART 1 (PIC18F4431 MICROCONTROLLER, PCPWM ISOLATORS, CURRENT COMPARATOR AND ASSOCIATED PARTS)



AN900

FIGURE A-3: BOARD SCHEMATIC, PART 2 (PIC18F2431 MICROCONTROLLER SOCKET, USART, CLOCK OSCILLATOR NETWORK AND OPTIONAL LIN INTERFACE)

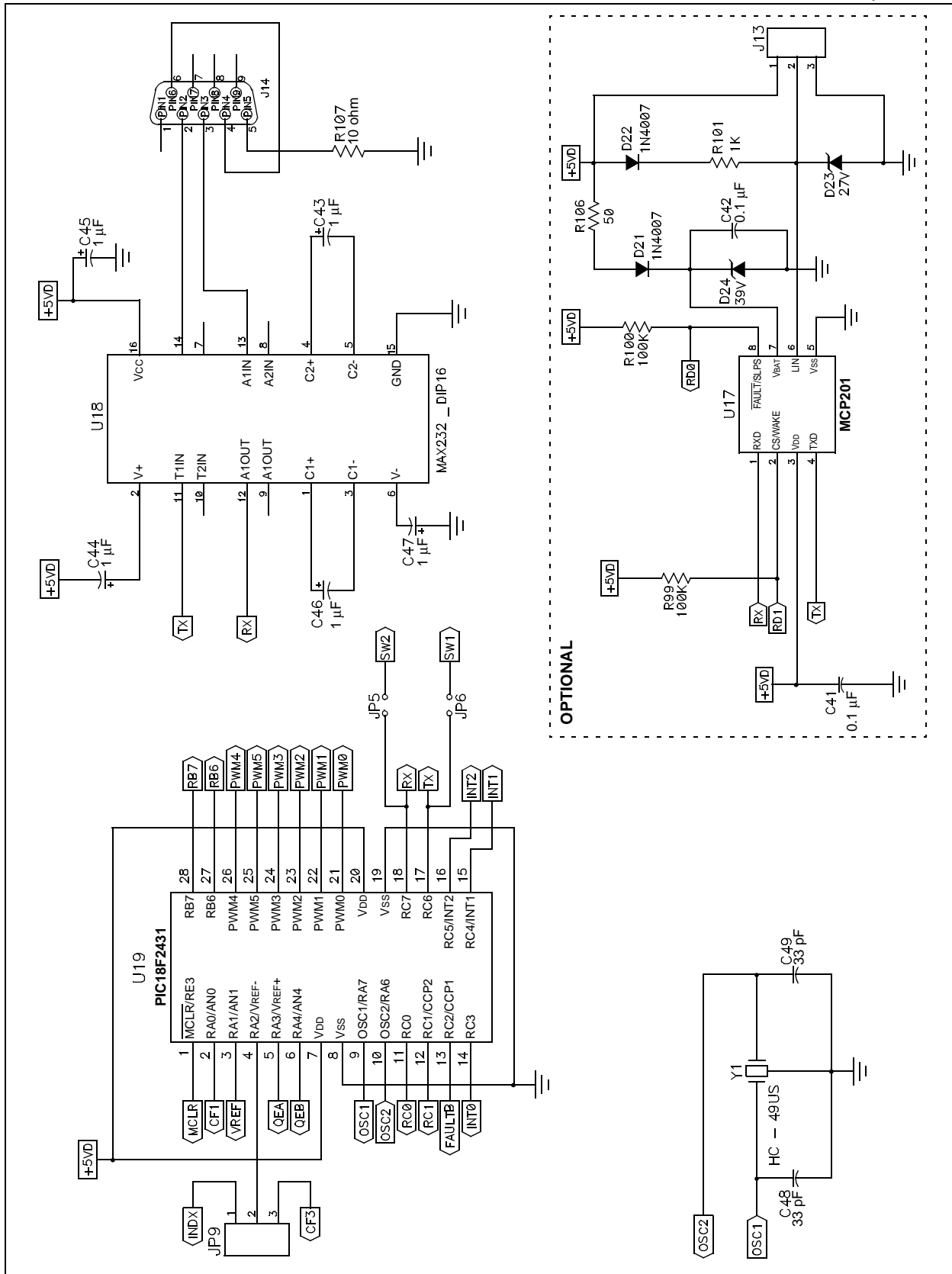


FIGURE A-4: BOARD SCHEMATIC, PART 3 (SENSOR AND MICROCONTROLLER HEADER CONNECTORS, MONITOR LEDS)

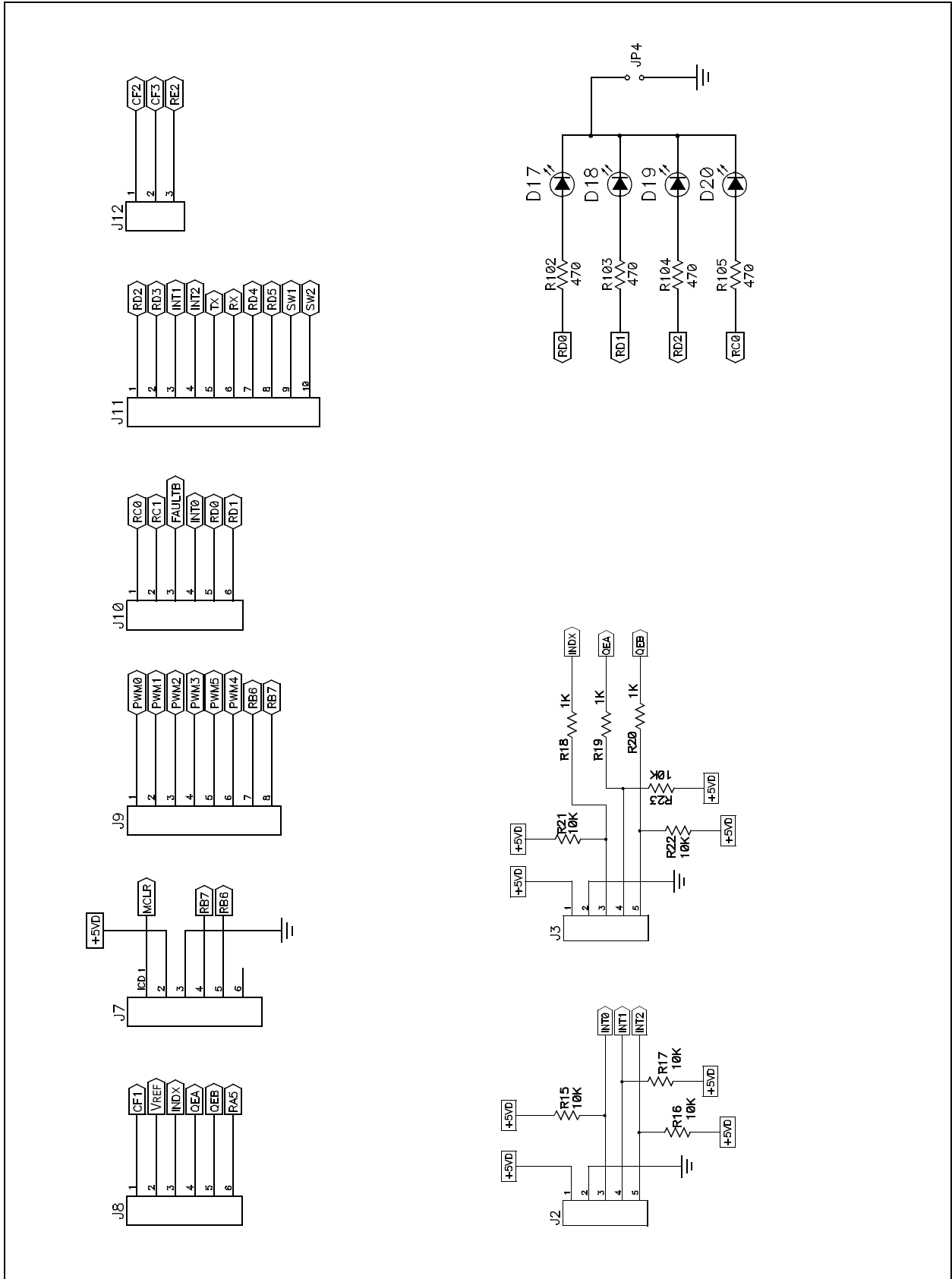


FIGURE A-5: BOARD SCHEMATIC, PART 4 (SIGNAL CONDITIONER FOR SENSORLESS BLDC OPERATION)

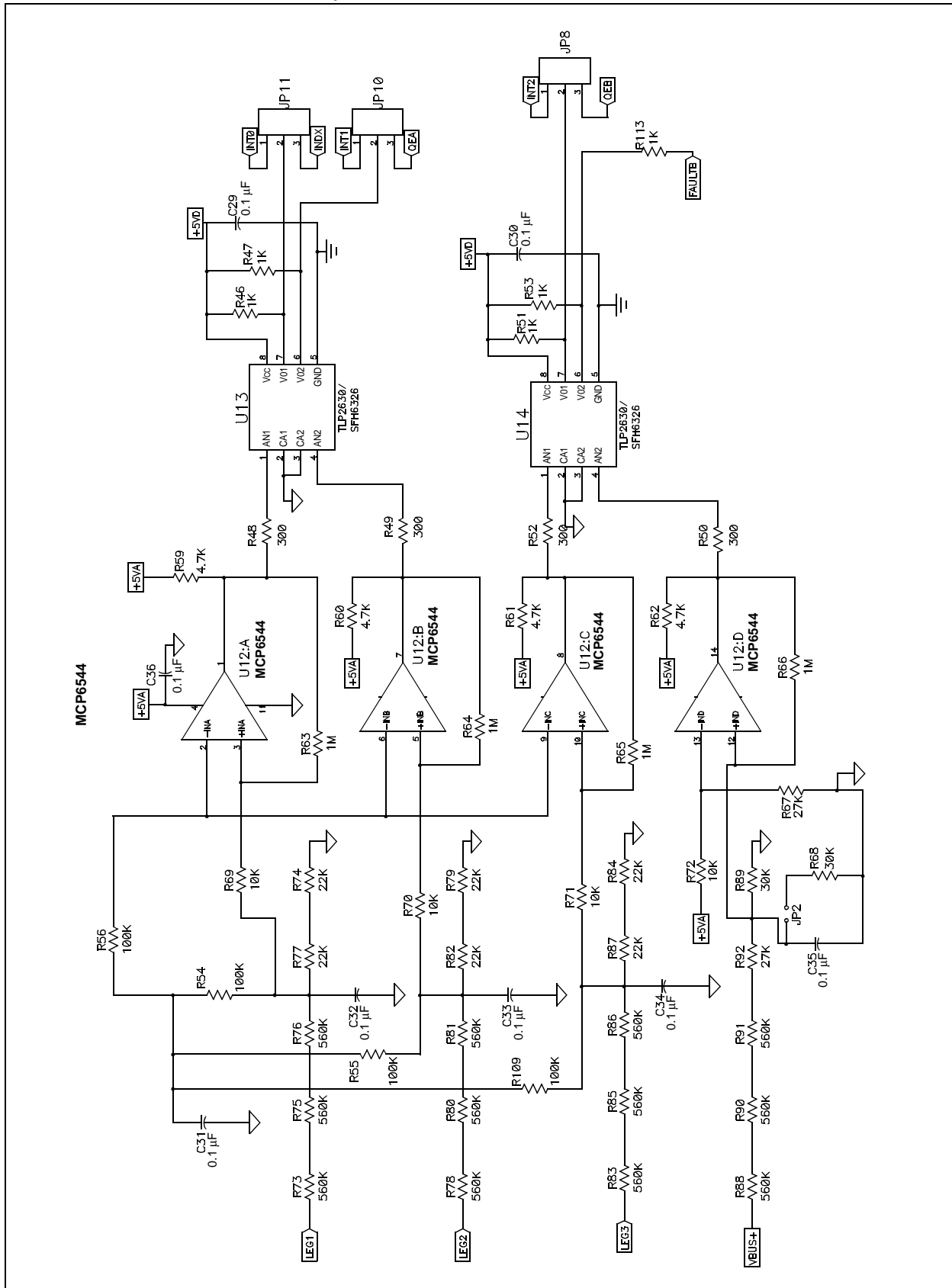
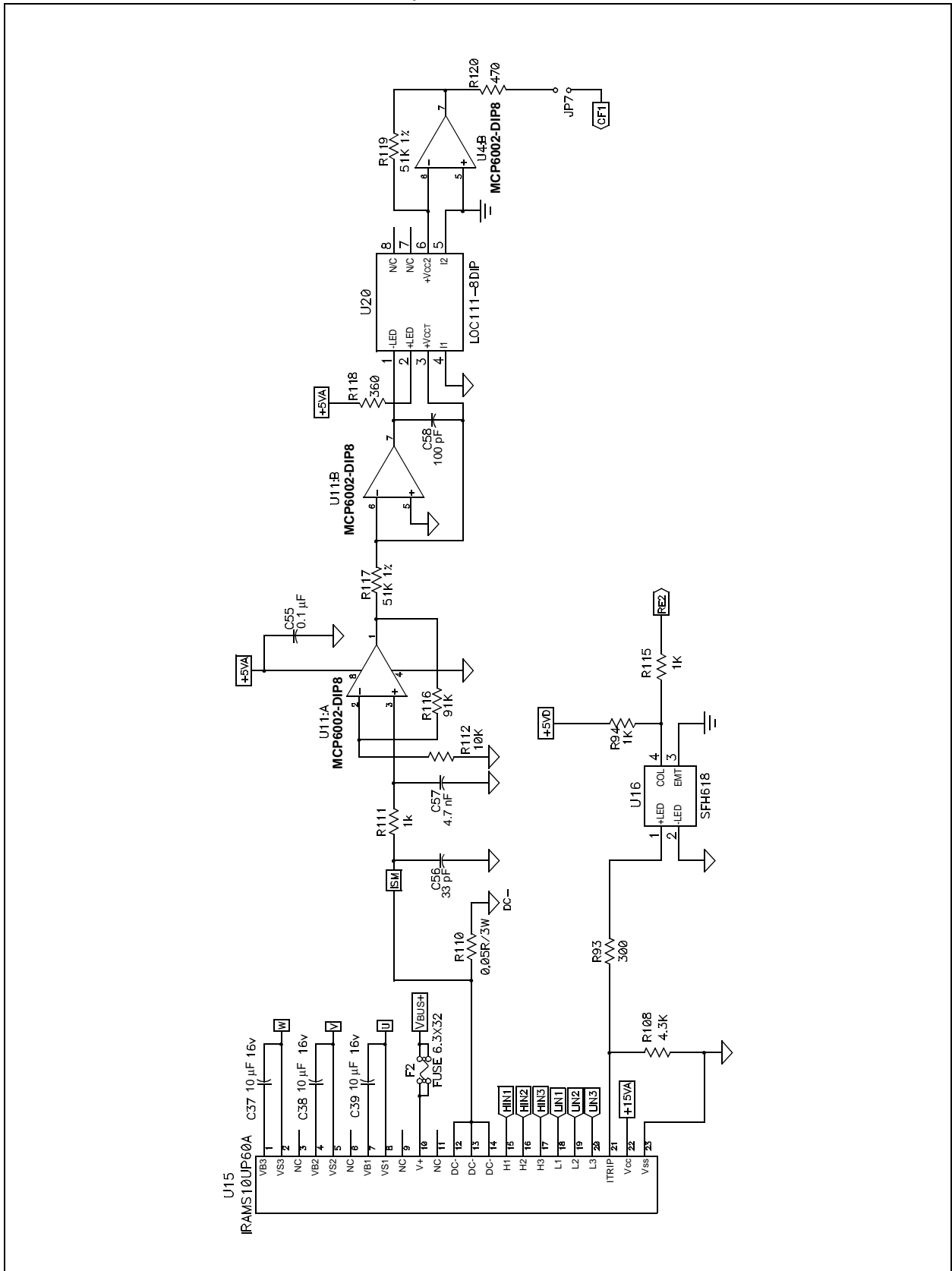
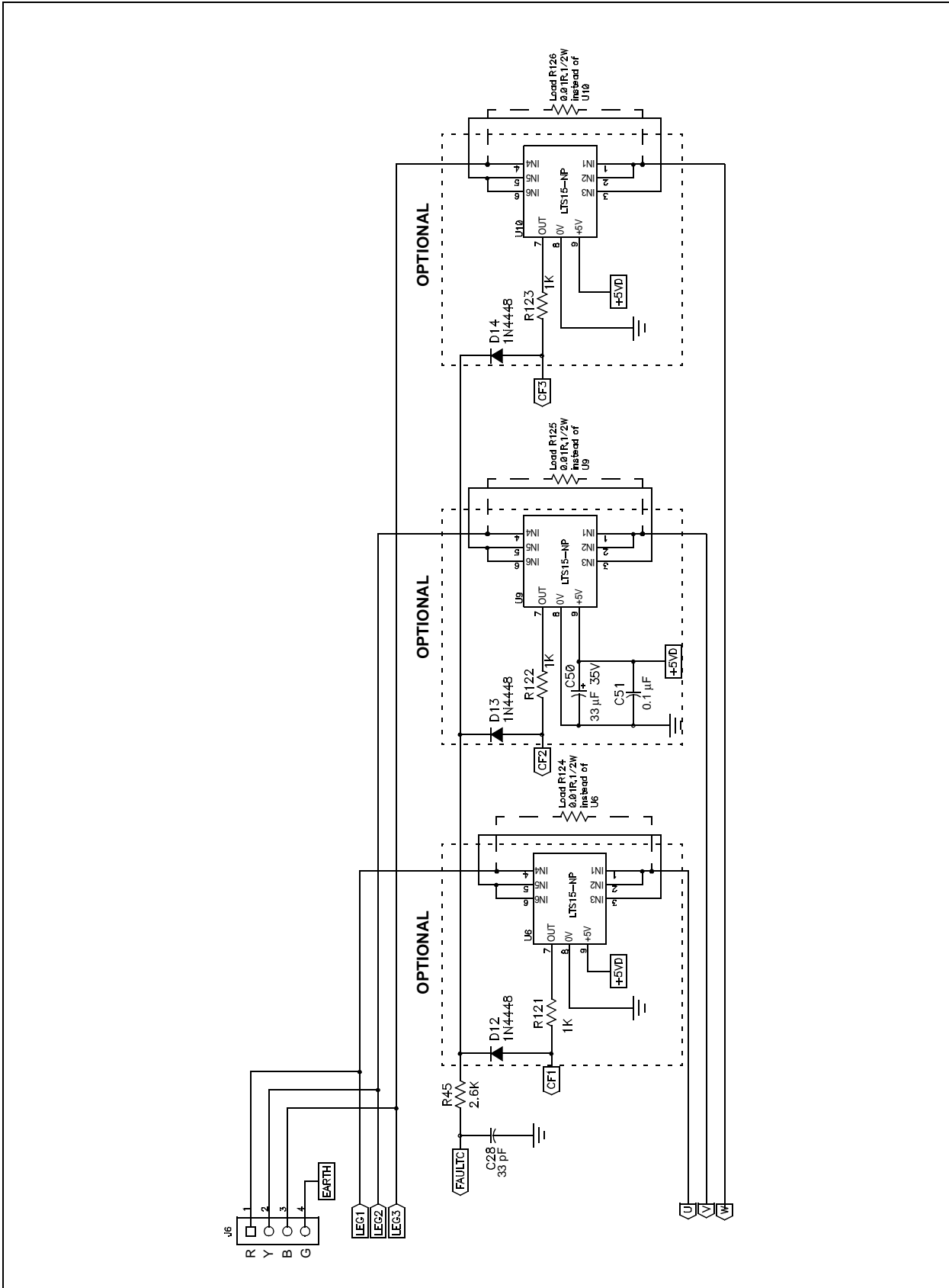


FIGURE A-6: BOARD SCHEMATIC, PART 5 (3-PHASE INVERTER POWER MODULE AND SHUNT CURRENT MEASUREMENT)



AN900

FIGURE A-7: BOARD SCHEMATIC, PART 6 (MOTOR TERMINAL BLOCK AND OPTIONAL CURRENT TRANSDUCER CIRCUITRY)



APPENDIX B: SOFTWARE DISCUSSED IN THIS APPLICATION NOTE

Due to size considerations, the complete source code listing for the applications described here is not included in the text. A complete version of the source code, with all required support files, is available for download as a Zip archive from the Microchip web site at:

www.microchip.com

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.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, Accuron, dsPIC, KEELOQ, microID, MPLAB, PIC, PICmicro, PICSTART, PRO MATE, PowerSmart, rfPIC, and SmartShunt are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.


AmpLab, FilterLab, MXDEV, MXLAB, PICMASTER, SEEVAL, SmartSensor and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, dsPICDEM, dsPICDEM.net, dsPICworks, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, PICkit, PICDEM, PICDEM.net, PICLAB, PICtail, PowerCal, PowerInfo, PowerMate, PowerTool, rLAB, rfPICDEM, Select Mode, Smart Serial, SmartTel and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2004, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

**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: 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'Shaughnessy 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 høj 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 - Ier 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

Waegenburghtplein 4
NL-5152 JR, 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

05/28/04