DESIGN OF LOW COST VARJABLE SPEED DRIVE FOR AIR BLOWER MOTOR

A dissertation submitted to the
Department of Electrical Engineering, University of Moratuwa
in partial fulfillment of the requirements for the
Degree of Master of Engineering

BY
Nihal Tashmantha Athapattu

Supervised by Dr. J. P. Karunadasa

Department of Electrical Engineering
University of Moratuwa, Sri Lanka

2006

Abstract

In Air conditioning air blowers are working at full speed although the conditional area at set temperature. Tins situation consumed considerable amount of energy to drive \ the fan or blower and running the refrigeration cycle itself The amount of energy which is used to drive the air blower and refrigeration cycle of the air conditioning system can be considered as loss of energy. This loss can be minimized by introducing the Variable Speed Drive (VSD) to drive the air blower or fan by sensing the conditional area temperature etc.

Cost of the VSD is very high in the local market, and it creates long pay back period for investment of installing a VSD for air conditioning system. Also electricity tariff in Sri Lanka is very high and has a trend of increasing. Therefore by introducing VSD the electrical energy used for Air conditioning can be minimized.

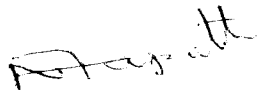
By considering the above facts low cost VSD is useful for the local market. In this project low cost Micro controller has been used as the motor control unit. Micro controller generates PWM to drive IGBTs in power module. The method used to vary the speed of induction motor is varying the voltage and frequency by keeping voltage/frequency ratio constant. Micro controller senses the variable voltage created by the potential meter or controller. TIns voltage is converted into form of binary value by one of analog to digital converter in the micro controller. This value is used to calculate pulse width and number of pulses in the period. The software program senses the ON/OFF switch input for the Micro controller and accordingly Run or Stop the motor.

The program can be further developed for improving reliability and safety of the

DECLARATION

The work submitted in this dissertation is the result of my own investigation, except where otherwise stated.

It has not already been accepted for any degree, and is also not being concurrently submitted for any other degree.



N. T. Athapattu
06 / 12 / 2006

I endorse the declaration by the candidate.

UOM Verified Signature

Dr. J. P. Karunadasa

Contents

Acknowledgement

Thanks are due first to my supervisor, Dr J. P. Karunadasa senior lecturer Department of Electrical Engineering, for his great insights, perspectives, guidance and sense of humor. My sincere thanks go to Professor Ranjth Perera, Head of Department Electrical Engineering, and Dr Lanka Udawatta lecturer Department of Electrical Engineering, for helping various ways to clarify the things related to my academic work with excellent cooperation and guidance. Sincere gratitude is also extend to Dr Thrishantha Nanayakkara, lecturer Department of Mechanical Engineering, for his guidance for programming of PIC micro controllers and related things. Finally, I should thank many individuals, friends and colleagues who have not been mentioned here personally in making this education process successful.

List of Figures

Figure **Page**

CHAPTER 1

Introduction

A variable speed drive regulates the speed of the motor, and in turn the speed of the pump or fan, by controlling the energy that goes into the motor, rather than restricting the flow of a process running constantly at full speed.

A variable speed drive can save over 60% of the energy. This is possible as it controls the energy at source, only using as much as is necessary to run the motor with the required speed and torque much in the same way as the accelerator in the car controls the engine. As acceptance of the technology increases, variable speed drives are gaining market share across a range of industries. During the last few decades, the variable speed controlling of AC motors have been popular due to the achievements of higher efficiency, smooth controlling, reliability and the expansion of lifetime of the equipments. Especially the variable speed controlling on induction machines have been very popular since the simplicity and the low cost of the machines itself. Therefore several controlling mechanisms had been introduced on controlling the induction machine. Mainly, these are open loop control systems. During this project work it is tried to check the availability methods and to introduce closed loop control system on three-phase induction motor. In addition to that at the steady state operation a novel approach is proposed to eliminate the torque ripple in the torque profile. The controller is planned to be implemented in a microcontroller.

Recently, electronic power and control systems have matured to allow these components to be used for motor control in place of mechanical gears. These electronics not only control

the motor's speed, but can improve the motor's dynamic and steady state characteristics. In addition, electronics can reduce the system's average power consumption and noise generation of the motor. Induction motor control is complex due to its nonlinear characteristics. While there are different methods for control, Variable Voltage Variable Frequency (VVVF) or V/*f* is the most common method of speed control in open loop. This method is most suitable for applications without position control requirements or the need for high accuracy of speed control. Examples of these applications include heating, air conditioning, fans and blowers. V/*f* control can be implemented by using low cost microcontrollers, rather than using costly digital signal processors (DSPs).

In many heating, ventilation, and air conditioning (HVAC) applications, air handler motors are either off, or on at full speed. However, by adding variable speed control to the air handler, significant energy savings over the standard on/off control can be realized, resulting in significantly reduced cost of operation. The task of designing the variable speed air handler is greatly simplified by using the Microchip microcontroller.

Achievement in brief

In particular, variable speed drives can be used to reduce energy consumption in fan and pump systems. A pump or fan running at half speed consumes only one-eighth of the energy of one running at full speed. The power required to run a pump or a fan is proportional to the cube of the speed. This means that if 100% flow requires full power, 75% requires $(0.75)^3$= 42% of full power, and 50% flow requires $(0.5)^3$= 12.5% of the power.

As a small reduction of the speed can make a big difference on the energy consumption, and as many fan and pump systems run at less than full capacity a lot of the time, a variable speed drive can make huge savings compared to a motor driving an application under mechanical control. A variable speed drive can also make it possible to stop a motor completely when it is not required as re-starting with a variable speed drive causes far less stress than starting direct on line - soft start is an inherent feature of the drive. Regulating the motor speed has added benefit of easily accommodating capacity rises without extra investment, as speed increases of 5-20% is no problem with an AC variable speed drive as long as there is enough spare capacity in the system.

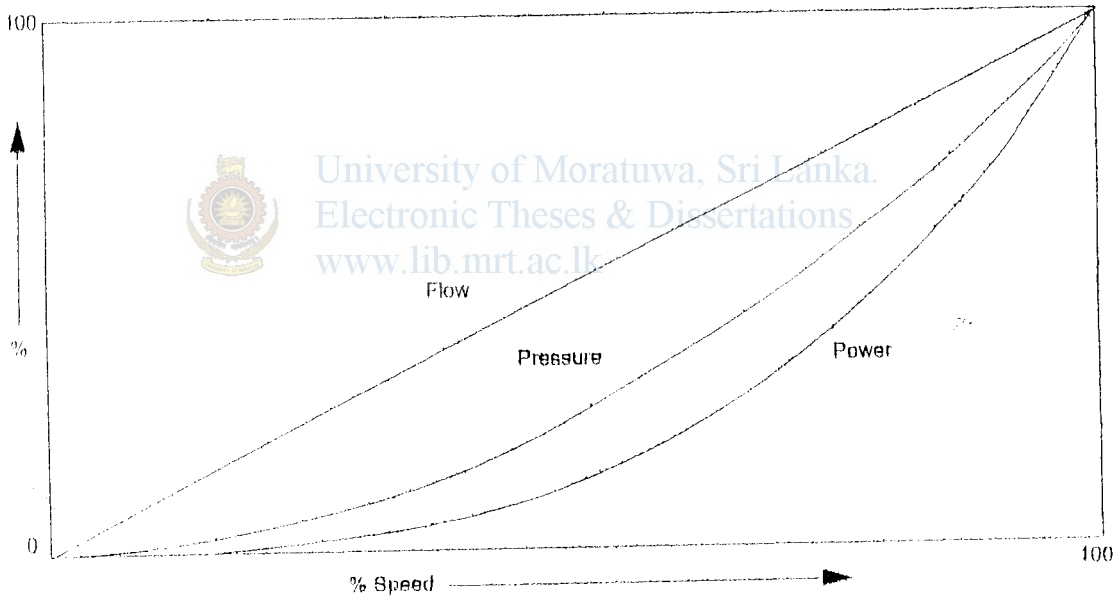


Figure 1.1 - Variable speed drive for Air flow controlling

By matching the performance of the motor to the needs of the process, variable speed drives can give major savings, compared to the wasteful practice of running the motor at full speed against a restriction to modulate output.

CHAPTER 2

Design Consideration of the Variable Speed Controller

2.1 Speed-Torque characteristics of the Induction motor

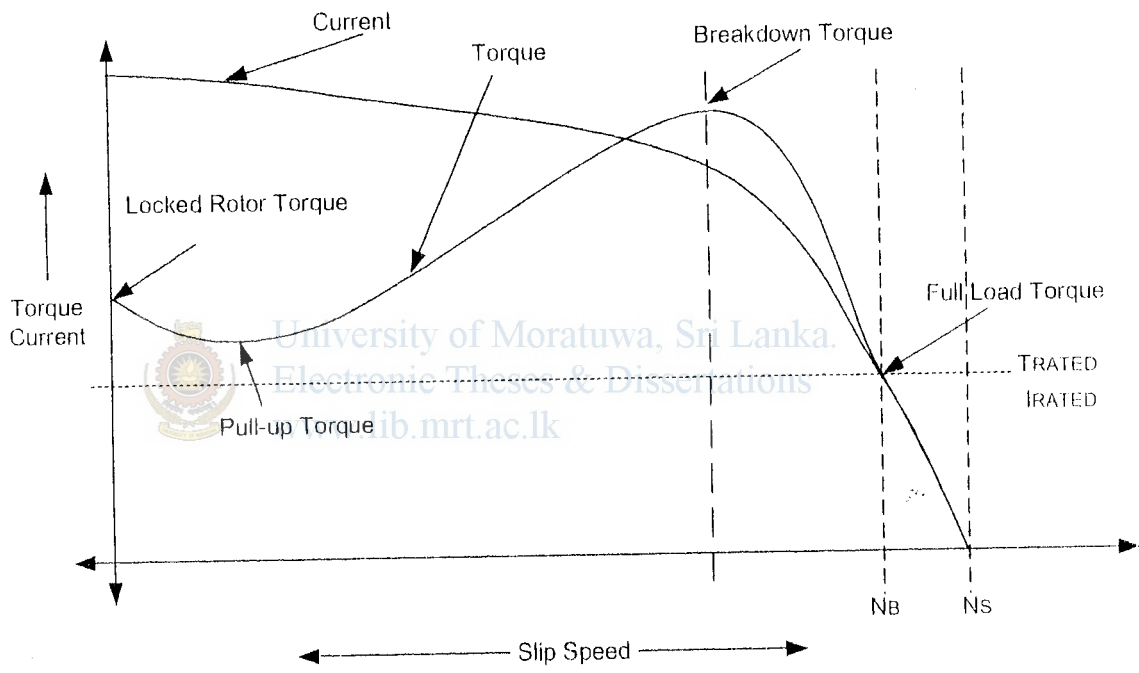


Figure 2.1- Speed-Torque characteristics of the Induction motor

Figure 2.1 shows the typical speed-torque characteristics of an induction motor. The X axis shows speed and slip. The Y axis shows the torque and current. The characteristics are drawn with rated voltage and frequency supplied to the stator. During start-up, the motor typically draws up to seven times the rated current. This high current is a result of stator and

rotor flux, the losses in the stator and rotor windings, and losses in the bearings due to friction. This high starting current overcomes these components and produces the momentum to rotate the rotor. At start-up, the motor delivers 1.5 times the rated torque of the motor. This starting torque is also called locked rotor torque (LRT). As the speed increases, the current drawn by the motor reduces slightly .

The current drops significantly when the motor speed approaches ~80% of the rated speed. At base speed, the motor draws the rated current and delivers the rated torque. At base speed, if the load on the motor shaft is increased beyond its rated torque, the speed starts dropping and slip increases. When the motor is running at approximately 80% of the synchronous speed, the load can increase up to 2.5 times the rated torque. This torque is called breakdown torque. If the load on the motor is increased further, it will not be able to take any further load and the motor will stall. In addition, when the load is increased beyond the rated load, the load current increases following the current characteristic path. Due to this higher current flow in the windings, inherent losses in the windings increase as well. This leads to a higher temperature in the motor windings. Motor windings can withstand different temperatures, based on the class of insulation used in the windings and cooling system used in the motor. Some motor manufacturers provide the data on overload capacity and load over duty cycle. If the motor is overloaded for longer than recommended, then the motor may burn out. As seen in the speed-torque characteristics, torque is highly nonlinear as the speed varies. In many applications, the speed needs to be varied, which makes the torque vary. A simple method of speed control called, Variable Voltage Variable Frequency (VVVF or V/f).

2.2 Load characteristics of the Induction motor

Constant Torque, Variable Speed Loads

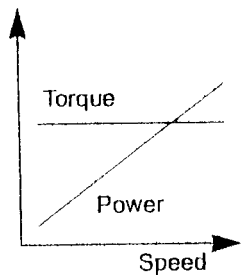


Figure 2.2 - Constant Torque, Variable Speed Loads

The torque required by this type of load is constant regardless of the speed. In contrast, the power is linearly proportional to the speed. Equipments such as screw compressors, conveyors and feeders have this type of characteristic.

Variable Torque, Variable Speed Loads

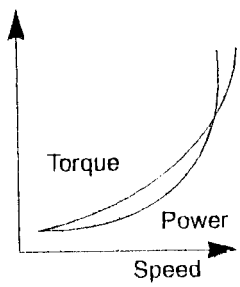


Figure 2.3 - Variable Torque, Variable Speed Loads

This is the most commonly found in the industry and sometimes is known as a quadratic torque load. The torque is proportional to the square of the speed, while the power is proportional to the cube of the speed. This is the typical torque-speed characteristic of a fan or a pump.

Constant Power Loads

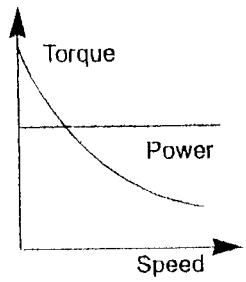


Figure 2.4 – Constant Power Loads

This type of load is rare and sometimes found in the industry. The power remains constant while the torque varies. The torque is inversely proportional to the speed, which theoretically means infinite torque at zero speed and zero torque at infinite speed. In practice, there is always a finite value to the breakaway torque, required. This type of load is characteristic of the traction drives, which require high torque at low speeds for the initial acceleration and then a much reduced torque when at running speed.

Constant Power, Constant Torque Loads

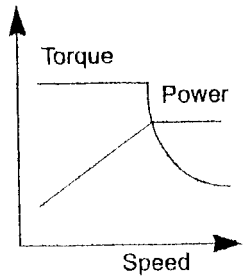


Figure 2.5 - Constant Power, Constant Torque Loads

Constant power , Constant torque load, as speed increases, the torque is constant while the power linearly increasing. When the torque starts to decrease, the power then remains constant.

2.3 V/*f* Control Theory

As the speed-torque characteristics, the induction motor draws the rated current and delivers the rated torque at the base speed. When the load is increased (over-rated load), while running at base speed, the speed drops and the slip increases. As we have seen in the earlier section, the motor can take up to 2.5 times the rated torque with around 20% drop in the speed. Any further increase of load on the shaft can stall the motor.

The torque developed by the motor is direct proportional to the magnetic field produced by the stator. So, the voltage applied to the stator is direct proportional to the product of stator flux and angular velocity. This makes the flux produced by the stator proportional to the ratio of applied voltage and frequency of supply. By varying the frequency, the speed of the motor can be varied. Therefore, by varying the voltage and frequency by the same ratio, flux and hence, the torque can be kept constant throughout the speed range.

Stator Voltage (V) α [Stator Flux(Φ)] x [Angular Velocity (ω)]

$V \alpha \Phi x 2\pi f$

$\Phi \alpha V/f$

This makes constant V/f the most common speed control of an induction motor. Figure 2.6 shows the relation between the voltage and torque versus frequency. Figure 2.6 demonstrates voltage and frequency being increased up to the base speed. At base speed, the voltage and frequency reach the rated values as listed in the nameplate. We can drive the motor beyond base speed by increasing the frequency further. However, the voltage applied cannot be increased beyond the rated voltage. Therefore, only the frequency can be increased, which results in the field weakening and the torque available being reduced. Above base speed, the factors governing torque become complex, since friction and windage losses increase significantly at higher speeds. Hence, the torque curve becomes nonlinear with respect to speed or frequency.

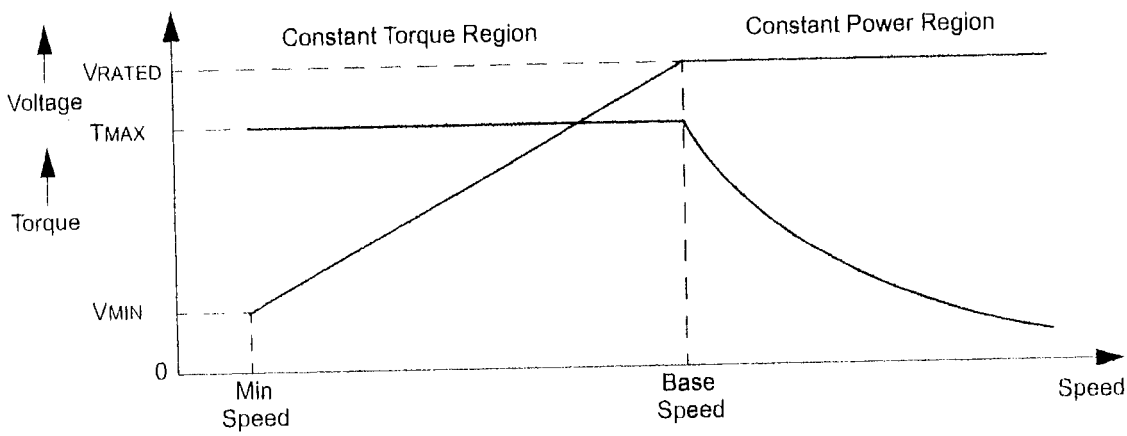


Figure 2.6 - Speed torque characteristics of induction motor

CHAPTER 3

Theoretical Developments

3.1 Principal of Sinusoidal PWM control

In sinusoidal PWM, the firing instants required to synthesize correctly the pulse width modulated wave are determined by comparing a triangular carrier wave and the reference modulation sine wave. The crossover points of the two waves determine the firing instants , as shown in the figure 3.1

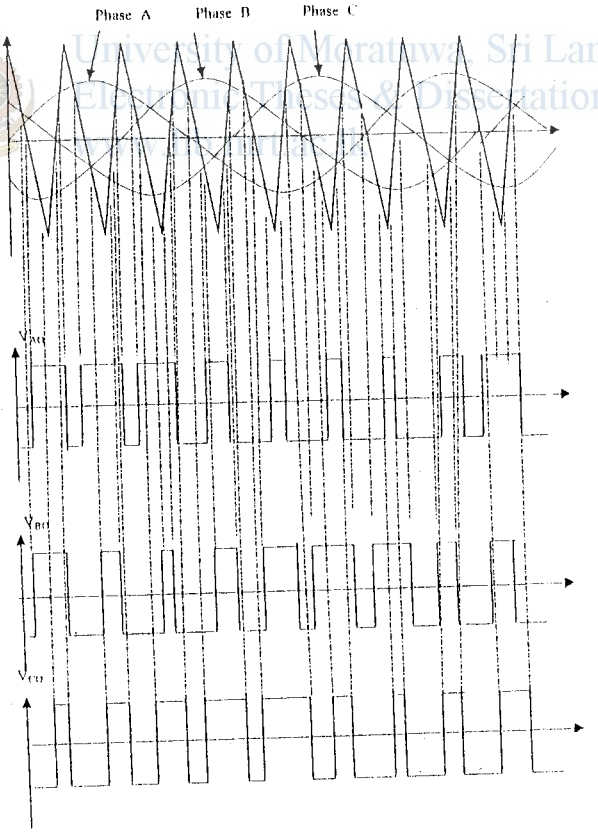


Figure 3.1- Generation of PWM wave

The pulse height of the pulse width modulated output signal is determined by the direct voltage on the supply side of the inverter. Also the pulse pattern is dependent on the ratio of the peak modulating voltage Vm, to the peak carrier voltage Vc. This ratio is called modulation index or modulation ratio, M

M = Vm / Vc =Modulation ratio

M is in its usual range 0<= M <=1

Another property of PWM waveform is the ratio between the frequencies of the carrier and modulating waveforms.

$$P = \frac{\text{Frequency of Carrier wave}}{\text{Frequency of Modulating Wave}}$$

= Carrier ratio

According to these frequencies there are re two methods of sinusoidal PWM.

1. Synchronous operation

   If the frequency of the triangular carrier wave is an integer multiple of the frequency of the modulating sine wave, then the modulation is synchronous.

2. Asynchronous operation

   If the carrier frequency is not a multiple of the modulating waveform frequency, then the modulation is asynchronous.

   Numbers of switching pulses in each half cycle depend on the carrier ratio. Figure 3.1 shows how an increase in carrier ratio changes the number of pulses within each half cycle.

3.2 Inverter Model

The attention is drawn to 3- phase inverter since it is the commonly used inverter in the industry. The 3-phase inverter with its freewheeling diodes can be represented as shown in figure 3.2.

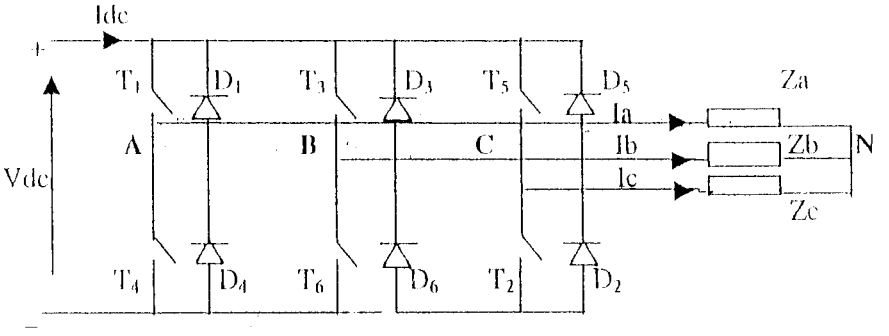


Figure 3.2 – Three phase inverter model with freewheeling diodes

T_1 to T_6 are the switching devices and D_1 to D_6 are the freewheeling diodes (used in order for inductive current to flow after particular switch is turned off) and Z_a , Z_b and Z_c represents the star connected load. The relevant phases are noted as A, B, and C and the neutral point as N.

The current flow is always taken towards the load and hence Ia , Ib and Ic take the direction as shown. For the analysis free wheeling diodes are neglected and the transistor switches are considered as ideal switches. Hence the simple model which we use is shown below;

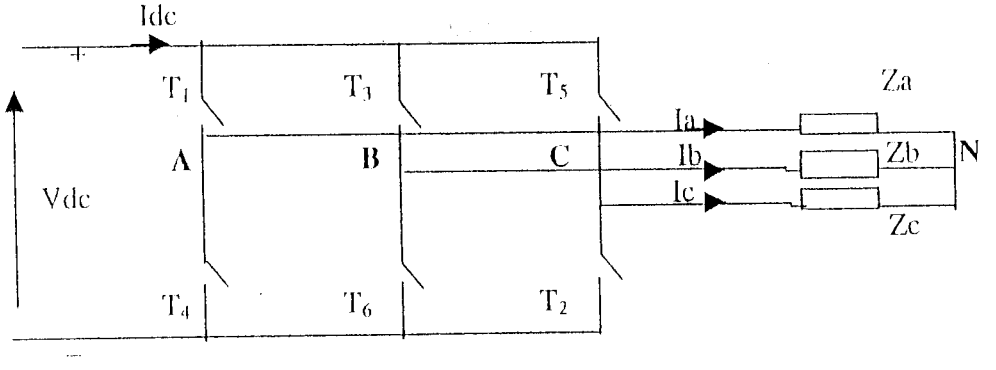


Figure 3.3 – Three phase inverter model without freewheeling diodes

The mid point of the voltage as "0" and hence the dc voltage is divided into two portions of Vdc / 2. The phase voltages are considered as Van, Vbn and Vcn.

3.3 Output Voltages

The output voltages are of three types; namely

1. Pole voltages : Vao , V bo , V co
2. Phase Voltages : Van , Vbn , Vca

These voltages can be obtained by analyzing the inverter model.

By applying Kirchof's voltage law;

$$Van = Vao + Von \quad (1)$$

$$Vba = Vbo + Von \quad (2)$$

$$Vcn = Vco + Von \quad (3)$$

$$(1)+(2)+(3)\ ;\ Van + Vba + Vcn = Vao + Vbo + Vco + 3Von$$

$$= 0 \text{ (because the output is considered as balance)}$$

$$\text{hence } Von = -(1/3)(Vao + Vbo + Vco) \quad (4)$$

By using the 4th equation in (1), (2) and (3);

$$Van = 2/3\ Van - 1/3\ Vbo - 1/3\ Vco \qquad (5)$$

$$Vba = -1/3\ Vao + 2/3\ Vbo - 1/3\ Vco \qquad (6)$$

$$Van = -1/3\ Vao - 1/3\ Vbo + 2/3\ Vco \qquad (7)$$

This can be represented in matrix notation as;

$$\begin{pmatrix} Van \\ Vbn \\ Vcn \end{pmatrix} = 1/3 \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} Vao \\ Vbo \\ Vco \end{pmatrix} \qquad (8)$$

The pole voltages Vao, Vbo and Vco solely depend on the switching state and it can be written as;

$$Vao = \begin{cases} Vdc/2 \text{ when } T_1 \text{ is ON} \\ -Vdc/2 \text{ when } T_4 \text{ is ON} \end{cases} \qquad (9)$$

$$Vbo = \begin{cases} Vdc/2 \text{ when } T_3 \text{ is ON} \\ -Vdc/2 \text{ when } T_6 \text{ is ON} \end{cases} \qquad (10)$$

$$Vco = \begin{cases} Vdc/2 \text{ when } T_5 \text{ is ON} \\ -Vdc/2 \text{ when } T_2 \text{ is ON} \end{cases} \quad (11)$$

The line voltage as usually can be derived using phase voltages and hence can be represented mathematically as;

$Van = Van - Vbn$

$Vbc = Vbn - Vcn$

$Vca = Vcn - Van$

3.4 Micro Controller unit (MCU) Implementation

In the MCU implementation, the software is responsible for continuously updating the PWM duty cycles.

The PWM duty cycle calculations are based on a point in the sine wave. At full modulation (Maximum voltage), 100 percent duty cycle corresponds to the positive peak of the sine wave, and 0 percent duty cycle is equivalent to the negative peak. The zero crossover point is represented by 50 percent duty cycle, as can be observed in figure 3.4

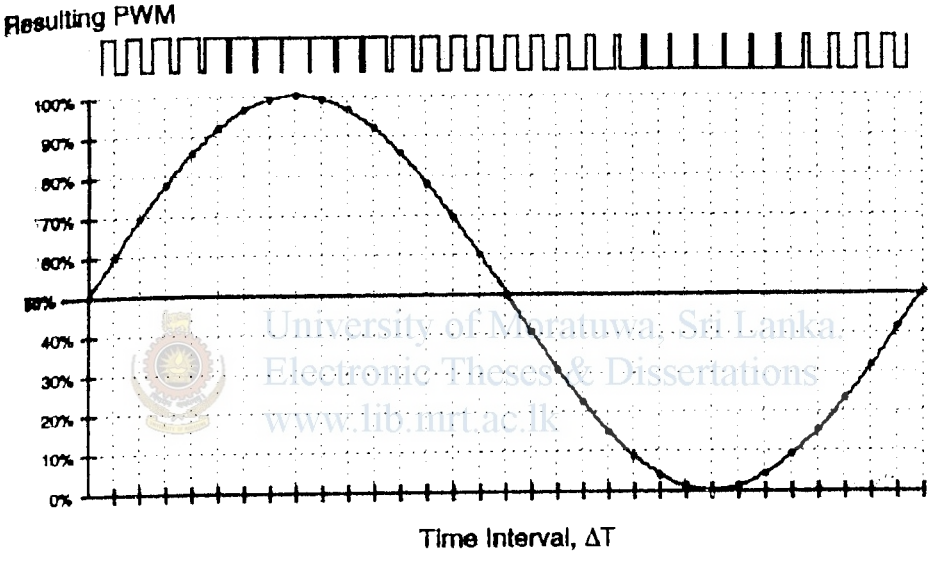


Figure 3.4 – Sinusoidal wave generation using PWM

A HIGH $= m.\ \sin(\Phi).50\% + 50\ \%$

B HIGH $= m.\ \sin(\Phi - \pi/3).50\% + 50\ \%$

C HIGH $= m.\ \sin(\Phi + \pi/3).50\% + 50\ \%$

Where m is the modulation index (0 to 1, 0= zero voltage, 1 = full voltage)

Since two PWM signals are required for each phase, the output duty cycle at the phase winding corresponds with the duty cycle of the top device. As an example, if the desired

output at the phase winding is 75 percent duty cycle, the top PWM would be 75 percent duty cycle and the bottom PWM would be at 25 percent in actual practice, the PWM at the phase winding will be slightly different because of the necessity for dead time.

3.5 Changing output frequency

The PWM duty cycles are normally updated at a periodic rate, and the frequency of the sine wave determined by this up date rate (ΔT) and the number of samples points in a cycle. The relationship is given by

Output frequency = 1 /(update rate X number of samples)

To vary the frequency, either the number of sample points must be changed or the time between updates must be changed. When the number of samples is changed, the update rate is kept constant. The sample points typically come from a table, the number of sample points are easily changed by varying the increment value through the table as given by equation (3.1) The resulting frequency can be calculated as shown in Equation (3.2)

$$\textbf{Number of samples} = \frac{\textbf{Table size}}{\textbf{Increment value}} \qquad (3.1)$$

$$\textbf{Output frequency} = \frac{\textbf{Increment value}}{\textbf{Update rate X table size}} \qquad (3.2)$$

When the update rate is changed, number of samples is kept constant. The update rate is calculated by software, according to the speed set value of controller.

CHAPTER 4

Micro Controller Unit for motor controls

4 .1 MCU Types for Motor Controls

Single chip microcontroller devices are ideal for motor control applications. The advantage of a MCU for motor control design is that the MCU's program can be changed revised by changing a few lines of text in the source code. This source code is then converted into machine code by software compiler or assembler and is programmed into MCU.

The microcontroller is a single-chip computer. It operates as stored program machine; that is , it must read its program code and data values from its memory in order to operate. Two common methods are used to accomplish this. One is called Von Neumann architecture and has been employed in many MCU's . This method uses one data bus and memory space for both program code and data values, saving cost but slowing down the code execution.

The other approach, called Harvard architecture, separates the program code and data values into two memory structures, allowing parallel loading of both at the same time. This technique speeds up execution time but requires more data pins. There are some modified MCU versions of the Harvard architecture that use only one external memory bus but use both program and data buses internally.

The MCU can measure the motor's current , voltage, speed, temperature, and even magnetic flux and can then compute the best operating strategy for the motor. The MCU can allow the motor's speed or torque to be managed with high precision and can protect against or at lest detect motor fault conditions.

4.2 PIC 18F4431 Microcontroller

PIC18F4431 microcontrollers offers high computational performance at an economical price, with the addition of high endurance enhanced Flash program memory and a high speed 10-bit A/D converter. On top of these features, the PIC18F4431 introduces design enhancements that make these microcontrollers a logical choice for many high performance, power control and motor control applications. These special peripherals include:

- 14-bit resolution Power Control PWM Module (PCPWM) with programmable dead time insertion
- Motion Feedback Module (MFM), including a 3-channel Input Capture (IC) Module and Quadrature Encoder Interface (QEI)
- High-speed 10-bit A/D Converter (HSADC)

The PCPWM can generate up to eight complementary PWM outputs with dead-band time insertion. Overdrive current is detected by off-chip analog comparators or the digital fault inputs. The MFM Quadrature Encoder Interface provides precise rotor position feedback and/or velocity measurement.

PIC18F4431 devices also feature Flash program memory and an internal RC oscillator with built-in Low power modes.

4.2.1 Features of 18F4431 Microcontroller

1. Nano Watt technology

The PIC18F4431 microcontroller incorporates a range of features that can significantly reduce power consumption during operation.

Key items include:

- **Alternate Run Modes:**

  By clocking the controller from the Timer1 source or the internal oscillator block, power consumption during code execution can be reduced by as much as 90%.

- **Multiple Idle Modes:**

  The controller can also run with its CPU core disabled, but the peripherals are still active. In these states, power consumption can be reduced even further, to as little as 4% of normal operation requirements.

- **On-the-fly Mode Switching:**

  The power-managed modes are invoked by user code during operation, allowing the user to incorporate power saving ideas into their application's software design.

- **Lower Consumption in Key Modules:**

  The power requirements for both Timer1 and the Watchdog Timer have been reduced by up to 80%, with typical values of 1.1 and 2.1 μA, respectively.

2. Multiple Oscillator options

The PIC18F4431 microcontroller offers nine different oscillator options, allowing users a wide range of choices in developing application hardware. These include:

- Four crystal modes, using crystals or ceramic resonators.
- Two external clock modes, offering the option of using two pins (oscillator input and a divide-by-4 clock output) or one pin (oscillator input, with the second pin reassigned as general I/O).
- Two external RC oscillator modes, with the same pin options as the external clock modes.
- An internal oscillator block, which provides an 8 MHz clock and an INTRC source (approximately 31 kHz, stable over temperature and VDD), as well as a range of 6 user selectable clock frequencies (from 125 kHz to 4 MHz) for a total of 8 clock frequencies.

Besides its availability as a clock source, the internal oscillator block provides a stable reference source that gives the family additional features for robust operation:

- **Fail-Safe Clock Monitor**

This option constantly monitors the main clock source against a reference signal provided by the internal oscillator. If a clock failure occurs, the controller is switched to the internal oscillator block, allowing for continued low speed operation or a safe application shutdown.

• **Two-Speed Start-up**

This option allows the internal oscillator to serve as the clock source from Power-on Reset or wake-up from Sleep mode, until the primary clock source is available. This allows for code execution during what would otherwise be the clock start-up interval, and can even allow an application to perform routine Background activities and return to Sleep without returning to full power operation.

3. Other Special Features

• **Memory Endurance:**

The enhanced Flash cells for both program memory and data EEPROM are rated to last for many thousands of erase/write cycles – up to 100,000 for program memory and 1,000,000 for EEPROM. Data retention without refresh is conservatively estimated to be greater than 100 years.

• **Self-programmability:**

These devices can write to their own program memory spaces under internal software control. By using a bootloader routine located in the protected Boot Block at the top of program memory, it becomes possible to create an application that can update itself in the field.

• **Power Control PWM Module:**

In PWM mode this module provides 1, 2 or 4 modulated outputs for controlling half-bridge and full-bridge drivers. Other features include Auto-Shutdown on fault detection and Auto-Restart to reactivate outputs once the condition has cleared.

• **Enhanced USART:**

This serial communication module is capable of standard RS-232 operation using the internal oscillator block, removing the need for an external crystal (and its accompanying power requirement) in applications that talk to the outside world.

• **High-speed 10-bit A/D Converter:**

This module incorporates Programmable Acquisition Time, allowing for a channel to be selected and a conversion to be initiated without waiting for a sampling period and thus, reducing code overhead.

• **Motion Feedback Module (MFM):**

This module features a Quadrature Encoder Interface (QEI) and an Input Capture (IC) module. The QEI accepts two phase inputs (QEA, QEB) and one index input (INDX) from an incremental encoder. The QEI supports high and low precision position tracking, direction status and change of direction interrupt, and velocity measurement. The input capture features 3 channels of independent input capture with Timer5 as the time base, a special event trigger to other modules, and an adjustable noise filter on each IC input.

• **Extended Watchdog Timer (WDT):**

This enhanced version incorporates a 16-bit prescaler, allowing a time-out range from 4 ms to over 2 minutes, that is stable across operating voltage and temperature.

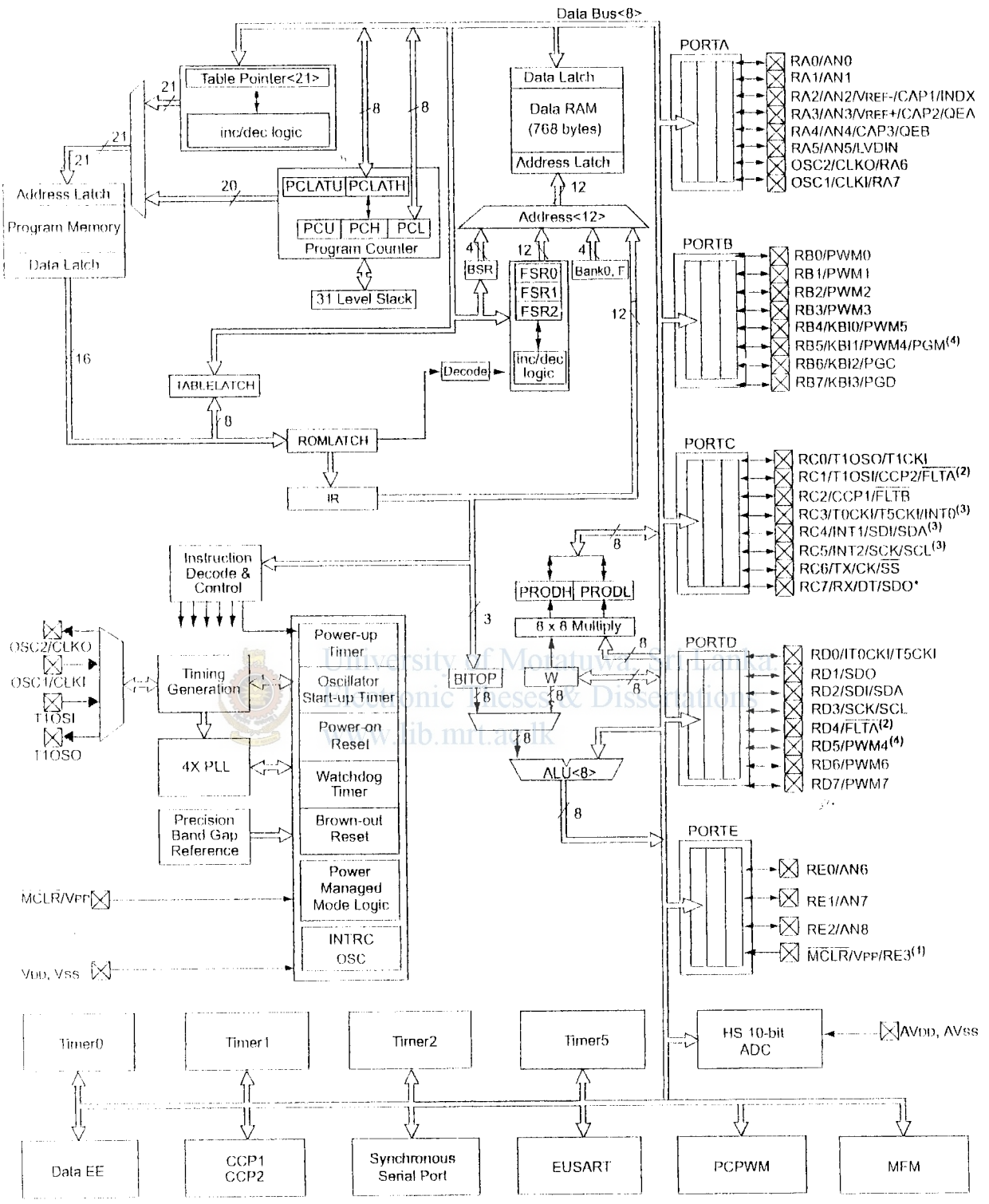


Figure 4.1- Block diagram of PIC 18F4431

4.2.2 Power Control PWM Module

The Power Control PWM module simplifies the task of generating multiple, synchronized pulse width modulated (PWM) outputs for use in the control of motor controllers and power conversion applications. In particular, the following power and motion control applications are supported by the PWM module:

- Three-phase and Single-phase AC Induction Motors
- Switched Reluctance Motors
- Brushless DC (BLDC) Motors
- Uninterruptible Power Supplies (UPS)
- Multiple DC Brush Motors

The PWM module has the following features:

- Up to eight PWM I/O pins with four duty cycle generators. Pins can be paired to get a complete half-bridge control.
- Up to 14-bit resolution, depending upon the PWM period.
- "On-the-fly" PWM frequency changes.
- Edge- and Center-aligned Output modes.
- Single-pulse Generation mode.
- Programmable dead time control between paired PWMs.
- Interrupt support for asymmetrical updates in Center-aligned mode.

- Output overrides for Electrically Commutated Motor (ECM) operation; for example, Brushless DC motors.
- Special Event comparator for scheduling other peripheral events.
- PWM outputs disable feature sets PWM outputs to their inactive state when in Debug mode.

The Power Control PWM module supports four PWM generators and eight output channels on PIC18F4431 device. A simplified block diagram of the module is shown in Figure 4.2 shows how the module hardware is configured for each PWM output pair for the complementary output mode.

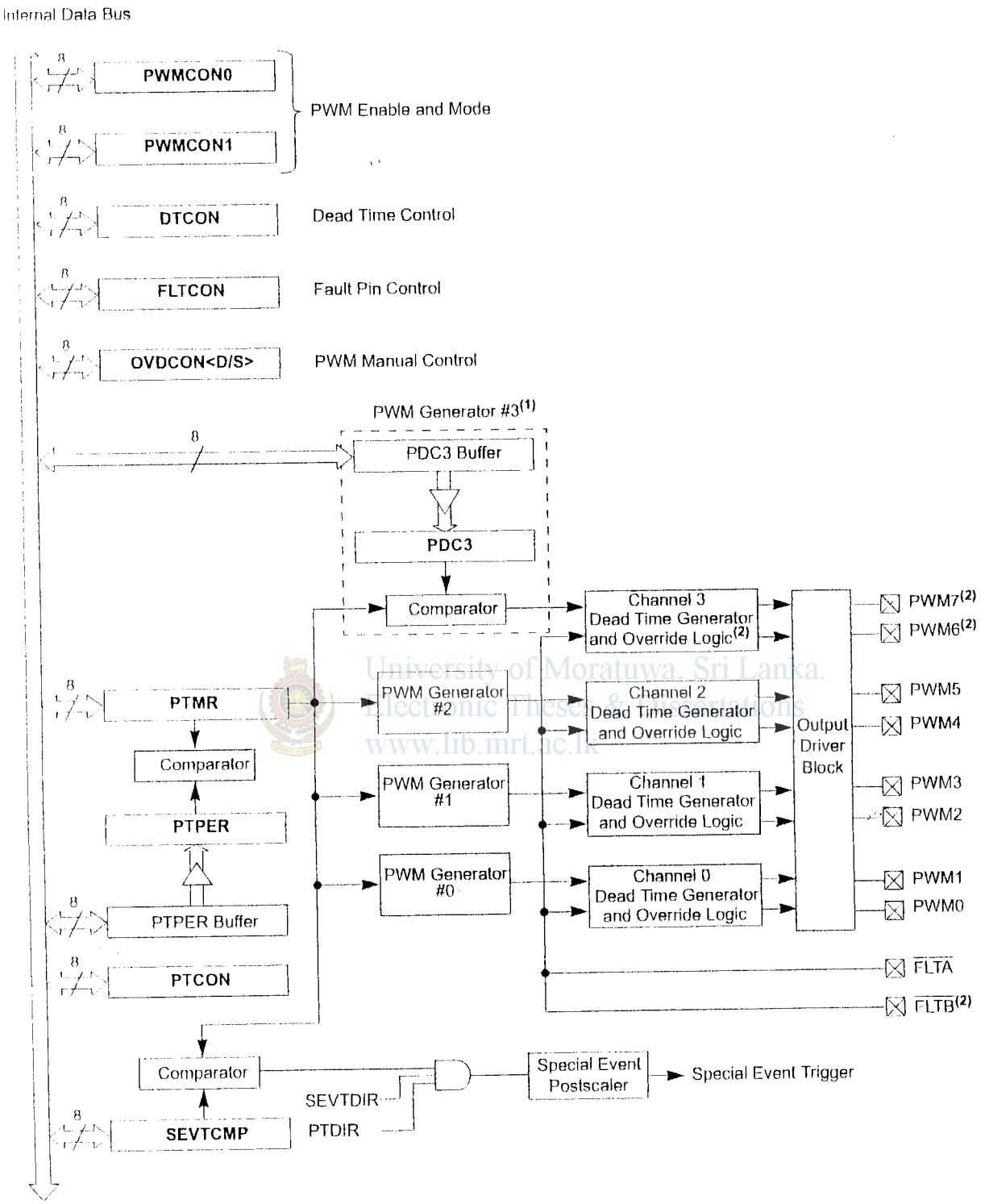


Figure 4.2 - Block diagram of Power Control PWM module

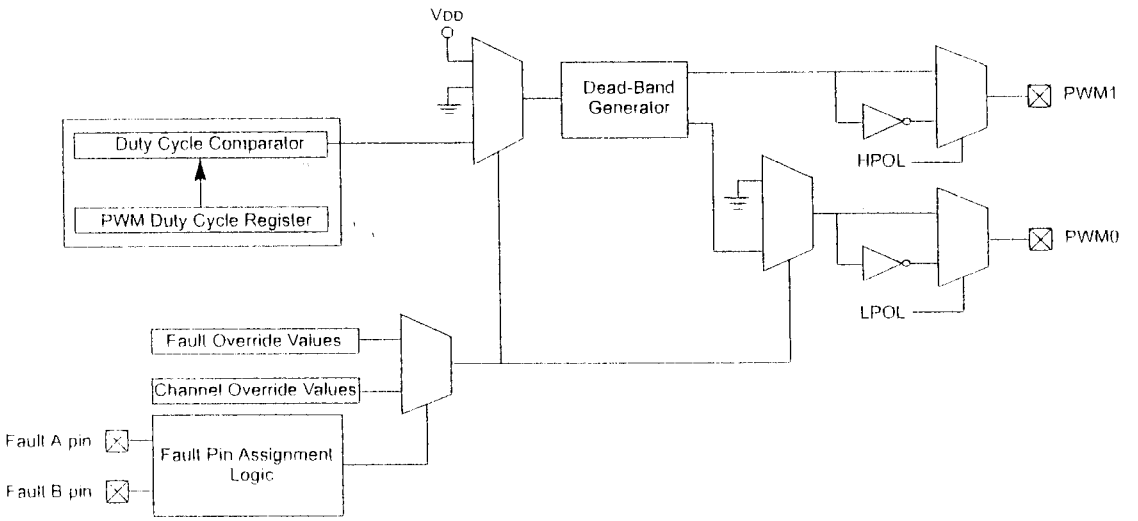


Figure 4.3 - PWM module block diagram, one output pair, Complementary

4.2.3 Control Registers

The operation of the PWM module is controlled by a total of 22 registers. Eight of these are used to configure the features of the module:

- PWM Timer Control register 0 (PTCON0)
- PWM Timer Control register 1 (PTCON1)
- PWM Control register 0 (PWMCON0)
- PWM Control register 1 (PWMCON1)
- Dead Time Control register (DTCON)
- Output Override Control register (OVDCOND)
- Output State register (OVDCONS)
- Fault Configuration register (FLTCONFIG)

There are also 14 registers that are configured as seven register pairs of 16 bits. These are used for the configuration values of specific features. They are:

- PWM Time Base Registers (PTMRH and PTMRL)
- PWM Period Registers (PTPERH and PTPERL)
- PWM Special Event Compare Registers (SEVTCMPH and SEVTCMPL)
- PWM Duty Cycle #0 Registers (PDC0H and PDC0L)
- PWM Duty Cycle #1 Registers (PDC1H and PDC1L)
- PWM Duty Cycle #2 Registers (PDC2H and PDC2L)
- PWM Duty Cycle #3 registers (PDC3H and PDC3L)

All of these register pairs are double-buffered.

4.2.4 Module Functionality

The PWM module supports several modes of operation that are beneficial for specific power and motor control applications. Each mode of operation is described in subsequent sections. The PWM module is composed of several functional blocks. The operation of each is explained separately in relation to the several modes of operation:

- PWM Time Base

- PWM Time Base Interrupts
- PWM Period
- PWM Duty Cycle
- Dead Time Generators
- PWM Output Overrides
- PWM Fault Inputs
- PWM Special Event Trigger

4.2.5 PWM Time Base

The PWM time base is provided by a 12-bit timer with prescaler and postscaler functions. A simplified block diagram of the PWM time base is shown in Fig 4.4. The PWM time base is configured through the PTCON0 and PTCON1 registers. The time base is enabled or disabled by respectively setting or clearing the PTEN bit in the PTCON1 register.

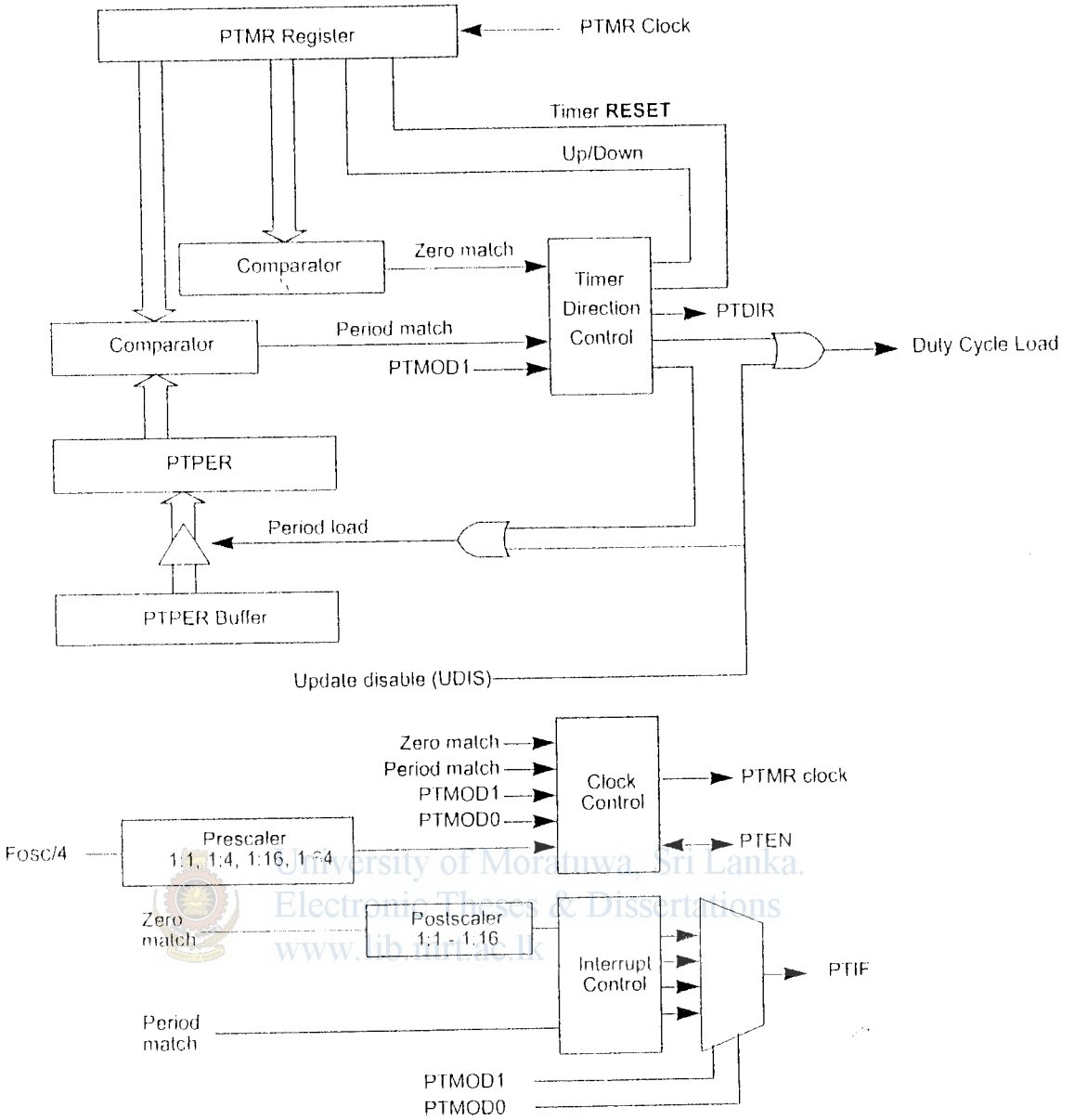


Figure 4.4 - PWM Time base block diagram

The PWM time base can be configured for four different modes of operation:

- Free Running mode
- Single-shot mode
- Continuous Up/Down Count mode
- Continuous Up/Down Count mode with interrupts for double updates

These four modes are selected by the PTMOD1:PTMOD0 bits in the PTCON0 register. The Free Running mode produces edge-aligned PWM generation. The up/down counting modes produce center-aligned PWM generation. The Single-shot mode allows the PWM module to support pulse control of certain electronically commutated motors (ECMs) and produces edge-aligned operation.

4.2.6 PWM Period

The PWM period is defined by the PTPER register pair (PTPERL and PTPERH). The PWM period has 12-bit resolution by combining 4 LSBs of PTPERH and 8-bits of PTPERL. PTPER is a double-buffered register used to set the counting period for the PWM time base. The PTPER buffer contents are loaded into the PTPER register at the following times:

- Free Running and Single-shot modes: when the PTMR register is reset to zero after a match with the PTPER register.
- Up/Down Counting modes: When the PTMR register is zero. The value held in the PTPER buffer is automatically loaded into the PTPER register when the PWM time base is disabled (PTEN = 0).

The PWM period can be calculated from the following formulas:

PWM period for free running mode

$$Tpwm = \frac{(PTPER + 1)}{Fosc/(PTMRps/4)}$$

or

$$Tpwm = \frac{(PTPER + 1) \times PTMRps}{Fosc/4}$$

PWM period for free Up/Down counting mode

$$TPWM = \frac{(2 \times PTPER)}{Fosc/(PTMRPS/4)}$$

The PWM frequency is the inverse of period; or

$$\text{PWM frequency} = \frac{1}{\text{PWM period}}$$

4.2.7 PWM Duty Cycle

PWM duty cycle is defined by PDCx (PDCxL and PDCxH) registers. There are a total of 4 PWM Duty Cycle registers for 4 pairs of PWM channels. The Duty Cycle registers have 14-bit resolution by combining 6 LSbs of PDCxH with the 8 bits of PDCxL. PDCx is a double-buffered register used to set the counting period for the PWM time base.

4.2.8 PWM Duty Cycle Registers

There are four 14-bit special function registers used to specify duty cycle values for the PWM module:

- PDC0 (PDC0L and PDC0H)
- PDC1 (PDC1L and PDC1H)
- PDC2 (PDC2L and PDC2H)
- PDC3 (PDC3L and PDC3H)

The value in each Duty Cycle register determines the amount of time that the PWM output is in the active state. The upper 12 bits of PDCn hold the actual duty cycle value from PTMRH/L<11:0>, while the lower 2 bits control which internal Q-clock the duty cycle match occurs.

In Edge-aligned mode, the PWM period starts at Q1 and ends when the Duty Cycle register matches the PTMR register as follows. The duty cycle match is considered when the upper 12 bits of the PDC is equal to the PTMR.

4.2.9 Edge-Aligned PWM

Edge-aligned PWM signals are produced by the module when the PWM time base is in the Free Running mode or the Single-shot mode. For edge-aligned PWM outputs, the output for a given PWM channel has a period specified by the value loaded in PTPER and a duty cycle specified by the appropriate Duty Cycle register (see Figure 4.5). The PWM output is driven active at the beginning of the period (PTMR = 0) and is driven inactive when the value in the Duty Cycle register matches PTMR. A new cycle is started when PTMR matches the PTPER as explained in the PWM period section. If the value in a particular Duty Cycle register is zero, then the output on the corresponding PWM pin will be inactive for the entire PWM period. In addition, the output on the PWM pin will be active for the entire PWM period if the value in the Duty Cycle register is greater than the value held in the PTPER register.

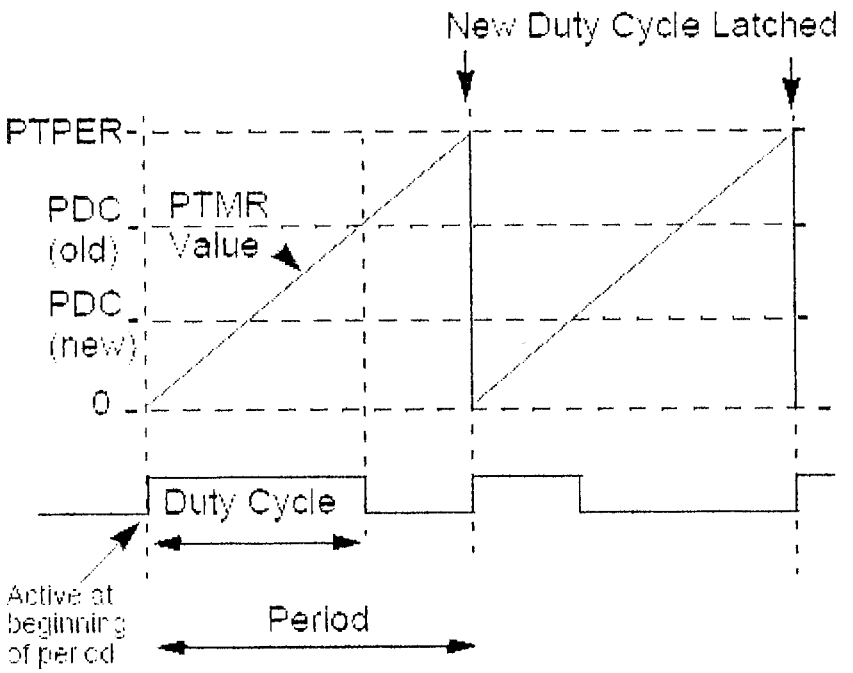


Figure 4.5 – Generation of PWM waveform

4.2.10 Complementary PWM Operation

The Complementary mode of PWM operation is useful to drive one or more power switches in half-bridge configuration as shown in Figure 4.6. This inverter topology is typical for a 3-phase induction motor, brushless DC motor or a three-phase Uninterruptible Power Supply (UPS) control applications. Each upper/lower power switch pair is fed by a complementary PWM signal. Dead time may be optionally inserted during device switching where both outputs are inactive for a short period. In Complementary mode, the duty cycle comparison units are assigned to the PWM outputs as follows:

- PDC0 register controls PWM1/PWM0 outputs
- PDC1 register controls PWM3/PWM2 outputs
- PDC2 register controls PWM5/PWM4 outputs
- PDC3 register controls PWM7/PWM6 outputs

PWM1/3/5/7 are the main PWMs that are controlled by the PDC registers and PWM0/2/4/6 are the complemented outputs. When using the PWMs to control the half bridge, the odd number PWMs can be used to control the upper power switch and the even numbered PWMs for the lower switches.

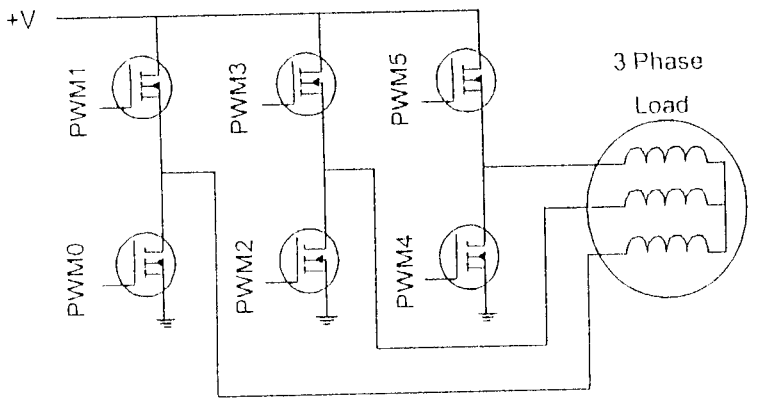


Figure 4.6 - Typical Load for Complementary PWM outputs

The Complementary mode is selected for each PWM I/O pin pair by clearing the appropriate PMODx bit in the PWMCON0 register. The PWM I/O pins are set to Complementary mode by default upon all kinds of device resets.

4.2.11 Dead Time Generators

In power inverter applications where the PWMs are used in Complementary mode to control the upper and lower switches of a half-bridge, a dead time insertion is highly recommended. The dead time insertion keeps both outputs in inactive state for a brief time. This avoids any overlap in the switching during the state change of the power devices due to TON and TOFF characteristics.

Because the power output devices cannot switch instantaneously, some amount of time must be provided between the turn-off event of one PWM output in a complementary pair and the turn-on event of the other transistor. The PWM module allows dead time to be programmed.

Dead Time Insertion

Each complementary output pair for the PWM module has a 6-bit down counter used to produce the dead time insertion. As shown in Figure 4.7, each dead time unit has a rising and falling edge detector connected to the duty cycle comparison output. The dead time is loaded into the timer on the detected PWM edge event. Depending on whether the edge is rising or falling, one of the transitions on the complementary outputs is delayed until the timer counts down to zero. A timing diagram indicating the dead time insertion for one pair of PWM outputs is shown in Figure 4.8.

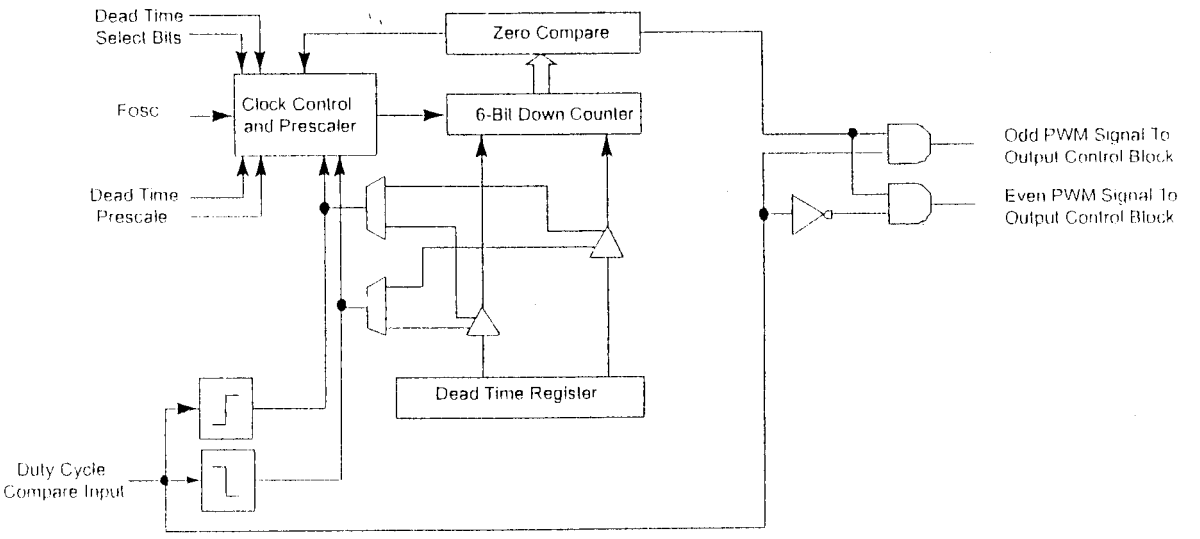


Figure 4.7 - Dead Time control unit block diagram for one PWM output pair

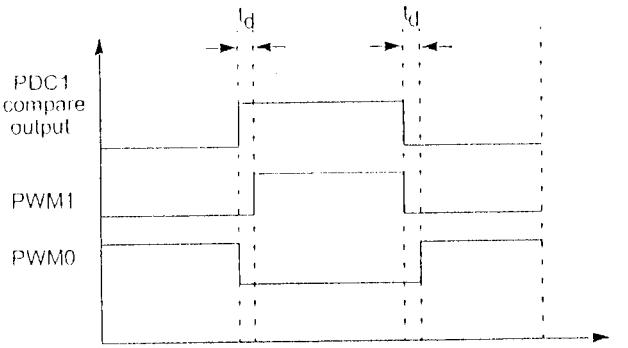


Figure 4.8 - Dead time insertion for complementary PWM

4.3 Control Strategy

4.3.1 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 4.9. 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.

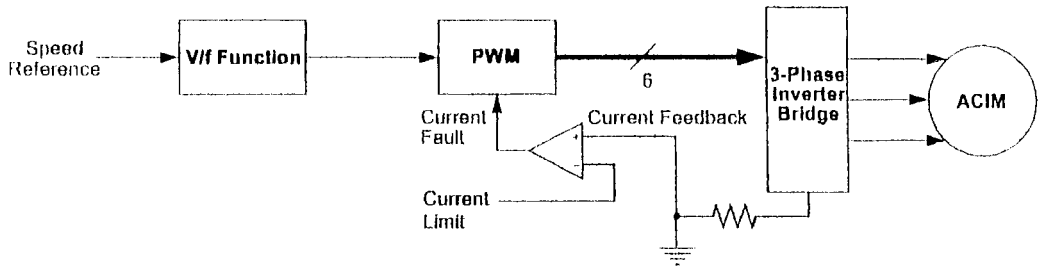


Figure 4.9 – Block diagram of VSD with current feed back

4.3.2 V/f Control of blower motor with Air temperature Feedback

Block diagram of closed-loop speed control of air blower is illustrated in Figure 4.10. The reference temperature is set by a potentiometer. This signal is used as reference value for the ADC module of microcontroller. Air temperature feed back signal is generated by temperature sensor and it is used as input to ADC module.

The temperature error signal is then used as an input to a proportional 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.

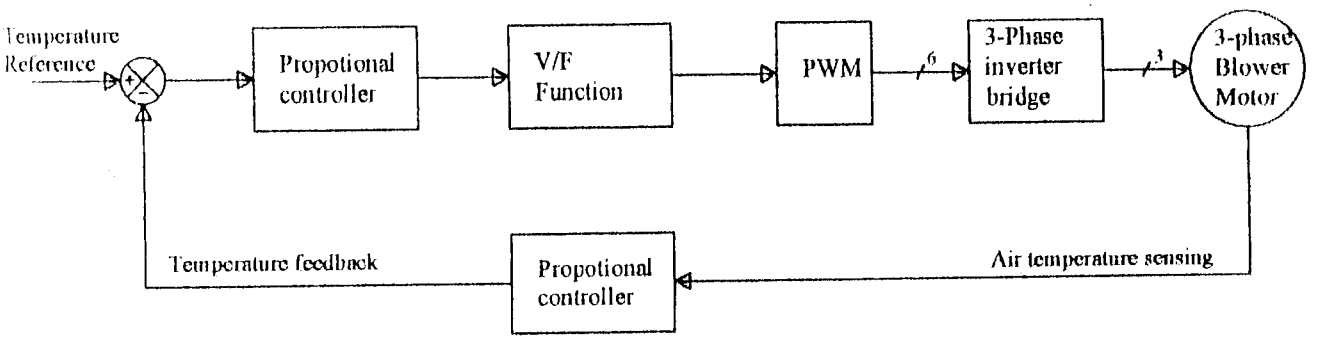


Figure 4.10 – Block diagram for closed loop Air blower speed control

4.4 Initializing The Power Control PWM Module And ADC Module

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. 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 (FOSC/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.

3. Configure the PCPWM output.

   a) Enable PWM0 through PWM7 as outputs.

   b) Set the PWM I/O pairs (PWM0/1, 2/3 and 6/7) 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 FOSC/2 as the dead-time prescaler.

   b) Load DTCON<5:0> with a dead-time value to achieve a 2 µs 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.

4.5 Initializing the ADC module

One analog values is measured in this application:

• AN1 (LM 35DZ temperature sensor input for the speed reference)

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 single-Channel mode.

c) Assign VREF+ and VREF-.

d) Enable the FIFO buffer.

e) Select the left-justified format for the A/D result.

f) Set the A/D acquisition time to 12 TAD (required for sequential conversion).

g) Set the A/D conversion clock to FOSC/32

h) Turn on the ADC.

4.6 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.

In this application, the potentiometer determines the target motor speed reference signal.

4.7 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:

1. When a Timer0 interrupt occurs, an updated target drive frequency is determined by F =(ADRESH/4)

2. The sine value for each phase is read from the sine table, pointed to by the offset value for that phase.

3. 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.

4. The offset values are updated for the next table access.

5. The Timer0 reload value is calculated based on the updated motor drive frequency , where f is the drive frequency. In this program , the number of sine table entries is set at 19. The reload value determines the value at which the PWM duty cycle is updated.

6. 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.

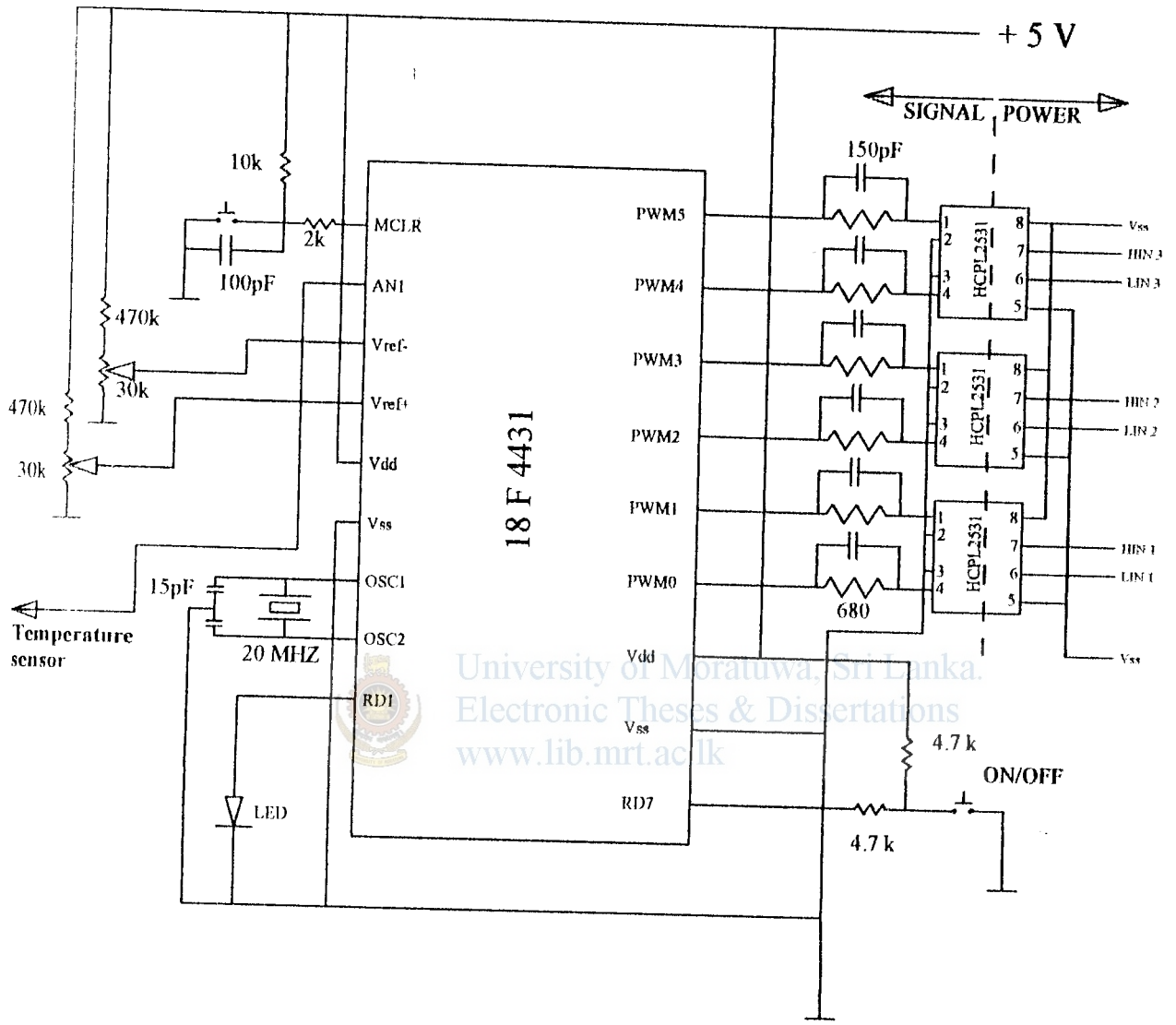


Figure 4.11 - Circuit diagram of Micro Controller

CHAPTER 5

Power Modules for Induction Motor Control

5.1 Motor Drive Requirements

Three-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 . 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 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 these cases, software program develops necessary dead time between the complimentary channels. Power modules are available in wide variety of configurations that accommodate most of the common motor drive topologies.

5.2 Six-Packs Configuration

A common circuit configuration that is used for three-phase induction motor is Six-pack. Six-pack is a three-phase bridge arrangement, which consists of six transistors. The collector of upper transistors are all connected together to a common positive bus terminal, the emitters of lower transistors are connected to a common negative terminal. The positive and negative terminals are connected to a DC source. The three output terminals are connected to a three-phase motor.

A six-pack module could consist of IGBTs, MOSFETs, bipolar transistors, or any other type of power switching transistors. Large high voltage motor often use an IGBT six-pack In PWM motor drive designs IGBTS are best suited for applications that require transistors rated at 400 V or higher, while MOSFETs are most effective for applications that require transistors rated less than 100 V .

Six- pack configuration is nearly universal in its suitability for different three-phase motor drive systems. High power six-packs often have a separate emitter terminal for the gate drive connection, as shown in figure 5.1. These terminals are often called an emitter-Kelvin terminals. The effect of emitter inductance can be minimized by making use of this terminal. The large currents switched by the power devices can create large voltage transients in the stray emitter inductance. The Kelvin terminals are not intended to be used for power connections, their internal wire bonds are not designed to handle high currents.

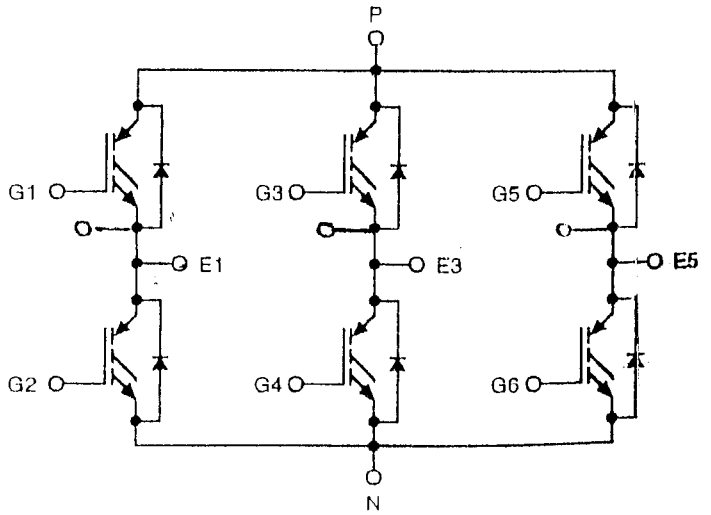


Figure 5.1 – Six-pack IGBT module with emitter Kelvins.

5.3 Gate Drive requirements

IGBTs commonly used in motor drives, UPS and converters operating at dc bus voltages up to 600VDC require voltage drive in order to achieve a saturated "ON" state condition. The drive signal must have the following characteristics

- An amplitude of 10V to 15V.
- A low source resistance for rapid charge and discharge of the gate capacitance.
- A floating output so that high side switches can be driven.

In addition to the above requirements the actual driver should be capable of driving combinations of devices in both low-side and high-side switch configurations. With this in mind the driver should also provide the following:

- Low internal power loss at high switching frequency and maximum offset voltage.
- Accept ground referenced logic level input signals.

- Protect the power switch from damage by clamping the gate signal to the low state in the event of gate under voltage or over voltage or if the load current exceeds a predetermined peak value.
- Protect the power switch by clamping the signal to the low state if the signal inputs are disconnected.

Traditionally the functions described above have required discrete circuits of some complexity but International Rectifier's IR2130 six-channel gate driver perform all the requirements for interfacing logic level control circuits to high power IGBTs in high-side/ low-side switch configurations using up to six devices.

5.3.1 IR2130 Block Diagram

As shown in Figure 5.2 the gate driver consists of six output drivers which receive their inputs from the three input signal generator blocks each providing two outputs. The three low-side output drivers are driven directly from the signal generators L1, L2 and L3 but the high-side drive signals H1, H2 and H3 must be level shifted before being applied to the high-side output drivers.

An under voltage detector circuit monitoring the VCC level provides an input to inhibit the six outputs of the signal generator circuits. In addition, there are individual under voltage lockout circuits for the high-side outputs should any of the floating bias supplies fall below a predetermined level.

The ITRIP signal which can be derived from a current sensor in the main power circuit of the equipment (current transformer, viewing resistor, etc.) is compared with a 0.5volt reference and is then "OR-ed" with the UV signal to inhibit the six signal generator outputs. A fault logic circuit set by the UV or ITRIP inputs provides an open drain TTL output for system indication or diagnostics. There is also an internal current amplifier in the IR2130 and IR2132 that provides an analog signal proportional to the voltage difference between VSS and VS0. Thus, a viewing resistor in the main power circuit can provide a positive voltage at VS0 and by suitable feedback resistors the current amplifier can be scaled to generate 0-5Vdc as a function of actual load current.

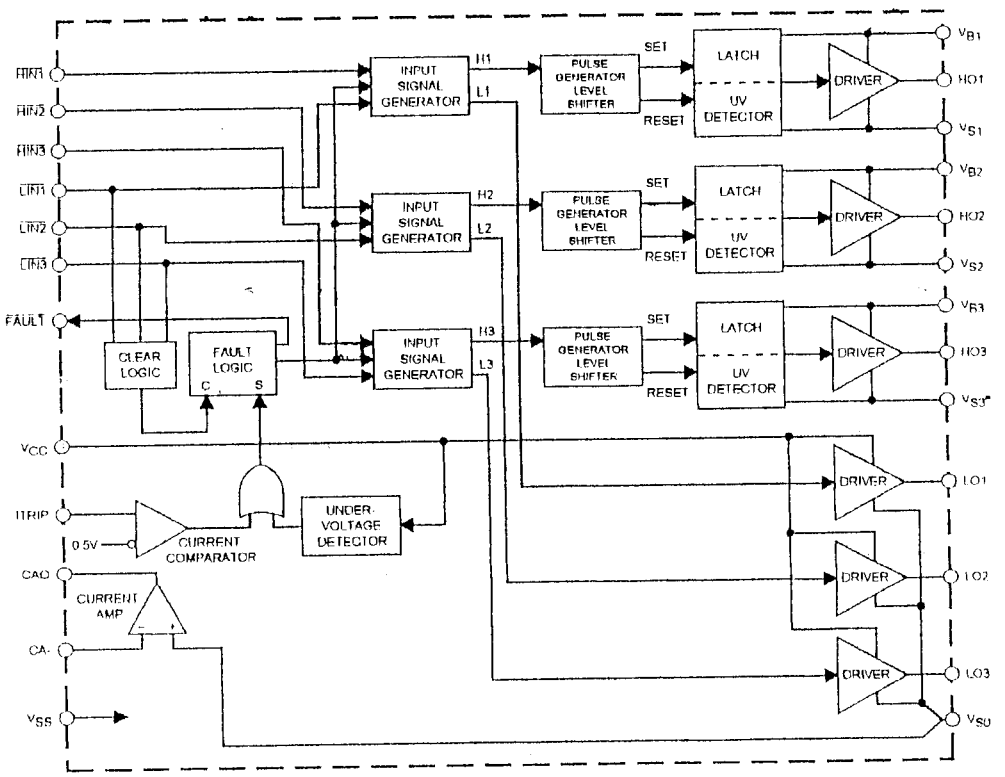


Figure 5.2 - Functional block diagram of the IR 2130

5.3.2 Protection Circuits and Fault Reporting

- **Under voltage Protection**

An under voltage condition on the VCC level, defined as less than 8.9V (as VCC is reduced) and less than 9.3V nominal (as VCC is increased) causes all outputs to shutdown With VCC at around 9 volts the drivers provide marginally adequate drive voltages to ensure full enhancement of the power switches for most applications. Separate UV lockout circuits are provided on the three high-side outputs. They also have a 0.4V hysteresis band with levels of 8.3 volts for a falling bias voltage and 8.7 volts for a rising voltage. Unlike the VCC UV circuit they inhibit only their particular high-side output and do not affect the operation of any other output.

- **Current Trip**

In the event of a shoot-through current or an output overload it is desirable to terminate all the output signals from the driver. This is accomplished through a current comparator circuit which monitors the voltage drop across a low side viewing resistor and compares it with a 0.5 volt reference level. The current comparator output is "OR-ed" with the VCC under voltage circuit output so that a fault condition of either type causes the fault logic circuit to actuate.

- **Fault Logic**

This circuit consists of a latch which is set by the conditions described incurrent trip and is reset by holding all three low-side inputs high for more than 10 microseconds or by

recycling the VCC bias supply. When the fault latch is set it produces two output signals. One is used to inhibit all three input signal generator circuits thus inhibiting all six outputs. The other output signal appears as a fault indicator which goes low in the presence of a fault condition. The active low condition can drive an LED fault indicator or external logic circuit.

- **Current Sensing in IR2130**

Using the same current viewing resistor the current sense voltage of 0-0.5V is amplified in the current amplifier to generate a 0-5V analog function for processing in an external control circuit. In actual operation the voltage difference between the Vso and Vss pins forms the input voltage for the non inverting amplifier although only the positive current is measured. Two resistors Rf and RIN set the gain of the amplifier as shown in Figure 5.3. Actual voltage gain is given by the relationship

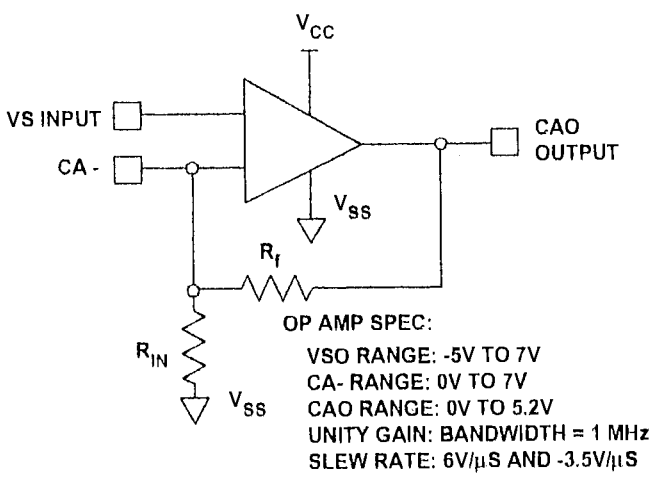


Figure 5.3 - Block diagram of the Current Sensing amplifier

$$A = \frac{R_f + R_{IN}}{R_{IN}}$$

for a gain of 10 with $R_{IN} = 1k$

$$10 = \frac{R_f + 1\,k}{1k}$$

$$R_f + 1\,k = 10\,k$$

$$R_f = 9\,k$$

Power for the current amplifier is supplied from VCC.

5.4 Heat Sink Calculation for IGBT module

Selection Details of Heat Sink

Thermal resistance of IGBT (TRG4 PC 30 KD)

Junction to Case thermal resistance of IGBT ($R_{\theta jc}$) =1.2 °C/W

Case to Sink, flat, greased surface thermal resistance ($R_{\theta cs}$) =0.24 °C/W

Thermal resistance of Mica sheet + Thermal compound = 1.6 °C/W

Switching Loss of IGBT (@ I=8A, Tj =120 °C) = 0.5 mJ

Total switching loss at 5 kHz $= 0.5 \times 10^{-3} \times 5 \times 10^{3}$

$= 2.5$ W

Total heat generated by 8 nos. of IGBTs (Pd) = 20 W

By Considering, surface Area of Heat Sink = A m^2

Thermal resistance of heat sink (R_{0sa}) =0.08/A

Maximum operating temperature of junction (Tjm)=150 °C

Maximum ambient temperature (Ta) = 30 °C

$$(Tjm - Ta) = (R_{0jc} + R_{0cs} + R_{0sa}) \times Pd$$

$$(150 - 30) = (1.2 + 0.24 + 0.125/A) \times 20$$

$$A = 0.0274 \ m^2 = 274.00 \ cm^2$$

Therefore Heat sink area should be more than the 274.00 cm^2

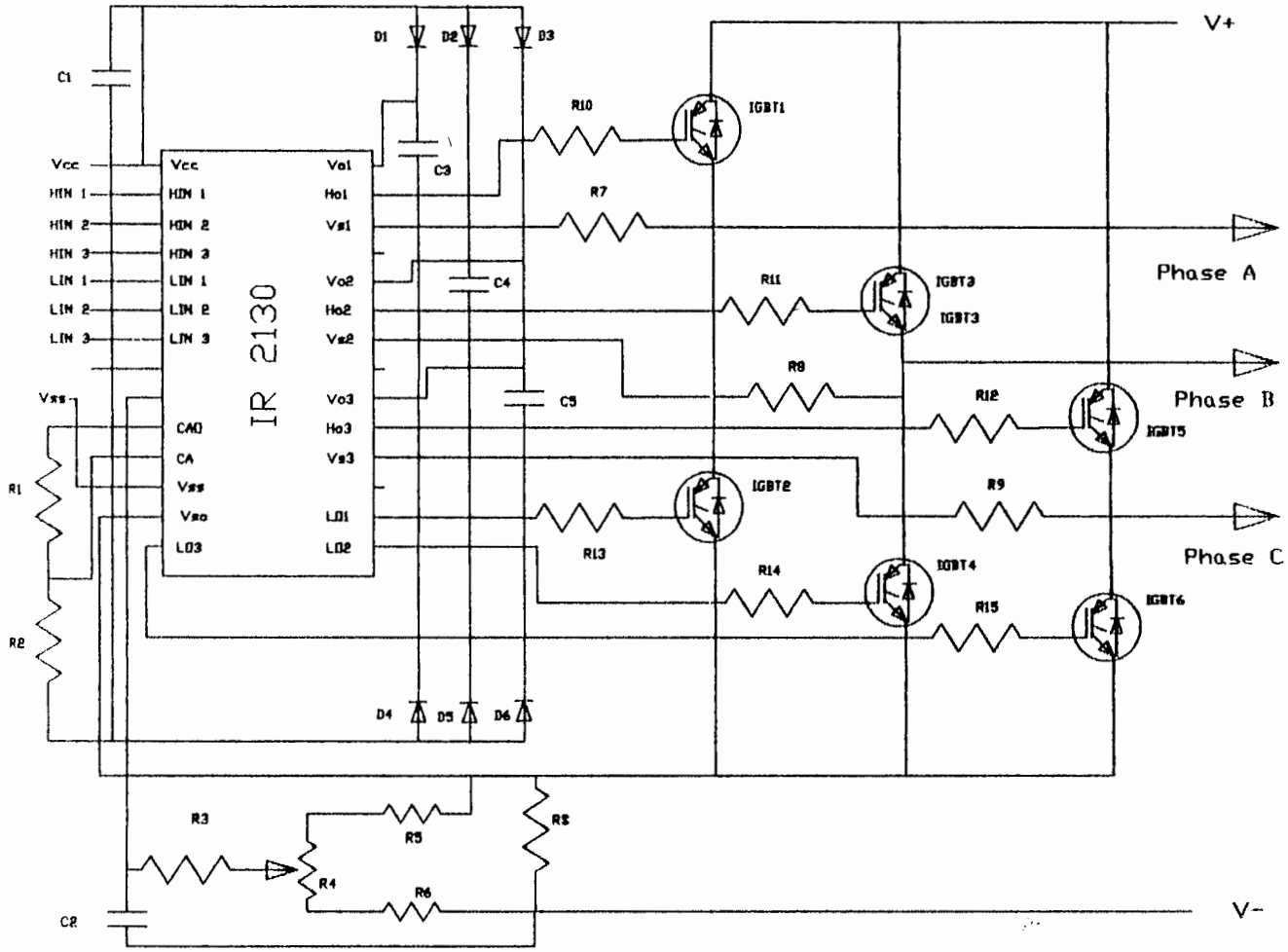
Selected Heat sink has area of 667 cm^2

Then operating junction temperature Tjo

$$(Tjo - Ta) = (R_{0jc} + R_{0cs} + R_{0sa}) \times Pd$$

$$(Tjo - 30) = (1.2 + 0.24 + 0.125/0.0667) \times 20$$

$$Tjo = 96.28 \ °C$$



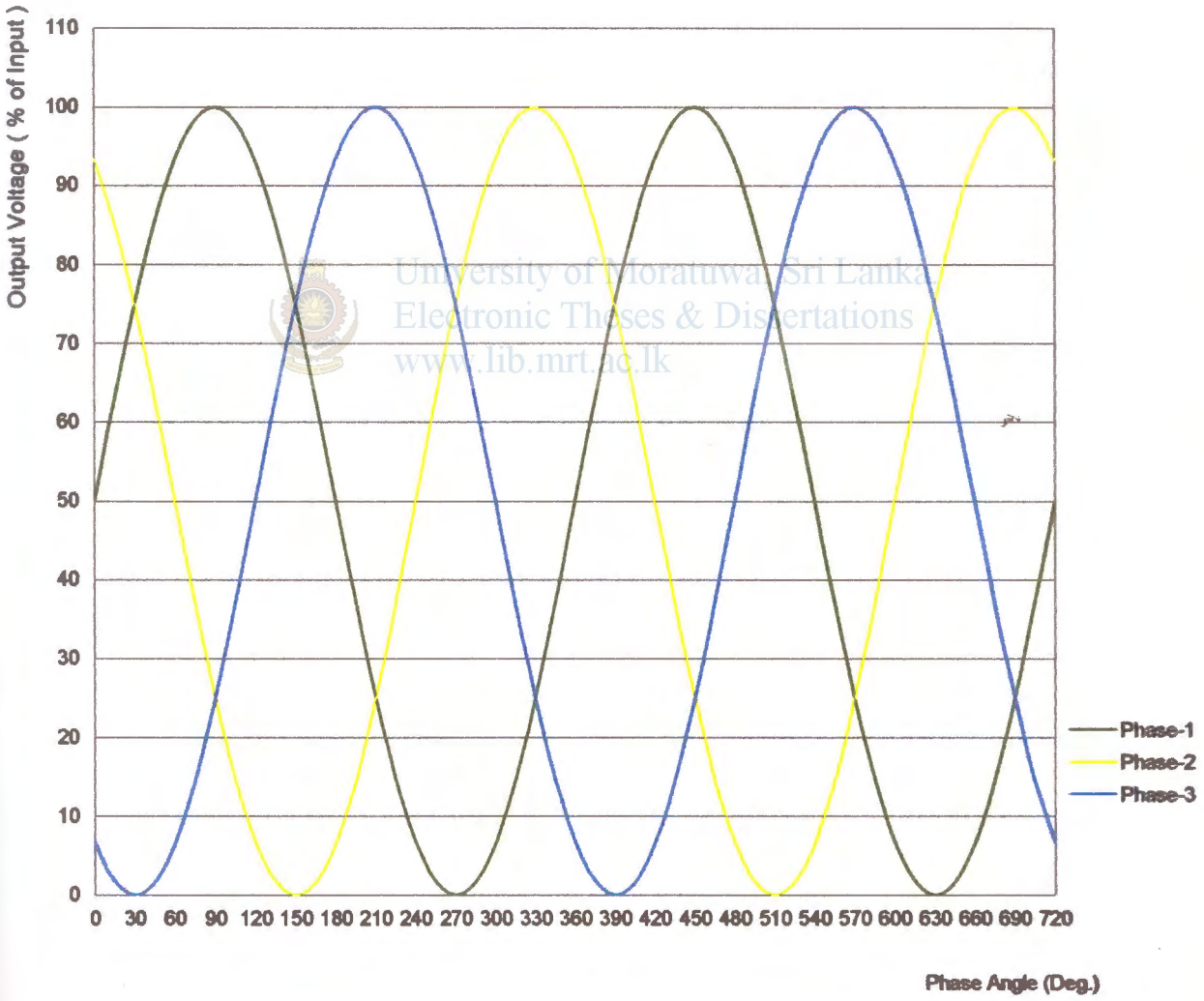
IGBT1-6 –IRG4PC30KD
R1 - 9.0 kΩ, 1W
R2, R3 – 1 kΩ, 1W
R4 – 50 Ω Trimmer type
R5, R6 – 10 Ω,1W
R10,R11,R12,R13,R14,R15 -100 Ω, 1W
R7,R8, R9 - 47 Ω, 1W
RS - .1Ω, 16 W
D0,..,D6 -10DF6 ,Ultra fast recovery diode
C1,C3,C4,C5 – 0.1 μF,50V, Ceramic capacitor
C2- 1 nF, 50 V, Ceramic capacitor

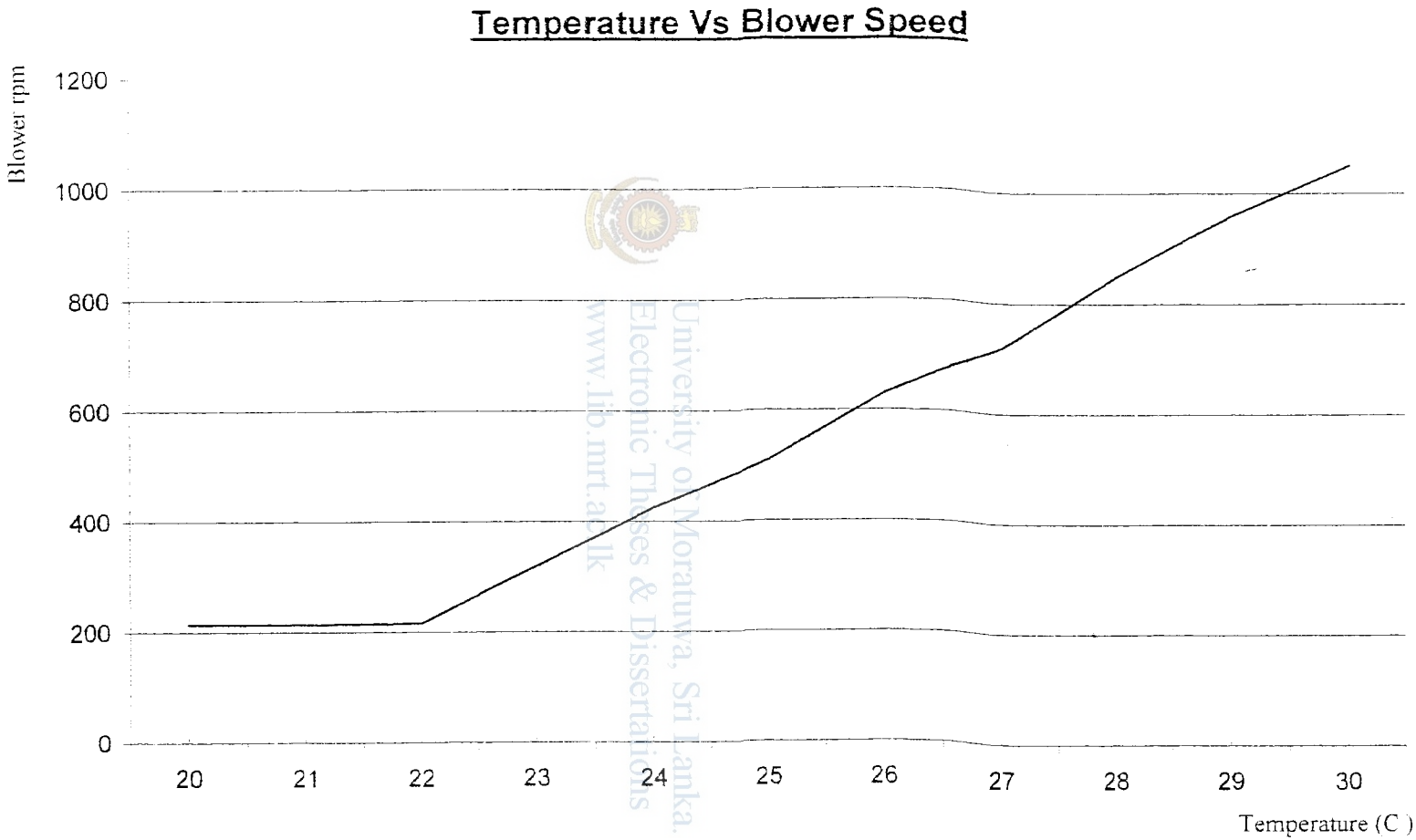
Figure 5.4 – Circuit diagram of IGBT driver & IGBT Power module

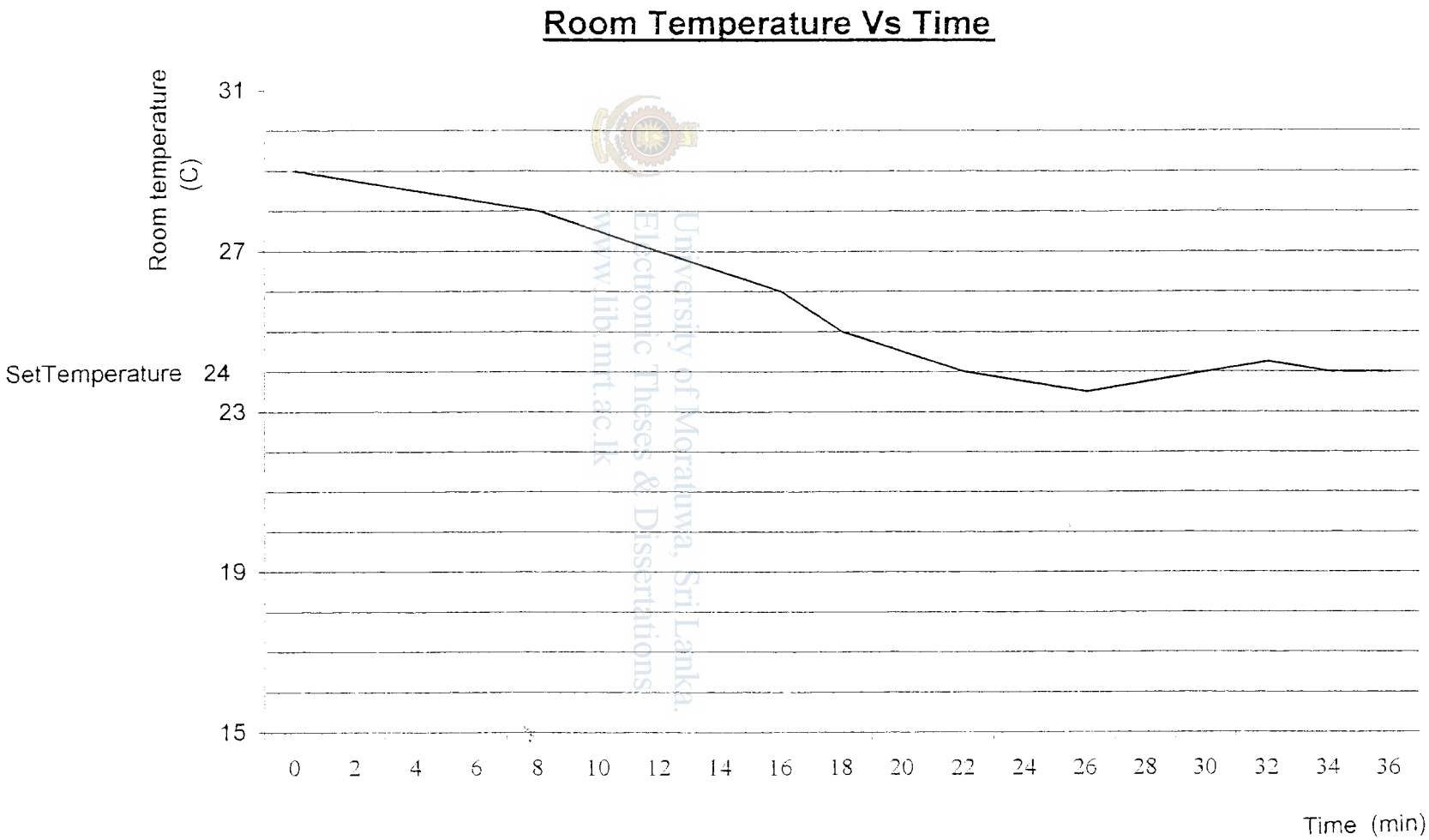
CHAPTER 6

Results

Output Voltages Generated by PWM







CHAPTER 7

Conclusion and Future Developments

7.1 Conclusion

Considering all the above research work and the results obtained, it can be concluded as follows. After power electronics start to pay a big role in the industry, VSD become much popular in speed control of Induction machines. Several advanced controlling techniques have been introduced and employed, especially for three phase induction machines during last few decades after rapid inventions in power electronics.

Variable Speed Drive constructed by this project with PIC 18F4431 which is dedicated for motor control applications. Software program generates PWM outputs to create required voltage wave forms and frequency at outputs of the IGBT power module. Output voltage and frequency can be set by varying the potentiometer position. Instant of potentiometer form of temperature controller can be used to vary output voltage and frequency.

7.2 V/f Control with Velocity Feedback and Current 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 7.1 The reference speed is still set by a potentiometer or any other controller, 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 a tachometer signal.

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.

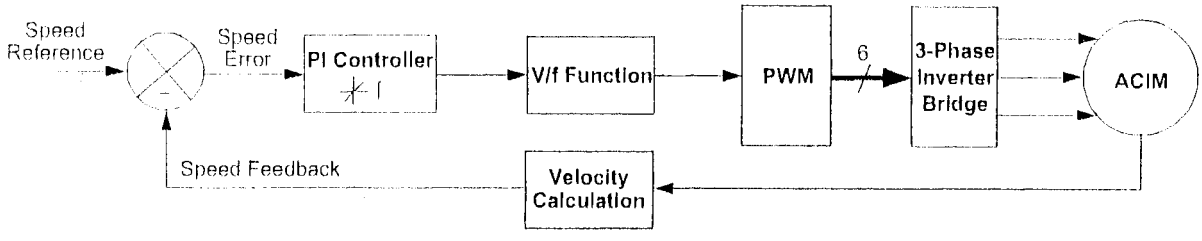


Figure 7.1 - Block diagram of VSD with feed back control

7.3 Additional features for VSD

Following features can be included in to the firm ware and software

Key activity

The main loop continuously checks key activity which 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.

Fault Signals Monitoring

Three fault signals can be monitored: overcurrent, overvoltage and over temperature. The overcurrent and overvoltage faults use the hardware fault inputs to directly inhibit the PCPWM outputs on a cycle-by-cycle basis.

Over Current 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. If the DC bus current signal exceeds the reference level, the fault input pin is driven low, indicating an overcurrent fault. Fault is indicated by blinking LED.

Over Voltage fault

The DC bus voltage is attenuated using a voltage divider and compared with a fixed reference signal using an external comparator. The fault input pin is used to monitor the overvoltage condition. Fault is indicated by blinking LED.

Over Temperature fault

The IGBT power module can have a Negative Temperature Coefficient (NTC) thermal sensor that monitors the junction temperature of the IGBTs. This NTC can be connected to analog input through an analog optocoupler and is continuously measured the junction temperature of the IGBTs. If it exceeds reference value, overtemperature fault is indicated by blinking LED.

7.4 Pay back period for VSD

Cost of Electronic Items =Rs 7,500.00

Cost of Electrical Items =Rs 2,000.00

Cost of casing ,Heat sink ect. =Rs 1,500.00

No of labour hours for assembling and testing of VSD = 20 hr.

Therefore cost of labour = 20 X 200.00 + Rs. 4,000.00

Other over heads cost = Rs 3,000.00

Total cost of VSD =18,000.00

By considering 20% profit = 3,600.00

Value of VSD =21,600.00

Simple Pay back period

By considering VSD will be installed for blower motor of package air conditioner having cooling capacity 50 kW, it will save the minimum of 5% of total electrical consumption.

Approximate Electrical energy saving per hour = 15 kW/hr X 0.05 =0.75 kW/hr

Electrical energy saving per day, by considering 6 hours operation =. 0.75 X 6 = 4.5 kW/hr

Cost of saving per day = 4.5 X 11.90 = Rs. 53.55

[Unit cost of Electricity is Rs. 11.90 (1 kW/hr = Rs. 11.90)]

Simple pay back period = 21,600.00/53.55 = 404 days

References:

[1] Richard Valantine, "Motor Control Electronics Handbook" ,1998, ISBN 0-07-066810-8

[2] David Bension, "PIC Micro Controller Application Guide", Version 2.0

[3] John B. Peatman "Design with PIC Microcontrollers", ISBN 81-7805-508-9

[4] Sanath Alahakoon, "Digital Motion Control Techniques for Electrical Drives", ISBN -91-7170-555-4

[5] Ned Moohan, Tore M. Undeland, William P. Robbins "Power Electronics" Second Edition

[6] Internationl Rectifier, "Motor Drive Control IC Designer's Manual "

[7] V. R. Moorthi "Power Electronics Devices, Circuits and Industrial applications", ISBN -10-0-19-567092-2

[8] Ashfaq Ahmed , "Power Electronics for Technology"', ISBN 81-297-0208-8

[9] R. Krishna, "Electric Motor Drives Modeling, Analysis and Control ", ISBN 81-297-0139-1

Software Program for Micro Controller

```
;*****************************************************************************
        include         <p18f4431.inc>
        include         <AA.inc>
;*****************************************************************************
        __CONFIG _CONFIG1H, 0x02
        __CONFIG _CONFIG2L, 0x0C
        __CONFIG _CONFIG2H, 0x3E
        __CONFIG _CONFIG3L, 0x3C

FLAGS bits
#define TIMER0_OV_FLAG 0
#define OFFSET1_FLAG    1
#define OFFSET2_FLAG    2
#define OFFSET3_FLAG    3
#define RUN             4
#defineSTOP             5
#defineFREQ_UPDATE      6

Keys parameters
#define KEY_PORT  PORTD
#define RUN_STOP_KEY  7

LED parameters
#define LED_PORT  PORTD
#define RUN_STOP_LED  0


Duty cycle limit definition,
#defineMINH_DUTY_CYCLE      0x00
#defineMINL_DUTY_CYCLE      0x3C



;*****************************************************************************
RAM locations in Access bank, initialized
;*****************************************************************************
UDATA_ACS
TABLE_OFFSET1           res     1
TABLE_OFFSET2           res     1
TABLE_OFFSET3           res     1
FLAGS                   res     1
FREQ_REF_H              res     1
```

```
FREQ_REF_L              res     1
FREQUENCY               res     1
TEMP                    res     1
TEMP1                   res     1
TEMP_LOCATION           res     2
PDC0L_TEMP              res     1
PDC0H_TEMP              res     1
PDC1L_TEMP              res     1
PDC1H_TEMP              res     1
PDC2L_TEMP              res     1
PDC2H_TEMP              res     1
SINE_TABLE              res     0x14    ;Sine table
temp                    res 1
temp1                   res 1


*****************************************************************************
                        RESET AND INTERRUPT VECTORS
*****************************************************************************
STARTUP     code 0x00
        goto    Start                   ;Reset Vector address
        CODE 0x08
        goto    ISR_HIGH                ;High priority ISR at 0x0008
PROG_LOW  CODE 0x018
        goto    ISR_LOW                 ;Low priority ISR at 0x0018



*****************************************************************************
                                INITIALIZATION
*****************************************************************************
Start

        clrf    FREQUENCY
        clrf    FLAGS
        call    INIT_PCPWM
        call    INIT_TMR0
        call    INIT_PORTD
        call    COPY_TABLE_TO_RAM
        call    INIT_MOTOR_START

WAIT_HERE
        call    KEY_CHECK
        btfss   FLAGS,RUN
        bra     WAIT_HERE
        clrf    FLAGS
        call    INIT_INTERRUPTS
```

```
*****************************************************************************
                                MAIN LOOP
*****************************************************************************

MAIN_LOOP
        btfss   FLAGS,TIMER0_OV_FLAG
        bra             bypass
        call    UPDATE_PWM_DUTYCYCLES
        call    UPDATE_TABLE_OFFSET
        bcf     FLAGS,TIMER0_OV_FLAG
bypass
        btfsc   FLAGS1,FREQ_UPDATE
        call    CALCULATE_TIMER0_RELOAD
        btfss   ADCON0, GO
        bsf     ADCON0, GO
        call    KEY_CHECK
        call    KEY_PRESSED

        bra     MAIN_LOOP


*****************************************************************************
                        INTERRUPT SERVICE ROUTINES
*****************************************************************************
ISR_HIGH
        btfsc   INTCON,TMR0IF
        bra             TIMER0_OVERFLOW
        RETFIE          FAST

TIMER0_OVERFLOW
        movff   FREQ_REF_H,TMR0H
        movff   FREQ_REF_L,TMR0L
        bsf             FLAGS,TIMER0_OV_FLAG
        bcf             INTCON,TMR0IF
        RETFIE          FAST

*****************************************************************************
                Low priority interrupt service routine
*****************************************************************************
ISR_LOW

        btfsc   PIR1,ADIF
        bra     READ_ADC_RESULTS

        RETFIE          FAST
```

```
READ_ADC_RESULTS

        movff   ADRESH,FREQUENCY
        movlw 0x30
        cpfsgt   FREQUENCY
        movwf FREQUENCY
        movlw 0xF0
        cpfslt  FREQUENCY
        movwf FREQUENCY
        bsf     FLAGS1, FREQ_UPDATE
        bcf     PIR1,ADIF
        RETFIE FAST


*****************************************************************************
UPDATE_PWM_DUTYCYCLES

*****************************************************************************

UPDATE_PWM_DUTYCYCLES

        movf   TABLE_OFFSET1,W
        movf   PLUSW0,W
        mulwf FREQUENCY, W
        movff  PRODH,PDC0H_TEMP
        movff  PRODL,PDC0L_TEMP

UPDATE_PWM2
        movf   TABLE_OFFSET2,W
        movf   PLUSW0,W
        mulwf FREQUENCY, W
        movff  PRODH,PDC1H_TEMP
        movff  PRODL,PDC1L_TEMP
UPDATE_PWM3
        movf   TABLE_OFFSET3,W
        movf   PLUSW0,W
        mulwf FREQUENCY, W
        movff  PRODH,PDC2H_TEMP
        movff  PRODL,PDC2L_TEMP


TRUNCATE_PWM123
        bcf     STATUS,C
        rlcf    PDC0L_TEMP,F
        rlcf    PDC0H_TEMP,F
        rlcf    PDC0L_TEMP,F
```

```
        rlcf    PDC0H_TEMP,F
        rlcf    PDC0L_TEMP,W
        andlw   0x3
        movff   PDC0H_TEMP,PDC0L_TEMP
        movwf   PDC0H_TEMP

        bcf     STATUS,C
        rlcf    PDC1L_TEMP,F
        rlcf    PDC1H_TEMP,F
        rlcf    PDC1L_TEMP,F
        rlcf    PDC1H_TEMP,F
        rlcf    PDC1L_TEMP,W
        andlw   0x3
        movff   PDC1H_TEMP,PDC1L_TEMP
        movwf   PDC1H_TEMP

        bcf     STATUS,C
        rlcf    PDC2L_TEMP,F
        rlcf    PDC2H_TEMP,F
        rlcf    PDC2L_TEMP,F
        rlcf    PDC2H_TEMP,F
        rlcf    PDC2L_TEMP,W
        andlw   0x3
        movff   PDC2H_TEMP,PDC2L_TEMP
        movwf   PDC2H_TEMP

        call    CHECK_LIMITS

        bsf     PWMCON1, UDIS
        movff   PDC0L_TEMP,PDC0L
        movff   PDC0H_TEMP,PDC0H
        movff   PDC1L_TEMP,PDC1L
        movff   PDC1H_TEMP,PDC1H
        movff   PDC2L_TEMP,PDC3L
        movff   PDC2H_TEMP,PDC3H

        bcf     PWMCON1, UDIS

        return

;*****************************************************************************
;TABLE_OFFSET
;*****************************************************************************

UPDATE_TABLE_OFFSET
        btfss   FLAGS,OFFSET1_FLAG
```

```
        bra     DECREMENT_OFFSET1
        movlw (SINE_TABLE_ENTRIES-1)
        cpfslt  TABLE_OFFSET1
        bra     CLEAR_OFFSET1_FLAG
        incf    TABLE_OFFSET1,F
        bra     UPDATE_OFFSET2

CLEAR_OFFSET1_FLAG
        bcf     FLAGS,OFFSET1_FLAG

DECREMENT_OFFSET1
        dcfsnz TABLE_OFFSET1,F
        bsf     FLAGS,OFFSET1_FLAG

UPDATE_OFFSET2
        btfss   FLAGS,OFFSET2_FLAG
        bra     DECREMENT_OFFSET2
        movlw (SINE_TABLE_ENTRIES-1)
        cpfslt  TABLE_OFFSET2
        bra     CLEAR_OFFSET2_FLAG
        incf    TABLE_OFFSET2,F
        bra     UPDATE_OFFSET3

CLEAR_OFFSET2_FLAG
        bcf     FLAGS,OFFSET2_FLAG

DECREMENT_OFFSET2
        dcfsnz TABLE_OFFSET2,F
        bsf     FLAGS,OFFSET2_FLAG

UPDATE_OFFSET3
        btfss   FLAGS,OFFSET3_FLAG
        bra     DECREMENT_OFFSET3
        movlw (SINE_TABLE_ENTRIES-1)
        cpfslt  TABLE_OFFSET3
        bra     CLEAR_OFFSET3_FLAG
        incf    TABLE_OFFSET3,F
        return

CLEAR_OFFSET3_FLAG
        bcf     FLAGS,OFFSET3_FLAG

DECREMENT_OFFSET3
        dcfsnz TABLE_OFFSET3,F
        bsf     FLAGS,OFFSET3_FLAG
        return
```

```
*****************************************************************************
CALCULATE_TIMER0_RELOAD

*****************************************************************************

CALCULATE_TIMER0_RELOAD

        bcf     FLAGS1,FREQ_UPDATE
        clrf    TEMP
        clrf    TEMP1
        movlw HIGH(FREQUENCY_SCALE)
        movwf TEMP_LOCATION
        movlw LOW(FREQUENCY_SCALE)
        movwf TEMP_LOCATION+1

continue_subtraction

        movf    FREQUENCY,W
        btfsc   STATUS,Z
        return
        bsf     STATUS,C
        movf    FREQUENCY,W
        subwfb TEMP_LOCATION+1,F
        clrf    WREG
        subwfb TEMP_LOCATION,F
        btfss   STATUS,C
        goto    Result
        incf    TEMP,F
        btfsc   STATUS,C
        incf    TEMP1,F
        goto    continue_subtraction

Result

        bsf     STATUS,C
        movlw 0xFF
        subfwb TEMP,F
        subfwb TEMP1,F
        movff  TEMP1,FREQ_REF_H
        movff  TEMP,FREQ_REF_L

        return
```

```
;*****************************************************************************
; Check limits routine
;*****************************************************************************
CHECK_LIMITS

CHK_PWM0_MIN
        movf   PDC0H_TEMP, F
        bnz    CHK_PWM1_MIN
        movlw MINL_DUTY_CYCLE
        cpfsgt  PDC0L_TEMP
         movwf PDC0L_TEMP

 CHK_PWM1_MIN
         movf   PDC1H_TEMP, F
         bnz    CHK_PWM2_MIN
         movlw MINL_DUTY_CYCLE
         cpfsgt  PDC1L_TEMP
         movwf PDC1L_TEMP

 CHK_PWM2_MIN
         movf   PDC2H_TEMP, F
         bnz    DONE_CHECK_LIMITS
         movlw MINL_DUTY_CYCLE
         cpfsgt  PDC2L_TEMP
         movwf PDC2L_TEMP
 DONE_CHECK_LIMIT
         return
 ;*****************************************************************************
 ; Stops the motor by driving the PWMs to 0% duty cycle.
 ;*****************************************************************************

 STOP_MOTOR
         bcf    PIE1,ADIE
         bcf    INTCON,TMR0IE
         clrf   OVDCOND
         clrf   TABLE_OFFSET1
         clrf   TABLE_OFFSET2
         clrf   TABLE_OFFSET3
         bcf    FLAGS, TIMER0_OV_FLAG
          bsf   FLAGS1,STOP

         return
```

```
;*****************************************************************************
; Starts motor from previous stop with motor parameters initialized
;*****************************************************************************
RUN_MOTOR
        bsf     FLAGS1,RUN
        bcf     FLAGS,FLAG_FAULT
        bsf     PIE1,ADIE
        call    INIT_MOTOR_START
        call    UPDATE_PWM_DUTYCYCLES
        call    UPDATE_TABLE_OFFSET
        bsf     INTCON,TMR0IE
        movlw   b'11111111'
        movwf   OVDCOND
        return

;*****************************************************************************
; KEY SWITCH SUBROUTINES
;*****************************************************************************
KEY_CHECK

        btfss   KEY_PORT, RUN_STOP_KEY
        return
        bsf             FLAGS,RUN
        return

KEY_PRESSED

        btfss   KEY_PORT, RUN_STOP_KEY
        goto    STOP_MOTOR_NOW
        btfss   FLAGS,STOP
        return
        goto    RUN_MOTOR
        bsf     LED_PORT,RUN_STOP_LED
        return

STOP_MOTOR_NOW
        call    STOP_MOTOR
        bcf     LED_PORT,RUN_STOP_LED
        return

;*****************************************************************************
; Initialize High-Speed ADC
;*****************************************************************************
INIT_HSADC
        movlw b'00000000'
        movwf ADCON1
```

```
        movlw b'00110010'
        movwf ADCON2
        movlw b'00000000'
        movwf ADCON3
        movlw b'11001111'
        movwf ADCHS
        movlw b'00000010'
        movwf ANSEL0
        movlw b'00000010'
        movwf TRISA

        movlw b'00000101'
        movwf ADCON0
        return

;*********************************************************************
Initialize PCPWM
;*********************************************************************
INIT_PCPWM

        movlw b'00000000'
        movwf PTCON0

        movlw 0xF9
        movwf PTPERL
        movlw 0x00
        movwf PTPERH

        movlw b'01010000'
        movwf PWMCON0

        movlw b'00000001'
        movwf  PWMCON1

        movlw  b'00001010'
        movwf  DTCON
                movlw  b'11111111'
        movwf  OVDCOND
        movlw  b'00000000'
        movwf  OVDCONS
        movlw  0x00
        movwf  SEVTCMPL
        movlw  0x00
        movwf  SEVTCMPH
        bsf     PTCON1, PTEN
        return
```

```
;*****************************************************************************
;Initialize PORTC
;*****************************************************************************
INIT_PORTB
        movlw b'00000000'
        movwf TRISB
        return

;*****************************************************************************
;Initialize PORTD
;*****************************************************************************
INIT_PORTD
        movlw b'10000001'
        movwf TRISD
        return

;*****************************************************************************
;Initialize Timer0
;*****************************************************************************
INIT_TMR0
        movlw b'10000100'
        movwf T0CON
        movlw 0xF8
        movwf TMR0H
        movlw 0x5E
        movwf TMR0L
        return

;*****************************************************************************
;initialize interrupts
;*****************************************************************************
INIT_INTERRUPTS

        bsf     INTCON,TMR0IE
        bsf     INTCON2,TMR0IP
        bsf     PIE1,ADIE
        bcf     IPR1,ADIP

        movlw b'10010011'
        movwf RCON
        bsf     INTCON,GIEL
        bsf     INTCON,GIEH

        return
```

```
*****************************************************************************
Initialize Motor
*****************************************************************************
INIT_MOTOR_START
        movlw 0x09
        movwf TABLE_OFFSET1
        bsf     FLAGS,OFFSET1_FLAG
        movlw 0x03
        movwf TABLE_OFFSET2
        bcf     FLAGS,OFFSET2_FLAG
        movlw 0x0F
        movwf TABLE_OFFSET3
        bcf     FLAGS,OFFSET3_FLAG
        bsf     PORTC,0
        bra     CONT_INIT_MOT

CONT_INIT_MOT
        movlw 0x30
        movwf FREQUENCY
        movlw 0xFD
        movwf FREQ_REF_H
        movwf TMR0H
        movlw 0x2C
        movwf TMR0L
        movwf FREQ_REF_L
        bsf     FLAGS,TIMER0_OV_FLAG
        return

*****************************************************************************
COPY_TABLE_TO_RAM
*****************************************************************************
COPY_TABLE_TO_RAM
        movlw UPPER sine_table
        movwf TBLPTRU
        movlw HIGH sine_table
        movwf TBLPTRH
        movlw LOW sine_table
        movwf TBLPTRL
        movlw LOW(SINE_TABLE)
        movwf FSR0L
        movlw HIGH(SINE_TABLE)
        movwf FSR0H
        movlw 0x14
        movwf TEMP
```

```
OPY_AGAIN
        TBLRD*+
        movff  TABLAT,POSTINC0
        decfsz  TEMP,F
        bra     COPY_AGAIN

        movlw  LOW(SINE_TABLE)
        movwf  FSR0L
        movlw HIGH(SINE_TABLE)
        movwf  FSR0H
        return


*****************************************************************************
                                SINE TABLE
*****************************************************************************
ABLE            code 0x0600
ine  table
0x00,0x02,0x08,0x11,0x1E,0x2E,0x40,0x54,0x69,0x80,0x96,0xAB,0xBF,0xD1,0xE1,0xEE
0xF7,0xFD,0xFF

*****************************************************************************
        END
```

File AA.inc

****

Oscillator frequency

#define OSCILLATOR d'20000000'

****

Timer0 prescaler;

#defineTIMER0_PRESCALE d'16'

****

The sampling frequency

#defineSINE_TABLE_ENTRIES d'37'

****

***SAMPLES_PER_CYCLE = (SINE_TABLE_ENTRIES-1)*d'2'

INSTRUCTION_CYCLE = (OSCILLATOR)/d'4'

FREQUENCY_SCALE=

(INSTRUCTION_CYCLE/SAMPLES_PER_CYCLE)/(TIMER0_PRESCALE/4)

(Timer prescale/4 is done to compensate ADC multiplication factor of 4 to the frequency)

****

PWM frequency definition

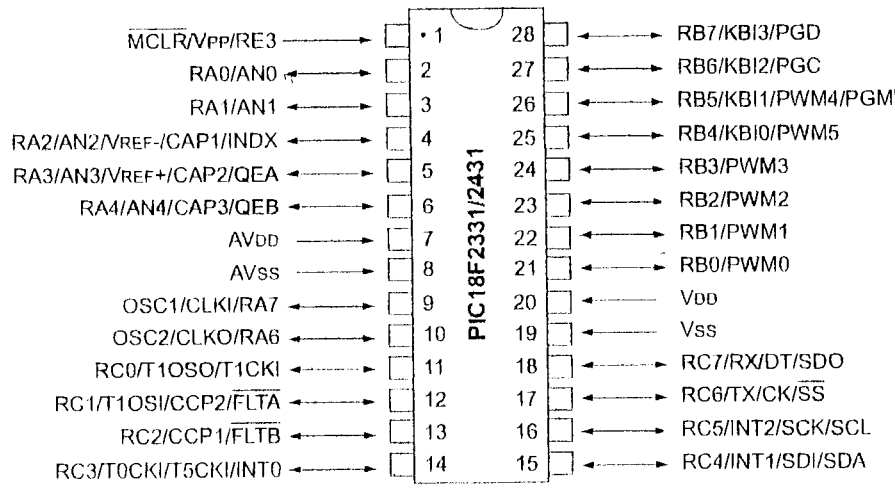
#definePWM_FREQUENCY d'5000'

****

```
// File: 18f4431.lkr
// Sample linker script for the PIC18F4431 processor

LIBPATH .

CODEPAGE   NAME=vectors    START=0x0          END=0x29        PROTECTED
CODEPAGE   NAME=page       START=0x2A         END=0x3FFF
CODEPAGE   NAME=idlocs     START=0x200000     END=0x200007    PROTECTED
CODEPAGE   NAME=config     START=0x300000     END=0x30000D
PROTECTED
CODEPAGE   NAME=devid      START=0x3FFFFE     END=0x3FFFFF
PROTECTED
CODEPAGE   NAME=cedata     START=0xF00000     END=0xF000FF
PROTECTED

ACCESSBANK NAME=accessram  START=0x0          END=0x5F
DATABANK   NAME=gpr0       START=0x60         END=0xFF
DATABANK   NAME=gpr1       START=0x100        END=0x1FF
DATABANK   NAME=gpr2       START=0x200        END=0x2FF
ACCESSBANK NAME=accesssfr  START=0xF60        END=0xFFF       PROTECTED

SECTION    NAME=CONFIG     ROM=config
```

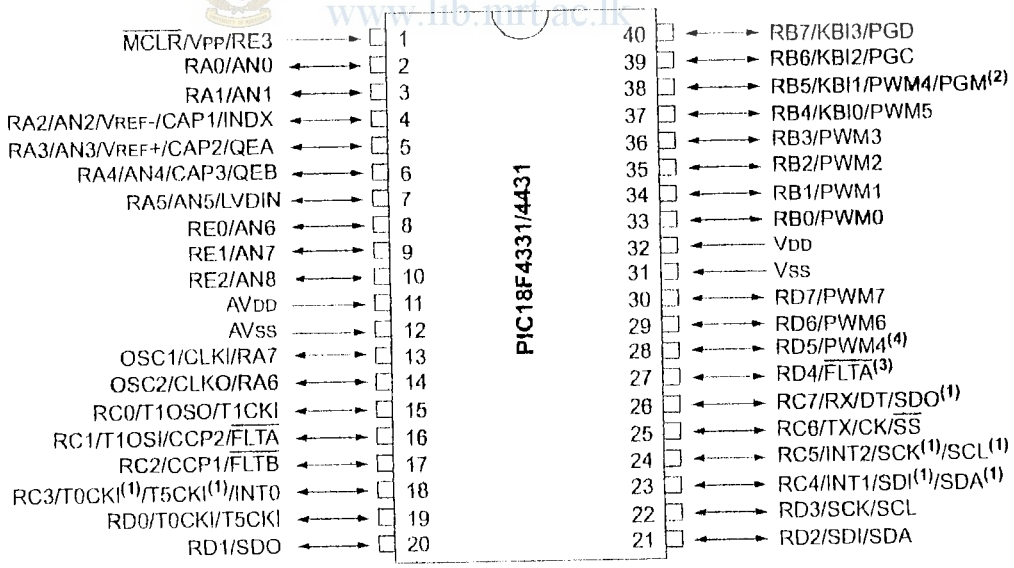
Pin Diagrams

28-Pin SDIP, SOIC



Note 1: Low-voltage programming must be enabled.

40-Pin PDIP



Note 1: RC3 is the alternate pin for T0CKI/T5CKI; RC4 is the alternate pin for SDI/SDA; RC5 is the alternate pin for SCK/SCL.

2: Low-voltage programming must be enabled.

3: RD4 is the alternate pin for FLTA.

4: RD5 is the alternate pin for PWM4.

PIC18F2331/2431/4331/4431

TABLE 1-3: PIC18F4331/4431 PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	TQFP	QFN			
MCLR/VPP/RE3	1	18	18			Master Clear (input) or programming voltage (input).
MCLR				I	ST	Master Clear (Reset) input. This pin is an active-low. Reset to the device.
VPP				P		Programming voltage input.
RE3				I	ST	Digital input. Available only when MCLR is disabled.
OSC1/CLKI/RA7	13	30	32			Oscillator crystal or external clock input.
OSC1				I	ST	Oscillator crystal input or external clock source input. ST buffer when configured in RC mode, CMOS otherwise.
CLKI				I	CMOS	External clock source input. Always associated with pin function OSC1. (See related OSC1/CLKI, OSC2/CLKO pins.)
RA7				I/O	TTL	General purpose I/O pin.
OSC2/CLKO/RA6	14	31	33			Oscillator crystal or clock output.
OSC2				O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKO				O	—	In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
RA6				I/O	TTL	General purpose I/O pin.
						PORTA is a bidirectional I/O port.
RA0/AN0	2	19	19			
RA0				I/O	TTL	Digital I/O.
AN0				I	Analog	Analog input 0.
RA1/AN1	3	20	20			
RA1				I/O	TTL	Digital I/O.
AN1				I	Analog	Analog input 1.
RA2/AN2/VREF-/CAP1/INDX	4	21	21			
RA2				I/O	TTL	Digital I/O.
AN2				I	Analog	Analog input 2.
VREF-				I	Analog	A/D Reference Voltage (Low) input.
CAP1				I	ST	Input capture pin 1.
INDX				I	ST	Quadrature Encoder Interface index input pin.
RA3/AN3/VREF+/CAP2/QEA	5	22	22			
RA3				I/O	TTL	Digital I/O.
AN3				I	Analog	Analog input 3.
VREF+				I	Analog	A/D Reference Voltage (High) input.
CAP2				I	ST	Input capture pin 2.
QEA				I	ST	Quadrature Encoder Interface channel A input pin.
RA4/AN4/CAP3/QEB	6	23	23			
RA4				I/O	TTL	Digital I/O.
AN4				I	Analog	Analog input 4.
CAP3				I	ST	Input capture pin 3.
QEB				I	ST	Quadrature Encoder Interface channel B input pin.
RA5/AN5/LVDIN	7	24	24			
RA5				I/O	TTL	Digital I/O.
AN5				I	Analog	Analog input 5.
LVDIN				I	Analog	Low-voltage Detect input.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power; OD = Open-Drain (no diode to VDD)

TABLE 1-3: PIC18F4331/4431 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	TQFP	QFN			
						PORTE is a bidirectional I/O port.
RE0/AN6	8	25	25			
RE0				I/O	ST	Digital I/O.
AN6				I	Analog	Analog input 6.
RE1/AN7	9	26	26			
RE1				I/O	ST	Digital I/O.
AN7				I	Analog	Analog input 7.
RE2/AN8	10	27	27			
RE2				I/O	ST	Digital I/O.
AN8				I	Analog	Analog input 8.
VSS	12, 31	6, 29	6, 30, 31	P	—	Ground reference for logic and I/O pins.
VDD	11, 32	7, 28	7, 8, 28, 29	P	—	Positive supply for logic and I/O pins.
NC	—	12, 13, 33, 34	13	NC	NC	No connect

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power; OD = Open-Drain (no diode to VDD)

TABLE 1-3: PIC18F4331/4431 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	TQFP	QFN			
						PORTD is a bidirectional I/O port, or a Parallel Slave Port (PSP) for interfacing to a microprocessor port. These pins have TTL input buffers when PSP module is enabled.
RD0/T0CKI/T5CKI	19	38	38			
RD0				I/O	ST	Digital I/O.
T0CKI				I	ST	Timer0 external clock input.
T5CKI				I	ST	Timer5 input clock.
RD1/SDO	20	39	39			
RD1				I/O	ST	Digital I/O.
SDO				O	—	SPI Data out.
RD2/SDI/SDA	21	40	40			
RD2				I/O	ST	Digital I/O.
SDI				I	ST	SPI Data in.
SDA				I/O	ST	I^2C Data I/O.
RD3/SCK/SCL	22	41	41			
RD3				I/O	ST	Digital I/O.
SCK				I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL				I/O	ST	Synchronous serial clock input/output for I^2C mode.
RD4/$\overline{FLTA}$	27	2	2			
RD4				I/O	ST	Digital I/O.
$\overline{FLTA}$				I	ST	Fault interrupt input pin.
RD5/PWM4	28	3	3			
RD5				I/O	ST	Digital I/O.
PWM4				O	TTL	PWM output 4.
RD6/PWM6	29	4	4			
RD6				I/O	ST	Digital I/O.
PWM6				O	TTL	PWM output 6.
RD7/PWM7	30	5	5			
RD7				I/O	ST	Digital I/O.
PWM7				O	TTL	PWM output 7.

Legend: TTL = TTL compatible input
CMOS = CMOS compatible input or output
ST = Schmitt Trigger input with CMOS levels
I = Input
O = Output
P = Power
OD = Open-Drain (no diode to VDD)

TABLE 1-3: PIC18F4331/4431 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	TQFP	QFN			
						PORTC is a bidirectional I/O port.
RC0/T1OSO/T1CKI	15	32	34			
RC0				I/O	ST	Digital I/O.
T1OSO				O	—	Timer1 oscillator output.
T1CKI				I	ST	Timer1 external clock input.
RC1/T1OSI/CCP2/$\overline{FLTA}$	16	35	35			
RC1				I/O	ST	Digital I/O.
T1OSI				I	CMOS	Timer1 oscillator input.
CCP2				I/O	ST	Capture2 input, Compare2 output, PWM2 output.
$\overline{FLTA}$				I	ST	Fault interrupt input pin.
RC2/CCP1/$\overline{FLTB}$	17	36	36			
RC2				I/O	ST	Digital I/O.
CCP1				I/O	ST	Capture1 input/Compare1 output/PWM1 output.
$\overline{FLTB}$				I	ST	Fault interrupt input pin.
RC3/T0CKI/T5CKI/INT0	18	37	37			
RC3				I/O	ST	Digital I/O.
T0CKI				I	ST	Timer0 alternate clock input.
T5CKI				I	ST	Timer5 alternate clock input.
INT0				I	ST	External interrupt 0.
RC4/INT1/SDI/SDA	23	42	42			
RC4				I/O	ST	Digital I/O.
INT1				I	ST	External interrupt 1.
SDI				I	ST	SPI Data in.
SDA				I/O	ST	I^2C Data I/O.
RC5/INT2/SCK/SCL	24	43	43			
RC5				I/O	ST	Digital I/O.
INT2				I	ST	External interrupt 2.
SCK				I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL				I/O	ST	Synchronous serial clock input/output for I^2C mode.
RC6/TX/CK/$\overline{SS}$	25	44	44			
RC6				I/O	ST	Digital I/O.
TX				O	—	USART Asynchronous Transmit.
CK				I/O	ST	USART Synchronous Clock (see related RX/DT).
$\overline{SS}$				I	ST	SPI Slave Select input.
RC7/RX/DT/SDO	26	1	1			
RC7				I/O	ST	Digital I/O.
RX				I	ST	USART Asynchronous Receive.
DT				I/O	ST	USART Synchronous Data (see related TX/CK).
SDO				O	—	SPI Data out.

Legend: TTL = TTL compatible input
ST = Schmitt Trigger input with CMOS levels
O = Output
OD = Open-Drain (no diode to VDD)
CMOS = CMOS compatible input or output
I = Input
P = Power

TABLE 1-3: PIC18F4331/4431 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	TQFP	QFN			
						PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0/PWM0	33	8	9			
RB0				I/O	TTL	Digital I/O.
PWM0				O	TTL	PWM output 0.
RB1/PWM1	34	9	10			
RB1				I/O	TTL	Digital I/O.
PWM1				O	TTL	PWM output 1.
RB2/PWM2	35	10	11			
RB2				I/O	TTL	Digital I/O.
PWM2				O	TTL	PWM output 2.
RB3/PWM3	36	11	12			
RB3				I/O	TTL	Digital I/O.
PWM3				O	TTL	PWM output 3.
RB4/KBI0/PWM5	37	14	14			
RB4				I/O	TTL	Digital I/O.
KBI0				I	TTL	Interrupt-on-change pin.
PWM5				O	TTL	PWM output 5.
RB5/KBI1/PWM4/ PGM	38	15	15			
RB5				I/O	TTL	Digital I/O.
KBI1				I	TTL	Interrupt-on-change pin.
PWM4				O	TTL	PWM output 4.
PGM				I/O	ST	Low-voltage ICSP programming entry pin.
RB6/KBI2/PGC	39	16	16			
RB6				I/O	TTL	Digital I/O.
KBI2				I	TTL	Interrupt-on-change pin.
PGC				I/O	ST	In-Circuit Debugger and ICSP programming clock pin.
RB7/KBI3/PGD	40	17	17			
RB7				I/O	TTL	Digital I/O.
KBI3				I	TTL	Interrupt-on-change pin.
PGD				I/O	ST	In-Circuit Debugger and ICSP programming data pin.

Legend: TTL = TTL compatible input
CMOS = CMOS compatible input or output
ST = Schmitt Trigger input with CMOS levels
I = Input
O = Output
P = Power
OD = Open-Drain (no diode to VDD)

International
IOR Rectifier

IR2130/IR2132

3-PHASE BRIDGE DRIVER

Features

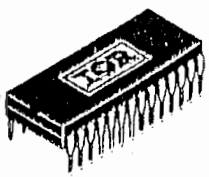
- Floating channel designed for bootstrap operation
  Fully operational to +600V
  Tolerant to negative transient voltage
  dV/dt immune
- Gate drive supply range from 10 to 20V
- Undervoltage lockout for all channels
- Over-current shutdown turns off all six drivers
- Independent half-bridge drivers
- Matched propagation delay for all channels
- Outputs out of phase with inputs
- Cross-conduction prevention logic

Product Summary

V_{OFFSET}	600V max.
$I_{O+/-}$	200 mA / 420 mA
V_{OUT}	10 - 20V
$t_{on/off}$ (typ.)	675 & 425 ns
Deadtime (typ.)	2.5 µs (IR2130) 0.8 µs (IR2132)

Packages


28 Lead SOIC

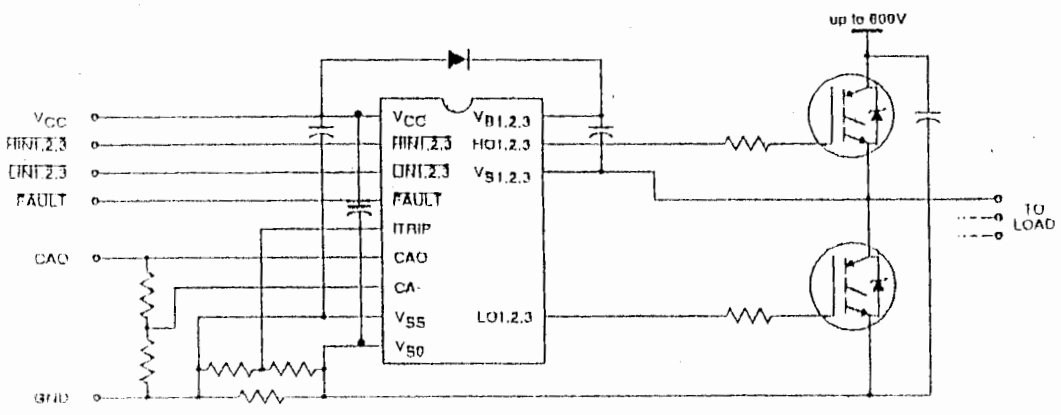

28 Lead PDIP


44 Lead PLCC w/o 12 Leads

Description

The IR2130/IR2132 is a high voltage, high speed power MOSFET and IGBT driver with three independent high and low side referenced output channels. Proprietary HVIC technology enables ruggedized monolithic construction. Logic inputs are compatible with 5V CMOS or LSTTL outputs. A ground-referenced operational amplifier provides analog feedback of bridge current via an external current sense resistor. A current trip function which terminates all six outputs is also derived from this resistor. An open drain $\overline{FAULT}$ signal indicates if an over-current or undervoltage shutdown has occurred. The output drivers feature a high pulse current buffer stage designed for minimum driver cross-conduction. Propagation delays are matched to simplify use at high frequencies. The floating channels can be used to drive N-channel power MOSFETs or IGBTs in the high configuration which operate up to 600 volts.

Typical Connection



Absolute Maximum Ratings

Absolute Maximum Ratings indicate sustained limits beyond which damage to the device may occur. All voltage parameters are absolute voltages referenced to V_{S0}. The Thermal Resistance and Power Dissipation ratings are measured under board mounted and still air conditions. Additional information is shown in Figures 50 through 53.

Symbol	Definition	Min.	Max.	Units
$V_{B1,2,3}$	High Side Floating Supply Voltage	-0.3	625	
$V_{S1,2,3}$	High Side Floating Offset Voltage	$V_{B1,2,3}$ - 25	$V_{B1,2,3}$ + 0.3	
$V_{HO1,2,3}$	High Side Floating Output Voltage	$V_{S1,2,3}$ - 0.3	$V_{B1,2,3}$ + 0.3	
V_{CC}	Low Side and Logic Fixed Supply Voltage	-0.3	25	
V_{SS}	Logic Ground	V_{CC} - 25	V_{CC} + 0.3	
$V_{LO1,2,3}$	Low Side Output Voltage	-0.3	V_{CC} + 0.3	
V_{IN}	Logic Input Voltage ($\overline{HIN1,2,3}$, $\overline{LIN1,2,3}$ & ITRIP)	V_{SS} - 0.3	V_{CC} + 0.3	V
V_{FLT}	$\overline{FAULT}$ Output Voltage	V_{SS} - 0.3	V_{CC} + 0.3	
V_{CAO}	Operational Amplifier Output Voltage	V_{SS} - 0.3	V_{CC} + 0.3	
V_{CA-}	Operational Amplifier Inverting Input Voltage	V_{SS} - 0.3	V_{CC} + 0.3	
dV_S/dt	Allowable Offset Supply Voltage Transient	—	50	.....
P_D	Package Power Dissipation @ $T_A \leq +25°C$ (28 Lead DIP)	—	1.5	
	(28 Lead SOIC)	—	1.6	W
	(44 Lead PLCC)	—	2.0	
Rth_{JA}	Thermal Resistance, Junction to Ambient (28 Lead DIP)	—	83	
	(28 Lead SOIC)	—	78	°C/W
	(44 Lead PLCC)	—	63	
T_J	Junction Temperature	—	150	
T_S	Storage Temperature	-55	150	°C
T_L	Lead Temperature (Soldering, 10 seconds)	—	300	

Recommended Operating Conditions

The Input/Output logic timing diagram is shown in Figure 1. For proper operation the device should be used within the recommended conditions. All voltage parameters are absolute voltages referenced to V_{S0}. The V_S offset rating is tested with all supplies biased at 15V differential. Typical ratings at other bias conditions are shown in Figure 54.

Symbol	Definition	Min.	Max.	Units
$V_{B1,2,3}$	High Side Floating Supply Voltage	$V_{S1,2,3}$ + 10	$V_{S1,2,3}$ + 20	
$V_{S1,2,3}$	High Side Floating Offset Voltage	Note 1	600	
$V_{HO1,2,3}$	High Side Floating Output Voltage	$V_{S1,2,3}$	$V_{B1,2,3}$	
V_{CC}	Low Side and Logic Fixed Supply Voltage	10	20	
V_{SS}	Logic Ground	-5	5	
$V_{LO1,2,3}$	Low Side Output Voltage	0	V_{CC}	
V_{IN}	Logic Input Voltage ($\overline{HIN1,2,3}$, $\overline{LIN1,2,3}$ & ITRIP)	V_{SS}	V_{SS} + 5	V
V_{FLT}	$\overline{FAULT}$ Output Voltage	V_{SS}	V_{CC}	
V_{CAO}	Operational Amplifier Output Voltage	V_{SS}	5	
V_{CA-}	Operational Amplifier Inverting Input Voltage	V_{SS}	5	
T_A	Ambient Temperature	-40	125	°C

Note 1: Logic operational for V_S of (V_{S0} - 5V) to (V_{S0} + 600V). Logic state held for V_S of (V_{S0} - 5V) to (V_{S0} - V_{BS}).

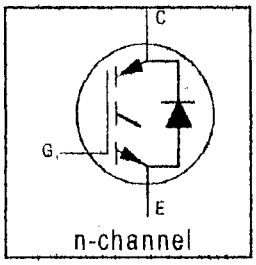
International
IOR Rectifier

PD -91587A

IRG4PC30KD

INSULATED GATE BIPOLAR TRANSISTOR WITH ULTRAFAST SOFT RECOVERY DIODE

Short Circuit Rated
UltraFast IGBT



V_{CES} = 600V

$V_{CE(on)}$ typ. = 2.21V

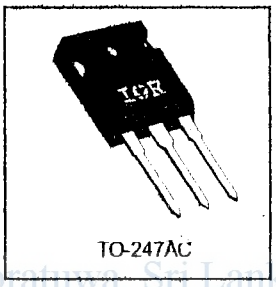
@V_{GE} = 15V, I_C = 16A

Features

- High short circuit rating optimized for motor control, t_{sc} =10µs, @360V V_{CE} (start), T_J = 125°C, V_{GE} = 15V
- Combines low conduction losses with high switching speed
- Tighter parameter distribution and higher efficiency than previous generations
- IGBT co-packaged with HEXFRED™ ultrafast, ultrasoft recovery antiparallel diodes

Benefits

- Latest generation 4 IGBTs offer highest power density motor controls possible
- HEXFRED™ diodes optimized for performance with IGBTs. Minimized recovery characteristics reduce noise, EMI and switching losses
- This part replaces the IRGBC30KD2 and IRGBC30MD2 products
- For hints see design tip 97003



TO-247AC

Absolute Maximum Ratings

	Parameter	Max.	Units
V_{CES}	Collector-to-Emitter Voltage	600	V
I_C @ T_C = 25°C	Continuous Collector Current	28	A
I_C @ T_C = 100°C	Continuous Collector Current	16	
I_{CM}	Pulsed Collector Current ①	58	
I_{LM}	Clamped Inductive Load Current ②	58	
I_F @ T_C = 100°C	Diode Continuous Forward Current	12	
I_{FM}	Diode Maximum Forward Current	58	
t_{sc}	Short Circuit Withstand Time	10	µs
V_{GE}	Gate-to-Emitter Voltage	± 20	V
P_D @ T_C = 25°C	Maximum Power Dissipation	100	W
P_D @ T_C = 100°C	Maximum Power Dissipation	42	
T_J T_{STG}	Operating Junction and Storage Temperature Range	-55 to +150	°C
	Soldering Temperature, for 10 sec.	300 (0.063 in. (1.6mm) from case)	
	Mounting Torque, 6-32 or M3 Screw.	10 lbf•in (1.1 N•m)	

Thermal Resistance

	Parameter	Min.	Typ.	Max.	Units
$R_{\theta JC}$	Junction-to-Case - IGBT	—	—	1.2	°C/W
$R_{\theta JC}$	Junction-to-Case - Diode	—	—	2.5	
$R_{\theta CS}$	Case-to-Sink, flat, greased surface	—	0.24	—	
$R_{\theta JA}$	Junction-to-Ambient, typical socket mount	—	—	40	
Wt	Weight	—	6 (0.21)	—	g (oz)

Electrical Characteristics @ T_J = 25°C (unless otherwise specified)

	Parameter	Min.	Typ.	Max.	Units	Conditions
$V_{(BR)CES}$	Collector-to-Emitter Breakdown Voltage③	600	—	—	V	V_{GE} = 0V, I_C = 250µA
$\Delta V_{(BR)CES}/\Delta T_J$	Temperature Coeff. of Breakdown Voltage	—	0.54	—	V/°C	V_{GE} = 0V, I_C = 1.0mA
$V_{CE(on)}$	Collector-to-Emitter Saturation Voltage	—	2.21	2.7	V	I_C = 16A; V_{GE} = 15V
		—	2.88	—		I_C = 28A; See Fig. 2, 5
		—	2.36	—		I_C = 16A, T_J = 150°C
$V_{GE(th)}$	Gate Threshold Voltage	3.0	—	6.0	V	V_{CE} = V_{GE}, I_C = 250µA
$\Delta V_{GE(th)}/\Delta T_J$	Temperature Coeff. of Threshold Voltage	—	-12	—	mV/°C	V_{CE} = V_{GE}, I_C = 250µA
g_{fe}	Forward Transconductance ④	5.4	8.1	—	S	V_{CE} = 100V, I_C = 16A
I_{CES}	Zero Gate Voltage Collector Current	—	—	250	µA	V_{GE} = 0V, V_{CE} = 600V
		—	—	2500		V_{GE} = 0V, V_{CE} = 600V, T_J = 150°C
V_{FM}	Diode Forward Voltage Drop	—	1.4	1.7	V	I_C = 12A; See Fig. 13
		—	1.3	1.6		I_C = 12A, T_J = 150°C
I_{GES}	Gate-to-Emitter Leakage Current	—	—	±100	nA	V_{GE} = ±20V

Switching Characteristics @ T_J = 25°C (unless otherwise specified)

	Parameter	Min.	Typ.	Max.	Units	Conditions
Q_g	Total Gate Charge (turn-on)	—	67	100	nC	I_C = 16A
Q_{ge}	Gate - Emitter Charge (turn-on)	—	11	16		V_{CC} = 400V See Fig.8
Q_{gc}	Gate - Collector Charge (turn-on)	—	25	37		V_{GE} = 15V
$t_{d(on)}$	Turn-On Delay Time	—	60	—	ns	T_J = 25°C
t_r	Rise Time	—	42	—		I_C = 16A, V_{CC} = 480V
$t_{d(off)}$	Turn-Off Delay Time	—	160	250		V_{GE} = 15V, R_G = 23Ω
t_f	Fall Time	—	80	120		Energy losses include "tail"
E_{on}	Turn-On Switching Loss	—	0.60	—	mJ	and diode reverse recovery
E_{off}	Turn-Off Switching Loss	—	0.58	—		See Fig. 9,10,14
E_{ts}	Total Switching Loss	—	1.18	1.8		
t_{sc}	Short Circuit Withstand Time	10	—	—	µs	V_{CC} = 360V, T_J = 125°C; V_{GE} = 15V, R_G = 10Ω , V_{CPK} < 500V
$t_{d(on)}$	Turn-On Delay Time	—	58	—	ns	T_J = 150°C, See Fig. 11,14
t_r	Rise Time	—	42	—		I_C = 16A, V_{CC} = 480V
$t_{d(off)}$	Turn-Off Delay Time	—	210	—		V_{GE} = 15V, R_G = 23Ω,
t_f	Fall Time	—	160	—		Energy losses include "tail"
E_{ts}	Total Switching Loss	—	1.69	—	mJ	and diode reverse recovery
L_E	Internal Emitter Inductance	—	13	—	nH	Measured 5mm from package
C_{ies}	Input Capacitance	—	920	—	pF	V_{GE} = 0V
C_{oes}	Output Capacitance	—	110	—		V_{CC} = 30V See Fig. 7
C_{res}	Reverse Transfer Capacitance	—	27	—		ƒ = 1.0MHz
t_{rr}	Diode Reverse Recovery Time	—	42	60	ns	T_J = 25°C See Fig. 14; I_F = 12A
		—	80	120		T_J = 125°C
I_{rr}	Diode Peak Reverse Recovery Current	—	3.5	6.0	A	T_J = 25°C See Fig. 15; V_R = 200V
		—	5.6	10		T_J = 125°C
Q_{rr}	Diode Reverse Recovery Charge	—	80	180	nC	T_J = 25°C See Fig. 16; dI/dt = 200A/µs
		—	220	600		T_J = 125°C
$di_{(rec)M}/dt$	Diode Peak Rate of Fall of Recovery During t_b	—	180	—	A/µs	T_J = 25°C See Fig. 17
		—	160	—		T_J = 125°C

FAIRCHILD
SEMICONDUCTOR

July 2005

Single-channel: 6N135, 6N136, HCPL-2503, HCPL-4502
Dual-Channel: HCPL-2530, HCPL-2531
High Speed Transistor Optocouplers

Features

- High speed-1 MBit/s
- Superior CMR-10 kV/µs
- Dual-Channel HCPL-2530/HCPL-2531
- Double working voltage-480V RMS
- CTR guaranteed 0-70°C
- U.L. recognized (File # E90700)

Applications

- Line receivers
- Pulse transformer replacement
- Output interface to CMOS-LSTTL-TTL
- Wide bandwidth analog coupling

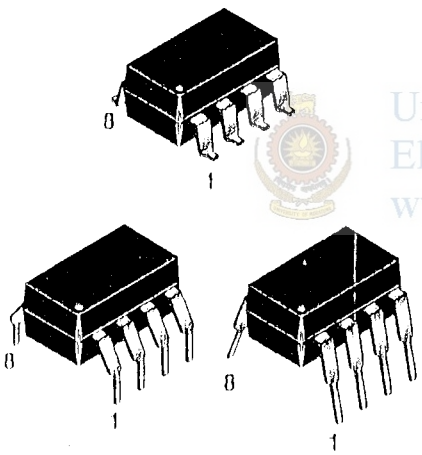
Description

The HCPL-4502/HCPL-2503, 6N135/6 and HCPL-2530/HCPL-2531 optocouplers consist of an AlGaAs LED optically coupled to a high speed photodetector transistor.

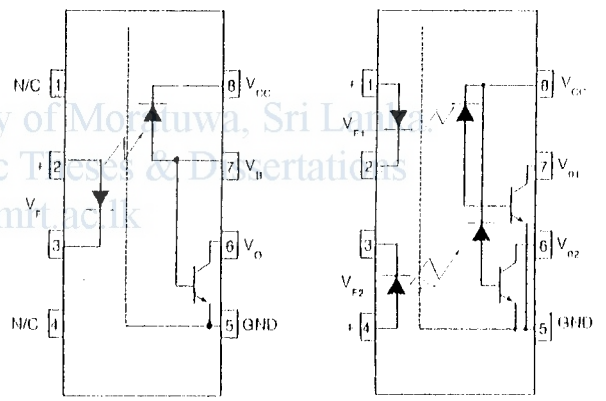
A separate connection for the bias of the photodiode improves the speed by several orders of magnitude over conventional phototransistor optocouplers by reducing the base-collector capacitance of the input transistor.

An internal noise shield provides superior common mode rejection of 10kV/µs. An improved package allows superior insulation permitting a 480 V working voltage compared to industry standard of 220 V.

Package



Schematic



6N135, 6N136, HCPL-2503, HCPL-4502

HCPL-2530/HCPL-2531

Pin 7 is not connected in Part Number HCPL-4502

Absolute Maximum Ratings (T_A = 25°C unless otherwise specified)

Parameter		Symbol	Value	Units
Storage Temperature		T_{STG}	-55 to +125	°C
Operating Temperature		T_{OPR}	-55 to +100	°C
Lead Solder Temperature		T_{SOL}	260 for 10 sec	°C
EMITTER				
DC/Average Forward Input Current	Each Channel (Note 1)	I_F (avg)	25	mA
Peak Forward Input Current (50% duty cycle, 1 ms P.W.)	Each Channel (Note 2)	I_F (pk)	50	mA
Peak Transient Input Current - (≤1 µs P.W., 300 pps)	Each Channel	I_F (trans)	1.0	A
Reverse Input Voltage	Each Channel	V_R	5	V
Input Power Dissipation	(6N135/6N136 and HCPL-2503/4502) (HCPL-2530/2531) Each Channel (Note 3)	P_D	100 45	mW
DETECTOR				
Average Output Current	Each Channel	I_O (avg)	8	mA
Peak Output Current	Each Channel	I_O (pk)	16	mA
Emitter-Base Reverse Voltage	(6N135, 6N136 and HCPL-2503 only)	V_{EBR}	5	V
Supply Voltage		V_{CC}	-0.5 to 30	V
Output Voltage		V_O	-0.5 to 20	V
Base Current	(6N135, 6N136 and HCPL-2503 only)	I_B	5	mA
Output power dissipation	(6N135, 6N136, HCPL-2503, HCPL-4502) (Note 4)	PD	100	mW
	(HCPL-2530, HCPL-2531) Each Channel		35	mW

Data Sheet

Temperature sensor ic LM35CZ and LM35DZ

RS stock numbers 317-954 and 317-960

The LM35 is a precision semiconductor temperature sensor giving an output of 10mV per degree Centigrade. Unlike devices with outputs proportional to the absolute temperature (in degrees Kelvin) there is no large offset voltage which, in most applications, will have to be removed.

Accuracies of ¼°C at room temperature or ¾°C over the full temperature range are typical.

Absolute maximum ratings (Note 10)

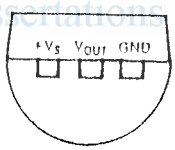
Supply voltage ..... +35V to -0.2V
Output voltage ..... +6V to -1.0V
Output current 10mA
Storage temperature, TO-92 package ..... -60°C to +150°C
Lead temperature (soldering, 10 seconds) ..... 260°C
Specified operating temperature range
T_{MIN} to T_{MAX} (Note 8)
LM35CZ ..... -40°C to +110°C
LM35DZ ..... 0°C to +100°C

Features

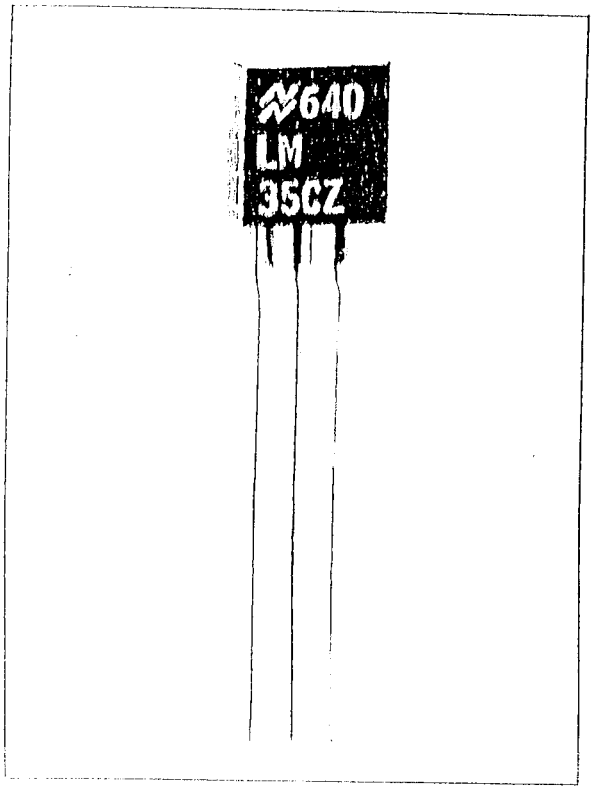
- Output proportional to °C
- Wide temperature range -40°C to +110°C (CZ version)
- Accurate ¼°C at room temperature typical
- Linear output 0.2°C typical
- Low current drain (60μA typical)
- Low self heating (0.08°C typical)
- Output impedance 0.1Ω at 1mA
- Standard T092 package.

Pin connections

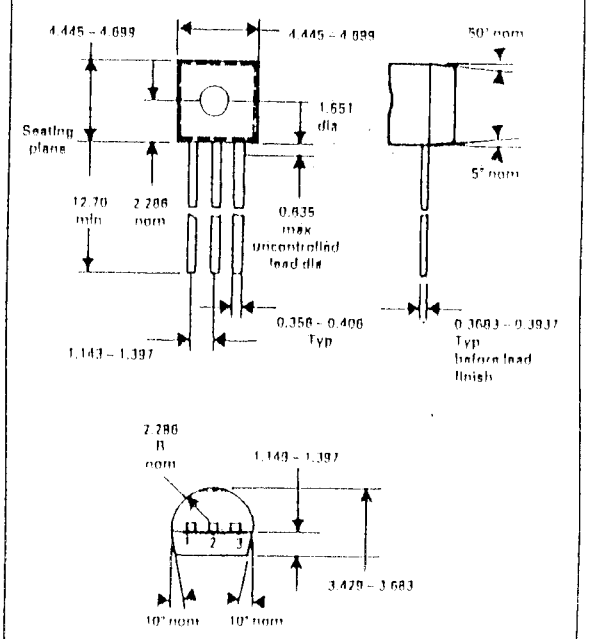
TO-92
Plastic package



BOTTOM VIEW

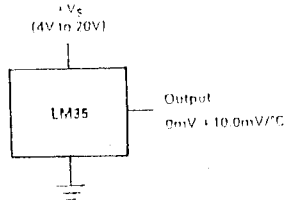


Package details



Application notes

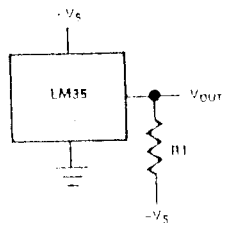
Figure 1



The circuit shown in Figure 1 is a basic single ended temperature sensor capable of measuring between +2°C and +100°C or +110°C depending on version.

To measure negative temperatures a negative supply is required as shown in Figure 2.

Figure 2



R1 should be selected as follows:

$$R1 = \frac{-V_s}{50 \times 10^{-6}}$$

Care must be taken when driving capacitive load, such as long cables or any load exceeding 50pF.

To remove the effect of capacitive loads the circuit shown Figure 3 should be used, however the resistor is added to the output impedance making this circuit suitable for connection to high impedance loads only.

Figure 4 shows a circuit when will overcome this problem and also give protection from radiated interference from relays or any other source of electrical noise.

Figure 3

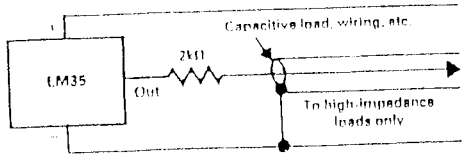
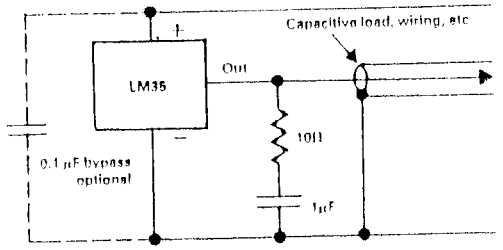


Figure 4



The circuits below show some typical applications of these temperature sensors.

Figure 5 Two-wire remote temperature sensor with sensor grounded

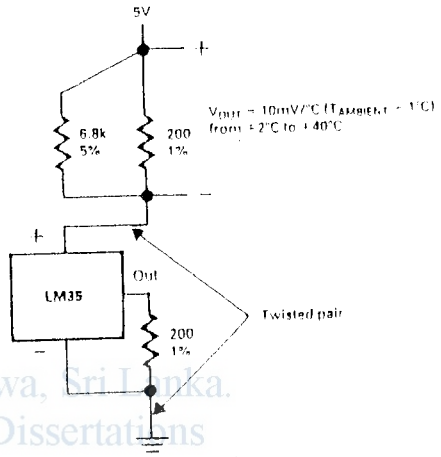


Figure 6 Two-wire remote temperature sensor