UNIVERSITI
TEKNOLOGI
PETRONAS

[DESIGN AND IMPLEMENTATION OF A MOBILE ROBOT]

by

[TRUONG NGUYEN VU]

FINAL YEAR PROJECT REPORT

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

DECEMBER 2004

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

[DESIGN AND IMPLEMENTATION OF A MOBILE ROBOT]

by

[TRUONG NGUYEN VU]

FINAL YEAR PROJECT REPORT
A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

DECEMBER 2004

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

[DESIGN AND IMPLEMENTATION OF A MOBILE ROBOT]

by

[Truong Nguyen Vu]

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

Mr. MOHD HARIS MD KHIR
Project Supervisor

UNIVERISITI TEKNOLOGI PETRONAS
31750 TRONOH, PERAK
DECEMBER 2004

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

TRUONG NGUYEN VU

ABSTRACT

Mobile robot nowadays has been used in various sectors of business ranged from sciences and industries to servicing and entertainment, and has posed very unique challenges to the researcher. Due to its autonomous nature, it requires an extensive sensing system to read the surrounding working environment, sufficient controller for data processing and decision-making, and also a very stable actuator system to actuate the robot response with respect to the working environment changes.

This report details the final year project of designing and implementing an autonomous wheel-based mobile robot, which has a capability to navigate according to a predetermined path while avoiding any obstacle on its route. It is also equipped with small manipulator to enable simple pick and place action: picking a small object from a predefined location and placing it at another predetermined location on the floor. Hence, in order for it to do all those task, the robot fundamentally consists of four main building functional blocks: (1) Navigation (2) Position tracking (3) Obstacle detection and avoidance (4) Manipulator & gripper control block (Since the robot movement is based on a predetermined plan, the path planner block is skipped in this project).

For further improvement, some fine-tuning on the manipulator structure and programming, as well as the development of a user interface to enable to user to select the robot movement path are highly recommended. In summary, the project has been successfully completed, providing a basement for future mobile robot development in UTP.

ACKNOWLEDGEMENTS

First and foremost, I would like to show my greatest appreciation to Mr. Mohd Haris Md Khir, my direct academic supervisor for accepting me as his final year project student as well as being very supportive throughout the project. He spent so much time guiding me through the project, giving me the direction whenever I get stuck in any related problems from the beginning towards the completion of the project. Working under him is really a great pleasure since I have learnt a great deal of knowledge, skills as well as building up some necessary precious experiences.

Huge thanks to Dr Mohd Noh Karsiti, Head of Electrical Engineering program for his constant advises and support on non-academic matters.

My thankfulness is also extended to the EE lab technicians, especially Miss Hawa, Mr. Azhar, Mr. Musa as well as Mr. Isnani for their constant support. They have shared with me not only the engineering related knowledge but also their valuable experiences, which help me to successfully complete my final year project.

Besides, I would like to express my sincere thanks to Mr. Zuki and Ms Nascreen for their successful organizing of the Electrical engineering program final year project.

Thanks to all whose names that are not mentioned here for providing assistance, support and cooperation in any form.

Last but not least, I would like to dedicate my deepest gratitude to my parents for all the moral supports, concerns and understandings.

Best Wishes

Truong Nguyen Vu - 3188

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND

For the last decades, many researchers have been dedicating their efforts in constructing revolutionary machines with embedded artificial intelligence to perform some risky, dangerous autonomous tasks (which were historically assigned to human beings). Among these machines, mobile robot is a typical example.

Mobile robot has constantly posed a unique challenge to artificial intelligence researchers. They are inherently autonomous, forcing the researchers to deal with those uncertainties (in both sensing and action), reliability as well as real time response. Hence, mobile robots require the integration of sensing, acting and planning within a single system. The sensing system critically helps the robot read the surrounding environment, enabling for path planning, finding, and navigation. It is really crucial in robot movement and acting to the changes of the environment.

Nowadays, mobile robots have been extensively used in various sectors of business ranged from industrial and scientific applications (i.e. petroleum path cleaning, underwater robot, space oriented robots, etc.) to servicing (i.e. patient assisting robot, trash collecting robot, etc). Although some of the researchers have successfully developed the legged-based robots, the most popular robot motion mechanism for the on-surface movement is wheeled based.

1.2 PROBLEM STATEMENT

The project objective is to build a complete wheel-based mobile robot, which is able to work in an indoor environment. It should basically be able to navigate, avoid any obstacles along its movement towards to the preset target. This action is performed through the application of appropriate algorithms, which can compute the optimized path for the robot movement. Thus, it requires an active sensing system to measure the environment, input to a microcontroller for processing and decision making, as a powerful mean for algorithm implementation.

1.3 OBJECTIVES AND SCOPE OF THE STUDY

The objectives of the project are defined as below

1. Construct a stable wheel-based mechanical structure, which is enabled to support up to 20 kg of load.
2. Design a mobile robot, being able to navigate according to a predetermined path while avoiding any obstacles and return to the original route.
3. Design a robot which has the capability to go to a pre-determined location, pick an object and place it at another predefined location on the floor. This could be seen as an extra feature added to the robot.

Due to the nature of the project (one-year project), the tasks have been clearly divided into two parts for two semesters with manageable targets as follow:

- **1st semester**: To develop the robot mechanical structure with motors and wheels mounted, drive circuit ready, so that the robot can perform some basic movements (i.e. forward, reverse or turning, etc)
- **2nd semester**: To develop the sensing system together with the microcontroller equipped to the robot. With the artificial intelligence embedded, the robot is able to operate autonomously on just a single **"start"**. A manipulator is an extra feature, enabling the robot to perform some pick and place tasks.

CHAPTER 2
LITERATURE REVIEW AND THEORY

2.1 INTRODUCTION TO MOBILE ROBOTICS

Mobile robot has continuously achieved more importance in industrial and technical application, for its reliability (uninterrupted, reliable execution of autonomous task), accessibility (ability to access to sites that are inaccessible to humans such as tight spaces, hazardous environments, etc). It has been widely used for surveillance, inspection and transportation tasks as well as for some entertainment areas. However, before any level tasks to be executed, it is critical that the robot can operate safely, a void obstacles, a nd perform s ome fundamental activities s uch a s contour f ollowing, d oor t raversal, etc w ithout p osing a ny risk t o the environment and people in it. In the past, to achieve that, people implemented t he robot w ith "hard wired" control program designed by a human operator which has the major disadvantages as follows:

- Due to the rigid nature of the control program, the robot is not able to adapt with the changes in the environment (i.e. rough surface, surface color or texture) or to the robot itself (i.e. failure of some individual sensors).
- Beside, the design of fixed control program is costly as well.

As a result, learning controller has been developed to the extent that it allows the robots to acquire the competences to interact with the environment. The result is the control process based on the perception of the robot about the environment and the specific properties of the sensors. The robot now is able to learn through the entire period of operation, it has the capability to adapt well to the changes in the working environment. Actually, for the latter versions of mobile robot, CCD cameras, active sensors are u sually e mployed t o capture advanced information about the working environment. That information will be then processed by a sufficiently strong processor (i.e. IBM base PC, laptop, etc) by the means of capable softwares (i.e. MATLAB). They are providing the computation platform for environment mapping construction and various optimal algorithms for robot navigation to be implemented.

2.2 16F84 PIC MICROCONTROLLER

PIC 16F84 PIC chip is a versatile microcontroller with flash memory. The on board flash memory can endure a minimum of 1000 erase-write cycles; meaning that it can be reprogrammed and reused at least 1000 times. The 18-pin 16F84 devotes 13 of its pins for I/O. Each pin may be independently programmed as input or output. The pin status (I/O direction control) may also be changed via programming (assembly or C).

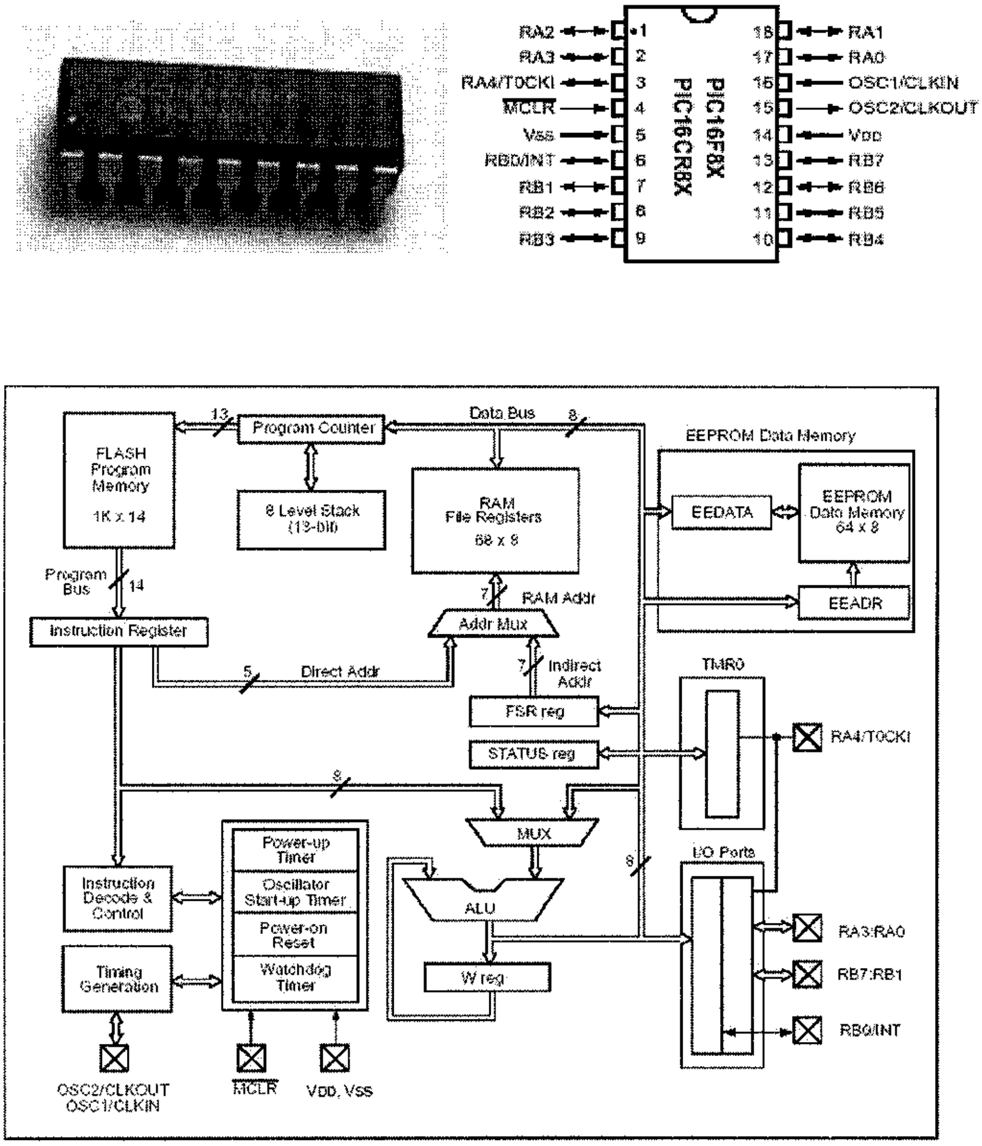


Figure 1 PIC 16F84 chip, IC pin out and block diagram

2.2.1 Programming 16F84 PIC to control servo motor

Servo motors are geared DC motors with a positional feedback control that allows the rotor to be accurately positioned. The motor shaft can be positioned through a minimum 90 degrees of rotation (from -90 to +90 degrees). To control the motor, the variable width pulse (PWM) needs to apply to the position control pin of the motor. The pulse is varied from 1 to 2ms. The width of the pulse controls the position of the shaft.

The 16F84 PIC can be programmed to control the servo motor. To be able to do so, a 1 to 2ms control pulse signal must be send from the microcontroller to the motor 50 to 60 times per second.

- 1ms pulse width will position the motor's shaft at the left position.
- 1.5 ms pulse for mid-position.
- 2ms pulse width for right position.

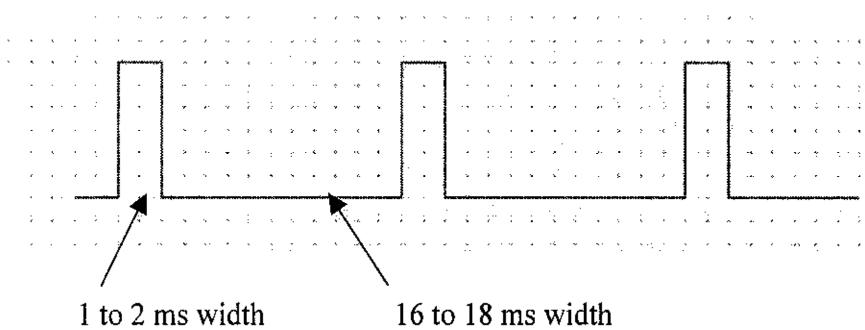


Figure 2 PWM pulse waveform for control servo motor positioning

2.3 DRIVE CIRCUIT

Normally when talking about motor directional rotation control, people refer to the H bridge circuit as shown below.

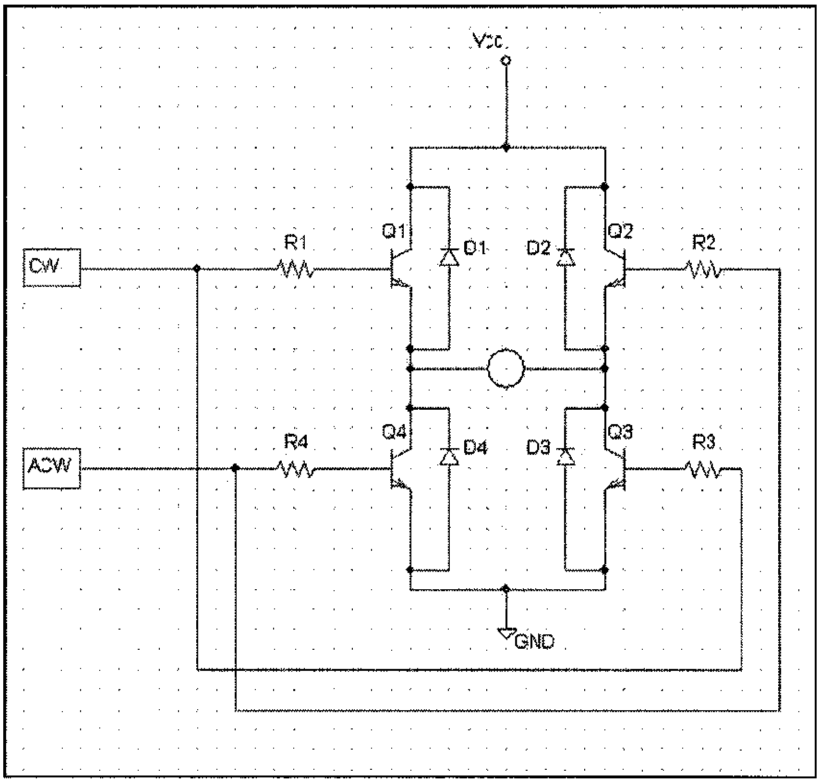


Figure 3 H- Bridge circuit

The operation of the circuit can be described through the following table.

Table 1 H-Bridge Truth Table

CW	AW	Q1	Q2	Q3	Q4	Motor rotating direction
On	Off	On	Off	On	Off	Clockwise
Off	On	Off	On	Off	On	Anticlockwise
Off	Off	Off	Off	Off	Off	Off
On	On	-	-	-	-	Not allowed

When CW and AW are both connected to 5 V, it is safe to have the motor at off state. The four diodes are to protect the transistor from the back EMF drawn back from the motor when it is rotating.

However, this circuit is not able to tolerate the large amount of current drawn from the motor under heavily loaded mode. Over-current can cause the driving transistors to be burnt. To overcome the problem, either power transistor or relay in place could be used. In the project, due to the very low on state resistance, which yields a quite high current rating (about 10 Amperes), MOSFET has been used to improve the efficiency of the bridge. The circuit schematic as well as operation is to be discussed in more details in the driving circuit section, **Chapter 4** (Results and Discussion).

2.4 MAPPING AND NAVIGATION

2.4.1 Mapping

Mapping refers to the construction of the environment model based on the measurements of sensors. This section describes two major map-constructing approaches: grid based and topological representation.

- **Grid based map**: represents the environment through the form of evenly spaced grids. Each grid cell contains a value, which indicates the presence or absence of an obstacle in the corresponding particular region of the environment. The grid-based map is considered easy to learn because they facilitate accurate localization and are easy to maintain.
- **Topological map**: represents the environment by graphs. The map consists of several nodes, which correspond to certain distinct location, situation on the robot navigating environment, connected by arcs if there is a direct path between them. The key advantage of the topological representation is its compactness, but requires fast planning.

There are few necessary concepts in map learning, and constructing:

- **Thresholding**: initially, each occupancy value in the grid is thresholded. Cells, whose occupancy values are less than the threshold, are considered free spaces (denoted as C). The others are considered occupied (denoted as D).
- **Voronoi diagram**: considering any point $(x, y) \in C$ in free space, the basic points of (x, y) are the closest points (x', y') in the occupied space (C'). This is referring to all points that minimize the Euclidean distance to (x, y). Those points are called basic points of (x, y) and their distance to (x, y) are called the clearance of (x, y). Voronoi diagram (which are represented in a form of skeletonization) is the set of points in free space that have at least two different basic points.
- **Critical points**: are the points on the Voronoi diagram that minimize the clearance locally. Finding the critical points is the key idea for partitioning the free space.

- **Critical lines**: are obtained by connecting each critical point with its basic points.

2.4.2 Localization

Localization is the process of aligning the robot local coordinate system with the global coordinate system of the map. It is really important for map-based approaches that lean their maps as the accuracy of the movement depends crucially on the alignment of the robot with its map. Traditionally, localization addresses two sub-problems: position tracking and global localization which are the two sides of the same coin: localization under uncertainty.

- **Position tracking**: is referred to the problem of estimating the location of the robot while it is moving with in its global map. It is particularly difficult to solve if the mapping is interleaved with localization.
- **Global localization**: is the problem of determining the position of the robot under global uncertainty. This problem is raised for instance when the robot uses a map that has been generated in the previous run, and is not informed about its initial location within the map.

2.4.3 Navigation

Navigation consists of two different modules: path planning and reactive collision avoidance. The path planning generates the minimum cost path to the target using the map (grid based or topological map). As a result, it communicates with the intermediate sub-goals generated by the path planner to the collision avoidance module which controls the velocity and exact motion direction of the robot based on the most recent measurement of sensors to avoid collision with obstacles. Both modules adjust their plans and control continuously in response to the current situation.

CHAPTER 3
METHODOLOGY AND PROJECT WORK

3.1 PROJECT METHODOLOGY

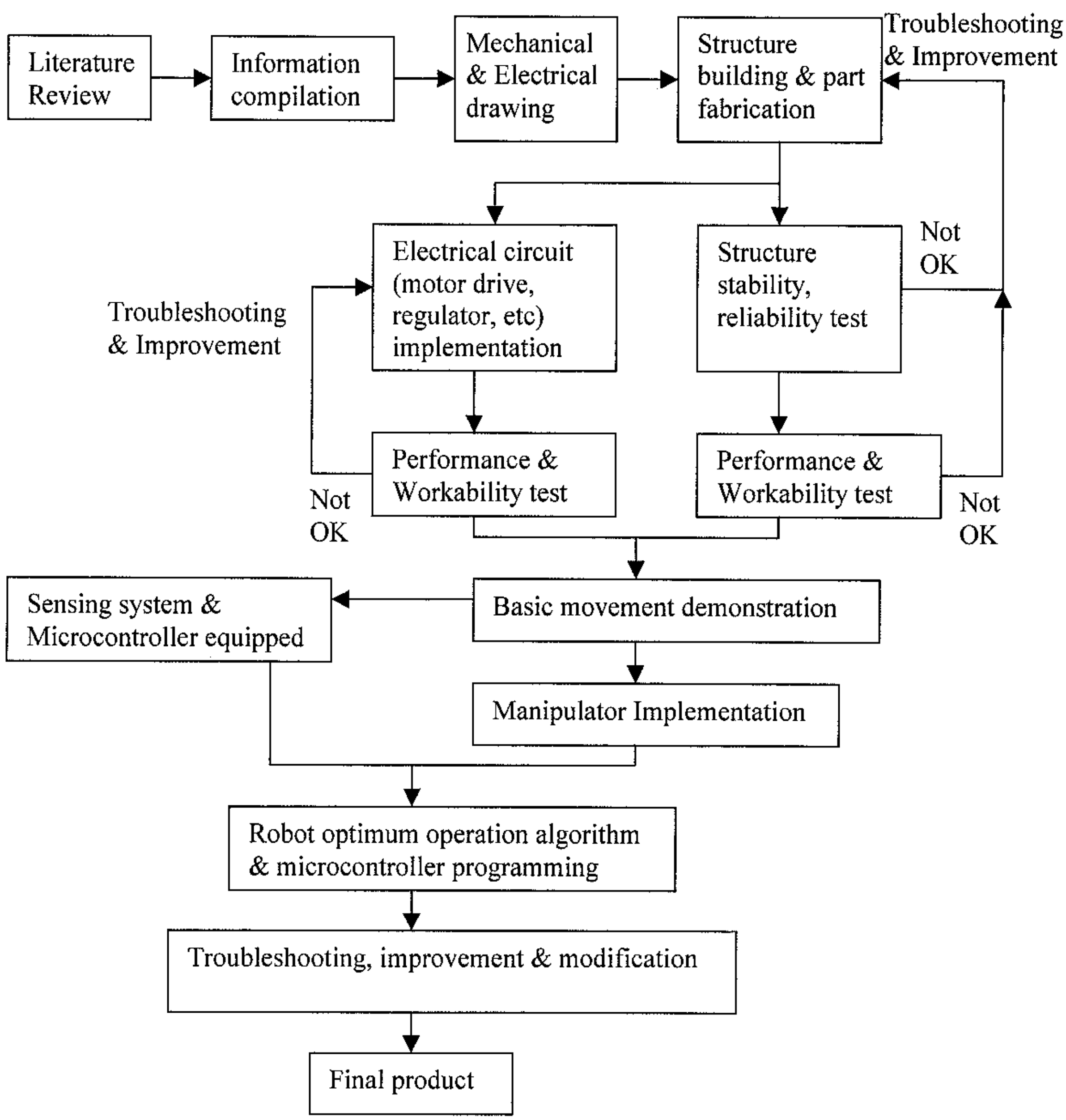


Figure 4 Project methodology

- **Literature Review & Information Compilation**: Extensive research has been done on the current mobile robot technology. Various resources (i.e. internet, related book on robotics, etc) have been accessed. This step is very critical, since it collected necessary information to prepare for further implementation steps of the project.
- **Electrical & Mechanical Drawing:** At this step, all the related electrical circuit (i.e. Drive circuit, regulator, etc) as well as the mechanical structure need to be drawn out. The mechanical drawing includes the detail on the robot structure (i.e. 3D with detailed dimension, etc).
- **Structure Building & Part Fabrication:** This is when the structure is built; motors and wheels are mounted to the robot base. This is followed by testing to access its stability and reliability in movement. Besides, the electrical implementation needs to be carried out in parallel to equip the robot with the necessary electrical hardware for the basic movement.
- **Performance & Workability Test:** is to test whether or not the built model (including the electrical circuits) is able to work according to the expectation.
- **Basic movement demonstration:** is to enable the robot to perform some basic movement (i.e. moving forwardly / reversely or making a turn).
- **Sensing System & Microcontroller equipped:** is to equip the sensors and the controller to the robot. The sensing system could be seen as the eyes of the whole system to percept, and measure the environment. All these information will be processed by the microcontroller for decision making.
- **Manipulator Implementation:** is to develop a small manipulator, enabling the robot for the pick and placement action.
- **Robot Optimum Operation Algorithm Implementation:** This is when all the needed algorithms (i.e. obstacle avoidance, navigation, etc) are embedded to the system where the microcontroller is considered as the tool, the platform to be implemented on.
- **Troubleshooting & Improvement:** needs to be done regularly to ensure a good product eventually.

Besides, to ensure the project to be able to meet the target, a project schedule has been planed out. The detail can be referred at **APPENDIX A**.

3.2 COMPONENTS

3.2.1 Mechanical Components

- 15cm diameter Custer wheels (2)
- 6 cm diameter standard trolley rubber covered wheel (1)
- Aluminum Alloy Coupling (3)
- Metal screws, compression springs, nuts.
- L shape angle mild iron bar with predrilled holes – 2 m.
- U shape hollow Aluminum Alloy bar – 1 m.

3.2.2 Electrical Components

- Futaba servo motor: 6 V (3), 12 V (1).
- 12 V RS DC motor (3)
- Microchip PIC (16F84-04 PIC) (3).
- 4 MHz crystal (1).
- N-channel MOSFET (4)
- P-channel MOSFET (4)
- BJT Q3904 transistor (4).
- RS single pole 6V relay (1).
- RS double pole 6V relay (1).
- LM7806 (1).
- LM7906 (1).
- Infrared sensors (3).
- Ultrasonic transducer (1).
- 741 Op-amp (4).
- LM339 (1).
- Capacitors, BJT transistors, resistors, switches, LEDs, etc (as per needed).
- Battery pack: 12 V 2A lead –acid standard motorbike battery source (1).

CHAPTER 4

RESULTS AND DISCUSSION

4.1 MECHANICAL STRUCTURE

In order to obtain the optimized structure for the robot, several mechanical layouts have been designed and put in consideration.

4.1.1 The first approach

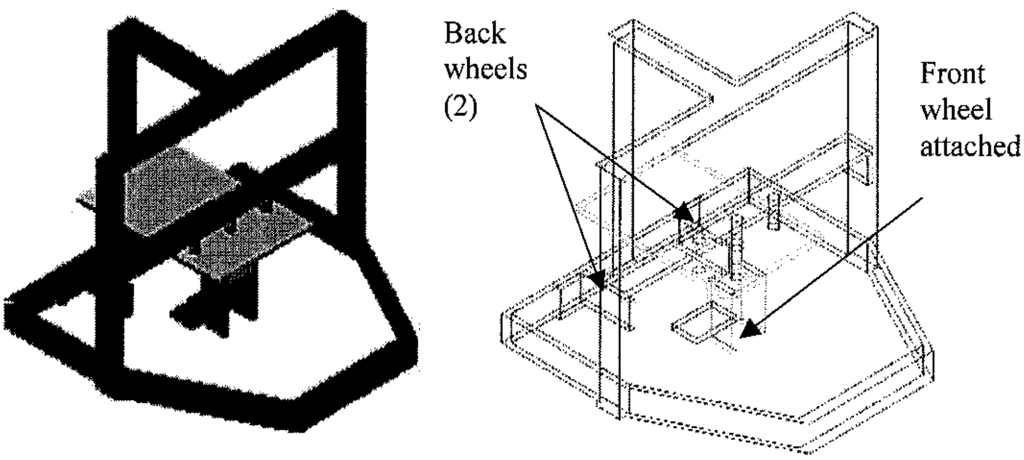


Figure 5 The first robot structure (without wheels)

This structure contains of 1 controllable front wheel (controlled via servo motor) for changing the direction of the robot; and two fixed position back wheels driven by DC motors. The whole structure could be implemented using solid aluminum alloy bar. Even though this design offers the strength as well as the light weight to the robot, the implementation also issues the associated cost (solid aluminum alloy bar is a relatively costly material for the project). The main problems raised are the difficulties in bending the aluminum alloy bar to the designed shape, also the balancing of the whole structure while carrying the heavy load (the structure does not have sufficient space to locate the load evenly). Therefore, it is decided not to further follow this layout to build robot base structure, instead seeking for the cheaper version while still offering the stability and the ease in implementation.

4.1.2 The second approach

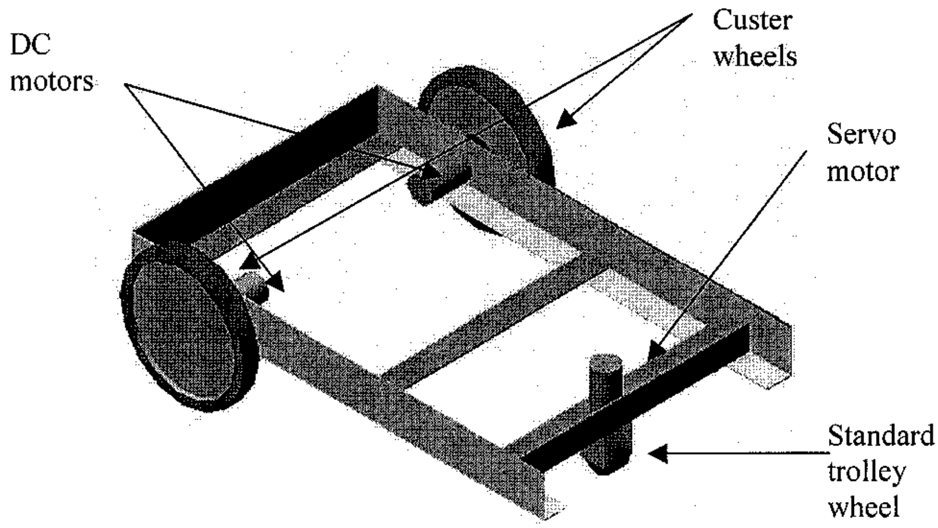


Figure 6 Base structure layout

The above structure comes with the mild iron bar skeleton. It is really convenience to be assembled since the mild iron bar comes in the shape of predrilled thin L shape bar which offering the strength and flexibility, but sufficiently light weighted. Those predrilled holes are made in various shapes and dimensions, which in turn let the motor be mounted on quite easily. Thus the implementation issue can be overcome. Moreover, this type of material is at a very cheaper price compared with the solid aluminum alloy bar, hence reduces the project cost.

The wheels are attached to the motor shafts through the Aluminum alloy couplings treated as the shaft extension. They are tightly screwed together, resulting in the stable and reliable movement as well as synchronizations. The detail drawings are shown in **APPENDIX B**.

For the mentioned above advantages, the second approach was chosen. The practical implementation showed the promised results. The structure is quite stable during the robot movement with sufficiently heavy load.

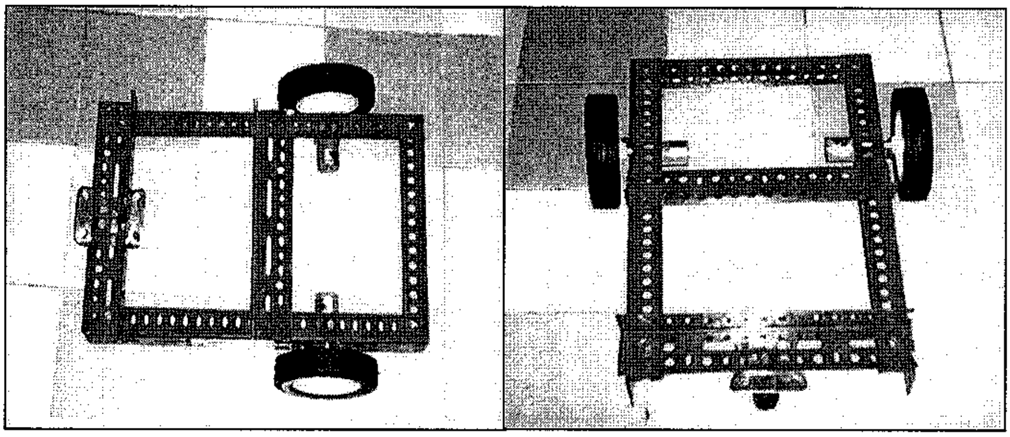


Figure 7 Robot Base structure in implementation.

4.2 MOTOR DRIVE CIRCUIT

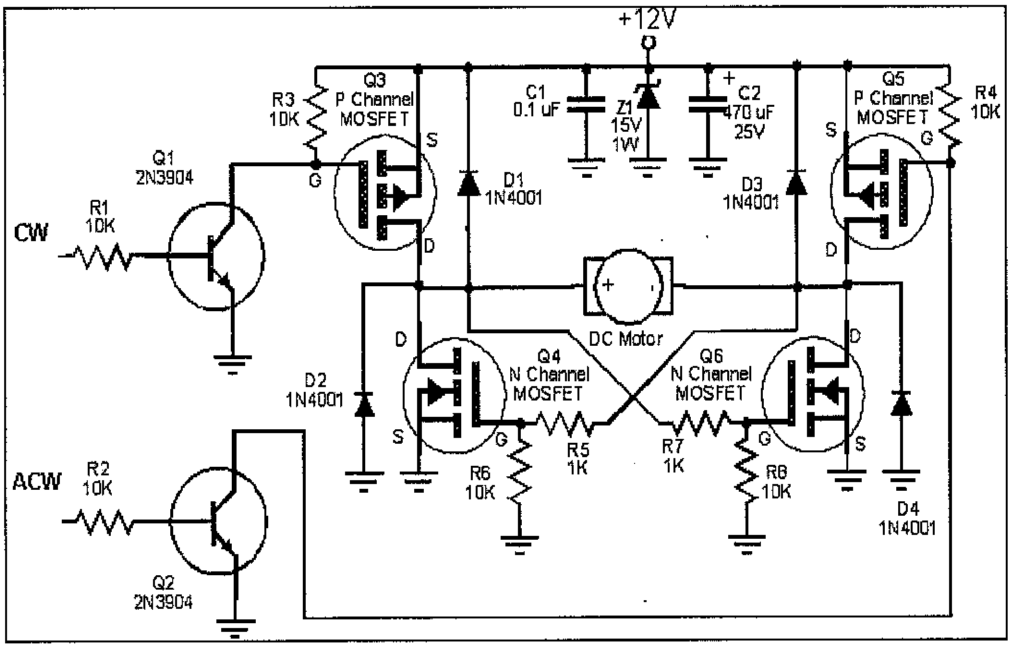


Figure 8 H-Bridge employed MOSFET as switch

As mentioned earlier, to be able to improve the efficiency as well as enable the circuit to be capable to handle higher current rating, MOSFETs have been employed as the major switching devices in the H-bridge circuit. The following table details the circuit operation.

Table 2 H-bridge operating true table

CW	AW	Q1	Q2	Q3	Q4	Q5	Q6	Motor rotating Mode
On	Off	On	Off	On	Off	Off	On	Clockwise
Off	On	Off	On	Off	On	On	Off	Anticlockwise
Off	Off	Off	Off	Off	Off	Off	Off	Stop
On	On	-	-	-	-	-	-	Not allowed

Stop mode

When CW=0 and ACW=0, the motor is stopped. R3 and R4 pull up the Gates of Q3 and Q5 respectively and turn off the MOSFETs.

Anti-clockwise mode

When CW=0 and ACW=1 (+5V), the motor is in anti-clockwise direction. Q1 is turned off and Q3 is turned off due to R3. Q2 is turned on by the voltage at B. Q2's collector pulls Q5's Gate to ground. This turns on Q5 (P channel needs more - negative voltage than Source to turn on). The negative side of the motor is raised to +12V. R5 raises Q4's Gate to around +11.5 V which turns on Q4. Q4's Drain goes to ground which makes the positive side of the motor go to ground. R7 is also connected to the positive side of the motor which pulls down Q6's Gate and makes sure that it is turned off. The current path for the motor is from +12V to Q5 to negative contact to positive contact to Q4 to ground.

Clockwise mode

When CW=1 and ACW=0, the motor is in clockwise direction. Q2 is turned off and Q5 is turned off due to R4. Q1 is turned on due to the voltage at A and Q1's collector goes to ground. This turns on Q3 raising the motor's positive side to +12V. R7 raises Q6's Gate voltage and turns it on. When Q6 turns on, R5 makes sure that Q4 remains off. The current path for the motor is from +12V to Q3 to positive contact to negative contact to Q6 to ground.

Not allowed mode

If CW=1 and ACW=1 then all MOSFETs turn on which shorts out the power supply among other things. This situation is not allowed to be occurred.

4.3 MANIPULATOR DESIGN AND IMPLEMENTATION

To enable the robot to have a simple pick and place capability, in this project we also include the design and implementation of a small manipulator, which is designed to have 2.5 DOF. What meant by 2.5 DOF is that it has two motor-controlled joints and one dependent joint which could be counted as half DOF as shown below.

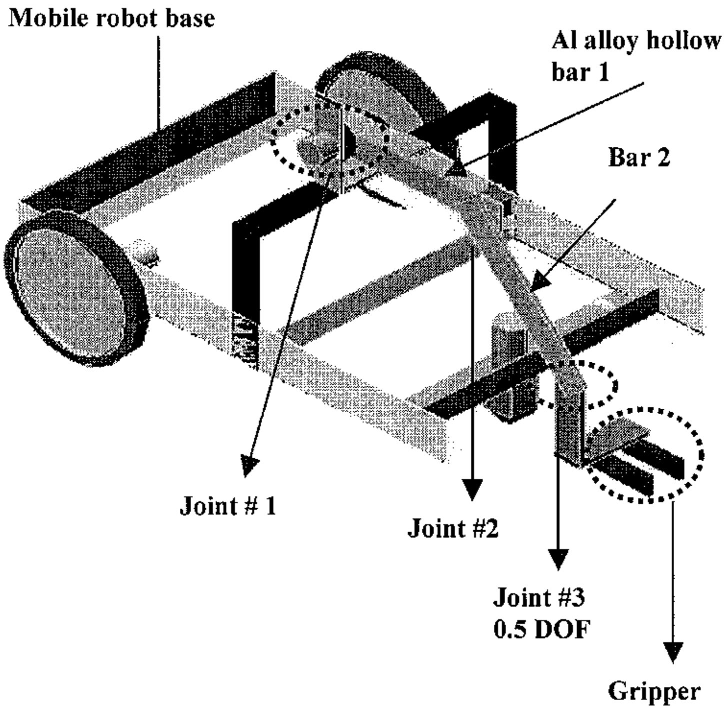


Figure 9 Manipulator structure layout

4.3.1 Joint #1

This joint is very critical since it withstands the whole manipulator load. Hence geared heavy duty 12 V RS DC motor is used here to be able to bring up and down the manipulator during pick up and placement of the object. For this situation, two stoppers are used at the back and in front of the joint to position the attached Al alloy bar when it hits the stoppers. In order to minimize the shaking/vibration of the structure when the Al Alloy bar hits the stoppers, the following relay circuit (which employs *latching circuits* to interface PIC output with relays) was implemented.

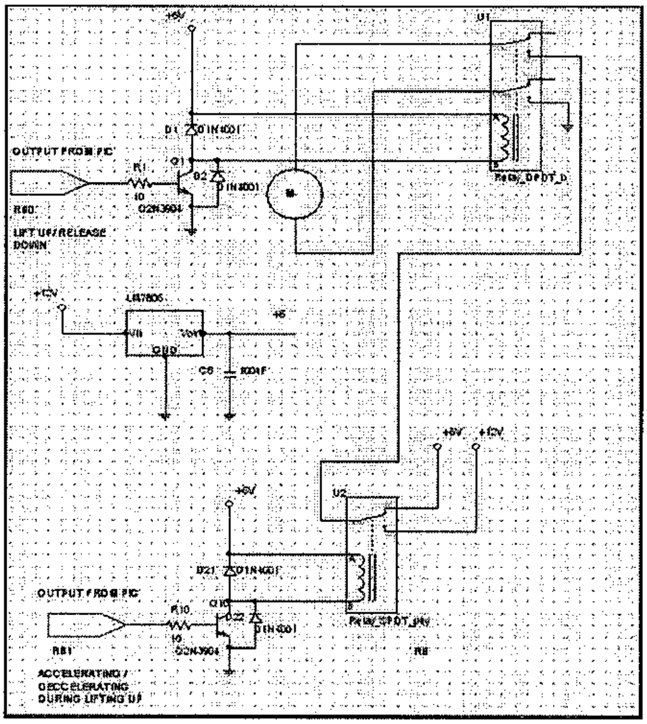


Figure 10 Joint #1 relay control circuit

4.3.1.1 Latching circuit

Due to the fact that the output from the PIC microcontroller is less than 20mA which is not sufficient to directly turn on / off the relay, a circuit network consisted of a BJT transistor and 2 D1N4001 Diodes is used for interfacing purpose between the PIC and the controlled relays.

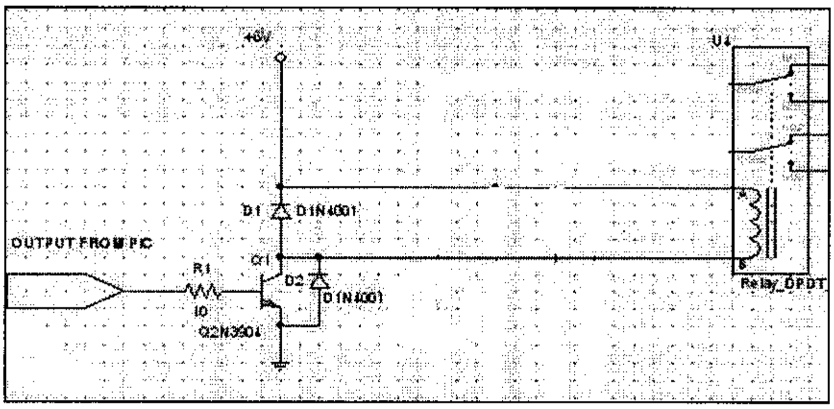


Figure 11 Latching circuit

The circuit operation can be described as follows:

- When the control signal (CS) is turned high (+5V), Q1 on, current will flow from the external source through the relay coil, passing the transistor to ground→ the coil is energized, switching on the relay.
- When CS is low, Q1 off, the relay coil is de-energized, relay off.

With this circuit, the drive (Figure 10) can easily accept the TTL logic inputs to control the motor motion as described below.

4.3.1.2 Joint #1 Relay control circuit operation

In order to lift up the manipulator during pick up operation, RB0 (output from PIC 16F84) goes high, activating the U1 relay to on position. At the same time, RB1 goes high, turns on U2 relay, and connects the motor terminal to the 12 V 2A battery source. The DC motor will rotate clockwise for about 2 seconds to lift the arm up. At this point of time, the Al alloy bar should have almost hit the back stopper, RB1 goes low, switches to 6V power source (output from LM7806 regulator) to be applied to the motor terminal. The motor now acts as a generator supplying back the current to the source which in turn brakes and slows down significantly the motor speed and torque so that when the Bar hits the stopper, the vibration is reduced.

To bring the arm down during placement action, RB0 goes low to turn off the DC motor. The arm will freely drop down due to gravity for about 1 second. After 1 second, RB0 goes high again. At this moment, RB1 has been already low. The motor terminal is applied to the 6V regulated source (which is not sufficient to lift the arm up again) to help the motor fight against the gravity. Therefore, it could act as a brake to reduce the dropping speed of the arm which in turn significantly reduces the vibration while the arm hits the front stopper.

4.3.2 Joint #2

A 12V Futaba coreless servo motor is used at this critical joint as it requires accuracy in positioning. One output (i.e. RB0) from another PIC 16F84 microcontroller is connected to the positioning control terminal of the servo motor.

During lifting up the object, PWM pulse of 1.6ms is given to the motor, let it bring the object a little bit from the floor (with the gripper holding). Then the whole manipulator is lifted up by the DC motor, PWM pulse of 1.1 ms is applied to bring down and place the object on the carrier platform of the robot (with the gripper releasing).

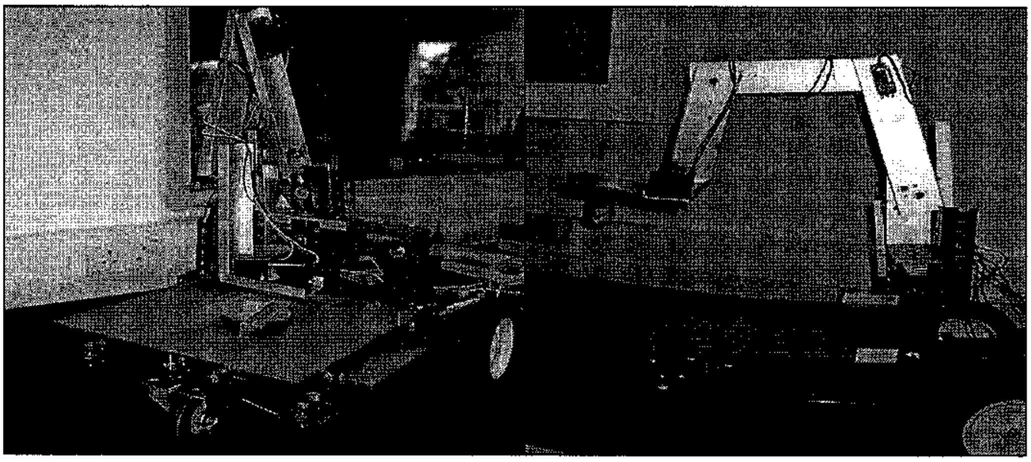
In order to bring down and place the object on the floor, after the whole arm (with the gripper holding) is lifted up by the DC motor. PWM pulse of 1.6 ms is applied on the servo motor terminal, bringing up the object out of the carrier platform. The DC motor then releases the arm down and brakes it accordingly until it reaches the front stopper. PWM pulse of 1.4 ms is now applied to finely bring down and place the object on the floor (with the gripper releasing).

4.3.3 Joint #3 (0.5 DOF)

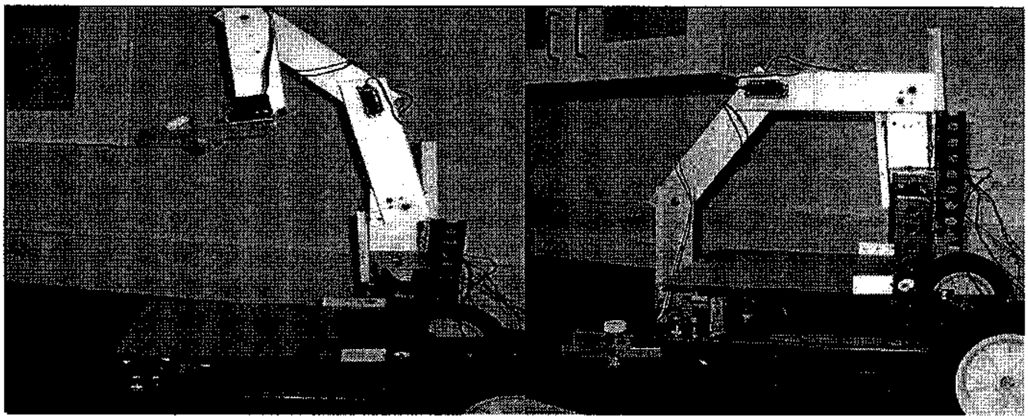
This is a dependent joint which is loosely screwed to the end of **Bar 2** (figure 9). Because of that, the gravity will play an active role to keep the gripper always perpendicular to the floor (horizontal axis) during pick up and placement. The gripper is also made from Al alloy hollow bar attached to this particular 0.5 DOF joint. The opening and closing of the two fingers are basically controlled through the use of two 6V Futaba indirect drive servo motors. These motors will help the fingers hold the object tightly by moving accurately towards each other. An infrared sensing circuit is also attached to the gripper letting it detect the desired object to pick up.

4.3.4 Manipulator in implementation

In this part, the manipulator in picking up and placing a small cylindrical box located from a known location on the floor was tested. The operation was so smooth that it was capable to handle the job well. First the object was picked up and put inside the carrier platform on the robot from the floor, then brought up and place back at predefined location. The followings are some photos taken of the manipulator in action as demonstrated below.



Manipulator stationed on carrier platform   Manipulator during placement operation



Manipulator during picking up   Object are placed back on the table

Figure 12 Manipulator in action

4.4 MOTOR SHAFT ENCODER

In order to record the movement of the wheel so that the position of the robot can be tracked, the motor shaft encoders are required. Unfortunately, the currently used RS DC motor in the robot does not consist of an internal encoder, thus it is necessary to construct our own ones. The basic mechanism is described as follows:

- The wheel is divided evenly into bright and dark sectors as in figure 13.
- IR (infrared) sensor (circuit is as shown in figure 14) is then used to sense the changing on brightness level while the wheel rotates which in turn keeps track for the number of rotations of the wheel so that we could know how far the robot has traveled.

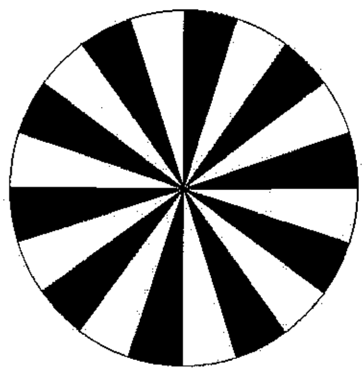


Figure 13 The wheel is equally divided into dark and bright section

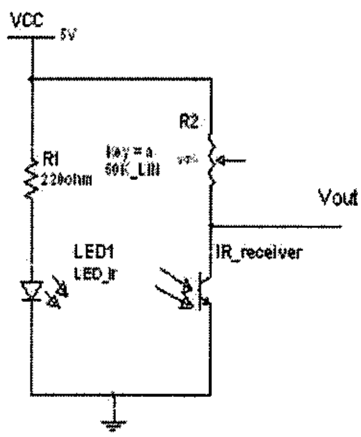


Figure 14 IR sensing circuit

4.4.1 IR (infrared) sensing circuit operation

- When the sensor encounters the dark surface on the wheel which actually absorbs the light transmitted from the IR emitter , thus there is no IR light reflected back to the receiver, the receiver is then acting as an open circuit→ $V_{out} = 5$ V
- When the sensor encounters the bright surface on the wheel which actually reflects back the light transmitted from the IR transmitter to the receiver, the receiver is then acting as a short circuit→ $V_{out} = 0$V
- The transition between low and high (or high and low) is then counted through the mean of a microcontroller. The number of pulses counted is actually related to the distance traveled by the robot through the following formula:

Distance = (number of pulses) x (pi x diameter) / (number of sectors)

In our case, number of sectors is 20; diameter is 15 cm, therefore:

Distance = (number of pulses) (3.14) (15cm)/ (20)
= 2.355(number of pulses) – cm

which yields a maximum possible tolerance of about 2.355 cm.

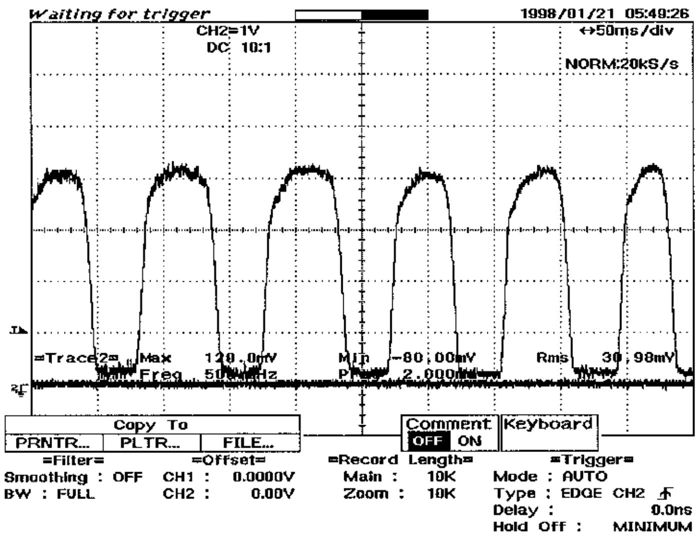


Figure 15 Wheel encoder output pulses

4.5 OBSTACLE DETECTION

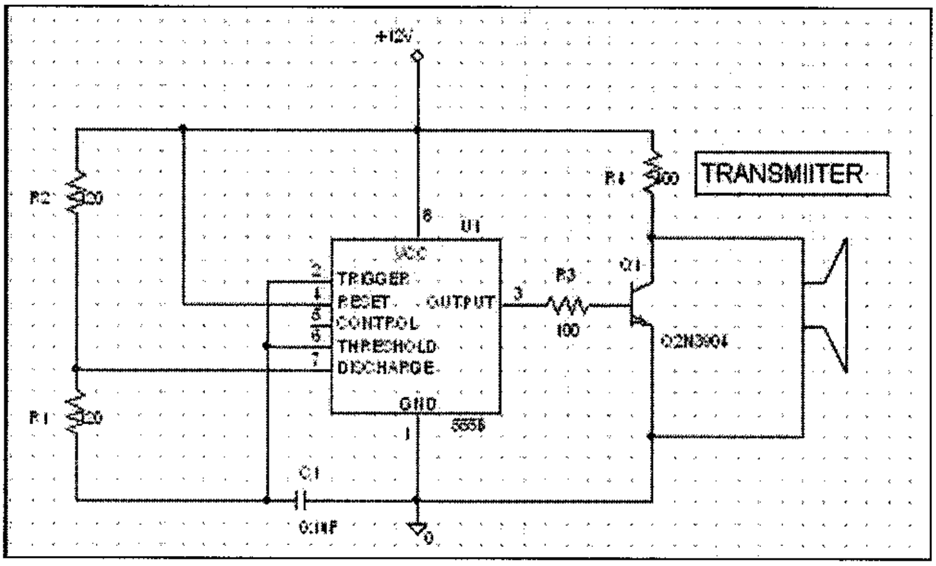


Figure 16 Ultrasonic transmitter (40KHz)

In this project, to enable the robot to detect any obstacles blocking its route, an ultrasonic sensing system is designed and implemented. Figure 16 shows the transmitter circuit.

4.5.1 Ultrasonic Transmitter

The 40 KHz Ultrasonic sound is generated through the clock signal produced from a 555 timer circuit.

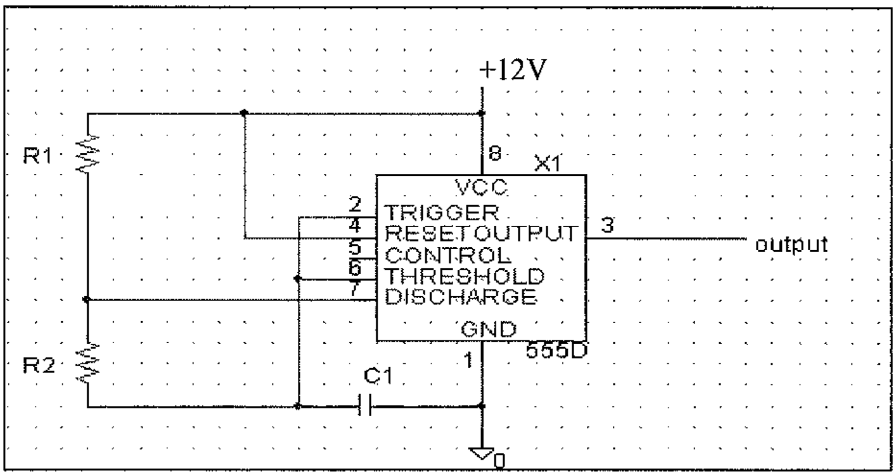


Figure 17 555 timer network

The circuit parameters are calculated below to produce a 40 KHz clock signal.

$$f = \frac{1.44}{(R_1 + 2R_2)C}$$

$$D = \frac{R_1}{R_1 + R_2} 100\% = 50\% \Rightarrow R_1 = R_2 = R$$

$$\rightarrow f = \frac{1.44}{3RC} = 40KHz$$

Choose C = 0.1μF, then

$R = 120\Omega$

Hence, in order to produce a 40 KHz, 50% duty cycle clock signal

$R_1 = \mathrm{R}_2 = 120\Omega$

$C = 0.1\mu F$

The generated clock signal (40 KHz) is then used to trigger an NPN BJT transistor network (as shown in figure 16) so that it could be amplified before reaching the ultrasonic transmitter unit. The transmitted sound is then propagated in the air, and will be reflected back upon hitting any object within its operating range.

4.5.2 Ultrasonic Receiver

This unit is to detect the reflected ultrasonic sound, and induce an AC voltage with according to how strong the reflect signal is. It means that the induced voltage across the receiver (terminals) will be varied accordingly to how far the object apart from the sensing system (mounted in front of the robot). The following table shows the experimental result obtained by measuring the induced voltage across the receiver terminals while varying the position of the object.

Table 3 Induced voltage (Peak Voltage- mV) versus Sensing distance

Sensing distance	Induced Voltage	Sensing distance	Induced Voltage
10 cm	400 mV	60 cm	200 mV
20 cm	368 mV	70 cm	120 mV
30 cm	340 mV	80 cm	40mV
40 cm	300 mV	90 cm	10mV
50 cm	248 mV	100 cm	10mV

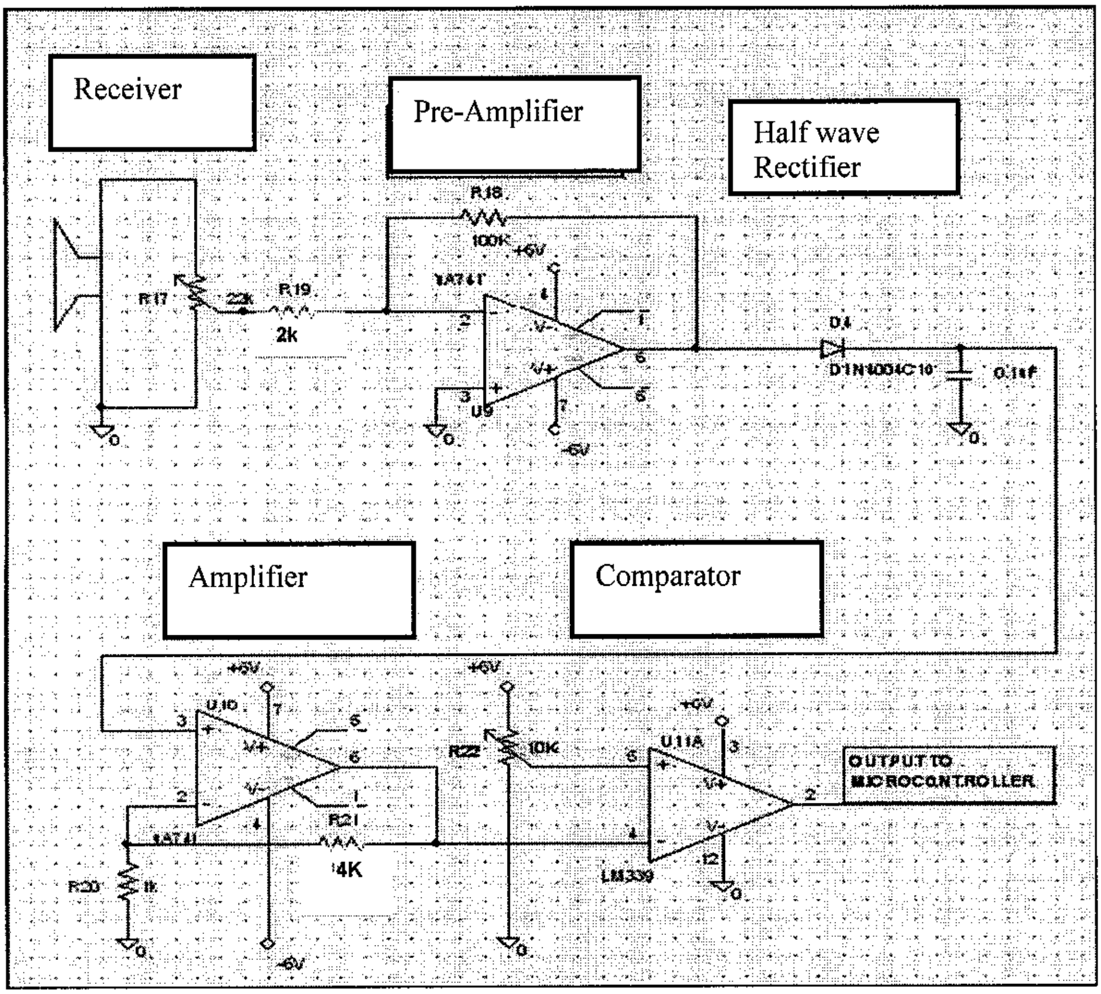


Figure 18 Ultrasonic receiver

Based on the experimental results as shown in table 3, the receiver circuit is designed carefully in the way that it could detect the object as far as 80 cm apart from its mounted position in front of the robot. Figure 18 shows the circuit schematic of the ultrasonic receiver which includes 5 processing stages as discussed below:

- **Receiving stage**: Once the receiver unit receives the reflected sound, an AC voltage (sinusoidal) is induced across its terminal.
- **Pre-Amplify**: The induced voltage is then amplified through an inverting amplifying op-amp network at a gain of -50.

$Gain = -\frac{R18}{R19}$, if R18 = 100 KΩ, R19 = 2KΩ, then $Gain = -50$

If the induced peak voltage is 40 mV at a sensing distance of 80 cm, then after this stage, it will be amplified to 40(50)=2000 mV =2V but 180 degrees out of phase. However, since the supplies for the op-amp are only +6 V and -6V, the amplified signal is limited or saturated at +6V and -6V as shown in the following table.

Table 4 Pre-amplified signal (peak value)

Sensing distance	Induced Voltage	Sensing distance	Induced Voltage
10 cm	6V	60 cm	6V
20 cm	6V	70 cm	6V
30 cm	6V	80 cm	2V
40 cm	6V	90 cm	0.5 V
50 cm	6V	100 cm	0.5V

- **Rectifying stage**: is to convert the pre-amplified signal to DC voltage through a simple half wave diode rectifier circuit. The average DC output is calculated through the following formula:

$$V_0 = \frac{V_{in} - 0.7}{\Pi} \cong \frac{V_{in} - 0.7}{3.14}$$

Where V_{in} is the peak value of the input signal. 0.7 refers to the voltage dropped across the diode

Table 5 Rectified voltage versus sensing distance

Sensing distance	Rectified Voltage	Sensing distance	Rectified Voltage
10 cm	1.68 V	60 cm	1.68 V
20 cm	1.68 V	70 cm	1.68V
30 cm	1.68 V	80 cm	0.41 V
40 cm	1.68 V	90 cm	0 V
50 cm	1.68 V	100 cm	0 V

- **Amplifying stage:** this stage is to amplify the rectified signal to a higher level. In order to do this, a non-inverting amplifying op-amp network is designed at a gain of 5.

$$Gain = 1 + \frac{R21}{R20}, \text{if } R20 = 1\,K\Omega, R21 = 4K\Omega, \text{then } Gain = 5$$

- **Comparing stage:** this DC amplified signal will be then compared a fixed voltage which could be varied through the variable resistor (R22) (connected to the non-inverting pin of the LM 339 comparator). If R22 is chosen so that the voltage applied to the non-inverting pin is slightly smaller than 0.41 x 5 = 2.05 V, then whenever an object appears within the sensing distance from (10 cm to 80 cm), the output of the comparator will go low and vice versa. This output is then tapped to the PIC 16F84 controller for obstacle detection and avoidance subroutine to be executed.

Table 6 Sensing Distance versus Output Voltage from the comparator

Sensing distance	**Output voltage**	**Sensing distance**	**Output voltage**
10 cm	0	60 cm	0
20 cm	0	70 cm	0
30 cm	0	80 cm	0
40 cm	0	90 cm	6V
50 cm	0	100 cm	6V

4.6 PIC 16F84 MICROCONTROLLER AND ROBOT MOVEMENT

In this project, in order to incorporate the robot movement and the manipulator operation together, three units of PIC 16F84 as the controllers are employed. These three communicate to each other through the relationship of master and slave. The master is solely in charged of the robot movement (i.e. move straight, make a turn, obstacle avoidance) while the slaves are used to control the operation of the manipulator upon getting the triggering signal sent from the master. One is to control the positioning of manipulator arm during pick and placement, the other is in charged of controlling the opening and closing action of the gripper attached to the end of bar 2 (refer to figure 9).

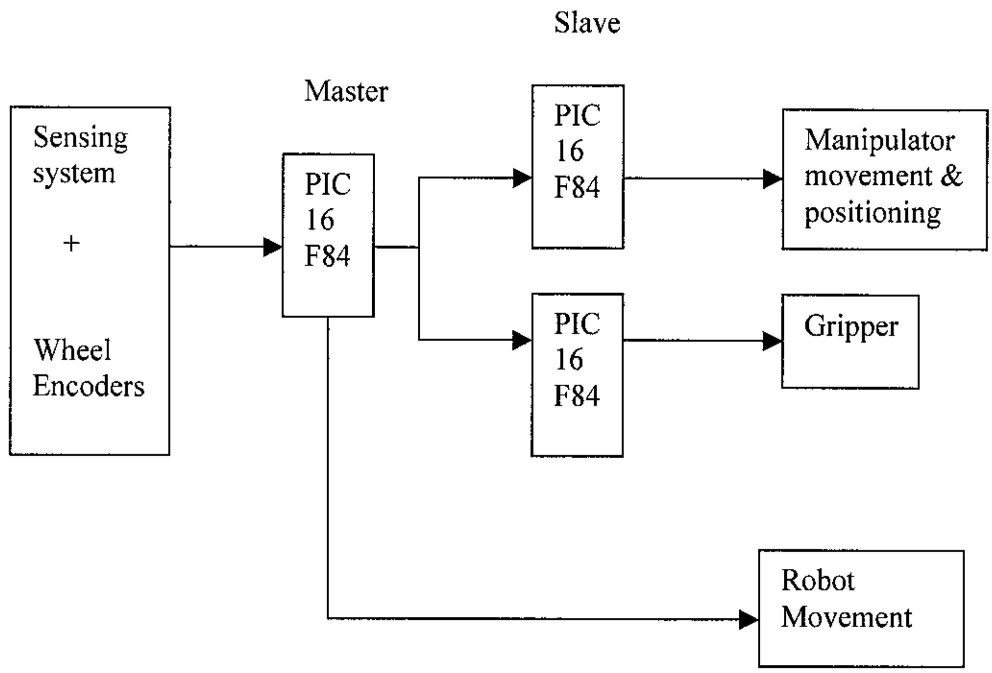


Figure 19 Microcontroller communication block diagram

4.6.1 Robot Programming Functional Blocks

As already discussed, the robot should be able to maneuver according to a predetermined path while avoiding any obstacles blocking its movement. Besides, it is also required to perform simple pick and place tasks. Therefore, the robot programming model should consist of the following functional blocks:

- Navigation: is in charged of all on-surface motion of the robot while communicating with two modules: path planning and obstacle detection and avoidance. However, in this project, the robot movement path is preplanned; therefore, the path planning module is skipped.
- Position tracking: is involved with the problem of estimating the robot location while it is moving within its movement plan. This is done through counting the input pulses from the motor shaft encoders as already discussed in section 4.
- Obstacle detection & Avoidance: is in charged of obstacle detection and avoidance while communicating with the navigation module as well as position tracking module.
- Manipulator & Gripper Control block: is in charged of controlling the pick and place operation of the robot (discussed in section 4.3). It also communicates with the navigation module and also the position tracking module in completing the tasks.

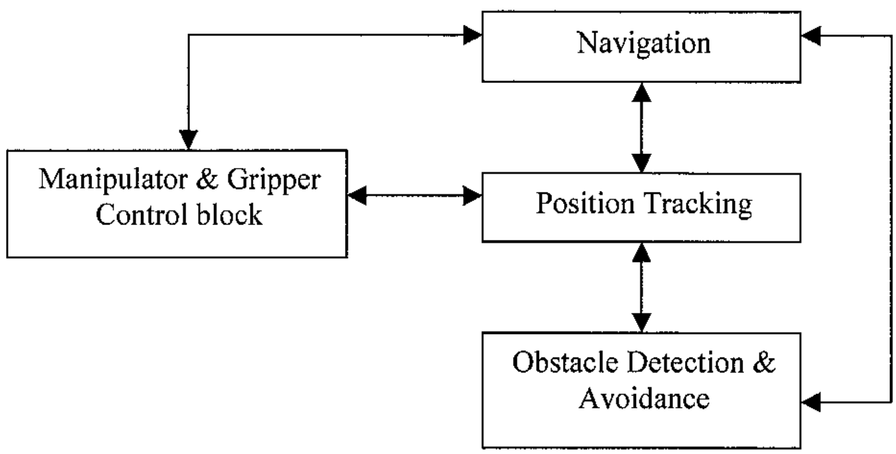


Figure 20 Robot Programming Functional blocks

4.6.2 Movement Plan and Robot Navigation

We have set up a movement plan to which the robot is able to follow.

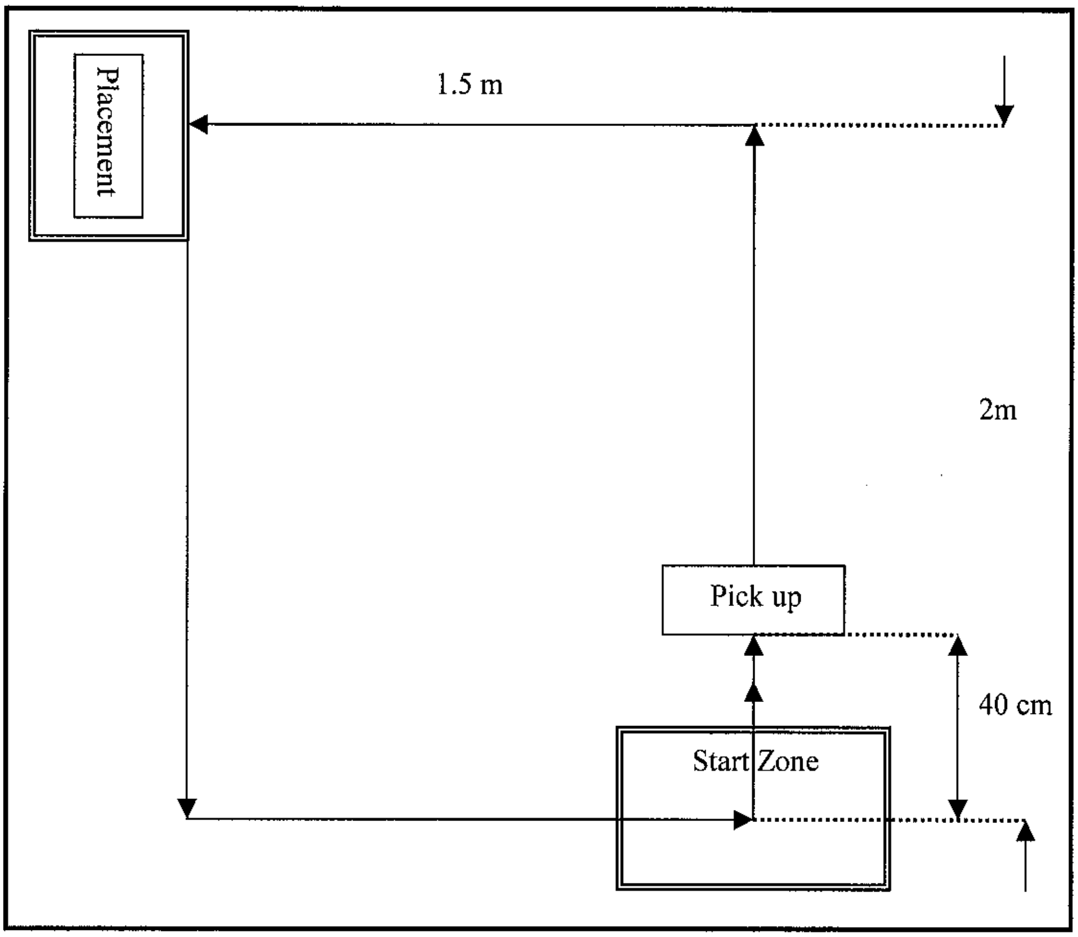


Figure 21 Movement plan

Description:

- The robot starts from its start zone, moves forward about 40 cm to pick up a small object placed anywhere within from 20 cm to 40 cm in front of the robot.
- Upon picking up, it places the object on its carrier platform, travels 160 cm more to complete the 2 m route, makes a left turn, and moves toward to left side for 150 cm.
- The robot is now at its placement location. It places the object well inside the predetermined boundaries, moves reversely for 4 cm, brings its arm up, makes a left turn and returns to its parking position (start zone).

4.6.3 Obstacle Avoidance

While moving accordingly to its predetermined path toward to destination or returning to the parking position, the robot might encounter obstacles. In these conditions, the ultrasonic sensing unit mounted in front of the robot will detect the existence of the obstacle within a sensing distance of 80 cm. An active low signal then triggers the microcontroller to activate its obstacle avoidance subroutine.

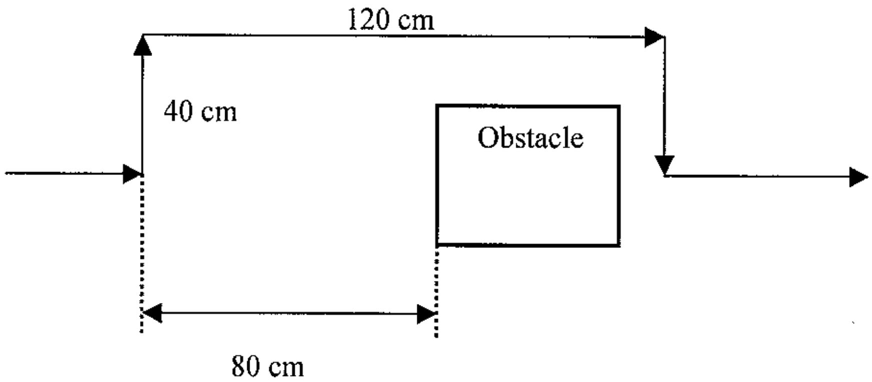


Figure 22 A Simple Obstacle Avoidance mechanism

To simulate the situation, the obstacles used to test the obstacle avoidance capability of the robot are boxes; with the fixed size of 30 x 30 x 40 cm. Figure 22 depicts a simple obstacle avoidance mechanism:

- Upon sensing unit detects the obstacle within a distance of 80 cm, the robot stops. At this point of time, it needs to identify its position referred to the movement plan through counting the pulses input from the wheel encoders.
- To avoid the obstacle, the robot then makes a left turn and moves forward to 40 cm. Based on the object dimension, the robot now escapes from being locked by the obstacle.
- It makes a right turn, moves forward for 120 cm to overcome the obstacle, turns right, again moves forward for 40 cm and makes a left turn eventually to return to its route. At this moment, the robot again needs to identify its position with referring to the movement plan so that it will know how far it has been moving from the start zone as well as how far it is from the destination, and then it only continues to move toward the target.

4.6.4 Some pictures of the robot at work

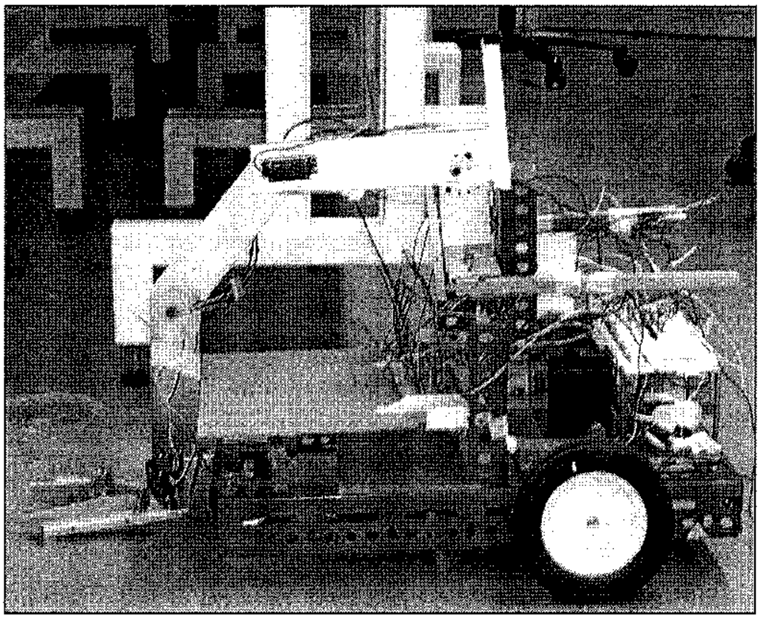


Figure 23 The robot grasps the object on the floor

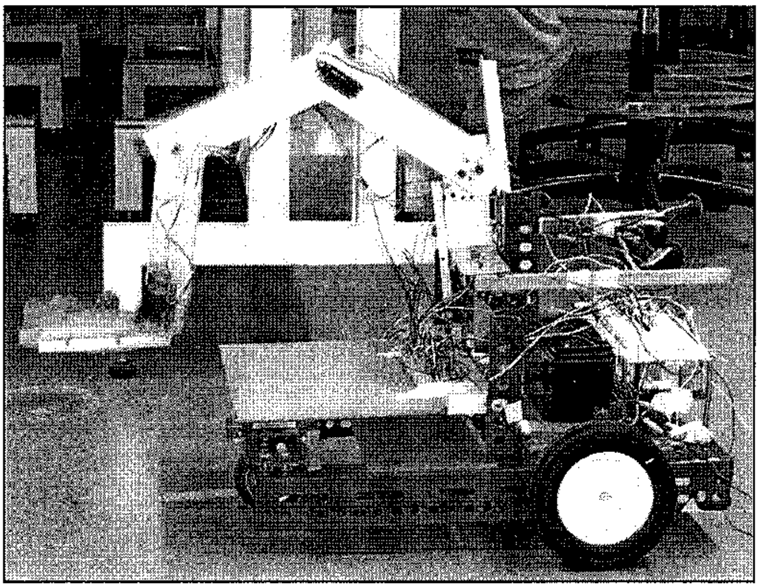


Figure 24 During picking up

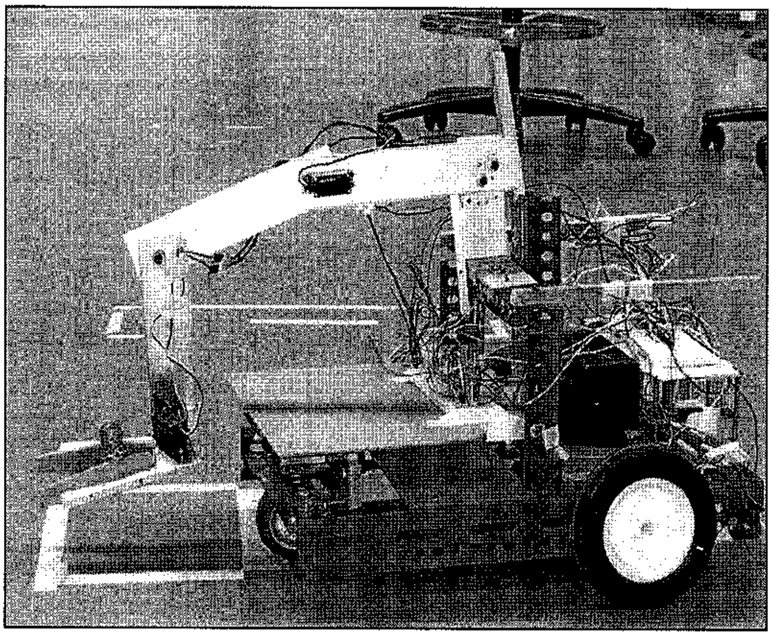


Figure 25 The robot places the object into a predetermined location

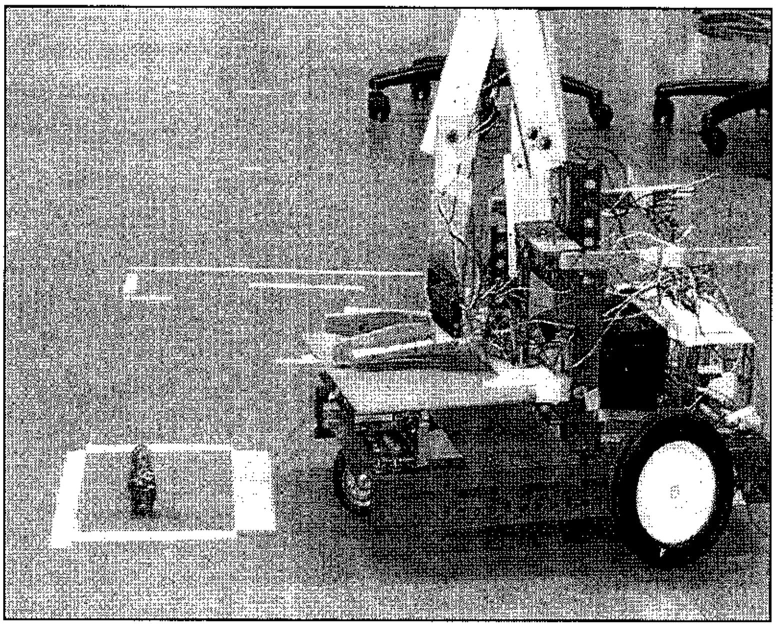


Figure 26 After placement

CHAPTER 5
CONCLUSION

The project objectives have been successfully met. The robot is not only able to move according to a predetermined plan while avoiding any obstacle blocking its route; but also able to do some simple pick and place tasks. To do all those, the robot programming is consisted of four building functional blocks: (1) Navigation (2) Position tracking (3) Obstacle detection and avoidance (4) Manipulator and gripper control block. The robot has incorporated these mentioned functional blocks by using 3 different units of master-slave communicated PIC 16F84 microcontrollers. The three units of microcontrollers have been working and communicating to each other accordingly, which in turn yields quite a smooth operation of the robot.

To further improve the performance, there are some recommendations as following:

- The manipulator programming needs some fine tuning so that it could operate more smoothly during picking up.
- Currently, there are several types of built-in two-finger gripper available in the market. Therefore, if possible, we could get one to equip to the robot manipulator to enhance the pick and place capability.
- More complicated obstacle avoidance algorithm needs to be implemented so that the robot would be able to react more precisely and effectively towards the changes in its working environment.
- A user interface is also in need to enable the user to select the robot path.

Besides, since this is the first time we have involved in implementing a big-scaled mobile robot, we have encountered a lot of issues, especially with the mechanical structure fabrication. Thus, it would be better if we could be provided a good robot base structure (with the supporting from the UTP mechanical department). Then we would proceed with all the related electrical work instead of implementing the base structure by ourselves (which might contain some mechanical defects, causing some misalignment in the robot movement latter).

REFERENCES

[1] Kortenkamp, Bonasso and Murphy, Artificial Intelligence and Mobile Robot, 1998, AAAI Press / MIT Press.

[2] Jose A. Castellanos, Mobile Robot Localization and Map Building, 2000, KAP

[3] Penelope Robert Smith, Active sensors for local planning in mobile robot, 2001, World Scientific.

[4] Martin David Adams, Sensor Modeling, Design and Data processing for autonomous navigation, 1999, World Scientific.

[5] Joseph L.Jones, Mobile Robots, 2^{nd} edition, 1998, AK Peters

APPENDIX A

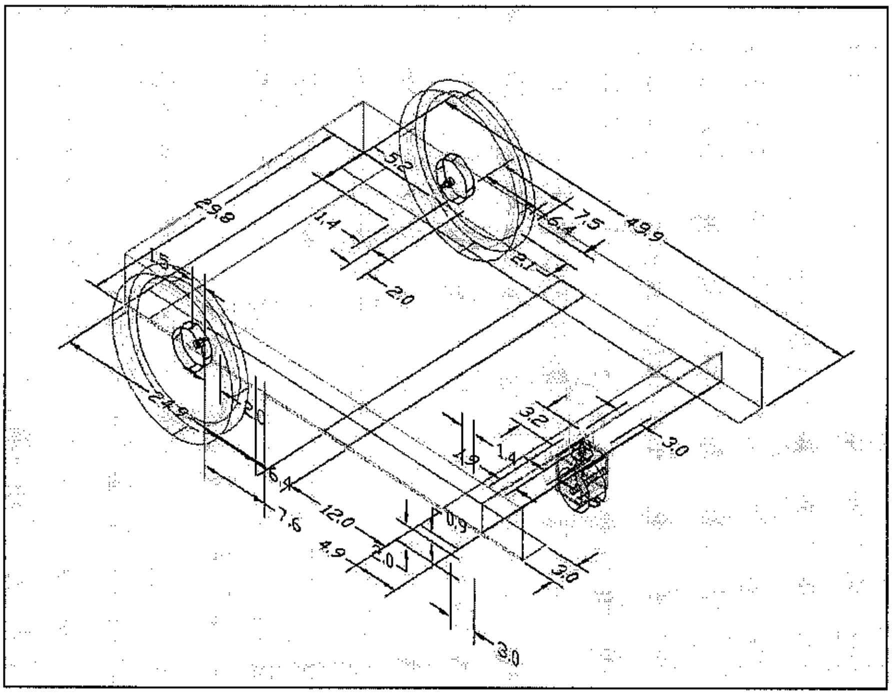
PROJECT GENERAL SCHEDULE

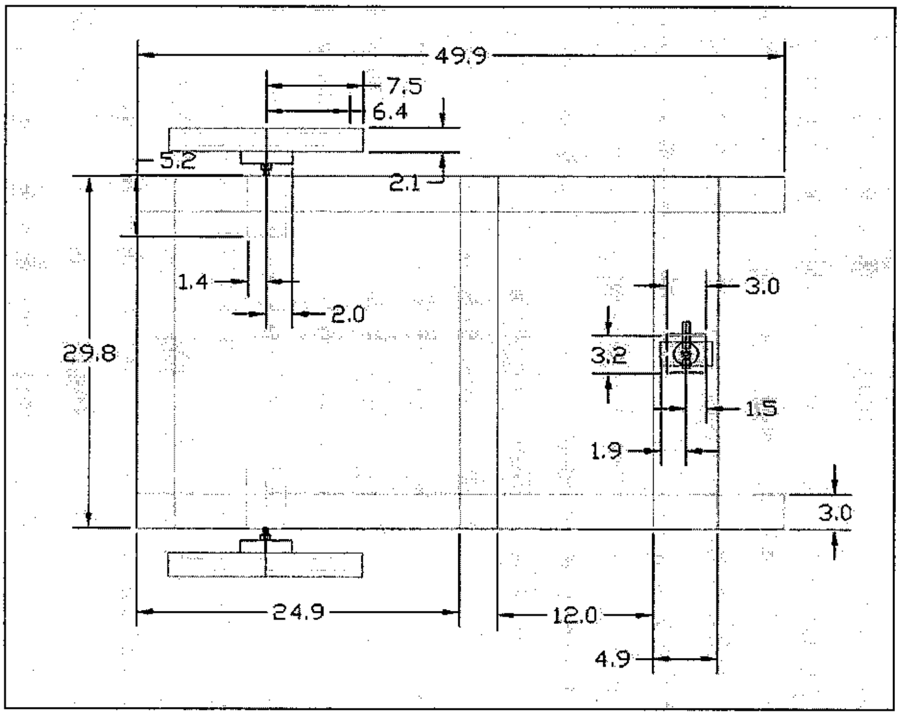
Week / Tasks	1	2	3	4	5	6	7	8	9	10	11	12	13	14	Break
Literature review on mobile robot technology.															
Information compilation & basic conceptual layout.															
Detail mechanical & electrical wiring drawing.															
Robot assembled, structure built & motor drives & rotational parts design.															
Structure stability, reliability, workability test & trouble shooting.															
Manual control development															
Manipulator Implementation															
Sensing system + microcontroller implementation. Robot optimum operating algorithms & microcontroller programming.															
Improvement & modification															

APPENDIX B

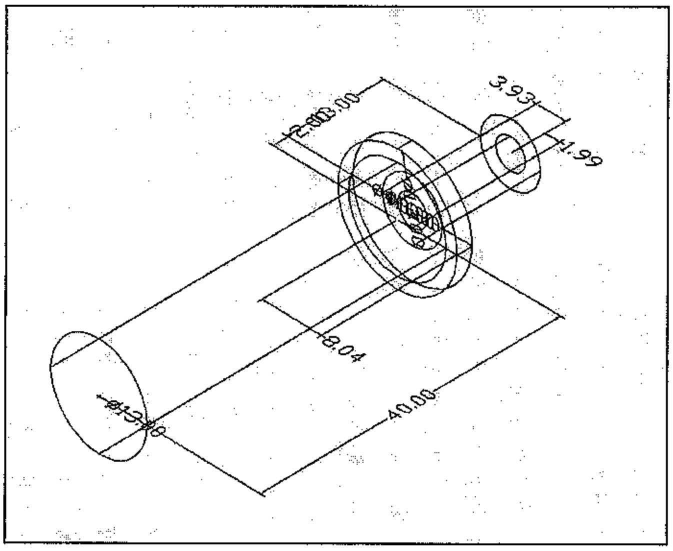
MECHANICAL DETAIL DRAWING

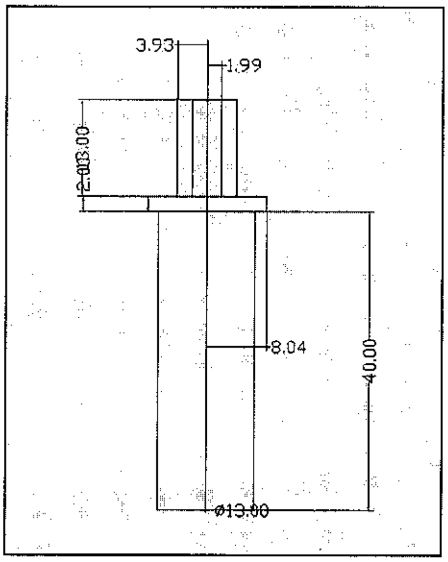
B1. Base Structure Drawing (Dimension in cm)





B2. Coupling Drawing (Dimension in mm)





APPENDIX C

PIC 16F84 DATASHEET

MICROCHIP

PIC16F84A
Data Sheet

18-pin Enhanced FLASH/EEPROM
8-bit Microcontroller

Note the following details of the code protection feature on PICmicro® MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable".
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

Trademarks

The Microchip name and logo, the Microchip logo, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, KEELOQ, SEEVAL, MPLAB and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

Total Endurance, ICSP, In-Circuit Serial Programming, Filter-Lab, MXDEV, microID, *Flex*ROM, *fuzzy*LAB, MPASM, MPLINK, MPLIB, PICC, PICDEM, PICDEM.net, ICEPIC, Migratable Memory, FanSense, ECONOMONITOR, Select Mode and microPort are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Term Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

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

Printed on recycled paper.

DNV Certification, Inc. USA
DNV MSC The Netherlands Accredited by the RvA
ISO 9001 / QS-9000 REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.

PIC16F84A

MICROCHIP

18-pin *Enhanced* FLASH/EEPROM 8-Bit Microcontroller

High Performance RISC CPU Features:

- Only 35 single word instructions to learn
- All instructions single-cycle except for program branches which are two-cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle
- 1024 words of program memory
- 68 bytes of Data RAM
- 64 bytes of Data EEPROM
- 14-bit wide instruction words
- 8-bit wide data bytes
- 15 Special Function Hardware registers
- Eight-level deep hardware stack
- Direct, indirect and relative addressing modes
- Four interrupt sources:
  - External RB0/INT pin
  - TMR0 timer overflow
  - PORTB<7:4> interrupt-on-change
  - Data EEPROM write complete

Peripheral Features:

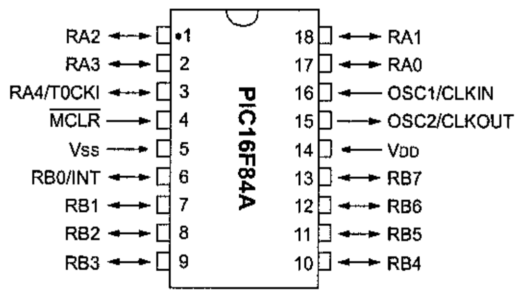
- 13 I/O pins with individual direction control
- High current sink/source for direct LED drive
  - 25 mA sink max. per pin
  - 25 mA source max. per pin
- TMR0: 8-bit timer/counter with 8-bit programmable prescaler

Special Microcontroller Features:

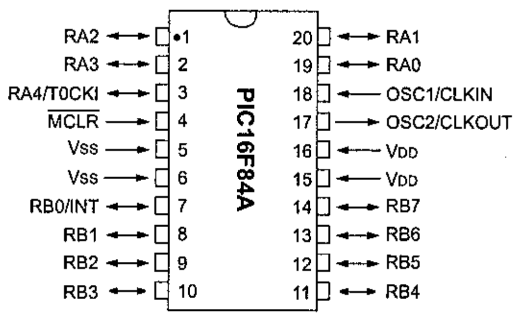
- 10,000 erase/write cycles *Enhanced* FLASH Program memory typical
- 10,000,000 typical erase/write cycles EEPROM Data memory typical
- EEPROM Data Retention > 40 years
- In-Circuit Serial Programming™ (ICSP™) - via two pins
- Power-on Reset (POR), Power-up Timer (PWRT), Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own On-Chip RC Oscillator for reliable operation
- Code protection
- Power saving SLEEP mode
- Selectable oscillator options

Pin Diagrams

PDIP, SOIC



SSOP



CMOS Enhanced FLASH/EEPROM Technology:

- Low power, high speed technology
- Fully static design
- Wide operating voltage range:
  - Commercial: 2.0V to 5.5V
  - Industrial: 2.0V to 5.5V
- Low power consumption:
  - < 2 mA typical @ 5V, 4 MHz
  - 15 µA typical @ 2V, 32 kHz
  - < 0.5 µA typical standby current @ 2V

Table of Contents

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@mail.microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com/cn** to receive the most current information on all of our products.

1.0 DEVICE OVERVIEW

This document contains device specific information for the operation of the PIC16F84A device. Additional information may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023), which may be downloaded from the Microchip website. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

The PIC16F84A belongs to the mid-range family of the PICmicro® microcontroller devices. A block diagram of the device is shown in Figure 1-1.

The program memory contains 1K words, which translates to 1024 instructions, since each 14-bit program memory word is the same width as each device instruction. The data memory (RAM) contains 68 bytes. Data EEPROM is 64 bytes.

There are also 13 I/O pins that are user-configured on a pin-to-pin basis. Some pins are multiplexed with other device functions. These functions include:

- External interrupt
- Change on PORTB interrupt
- Timer0 clock input

Table 1-1 details the pinout of the device with descriptions and details for each pin.

FIGURE 1-1: PIC16F84A BLOCK DIAGRAM

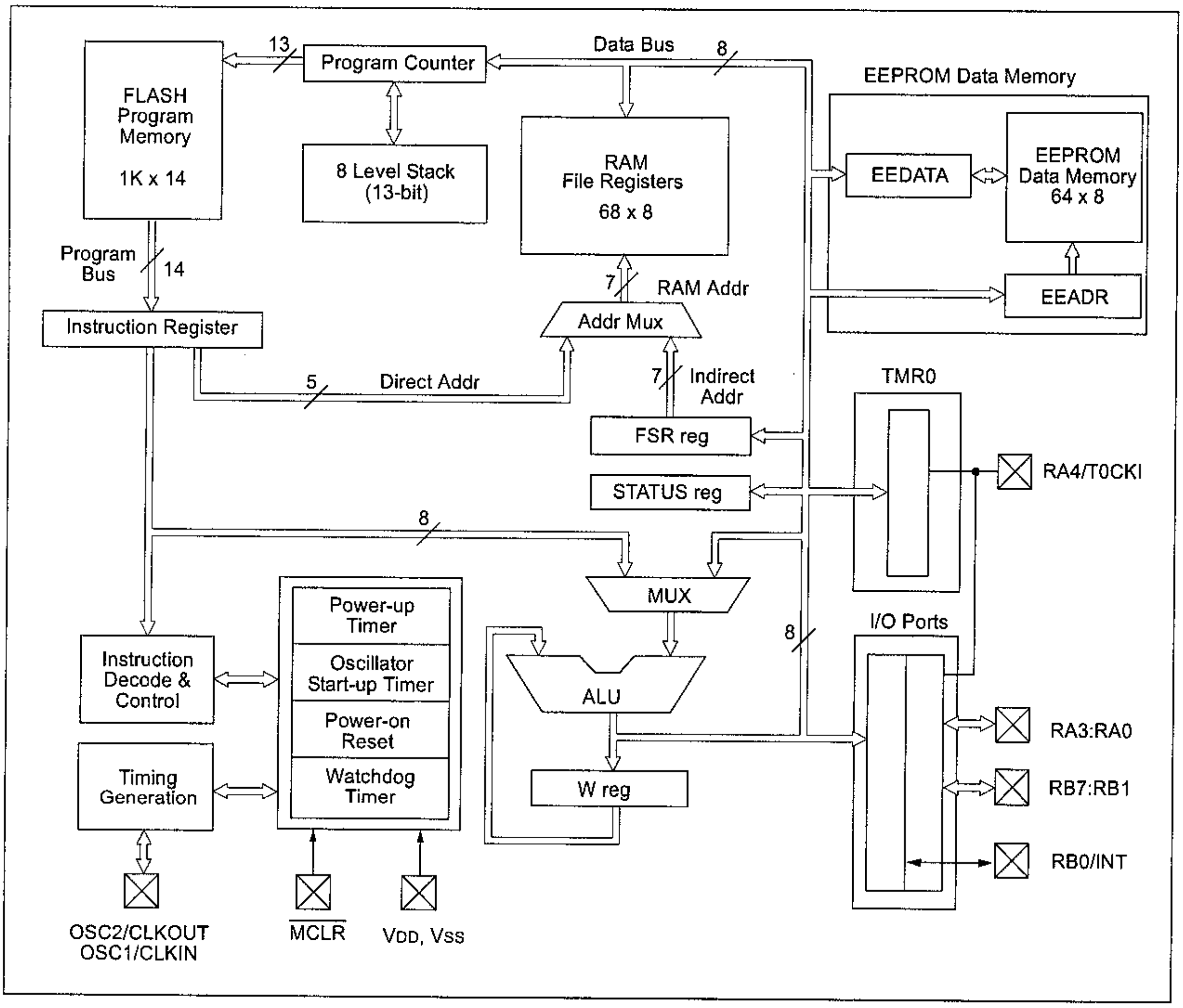


TABLE 1-1: PIC16F84A PINOUT DESCRIPTION

Pin Name	PDIP No.	SOIC No.	SSOP No.	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	16	16	18	I	ST/CMOS(3)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	15	15	19	O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKOUT, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
$\overline{MCLR}$	4	4	4	I/P	ST	Master Clear (Reset) input/programming voltage input. This pin is an active low RESET to the device.
						PORTA is a bi-directional I/O port.
RA0	17	17	19	I/O	TTL	
RA1	18	18	20	I/O	TTL	
RA2	1	1	1	I/O	TTL	
RA3	2	2	2	I/O	TTL	
RA4/T0CKI	3	3	3	I/O	ST	Can also be selected to be the clock input to the TMR0 timer/counter. Output is open drain type.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	6	6	7	I/O	TTL/ST(1)	RB0/INT can also be selected as an external interrupt pin.
RB1	7	7	8	I/O	TTL	
RB2	8	8	9	I/O	TTL	
RB3	9	9	10	I/O	TTL	
RB4	10	10	11	I/O	TTL	Interrupt-on-change pin.
RB5	11	11	12	I/O	TTL	Interrupt-on-change pin.
RB6	12	12	13	I/O	TTL/ST(2)	Interrupt-on-change pin. Serial programming clock.
RB7	13	13	14	I/O	TTL/ST(2)	Interrupt-on-change pin. Serial programming data.
V_{SS}	5	5	5,6	P	—	Ground reference for logic and I/O pins.
V_{DD}	14	14	15,16	P	—	Positive supply for logic and I/O pins.

Legend: I= input O = Output I/O = Input/Output P = Power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

2.0 MEMORY ORGANIZATION

There are two memory blocks in the PIC16F84A. These are the program memory and the data memory. Each block has its own bus, so that access to each block can occur during the same oscillator cycle.

The data memory can further be broken down into the general purpose RAM and the Special Function Registers (SFRs). The operation of the SFRs that control the "core" are described here. The SFRs used to control the peripheral modules are described in the section discussing each individual peripheral module.

The data memory area also contains the data EEPROM memory. This memory is not directly mapped into the data memory, but is indirectly mapped. That is, an indirect address pointer specifies the address of the data EEPROM memory to read/write. The 64 bytes of data EEPROM memory have the address range 0h-3Fh. More details on the EEPROM memory can be found in Section 3.0.

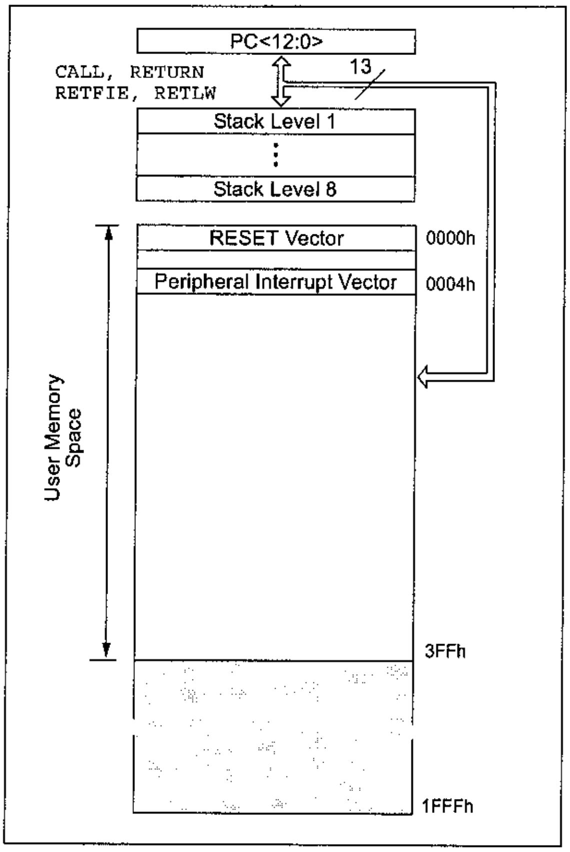
Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

2.1 Program Memory Organization

The PIC16FXX has a 13-bit program counter capable of addressing an 8K x 14 program memory space. For the PIC16F84A, the first 1K x 14 (0000h-03FFh) are physically implemented (Figure 2-1). Accessing a location above the physically implemented address will cause a wraparound. For example, for locations 20h, 420h, 820h, C20h, 1020h, 1420h, 1820h, and 1C20h, the instruction will be the same.

The RESET vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK - PIC16F84A



2.2 Data Memory Organization

The data memory is partitioned into two areas. The first is the Special Function Registers (SFR) area, while the second is the General Purpose Registers (GPR) area. The SFRs control the operation of the device.

Portions of data memory are banked. This is for both the SFR area and the GPR area. The GPR area is banked to allow greater than 116 bytes of general purpose RAM. The banked areas of the SFR are for the registers that control the peripheral functions. Banking requires the use of control bits for bank selection. These control bits are located in the STATUS Register. Figure 2-2 shows the data memory map organization.

Instructions `MOVWF` and `MOVF` can move values from the W register to any location in the register file ("F"), and vice-versa.

The entire data memory can be accessed either directly using the absolute address of each register file or indirectly through the File Select Register (FSR) (Section 2.5). Indirect addressing uses the present value of the RP0 bit for access into the banked areas of data memory.

Data memory is partitioned into two banks which contain the general purpose registers and the special function registers. Bank 0 is selected by clearing the RP0 bit (STATUS<5>). Setting the RP0 bit selects Bank 1. Each Bank extends up to 7Fh (128 bytes). The first twelve locations of each Bank are reserved for the Special Function Registers. The remainder are General Purpose Registers, implemented as static RAM.

2.2.1 GENERAL PURPOSE REGISTER FILE

Each General Purpose Register (GPR) is 8-bits wide and is accessed either directly or indirectly through the FSR (Section 2.5).

The GPR addresses in Bank 1 are mapped to addresses in Bank 0. As an example, addressing location 0Ch or 8Ch will access the same GPR.

FIGURE 2-2: REGISTER FILE MAP - PIC16F84A

File Address	Bank 0	Bank 1	File Address
00h	Indirect addr.(1)	Indirect addr.(1)	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2(1)	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch–4Fh	68 General Purpose Registers (SRAM)	Mapped (accesses) in Bank 0	8Ch–CFh
50h–7Fh			D0h–FFh

☐ Unimplemented data memory location, read as '0'.

Note 1: Not a physical register.

2.3 Special Function Registers

The Special Function Registers (Figure 2-2 and Table 2-1) are used by the CPU and Peripheral functions to control the device operation. These registers are static RAM.

The special function registers can be classified into two sets, core and peripheral. Those associated with the core functions are described in this section. Those related to the operation of the peripheral features are described in the section for that specific feature.

TABLE 2-1: SPECIAL FUNCTION REGISTER FILE SUMMARY

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on RESET	Details on page
Bank 0											
00h	INDF	Uses contents of FSR to address Data Memory (not a physical register)								---- ----	11
01h	TMR0	8-bit Real-Time Clock/Counter								xxxx xxxx	20
02h	PCL	Low Order 8 bits of the Program Counter (PC)								0000 0000	11
03h	STATUS[2]	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	8
04h	FSR	Indirect Data Memory Address Pointer 0								xxxx xxxx	11
05h	PORTA[4]	—	—	—	RA4/T0CKI	RA3	RA2	RA1	RA0	---x xxxx	16
06h	PORTB[5]	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	xxxx xxxx	18
07h	—	Unimplemented location, read as '0'								—	—
08h	EEDATA	EEPROM Data Register								xxxx xxxx	13,14
09h	EEADR	EEPROM Address Register								xxxx xxxx	13,14
0Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of the PC[1]					---0 0000	11
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	10
Bank 1											
80h	INDF	Uses Contents of FSR to address Data Memory (not a physical register)								---- ----	11
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	9
82h	PCL	Low order 8 bits of Program Counter (PC)								0000 0000	11
83h	STATUS [2]	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	8
84h	FSR	Indirect data memory address pointer 0								xxxx xxxx	11
85h	TRISA	—	—	—	PORTA Data Direction Register					---1 1111	16
86h	TRISB	PORTB Data Direction Register								1111 1111	18
87h	—	Unimplemented location, read as '0'								—	—
88h	EECON1	—	—	—	EEIF	WRERR	WREN	WR	RD	---0 x000	13
89h	EECON2	EEPROM Control Register 2 (not a physical register)								---- ----	14
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of the PC[1]					---0 0000	11
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	10

Legend: x = unknown, u = unchanged. - = unimplemented, read as '0', q = value depends on condition

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a slave register for PC<12:8>. The contents of PCLATH can be transferred to the upper byte of the program counter, but the contents of PC<12:8> are never transferred to PCLATH.

2: The $\overline{TO}$ and $\overline{PD}$ status bits in the STATUS register are not affected by a $\overline{MCLR}$ Reset.

3: Other (non power-up) RESETS include: external RESET through $\overline{MCLR}$ and the Watchdog Timer Reset.

4: On any device RESET, these pins are configured as inputs.

5: This is the value that will be in the port output latch.

2.3.1 STATUS REGISTER

The STATUS register contains the arithmetic status of the ALU, the RESET status and the bank select bit for data memory.

As with any register, the STATUS register can be the destination for any instruction. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to device logic. Furthermore, the $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

Only the `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions should be used to alter the STATUS register (Table 7-2), because these instructions do not affect any status bit.

Note 1: The IRP and RP1 bits (STATUS<7:6>) are not used by the PIC16F84A and should be programmed as cleared. Use of these bits as general purpose R/W bits is NOT recommended, since this may affect upward compatibility with future products.

2: The C and DC bits operate as a $\overline{\text{borrow}}$ and digit $\overline{\text{borrow}}$ out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

3: When the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. The specified bit(s) will be updated according to device logic

REGISTER 2-1: STATUS REGISTER (ADDRESS 03h, 83h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
bit 7							bit 0

bit 7-6 **Unimplemented:** Maintain as '0'

bit 5 **RP0**: Register Bank Select bits (used for direct addressing)
`01` = Bank 1 (80h - FFh)
`00` = Bank 0 (00h - 7Fh)

bit 4 **$\overline{\text{TO}}$**: Time-out bit
`1` = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
`0` = A WDT time-out occurred

bit 3 **$\overline{\text{PD}}$**: Power-down bit
`1` = After power-up or by the `CLRWDT` instruction
`0` = By execution of the `SLEEP` instruction

bit 2 **Z**: Zero bit
`1` = The result of an arithmetic or logic operation is zero
`0` = The result of an arithmetic or logic operation is not zero

bit 1 **DC**: Digit carry/$\overline{\text{borrow}}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions) (for $\overline{\text{borrow}}$, the polarity is reversed)
`1` = A carry-out from the 4th low order bit of the result occurred
`0` = No carry-out from the 4th low order bit of the result

bit 0 **C**: Carry/$\overline{\text{borrow}}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions) (for $\overline{\text{borrow}}$, the polarity is reversed)
`1` = A carry-out from the Most Significant bit of the result occurred
`0` = No carry-out from the Most Significant bit of the result occurred

Note: A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high or low order bit of the source register.

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

2.3.2 OPTION REGISTER

The OPTION register is a readable and writable register which contains various control bits to configure the TMR0/WDT prescaler, the external INT interrupt, TMR0, and the weak pull-ups on PORTB.

> **Note:** When the prescaler is assigned to the WDT (PSA = '1'), TMR0 has a 1:1 prescaler assignment.

REGISTER 2-2: OPTION REGISTER (ADDRESS 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

bit 7 **$\overline{RBPU}$:** PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6 **INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5 **T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3 **PSA**: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS2:PS0**: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared x = Bit is unknown

2.3.3 INTCON REGISTER

The INTCON register is a readable and writable register that contains the various enable bits for all interrupt sources.

> **Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>).

REGISTER 2-3: INTCON REGISTER (ADDRESS 0Bh, 8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit 7							bit 0

bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **EEIE:** EE Write Complete Interrupt Enable bit
1 = Enables the EE Write Complete interrupts
0 = Disables the EE Write Complete interrupt

bit 5 **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3 **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0 **RBIF:** RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

2.4 PCL and PCLATH

The program counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register. This register is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly readable or writable. If the program counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`. All updates to the PCH register go through the PCLATH register.

2.4.1 STACK

The stack allows a combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Mid-range devices have an 8 level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a `CALL` instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a `RETURN`, `RETLW` or a `RETFIE` instruction execution. PCLATH is not modified when the stack is PUSHed or POPed.

After the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

2.5 Indirect Addressing; INDF and FSR Registers

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a *pointer*). This is indirect addressing.

EXAMPLE 2-1: INDIRECT ADDRESSING

- Register file 05 contains the value 10h
- Register file 06 contains the value 0Ah
- Load the value 05 into the FSR register
- A read of the INDF register will return the value of 10h
- Increment the value of the FSR register by one (FSR = 06)
- A read of the INDF register now will return the value of 0Ah.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

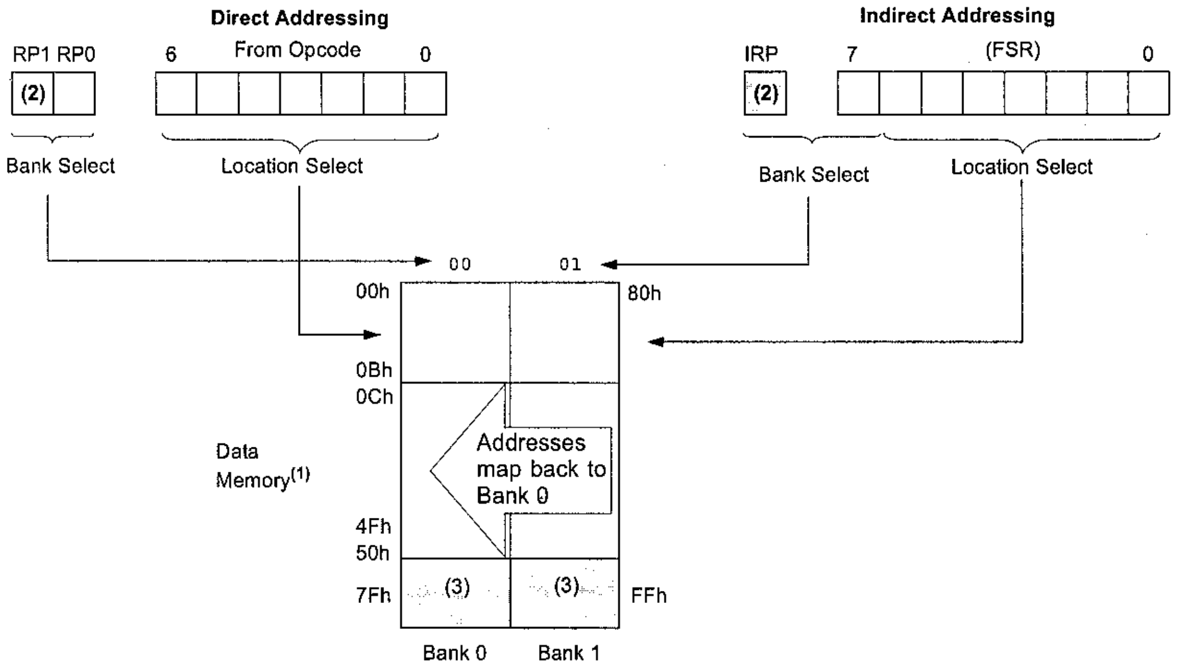
A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-2.

EXAMPLE 2-2: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

```
        movlw   0x20    ;initialize pointer
        movwf   FSR     ;to RAM
NEXT    clrf    INDF    ;clear INDF register
        incf    FSR     ;inc pointer
        btfss   FSR,4   ;all done?
        goto    NEXT    ;NO, clear next
CONTINUE
        :               ;YES, continue
```

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-3. However, IRP is not used in the PIC16F84A.

FIGURE 2-3: DIRECT/INDIRECT ADDRESSING



Note 1: For memory map detail, see Figure 2-2.

2: Maintain as clear for upward compatibility with future products.

3: Not implemented.

3.0 DATA EEPROM MEMORY

The EEPROM data memory is readable and writable during normal operation (full VDD range). This memory is not directly mapped in the register file space. Instead it is indirectly addressed through the Special Function Registers. There are four SFRs used to read and write this memory. These registers are:

- EECON1
- EECON2 (not a physically implemented register)
- EEDATA
- EEADR

EEDATA holds the 8-bit data for read/write, and EEADR holds the address of the EEPROM location being accessed. PIC16F84A devices have 64 bytes of data EEPROM with an address range from 0h to 3Fh.

The EEPROM data memory allows byte read and write. A byte write automatically erases the location and writes the new data (erase before write). The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write-time will vary with voltage and temperature as well as from chip to chip. Please refer to AC specifications for exact limits.

When the device is code protected, the CPU may continue to read and write the data EEPROM memory. The device programmer can no longer access this memory.

Additional information on the Data EEPROM is available in the PICmicro™ Mid-Range Reference Manual (DS33023).

REGISTER 3-1: EECON1 REGISTER (ADDRESS 88h)

U-0	U-0	U-0	R/W-0	R/W-x	R/W-0	R/S-0	R/S-0
—	—	—	EEIF	WRERR	WREN	WR	RD
bit 7							bit 0

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **EEIF:** EEPROM Write Operation Interrupt Flag bit
1 = The write operation completed (must be cleared in software)
0 = The write operation is not complete or has not been started

bit 3 **WRERR:** EEPROM Error Flag bit
1 = A write operation is prematurely terminated (any $\overline{MCLR}$ Reset or any WDT Reset during normal operation)
0 = The write operation completed

bit 2 **WREN:** EEPROM Write Enable bit
1 = Allows write cycles
0 = Inhibits write to the EEPROM

bit 1 **WR:** Write Control bit
1 = Initiates a write cycle. The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.
0 = Write cycle to the EEPROM is complete

bit 0 **RD:** Read Control bit
1 = Initiates an EEPROM read RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
0 = Does not initiate an EEPROM read

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

APPENDIX D

N CHANNEL MOSFET DATASHEET

MTP3055V

Preferred Device

Power MOSFET
12 Amps, 60 Volts
N–Channel TO–220

This Power MOSFET is designed to withstand high energy in the avalanche and commutation modes. Designed for low voltage, high speed switching applications in power supplies, converters and power motor controls, these devices are particularly well suited for bridge circuits where diode speed and commutating safe operating areas are critical and offer additional safety margin against unexpected voltage transients.

- On–resistance Area Product about One–half that of Standard MOSFETs with New Low Voltage, Low $R_{DS(on)}$ Technology
- Faster Switching than E–FET Predecessors
- Avalanche Energy Specified
- I_{DSS} and $V_{DS(on)}$ Specified at Elevated Temperature
- Static Parameters are the Same for both TMOS V and TMOS E–FET

MAXIMUM RATINGS (T_C = 25°C unless otherwise noted)

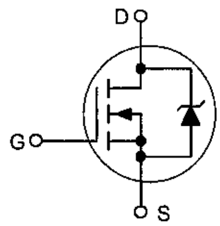
Rating	Symbol	Value	Unit
Drain–Source Voltage	V_{DSS}	60	Vdc
Drain–Gate Voltage (R_{GS} = 1.0 MΩ)	V_{DGR}	60	Vdc
Gate–Source Voltage – Continuous – Non–Repetitive ($t_p \le$ 10 ms)	 V_{GS} V_{GSM}	 ± 20 ± 25	 Vdc Vpk
Drain Current – Continuous @ 25°C – Continuous @ 100°C – Single Pulse ($t_p \le$ 10 μs)	I_D I_D I_{DM}	12 7.3 37	Adc Apk
Total Power Dissipation @ 25°C Derate above 25°C	P_D	48 0.32	Watts W/°C
Operating and Storage Temperature Range	T_J, T_{stg}	–55 to 175	°C
Single Pulse Drain–to–Source Avalanche Energy – Starting T_J = 25°C (V_{DD} = 25 Vdc, V_{GS} = 10 Vdc, I_L = 12 Apk, L = 1.0 mH, R_G = 25 Ω)	E_{AS}	72	mJ
Thermal Resistance – Junction to Case – Junction to Ambient	$R_{\theta JC}$ $R_{\theta JA}$	3.13 62.5	°C/W
Maximum Lead Temperature for Soldering Purposes, 1/8″ from case for 10 seconds	T_L	260	°C

ON Semiconductor™

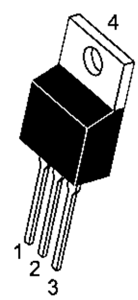
http://onsemi.com

12 AMPERES
50 VOLTS
$R_{DS(on)}$ = 150 mΩ

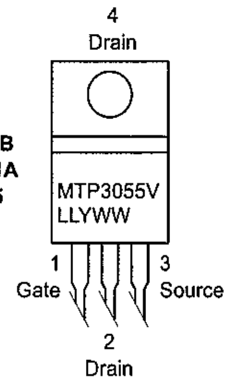
N–Channel



MARKING DIAGRAM & PIN ASSIGNMENT



TO–220AB
CASE 221A
STYLE 5



MTP3055V	=	Device Code
LL	=	Location Code
Y	=	Year
WW	=	Work Week

ORDERING INFORMATION

Device	Package	Shipping
MTP3055V	TO–220AB	50 Units/Rail

Preferred devices are recommended choices for future use and best overall value.

ELECTRICAL CHARACTERISTICS (T_J = 25°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
OFF CHARACTERISTICS						
Drain–Source Breakdown Voltage (V_{GS} = 0 Vdc, I_D = 250 μAdc)		$V_{(BR)DSS}$	60	–	–	Vdc
Temperature Coefficient (Positive)			–	65	–	mV/°C
Zero Gate Voltage Drain Current (V_{DS} = 60 Vdc, V_{GS} = 0 Vdc)		I_{DSS}	–	–	10	μAdc
(V_{DS} = 60 Vdc, V_{GS} = 0 Vdc, T_J = 150°C)			–	–	100	
Gate–Body Leakage Current (V_{GS} = ± 20 Vdc, V_{DS} = 0)		I_{GSS}	–	–	100	nAdc
ON CHARACTERISTICS (Note 1.)						
Gate Threshold Voltage (V_{DS} = V_{GS}, I_D = 250 μAdc)		$V_{GS(th)}$	2.0	2.7	4.0	Vdc
Temperature Coefficient (Negative)			–	5.4	–	mV/°C
Static Drain–Source On–Resistance (V_{GS} = 10 Vdc, I_D = 6.0 Adc)		$R_{DS(on)}$	–	0.10	0.15	Ohm
Drain–Source On–Voltage (V_{GS} = 10 Vdc) (I_D = 12 Adc)		$V_{DS(on)}$	–	1.3	2.2	Vdc
(I_D = 6.0 Adc, T_J = 150°C)			–	–	1.9	
Forward Transconductance (V_{DS} = 7.0 Vdc, I_D = 6.0 Adc)		g_{FS}	4.0	5.0	–	mhos
DYNAMIC CHARACTERISTICS						
Input Capacitance	(V_{DS} = 25 Vdc, V_{GS} = 0 Vdc, f = 1.0 MHz)	C_{iss}	–	410	500	pF
Output Capacitance		C_{oss}	–	130	180	
Reverse Transfer Capacitance		C_{rss}	–	25	50	
SWITCHING CHARACTERISTICS (Note 2.)						
Turn–On Delay Time	(V_{DD} = 30 Vdc, I_D = 12 Adc, V_{GS} = 10 Vdc, R_G = 9.1 Ω)	$t_{d(on)}$	–	7.0	10	ns
Rise Time		t_r	–	34	60	
Turn–Off Delay Time		$t_{d(off)}$	–	17	30	
Fall Time		t_f	–	18	50	
Gate Charge (See Figure 8)	(V_{DS} = 48 Vdc, I_D = 12 Adc, V_{GS} = 10 Vdc)	Q_T	–	12.2	17	nC
		Q_1	–	3.2	–	
		Q_2	–	5.2	–	
		Q_3	–	5.5	–	
SOURCE–DRAIN DIODE CHARACTERISTICS						
Forward On–Voltage (Note 1.)	(I_S = 12 Adc, V_{GS} = 0 Vdc)	V_{SD}	–	1.0	1.6	Vdc
	(I_S = 12 Adc, V_{GS} = 0 Vdc, T_J = 150°C)		–	0.91	–	
Reverse Recovery Time (See Figure 15)	(I_S = 12 Adc, V_{GS} = 0 Vdc, dI_S/dt = 100 A/μs)	t_{rr}	–	56	–	ns
		t_a	–	40	–	
		t_b	–	16	–	
Reverse Recovery Stored Charge		Q_{RR}	–	0.128	–	μC
INTERNAL PACKAGE INDUCTANCE						
Internal Drain Inductance (Measured from contact screw on tab to center of die)		L_D	–	3.5	–	nH
(Measured from the drain lead 0.25″ from package to center of die)				4.5		
Internal Source Inductance (Measured from the source lead 0.25″ from package to source bond pad)		L_S	–	7.5	–	nH

1. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2%.
2. Switching characteristics are independent of operating junction temperature.

TYPICAL ELECTRICAL CHARACTERISTICS

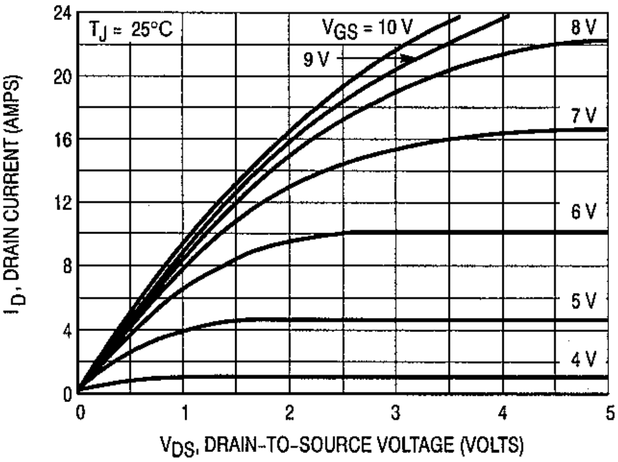


Figure 1. On–Region Characteristics

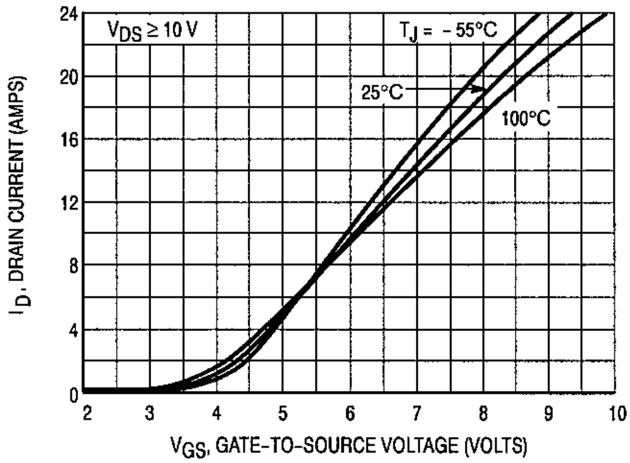


Figure 2. Transfer Characteristics

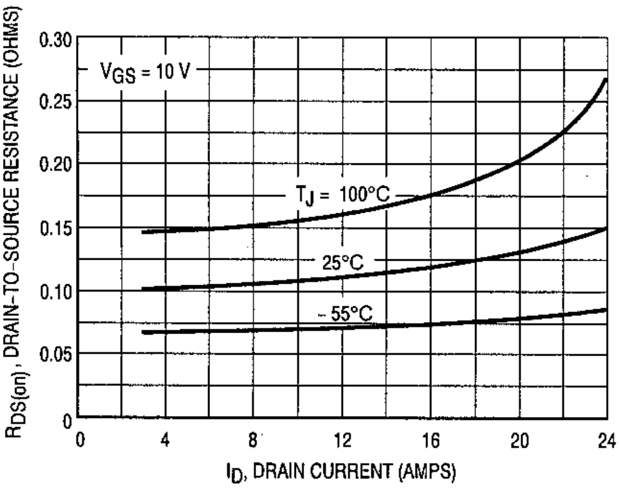


Figure 3. On–Resistance versus Drain Current and Temperature

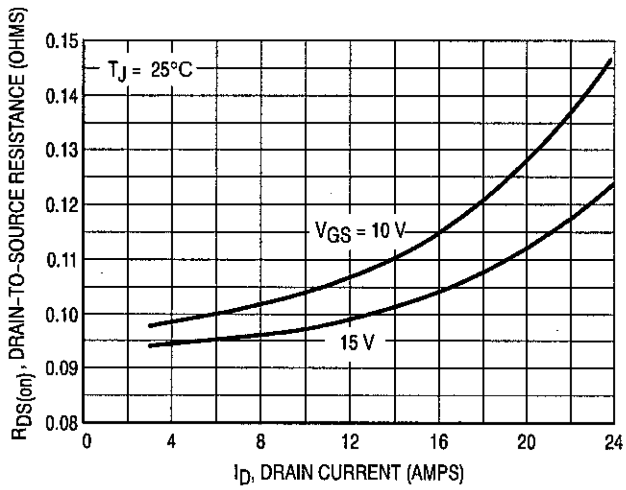


Figure 4. On–Resistance versus Drain Current and Gate Voltage

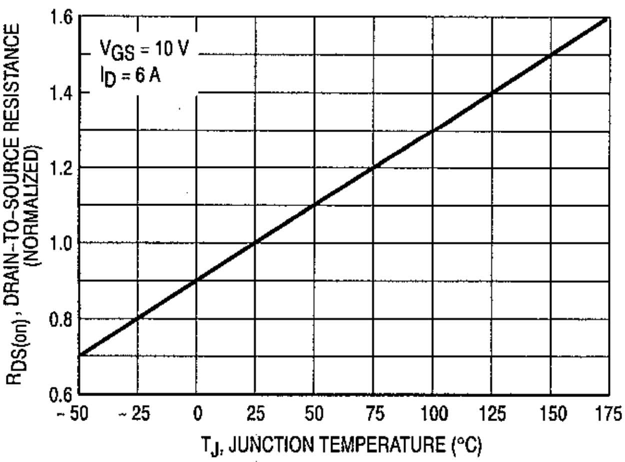


Figure 5. On–Resistance Variation with Temperature

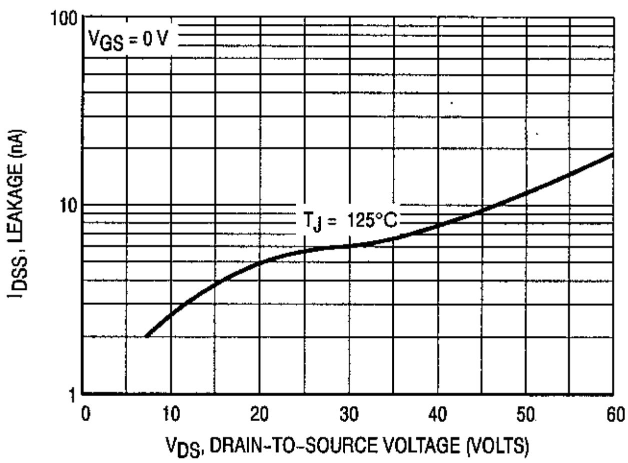


Figure 6. Drain–To–Source Leakage Current versus Voltage

POWER MOSFET SWITCHING

Switching behavior is most easily modeled and predicted by recognizing that the power MOSFET is charge controlled. The lengths of various switching intervals (Δt) are determined by how fast the FET input capacitance can be charged by current from the generator.

The published capacitance data is difficult to use for calculating rise and fall because drain–gate capacitance varies greatly with applied voltage. Accordingly, gate charge data is used. In most cases, a satisfactory estimate of average input current ($I_{G(AV)}$) can be made from a rudimentary analysis of the drive circuit so that

$t = Q/I_{G(AV)}$

During the rise and fall time interval when switching a resistive load, V_{GS} remains virtually constant at a level known as the plateau voltage, V_{SGP}. Therefore, rise and fall times may be approximated by the following:

$t_r = Q_2 \times R_G/(V_{GG} - V_{GSP})$

$t_f = Q_2 \times R_G/V_{GSP}$

where

V_{GG} = the gate drive voltage, which varies from zero to V_{GG}

R_G = the gate drive resistance

and Q_2 and V_{GSP} are read from the gate charge curve.

During the turn–on and turn–off delay times, gate current is not constant. The simplest calculation uses appropriate values from the capacitance curves in a standard equation for voltage change in an RC network. The equations are:

$t_{d(on)} = R_G\, C_{iss}\, \ln\, [V_{GG}/(V_{GG} - V_{GSP})]$

$t_{d(off)} = R_G\, C_{iss}\, \ln\, (V_{GG}/V_{GSP})$

The capacitance (C_{iss}) is read from the capacitance curve at a voltage corresponding to the off–state condition when calculating $t_{d(on)}$ and is read at a voltage corresponding to the on–state when calculating $t_{d(off)}$.

At high switching speeds, parasitic circuit elements complicate the analysis. The inductance of the MOSFET source lead, inside the package and in the circuit wiring which is common to both the drain and gate current paths, produces a voltage at the source which reduces the gate drive current. The voltage is determined by Ldi/dt, but since di/dt is a function of drain current, the mathematical solution is complex. The MOSFET output capacitance also complicates the mathematics. And finally, MOSFETs have finite internal gate resistance which effectively adds to the resistance of the driving source, but the internal resistance is difficult to measure and, consequently, is not specified.

The resistive switching time variation versus gate resistance (Figure 9) shows how typical switching performance is affected by the parasitic circuit elements. If the parasitics were not present, the slope of the curves would maintain a value of unity regardless of the switching speed. The circuit used to obtain the data is constructed to minimize common inductance in the drain and gate circuit loops and is believed readily achievable with board mounted components. Most power electronic loads are inductive; the data in the figure is taken with a resistive load, which approximates an optimally snubbed inductive load. Power MOSFETs may be safely operated into an inductive load; however, snubbing reduces switching losses.

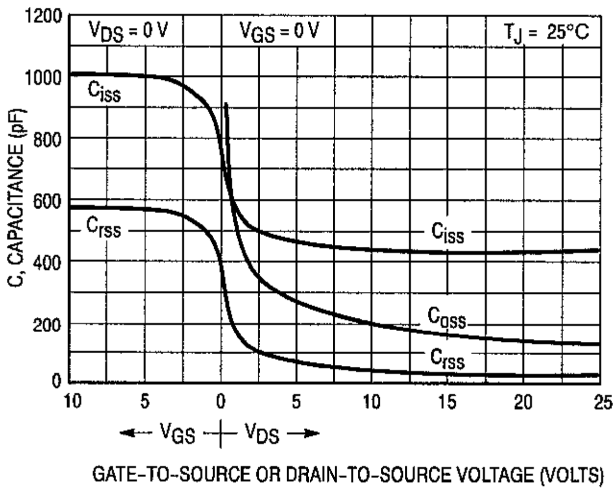


Figure 7. Capacitance Variation

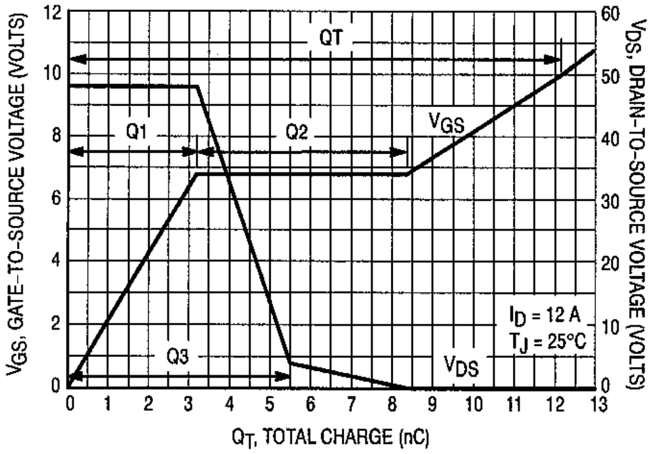


Figure 8. Gate–To–Source and Drain–To–Source Voltage versus Total Charge

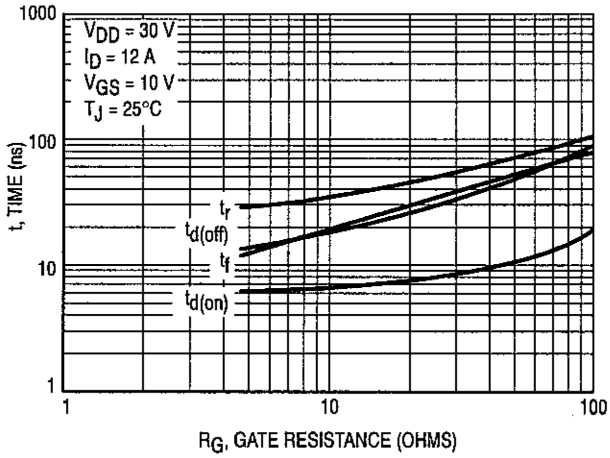


Figure 9. Resistive Switching Time Variation versus Gate Resistance

DRAIN–TO–SOURCE DIODE CHARACTERISTICS

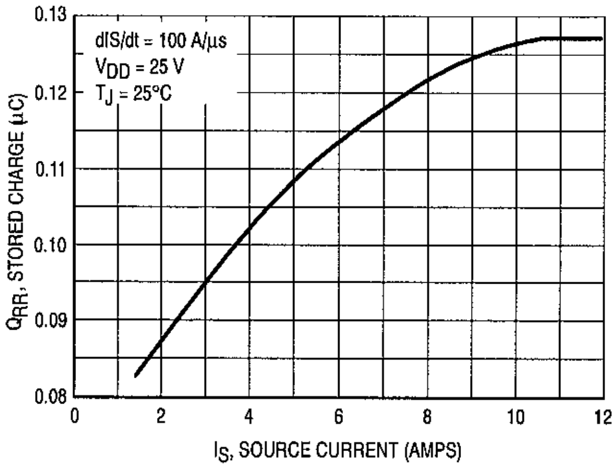


Figure 10. Stored Charge

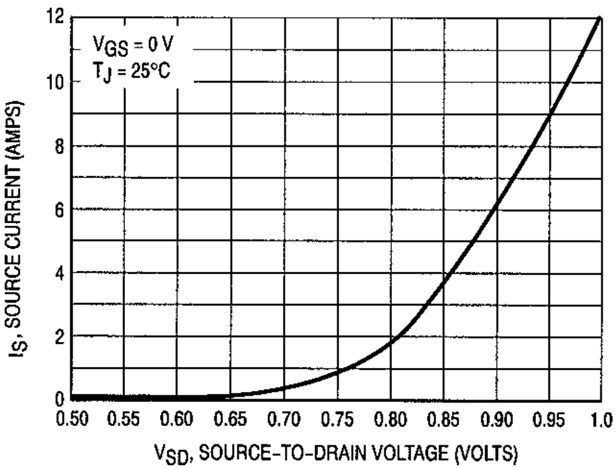


Figure 11. Diode Forward Voltage versus Current

SAFE OPERATING AREA

The Forward Biased Safe Operating Area curves define the maximum simultaneous drain–to–source voltage and drain current that a transistor can handle safely when it is forward biased. Curves are based upon maximum peak junction temperature and a case temperature (T_C) of 25°C. Peak repetitive pulsed power limits are determined by using the thermal response data in conjunction with the procedures discussed in AN569, "Transient Thermal Resistance–General Data and Its Use."

Switching between the off–state and the on–state may traverse any load line provided neither rated peak current (I_{DM}) nor rated voltage (V_{DSS}) is exceeded and the transition time (t_r,t_f) do not exceed 10 µs. In addition the total power averaged over a complete switching cycle must not exceed $(T_{J(MAX)} - T_C)/(R_{\theta JC})$.

A Power MOSFET designated E–FET can be safely used in switching circuits with unclamped inductive loads. For reliable operation, the stored energy from circuit inductance dissipated in the transistor while in avalanche must be less than the rated limit and adjusted for operating conditions differing from those specified. Although industry practice is to rate in terms of energy, avalanche energy capability is not a constant. The energy rating decreases non–linearly with an increase of peak current in avalanche and peak junction temperature.

Although many E–FETs can withstand the stress of drain–to–source avalanche at currents up to rated pulsed current (I_{DM}), the energy rating is specified at rated continuous current (I_D), in accordance with industry custom. The energy rating must be derated for temperature as shown in the accompanying graph (Figure 13). Maximum energy at currents below rated continuous I_D can safely be assumed to equal the values indicated.

SAFE OPERATING AREA

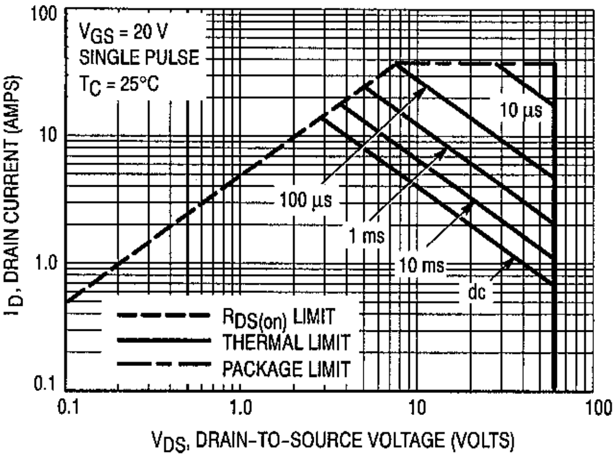


Figure 12. Maximum Rated Forward Biased Safe Operating Area

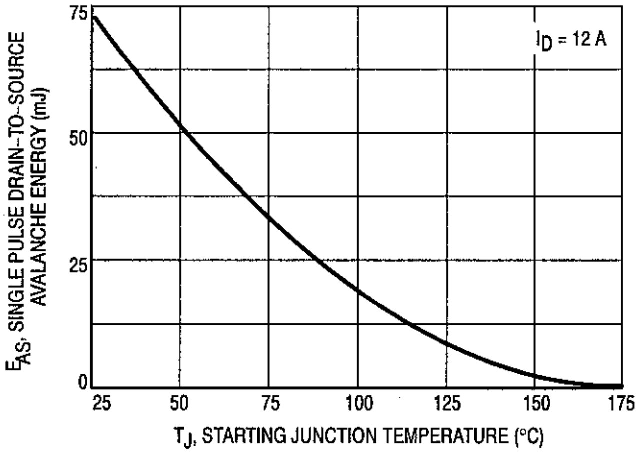


Figure 13. Maximum Avalanche Energy versus Starting Junction Temperature

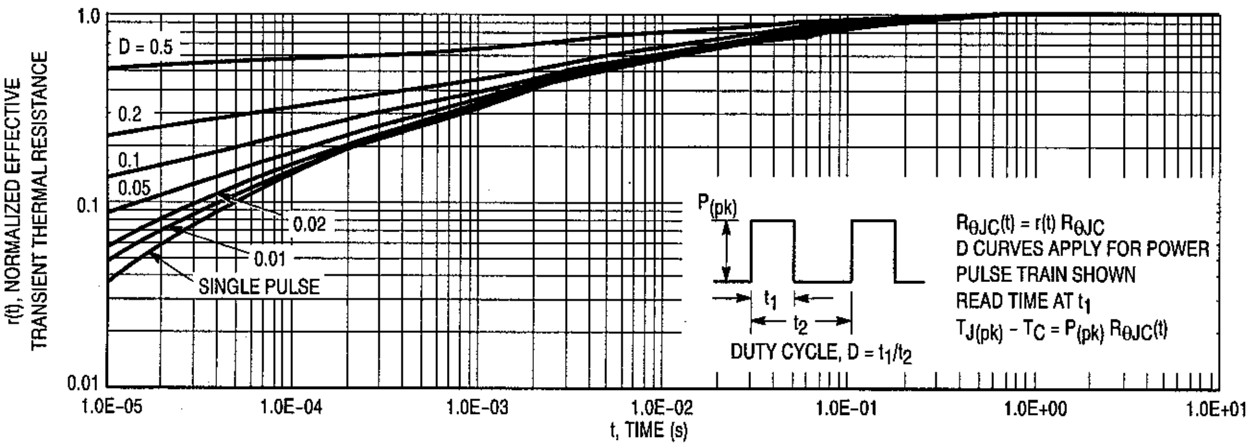


Figure 14. Thermal Response

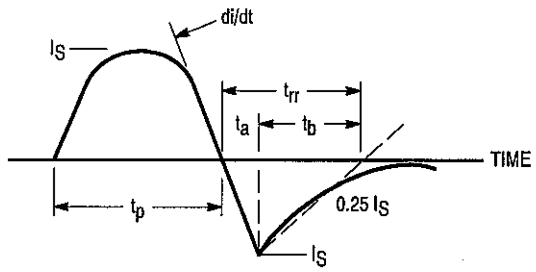
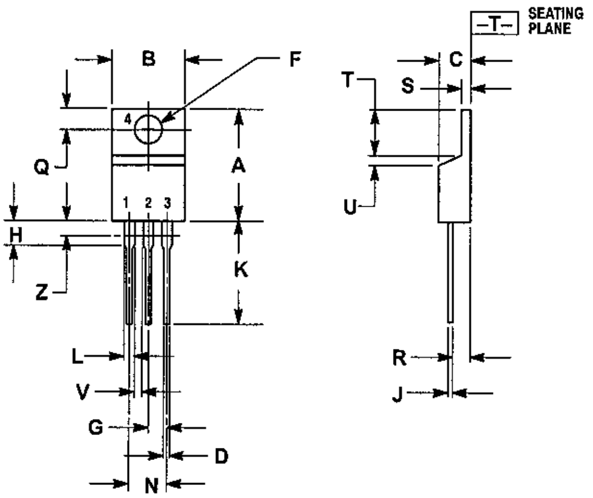


Figure 15. Diode Reverse Recovery Waveform

PACKAGE DIMENSIONS

TO–220 THREE–LEAD
TO–220AB
CASE 221A–09
ISSUE AA



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION Z DEFINES A ZONE WHERE ALL BODY AND LEAD IRREGULARITIES ARE ALLOWED.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.570	0.620	14.48	15.75
B	0.380	0.405	9.66	10.28
C	0.160	0.190	4.07	4.82
D	0.025	0.035	0.64	0.88
F	0.142	0.147	3.61	3.73
G	0.095	0.105	2.42	2.66
H	0.110	0.155	2.80	3.93
J	0.018	0.025	0.46	0.64
K	0.500	0.562	12.70	14.27
L	0.045	0.060	1.15	1.52
N	0.190	0.210	4.83	5.33
Q	0.100	0.120	2.54	3.04
R	0.080	0.110	2.04	2.79
S	0.045	0.055	1.15	1.39
T	0.235	0.255	5.97	6.47
U	0.000	0.050	0.00	1.27
V	0.045	---	1.15	---
Z	---	0.080	---	2.04

STYLE 5:
PIN 1. GATE
2. DRAIN
3. SOURCE
4. DRAIN

PUBLICATION ORDERING INFORMATION

NORTH AMERICA Literature Fulfillment:
Literature Distribution Center for ON Semiconductor
P.O. Box 5163, Denver, Colorado 80217 USA
Phone: 303–675–2175 or 800–344–3860 Toll Free USA/Canada
Fax: 303–675–2176 or 800–344–3867 Toll Free USA/Canada
Email: ONlit@hibbertco.com
Fax Response Line: 303–675–2167 or 800–344–3810 Toll Free USA/Canada

N. American Technical Support: 800–282–9855 Toll Free USA/Canada

EUROPE: LDC for ON Semiconductor – European Support
German Phone: (+1) 303–308–7140 (Mon–Fri 2:30pm to 7:00pm CET)
Email: ONlit–german@hibbertco.com
French Phone: (+1) 303–308–7141 (Mon–Fri 2:00pm to 7:00pm CET)
Email: ONlit–french@hibbertco.com
English Phone: (+1) 303–308–7142 (Mon–Fri 12:00pm to 5:00pm GMT)
Email: ONlit@hibbertco.com

EUROPEAN TOLL–FREE ACCESS*: 00–800–4422–3781
*Available from Germany, France, Italy, UK, Ireland

CENTRAL/SOUTH AMERICA:
Spanish Phone: 303–308–7143 (Mon–Fri 8:00am to 5:00pm MST)
Email: ONlit–spanish@hibbertco.com
Toll–Free from Mexico: Dial 01–800–288–2872 for Access – then Dial 866–297–9322

ASIA/PACIFIC: LDC for ON Semiconductor – Asia Support
Phone: 303–675–2121 (Tue–Fri 9:00am to 1:00pm, Hong Kong Time)
Toll Free from Hong Kong & Singapore:
001–800–4422–3781
Email: ONlit–asia@hibbertco.com

JAPAN: ON Semiconductor, Japan Customer Focus Center
4–32–1 Nishi–Gotanda, Shinagawa–ku, Tokyo, Japan 141–0031
Phone: 81–3–5740–2700
Email: r14525@onsemi.com

ON Semiconductor Website: http://onsemi.com

For additional information, please contact your local Sales Representative.

APPENDIX E

P CHANNEL MOSFET DATASHEET

MTP2955V

Preferred Device

Power MOSFET 12 Amps, 60 Volts

P–Channel TO–220

This Power MOSFET is designed to withstand high energy in the avalanche and commutation modes. Designed for low voltage, high speed switching applications in power supplies, converters and power motor controls, these devices are particularly well suited for bridge circuits where diode speed and commutating safe operating areas are critical and offer additional safety margin against unexpected voltage transients.

- Avalanche Energy Specified
- I_{DSS} and $V_{DS(on)}$ Specified at Elevated Temperature

MAXIMUM RATINGS (T_C = 25°C unless otherwise noted)

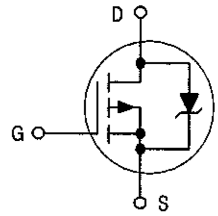
Rating	Symbol	Value	Unit
Drain–to–Source Voltage	V_{DSS}	60	Vdc
Drain–to–Gate Voltage (R_{GS} = 1.0 MΩ)	V_{DGR}	60	Vdc
Gate–to–Source Voltage – Continuous – Non–Repetitive ($t_p \le$ 10 ms)	 V_{GS} V_{GSM}	 ± 15 ± 25	 Vdc Vpk
Drain Current – Continuous – Continuous @ 100°C – Single Pulse ($t_p \le$ 10 μs)	I_D I_D I_{DM}	12 8.0 42	Adc Apk
Total Power Dissipation Derate above 25°C	P_D	60 0.40	Watts W/°C
Operating and Storage Temperature Range	T_J, T_{stg}	–55 to 175	°C
Single Pulse Drain–to–Source Avalanche Energy – Starting T_J = 25°C (V_{DD} = 25 Vdc, V_{GS} = 10 Vdc, Peak I_L = 12 Apk, L = 3.0 mH, R_G = 25 Ω)	E_{AS}	216	mJ
Thermal Resistance – Junction to Case – Junction to Ambient	 $R_{\theta JC}$ $R_{\theta JA}$	 2.5 62.5	°C/W
Maximum Lead Temperature for Soldering Purposes, 1/8″ from case for 10 seconds	T_L	260	°C

ON Semiconductor™

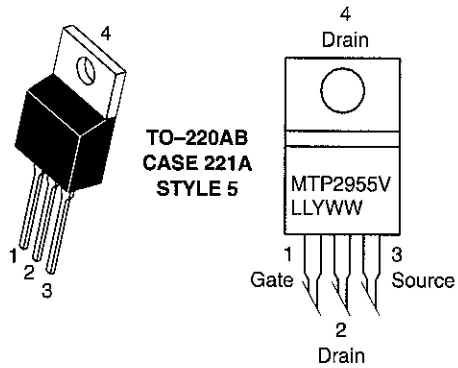
http://onsemi.com

12 AMPERES
60 VOLTS
$R_{DS(on)}$ = 230 mΩ

P–Channel



MARKING DIAGRAM & PIN ASSIGNMENT



MTP2955V	= Device Code
LL	= Location Code
Y	= Year
WW	= Work Week

ORDERING INFORMATION

Device	Package	Shipping
MTP2955V	TO–220AB	50 Units/Rail

Preferred devices are recommended choices for future use and best overall value.

ELECTRICAL CHARACTERISTICS (T_J = 25°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
OFF CHARACTERISTICS						
Drain–to–Source Breakdown Voltage (Cpk ≥ 2.0) (Note 3.) (V_{GS} = 0 Vdc, I_D = 0.25 mAdc) Temperature Coefficient (Positive)		$V_{(BR)DSS}$	60 –	– 58	– –	Vdc mV/°C
Zero Gate Voltage Drain Current (V_{DS} = 60 Vdc, V_{GS} = 0 Vdc) (V_{DS} = 60 Vdc, V_{GS} = 0 Vdc, T_J = 150°C)		I_{DSS}	– –	– –	10 100	μAdc
Gate–Body Leakage Current (V_{GS} = ± 15 Vdc, V_{DS} = 0 Vdc)		I_{GSS}	–	–	100	nAdc
ON CHARACTERISTICS (Note 1.)						
Gate Threshold Voltage (Cpk ≥ 2.0) (Note 3.) (V_{DS} = V_{GS}, I_D = 250 μAdc) Threshold Temperature Coefficient (Negative)		$V_{GS(th)}$	2.0 –	2.8 5.0	4.0 –	Vdc mV/°C
Static Drain–to–Source On–Resistance (Cpk ≥ 1.5) (Note 3.) (V_{GS} = 10 Vdc, I_D = 6.0 Adc)		$R_{DS(on)}$	–	0.185	0.230	Ohm
Drain–to–Source On–Voltage (V_{GS} = 10 Vdc, I_D = 12 Adc) (V_{GS} = 10 Vdc, I_D = 6.0 Adc, T_J = 150°C)		$V_{DS(on)}$	– –	– –	2.9 2.5	Vdc
Forward Transconductance (V_{DS} = 10 Vdc, I_D = 6.0 Adc)		g_{FS}	3.0	5.0	–	mhos
DYNAMIC CHARACTERISTICS						
Input Capacitance	(V_{DS} = 25 Vdc, V_{GS} = 0 Vdc, f = 1.0 MHz)	C_{iss}	–	550	700	pF
Output Capacitance		C_{oss}	–	200	280	
Reverse Transfer Capacitance		C_{rss}	–	50	100	
SWITCHING CHARACTERISTICS (Note 2.)						
Turn–On Delay Time	(V_{DD} = 30 Vdc, I_D = 12 Adc, V_{GS} = 10 Vdc, R_G = 9.1 Ω)	$t_{d(on)}$	–	15	30	ns
Rise Time		t_r	–	50	100	
Turn–Off Delay Time		$t_{d(off)}$	–	24	50	
Fall Time		t_f	–	39	80	
Gate Charge	(V_{DS} = 48 Vdc, I_D = 12 Adc, V_{GS} = 10 Vdc)	Q_T	–	19	30	nC
		Q_1	–	4.0	–	
		Q_2	–	9.0	–	
		Q_3	–	7.0	–	
SOURCE–DRAIN DIODE CHARACTERISTICS						
Forward On–Voltage (Note 1.)	(I_S = 12 Adc, V_{GS} = 0 Vdc) (I_S = 12 Adc, V_{GS} = 0 Vdc, T_J = 150°C)	V_{SD}	– –	1.8 1.5	3.0 –	Vdc
Reverse Recovery Time	(I_S = 12 Adc, V_{GS} = 0 Vdc, dI_S/dt = 100 A/μs)	t_{rr}	–	115	–	ns
		t_a	–	90	–	
		t_b	–	25	–	
Reverse Recovery Stored Charge		Q_{RR}	–	0.53	–	μC
INTERNAL PACKAGE INDUCTANCE						
Internal Drain Inductance (Measured from the drain lead 0.25″ from package to center of die)		L_D	–	4.5	–	nH
Internal Source Inductance (Measured from the source lead 0.25″ from package to source bond pad)		L_S	–	7.5	–	nH

1. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2%.
2. Switching characteristics are independent of operating junction temperature.
3. Reflects typical values. $C_{pk} = \left| \frac{\text{Max limit} - \text{Typ}}{3 \times \text{SIGMA}} \right|$

TYPICAL ELECTRICAL CHARACTERISTICS

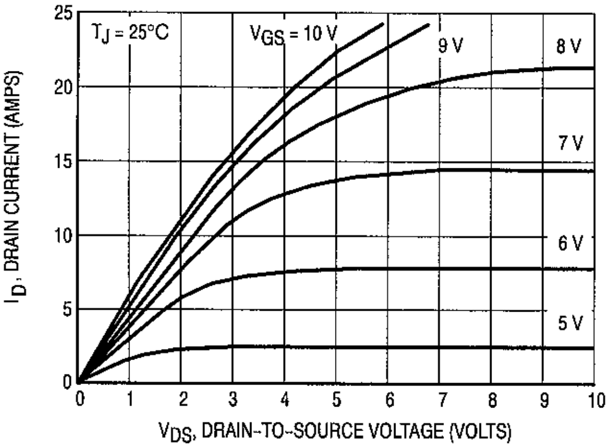


Figure 1. On–Region Characteristics

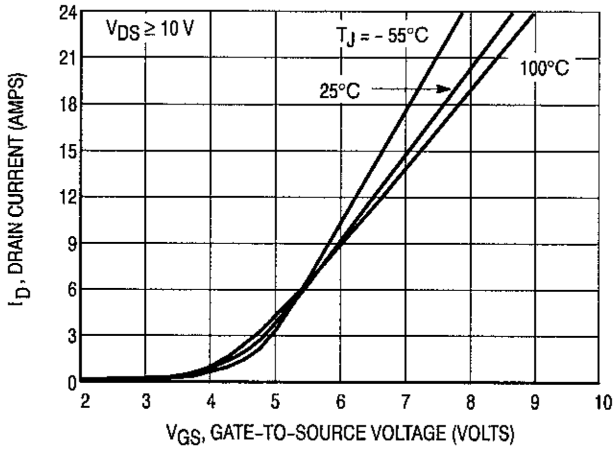


Figure 2. Transfer Characteristics

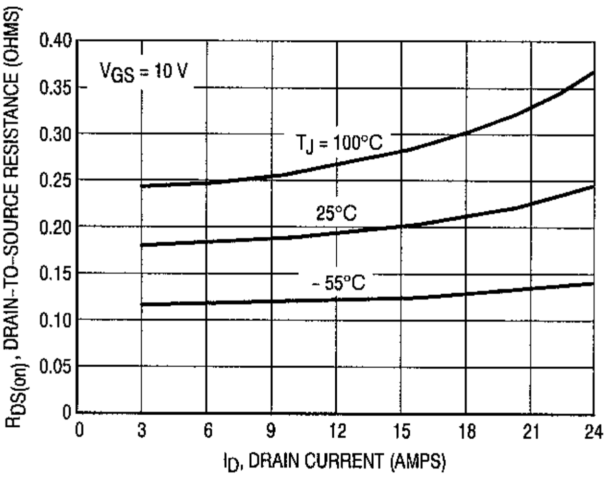


Figure 3. On–Resistance versus Drain Current and Temperature

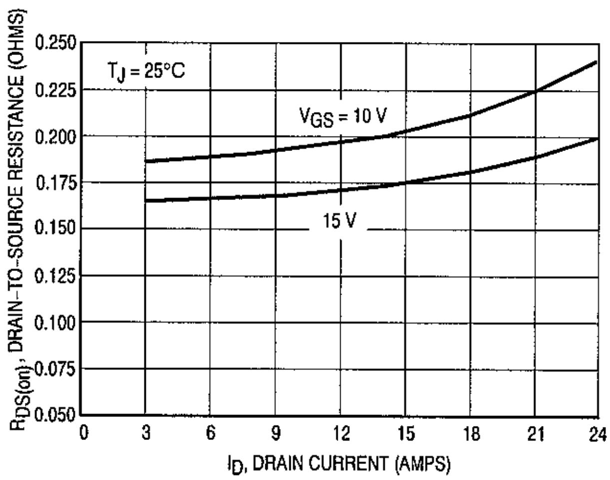


Figure 4. On–Resistance versus Drain Current and Gate Voltage

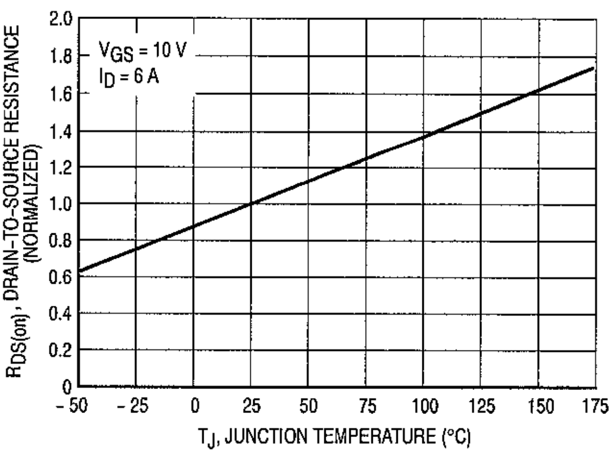


Figure 5. On–Resistance Variation with Temperature

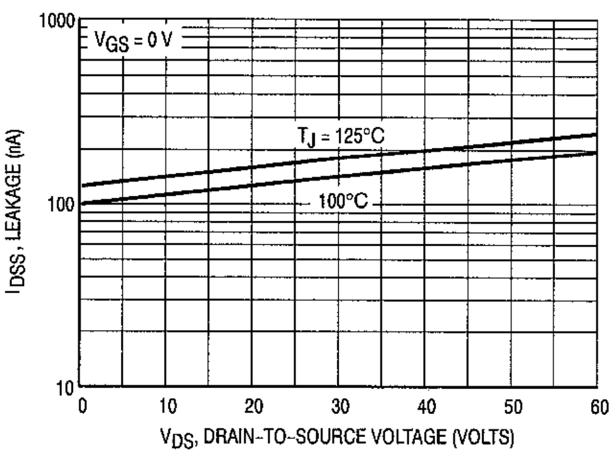


Figure 6. Drain–To–Source Leakage Current versus Voltage

POWER MOSFET SWITCHING

Switching behavior is most easily modeled and predicted by recognizing that the power MOSFET is charge controlled. The lengths of various switching intervals (Δt) are determined by how fast the FET input capacitance can be charged by current from the generator.

The published capacitance data is difficult to use for calculating rise and fall because drain–gate capacitance varies greatly with applied voltage. Accordingly, gate charge data is used. In most cases, a satisfactory estimate of average input current ($I_{G(AV)}$) can be made from a rudimentary analysis of the drive circuit so that

$t = Q/I_{G(AV)}$

During the rise and fall time interval when switching a resistive load, V_{GS} remains virtually constant at a level known as the plateau voltage, V_{SGP}. Therefore, rise and fall times may be approximated by the following:

$t_r = Q_2 \times R_G/(V_{GG} - V_{GSP})$

$t_f = Q_2 \times R_G/V_{GSP}$

where

V_{GG} = the gate drive voltage, which varies from zero to V_{GG}

R_G = the gate drive resistance

and Q_2 and V_{GSP} are read from the gate charge curve.

During the turn–on and turn–off delay times, gate current is not constant. The simplest calculation uses appropriate values from the capacitance curves in a standard equation for voltage change in an RC network. The equations are:

$t_{d(on)} = R_G\ C_{iss}\ \ln\ [V_{GG}/(V_{GG} - V_{GSP})]$

$t_{d(off)} = R_G\ C_{iss}\ \ln\ (V_{GG}/V_{GSP})$

The capacitance (C_{iss}) is read from the capacitance curve at a voltage corresponding to the off–state condition when calculating $t_{d(on)}$ and is read at a voltage corresponding to the on–state when calculating $t_{d(off)}$.

At high switching speeds, parasitic circuit elements complicate the analysis. The inductance of the MOSFET source lead, inside the package and in the circuit wiring which is common to both the drain and gate current paths, produces a voltage at the source which reduces the gate drive current. The voltage is determined by Ldi/dt, but since di/dt is a function of drain current, the mathematical solution is complex. The MOSFET output capacitance also complicates the mathematics. And finally, MOSFETs have finite internal gate resistance which effectively adds to the resistance of the driving source, but the internal resistance is difficult to measure and, consequently, is not specified.

The resistive switching time variation versus gate resistance (Figure 9) shows how typical switching performance is affected by the parasitic circuit elements. If the parasitics were not present, the slope of the curves would maintain a value of unity regardless of the switching speed. The circuit used to obtain the data is constructed to minimize common inductance in the drain and gate circuit loops and is believed readily achievable with board mounted components. Most power electronic loads are inductive; the data in the figure is taken with a resistive load, which approximates an optimally snubbed inductive load. Power MOSFETs may be safely operated into an inductive load; however, snubbing reduces switching losses.

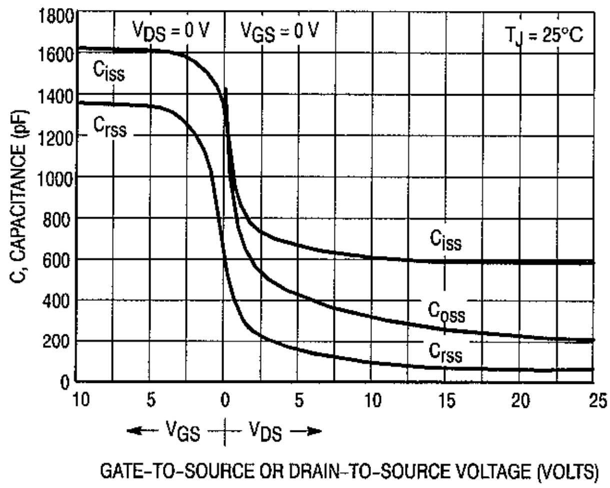


Figure 7. Capacitance Variation

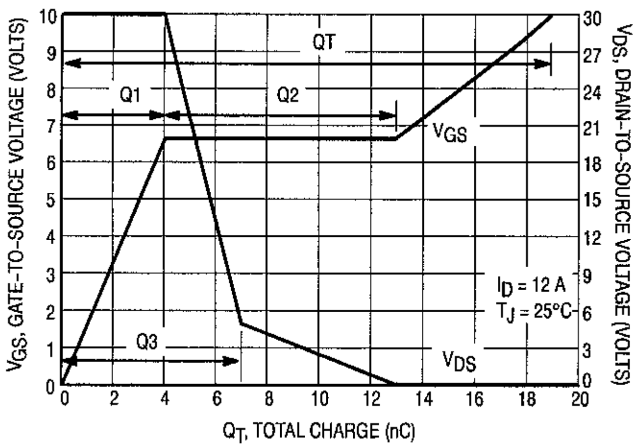


Figure 8. Gate–To–Source and Drain–To–Source Voltage versus Total Charge

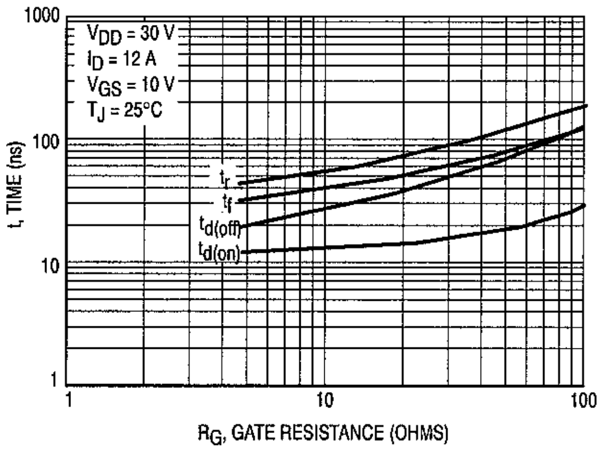


Figure 9. Resistive Switching Time Variation versus Gate Resistance

DRAIN–TO–SOURCE DIODE CHARACTERISTICS

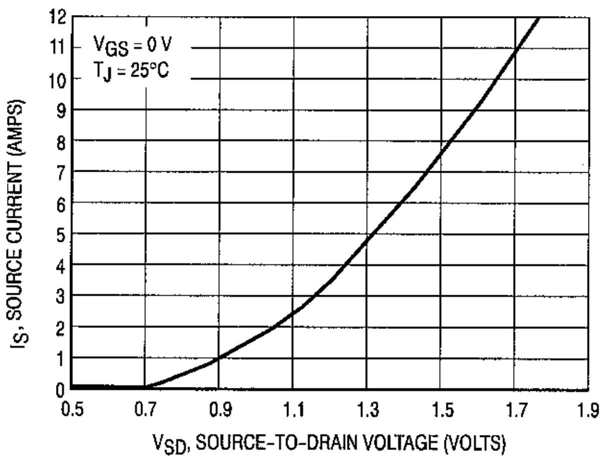


Figure 10. Diode Forward Voltage versus Current

SAFE OPERATING AREA

The Forward Biased Safe Operating Area curves define the maximum simultaneous drain–to–source voltage and drain current that a transistor can handle safely when it is forward biased. Curves are based upon maximum peak junction temperature and a case temperature (T_C) of 25°C. Peak repetitive pulsed power limits are determined by using the thermal response data in conjunction with the procedures discussed in AN569, "Transient Thermal Resistance–General Data and Its Use."

Switching between the off–state and the on–state may traverse any load line provided neither rated peak current (I_{DM}) nor rated voltage (V_{DSS}) is exceeded and the transition time (t_r,t_f) do not exceed 10 μs. In addition the total power averaged over a complete switching cycle must not exceed ($T_{J(MAX)} - T_C$)/($R_{\theta JC}$).

A Power MOSFET designated E–FET can be safely used in switching circuits with unclamped inductive loads. For reliable operation, the stored energy from circuit inductance dissipated in the transistor while in avalanche must be less than the rated limit and adjusted for operating conditions differing from those specified. Although industry practice is to rate in terms of energy, avalanche energy capability is not a constant. The energy rating decreases non–linearly with an increase of peak current in avalanche and peak junction temperature.

Although many E–FETs can withstand the stress of drain–to–source avalanche at currents up to rated pulsed current (I_{DM}), the energy rating is specified at rated continuous current (I_D), in accordance with industry custom. The energy rating must be derated for temperature as shown in the accompanying graph (Figure 13). Maximum energy at currents below rated continuous I_D can safely be assumed to equal the values indicated.

SAFE OPERATING AREA

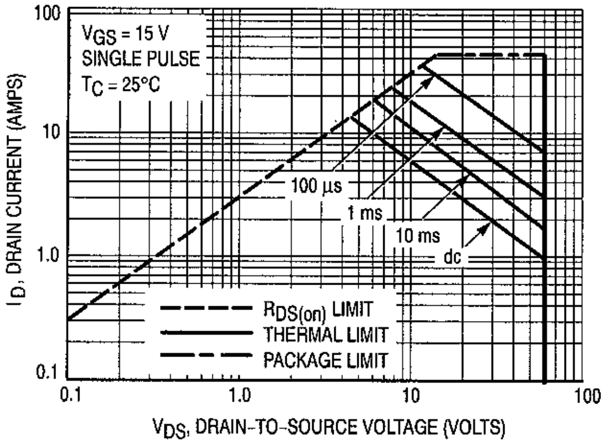


Figure 11. Maximum Rated Forward Biased Safe Operating Area

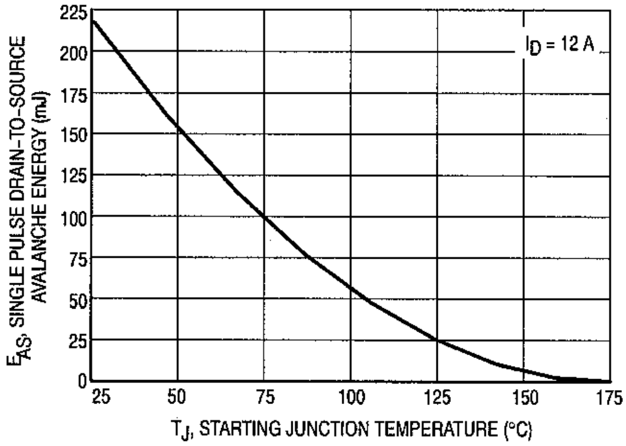


Figure 12. Maximum Avalanche Energy versus Starting Junction Temperature

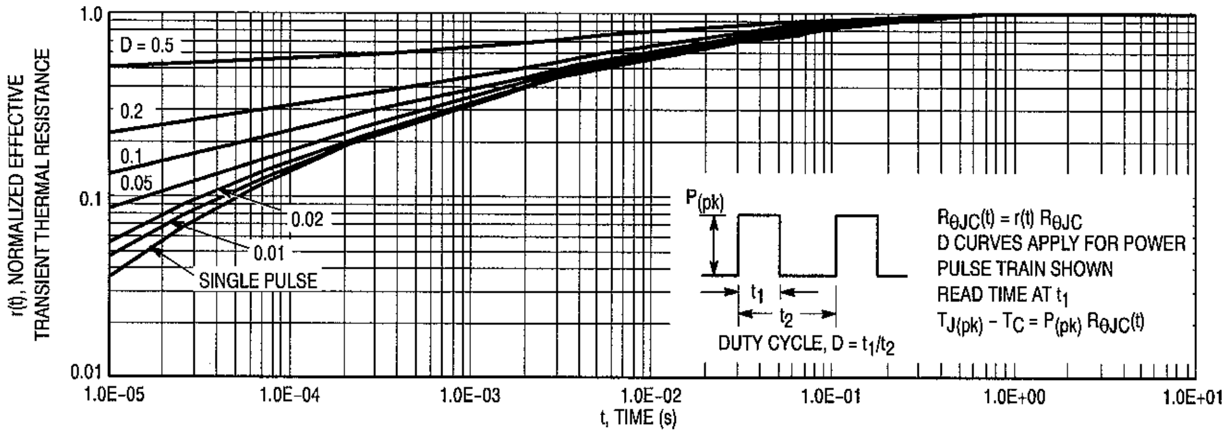


Figure 13. Thermal Response

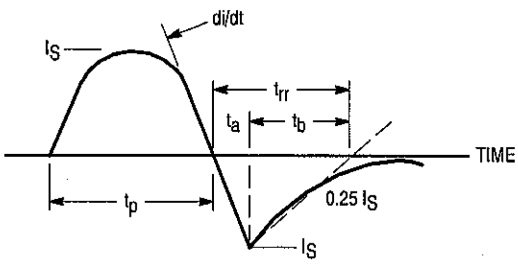
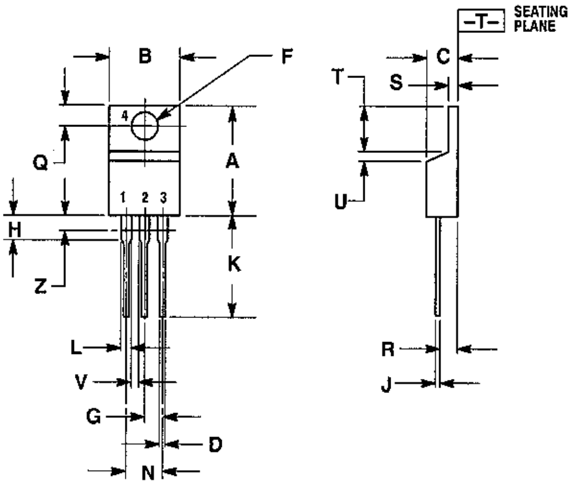


Figure 14. Diode Reverse Recovery Waveform

PACKAGE DIMENSIONS

TO–220 THREE–LEAD
TO–220AB
CASE 221A–09
ISSUE AA



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION Z DEFINES A ZONE WHERE ALL BODY AND LEAD IRREGULARITIES ARE ALLOWED.

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.570	0.620	14.48	15.75
B	0.380	0.405	9.66	10.28
C	0.160	0.190	4.07	4.82
D	0.025	0.035	0.64	0.88
F	0.142	0.147	3.61	3.73
G	0.095	0.105	2.42	2.66
H	0.110	0.155	2.80	3.93
J	0.018	0.025	0.46	0.64
K	0.500	0.562	12.70	14.27
L	0.045	0.060	1.15	1.52
N	0.190	0.210	4.83	5.33
Q	0.100	0.120	2.54	3.04
R	0.080	0.110	2.04	2.79
S	0.045	0.055	1.15	1.39
T	0.235	0.255	5.97	6.47
U	0.000	0.050	0.00	1.27
V	0.045	---	1.15	---
Z	---	0.080	---	2.04

STYLE 5:
PIN 1. GATE
2. DRAIN
3. SOURCE
4. DRAIN

PUBLICATION ORDERING INFORMATION

NORTH AMERICA Literature Fulfillment:
Literature Distribution Center for ON Semiconductor
P.O. Box 5163, Denver, Colorado 80217 USA
Phone: 303–675–2175 or 800–344–3860 Toll Free USA/Canada
Fax: 303–675–2176 or 800–344–3867 Toll Free USA/Canada
Email: ONlit@hibbertco.com
Fax Response Line: 303–675–2167 or 800–344–3810 Toll Free USA/Canada

N. American Technical Support: 800–282–9855 Toll Free USA/Canada

EUROPE: LDC for ON Semiconductor – European Support
German Phone: (+1) 303–308–7140 (Mon–Fri 2:30pm to 7:00pm CET)
Email: ONlit–german@hibbertco.com
French Phone: (+1) 303–308–7141 (Mon–Fri 2:00pm to 7:00pm CET)
Email: ONlit–french@hibbertco.com
English Phone: (+1) 303–308–7142 (Mon–Fri 12:00pm to 5:00pm GMT)
Email: ONlit@hibbertco.com

EUROPEAN TOLL–FREE ACCESS*: 00–800–4422–3781
*Available from Germany, France, Italy, UK, Ireland

CENTRAL/SOUTH AMERICA:
Spanish Phone: 303–308–7143 (Mon–Fri 8:00am to 5:00pm MST)
Email: ONlit–spanish@hibbertco.com
Toll–Free from Mexico: Dial 01–800–288–2872 for Access – then Dial 866–297–9322

ASIA/PACIFIC: LDC for ON Semiconductor – Asia Support
Phone: 303–675–2121 (Tue–Fri 9:00am to 1:00pm, Hong Kong Time)
Toll Free from Hong Kong & Singapore:
001–800–4422–3781
Email: ONlit–asia@hibbertco.com

JAPAN: ON Semiconductor, Japan Customer Focus Center
4–32–1 Nishi–Gotanda, Shinagawa–ku, Tokyo, Japan 141–0031
Phone: 81–3–5740–2700
Email: r14525@onsemi.com

ON Semiconductor Website: http://onsemi.com

For additional information, please contact your local Sales Representative.

APPENDIX F

MICROCONTROLLER PIC C PROGRAM CODE

F1. Master Microcontroller PIC C code

```
#include <16F84A.h>
#fuses XT,NOWDT,NOPROTECT
#use delay(clock=4000000)
void wheelencod(int m, int n);
void wheelencod1(int m);
void wheelencod2(int m);
void wheelencod3(int m, int n);
void wheelencodR(int m, int n);
void center();
void left();
void right();
void turnleft(int m);
void turnright(int m);
void movestraight(int m, int n);
void movestraight1(int m, int n);
void obstacleavoid();
void armdown();
void com2gripper();
void closegripper();
void main() {
delay_ms(1000);
armdown();
com2gripper();
movestraight1(10,0);
delay_ms(6000);
movestraight1(100,10);
delay_ms(1000);
turnleft(34);
movestraight(70,0);
output_low(PIN_B7);
delay_ms(1000);
output_low(PIN_B7);
armdown();
delay_ms(4000);
wheelencodR(2,0);
delay_ms(4000);
turnleft(34);
movestraight1(100,0);
turnleft(34);
movestraight(80,0);
turnleft(34);
center();


}
void center(void )
{
int i;
for (i=0;i<25;i++)
{
output_high(PIN_B4);
delay_us(1200);
output_low(PIN_B4);
delay_ms(16);
}
}

void left(void )
{
int i;
for (i=0;i<25;i++)
{
```

```
output_high(PIN_B4);
delay_us(655);
output_low(PIN_B4);
delay_ms(16);
}
}

void right(void )
{

int i;
for (i=0;i<25;i++)
{
output_high(PIN_B4);
delay_us(1800);
output_low(PIN_B4);
delay_ms(16);
}
}

void wheelencod(int m,int n)
{
int t;
int count;
int var=0;
count=n;
while(true)
{

  t=!input(PIN_A1);
  while(t==!input(PIN_A1))
   {
  output_high(PIN_B0);
  output_high(PIN_B2);
   }
count=count+1;
if (count>m-1)
 {
  break;
 }
if ((!input(PIN_A2)) && (var==0))
{
output_low(PIN_B0);
output_low(PIN_B2);
delay_ms(1000);
obstacleavoid();
var=1;
count=count+20;
}

}
output_low(PIN_B0);
output_low(PIN_B2);
return;
}

void wheelencod1(int m)
{
int t;
int count=0;
while(true)
{

  t=!input(PIN_A0);
  while(t==!input(PIN_A0))
   {
  output_high(PIN_B0);
  output_low(PIN_B2);
   }
count=count+1;
if (count>m-1)
 {
  break;
 }
```

```
}
output_low(PIN_B0);
output_low(PIN_B2);
return;
}

void wheelencod2(int m)
{
int t;
int count=0;
while(true)
{

  t=!input(PIN_A1);
  while(t==!input(PIN_A1))
   {
  output_low(PIN_B0);
  output_high(PIN_B2);
   }
count=count+1;
if (count>m-1)
 {
  break;
 }

}
output_low(PIN_B0);
output_low(PIN_B2);
return;
}

void wheelencod3(int m,int n)
{
int t;
int count;
count=n;
while(true)
{

  t=!input(PIN_A1);
  while(t==!input(PIN_A1))
   {
  output_high(PIN_B0);
  output_high(PIN_B2);
   }
count=count+1;
if (count>m-1)
 {
  break;
 }
}
output_low(PIN_B0);
output_low(PIN_B2);
return;
}
void wheelencodR(int m,int n)
{
int t;
int count;
count=n;
while(true)
{

  t=!input(PIN_A1);
  while(t==!input(PIN_A1))
   {
  output_high(PIN_B1);
  output_high(PIN_B3);
   }
count=count+1;
if (count>m-1)
 {
  break;
 }
}
```

```
output_low(PIN_B1);
output_low(PIN_B3);
return;
}

void turnleft(int m)
{
delay_ms(1000);
left();
delay_ms(1000);
wheelencod2(m);//turn left
delay_ms(1000);
}
void turnright(int m)
{
delay_ms(1000);
right();
delay_ms(1000);
wheelencod1(m);//turn right
delay_ms(1000);
}
void movestraight(int m, int n)
{
delay_ms(1000);
center();
delay_ms(1000);
wheelencod(m,n);//move straight
delay_ms(1000);
}
void movestraight1(int m, int n)
{
center();
delay_ms(1000);
wheelencod3(m,n);//move straight
delay_ms(1000);
}

void obstacleavoid(void )
{
wheelencodR(5,0);
output_low(PIN_B1);
output_low(PIN_B3);
delay_ms(1000);
turnleft(34);
movestraight1(10,0);
turnright(34);
movestraight1(25,0);
turnright(34);
movestraight1(10,0);
turnleft(34);
center();
return;
}
void armdown(void )
{
int i;
for(i=0;i<50;i++)
{
output_high(PIN_B5);
delay_us(800);
output_low(PIN_B5);
delay_ms(16);
}
for(i=0;i<50;i++)
{
output_high(PIN_B5);
delay_us(850);
output_low(PIN_B5);
delay_ms(16);
}

for(i=0;i<50;i++)
{
output_high(PIN_B5);
delay_us(1750);
```

```
output_low(PIN_B5);
delay_ms(16);
}

for(i=0;i<25;i++)
{
output_high(PIN_B5);
delay_us(1480);
output_low(PIN_B5);
delay_ms(16);
}

return;
}

void com2gripper(void )
{
delay_ms(3000);
output_low(PIN_B6);
delay_ms(1000);
output_high(PIN_B6);
return;
}
void closegripper(void )
{
output_low(PIN_B6);
delay_ms(1000);
output_float(PIN_B6);
}
```

F2. Slave Microcontroller 1 (Manipulator) PIC C code

```
#include <16F84.h>
#fuses XT,NOWDT,NOPROTECT
#use delay(clock=4000000)

void main() {
while(true)
{
//move arm down to pick up
while(input(PIN_A1))
{
;
}
if (!input(PIN_A1))
  {
  delay_ms(8000);
  output_high(PIN_B0);
  output_high(PIN_B1);
  delay_ms(220);
  output_low(PIN_B1);
  delay_ms(2000);
  output_low(PIN_B0);
  delay_ms(350);
  output_high(PIN_B0);
  delay_ms(1000);
  output_low(PIN_B0);

  }
while(input(PIN_A0))
{
;
}

if (!input(PIN_A0))
  {
  delay_ms(10000);
  output_high(PIN_B0);
  output_high(PIN_B1);
  delay_ms(220);
  output_low(PIN_B1);
  delay_ms(2000);
  output_low(PIN_B0);
  delay_ms(350);
  output_high(PIN_B0);
  delay_ms(1000);
  output_low(PIN_B0);

  }
}

}
```

F3. Slave microcontroller 2 (Gripper) PIC C code

```
#include <16F84.h>
#fuses XT,NOWDT,NOPROTECT
#use delay(clock=4000000)

void main() {
int i;
while(true)
{
while(input(PIN_A1))
{
;
}
if (!input(PIN_A1))
{
for(i=0;i<25;i++) //open gripper
{
output_high(PIN_B0);
delay_us(1800);
output_low(PIN_B0);
delay_ms(16);
}
delay_ms(5000);
for(i=0;i<250;i++) //close
{
output_high(PIN_B0);
delay_us(2000);
output_low(PIN_B0);
delay_ms(16);
}

}
while(input(PIN_A0))
{
;
}
if (!input(PIN_A0))
{
for(i=0;i<250;i++) //close gripper
{
output_high(PIN_B0);
delay_us(2000);
output_low(PIN_B0);
delay_ms(16);
}
delay_ms(2000);
for(i=0;i<25;i++) //open gripper
{
output_high(PIN_B0);
delay_us(2000);
output_low(PIN_B0);
delay_ms(16);
}

}
}
}
```

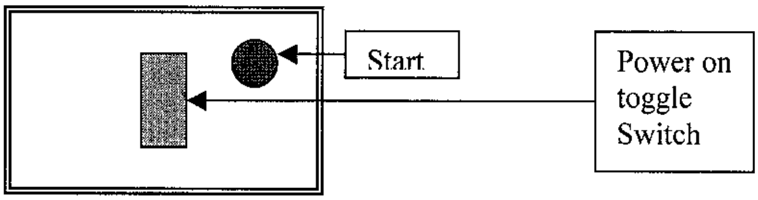
APPENDIX G

HOW TO OPERATE THE ROBOT

The following is the procedure on how to operate the robot:

G1. Starting operation

1. Plug on the power supply (12 V battery source) to the system power plugs.
2. Align the robot properly at the start zone.
3. The desired object to be picked up is placed any where from 20 cm to 40 cm in front of the robot.
4. Switch on the **power on toggle switch** on the control panel whose layout is as shown below.



5. Press **Start** to start the operation.
6. The operation can be stopped at any time by switching off the **power on toggle switch**. To start the operation again, turn on the **power on toggle switch** and press **Start**.

G2. Battery charging

1. Switch off the **Power on toggle switch.**
2. Unplug the system from the battery source.
3. Adjust the external power supply to 12 V.
4. Connect the battery source to the external supply for charging.
5. The charging is completed if the current value displayed on the external power supply reduces to 0.1 A.
6. Switch off the external power supply.
7. Disconnect the battery from the external power supply.
8. The battery is now ready to use in approximately 1 hour.

APPENDIX H

I/O ASSIGNMENT

Pin number	Description	Master	Slave 1	Slave 2
RB0	Right wheel forward	X		
RB1	Right wheel reverse	X		
RB2	Left wheel forward	X		
RB3	Left wheel reverse	X		
RB4	Front wheel controlling output	X		
RB5	Manipulator Joint 2 control	X		
RB6	Communication to Slave 1 & 2	X		
RB7	Communication to Slave 1 & 2	X		
RA0	Left wheel encoder	X		
RA1	Right wheel encoder	X		
RA2	Ultrasonic sensor triggering input	X		
RA0	Connected to Master RB7		X	
RA1	Connected to Master RB6		X	
RB0	To control relay U1 (Joint 1)		X	
RB1	To control relay U2 (Joint 1)		X	
RA0	Connected to Master RB7			X
RA1	Connected to Master RB6			X
RB0	To control gripper operation			X