SOFTWARE DEVELOPMENT OF SEARCH AND RETRIEVAL, OBJECT DETECTION AND OBSTACLES AVOIDANCE ALGORITHM FOR A MOBILE ROBOT

ZARIDAH BINTI MAT ZAIN

9 NOVEMBER 2005
SOFTWARE DEVELOPMENT OF SEARCH AND RETRIEVAL, OBJECT
DETECTION AND OBSTACLES AVOIDANCE ALGORITHM FOR A
MOBILE ROBOT

ZARIDAH BINTI MAT ZAIN

This Report Is Submitted In Partial Fulfillment Of Requirements For
The Degree of Bachelor In Electrical Engineering (Industry Power)

Fakulti Kejuruteraan Elektrik
Kolej Universiti Teknikal Kebangsaan Malaysia

9 NOVEMBER 2005
“I hereby declared that I have read through this report and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Electrical Engineering (Industrial Power).”

Signature : ........................................................................

MR. FARIZ BIN ALI @ IBRAHIM

Supervisor’s Name : ........................................................................

Date : 17 NOVEMBER 2005

........................................................................
"I hereby declared that this report is a result of my own work except for the excerpts that have been cited clearly in the references."

Signature : ..........................................................  
ZARIDAH BINTI MAT ZAIN  
Name : ..........................................................  
9 NOVEMBER 2005  
Date : ..........................................................
Special dedicated to my dearest parent, sisters and brothers
ACKNOWLEDGEMENTS

Alhamdulillah, thanks to the Almighty with his permission I have finished my Projek Sarjana Muda II within the given time. Firstly, I would like to express my gratitude and thanks to my project supervisor, Mr. Fariz Bin Ali @ Ibrahim for helping me without ceasing in guiding me through the documentation and information needed. Besides, I would like to say thanks to my friends for their support throughout this project.

My warmest gratitude also goes to all my colleagues and others that may not mentioned their name here, thanks for giving me all these supports. Without it my project may not be completed successfully.

Last but not least, to all my beloved family members, who were there to give their full support when I most need it.
ABSTRACT

This project is namely software development of search and retrieval, object detection and obstacles avoidance algorithm for a mobile robot is to build a mobile robot by using microcontroller PIC16F877 that capable of searching an object within three meters by using infrared sensor without any external connection. I’ve used PicBasic PRO for the programming and built a microcontroller circuit by using PIC16F877 microcontroller. Supposedly the robot will move towards the object after receiving signal from IR transmitter.
ABSTRAK

Projek ini dinamakan pembinaan perisian untuk robot mencari dan mengesan objek dengan menggunakan microcontroller PIC16F877 yang berkemampuan mencari objek yang ditetapkan pada jarak tiga meter dengan sendirinya dengan menggunakan infra red tanpa sambungan dari perkakasan luar. Saya telah menggunakan perisian PicBasic PRO untuk menulis aturcara dan telah membina perkakasan iaitu litar microcontroller untuk PIC16F877. robot ini sepatutnya akan bergerak ke hadapan ke arah objek selepas menerima isyarat daripada IR transmitter.
# CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>TITLE</td>
<td>i</td>
<td></td>
</tr>
<tr>
<td>DECLARATION</td>
<td>ii</td>
<td></td>
</tr>
<tr>
<td>DEDICATION</td>
<td>iii</td>
<td></td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>iv</td>
<td></td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>v</td>
<td></td>
</tr>
<tr>
<td>ABSTRAK</td>
<td>vi</td>
<td></td>
</tr>
<tr>
<td>CONTENTS</td>
<td>vii</td>
<td></td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xi</td>
<td></td>
</tr>
<tr>
<td>LIST OF APPENDICES</td>
<td>xiii</td>
<td></td>
</tr>
</tbody>
</table>

1 INTRODUCTION  
1.1 Overview    1  
1.2 Title and Objectives 2  
1.3 Scopes 3  
1.4 Project Workflow 3  
1.5 Work Schedules 5  

2 LITERATURE REVIEW  
2.1 Introduction 6  
2.2 What Is A Microcontroller 6  
2.3 Why Use A Microcontroller 8  
2.4 PIC Microcontroller Overview 10  
2.4.1 12-bit instruction core 11
2.4.2 14-bit instruction core 11
2.4.3 16-bit instruction core 13
2.4.4 Pic16F877 microcontroller 13
   2.4.4.1 CPU and ALU 14
   2.4.4.2 Program memory 15
   2.4.4.3 Hardware stack 15
   2.4.4.4 Data memory 15
   2.4.4.5 I/O ports 16
   2.4.4.6 EEPROM 17
   2.4.4.7 Timer 17
   2.4.4.8 PWM 18
   2.4.4.9 Synchronous serial port 18
   2.4.4.10 Interrupts 19
   2.4.4.11 USART 20
   2.4.4.12 Analogue to digital converter 20

2.4.5 Software for PIC 21
2.4.6 Assembly language 21
2.4.7 PicBasic compiler 22

III HARDWARE DEVELOPMENT 24
3.1 Introduction 24
3.2 Circuit Design And Explanations 25
   3.2.1 PIC microcontroller circuit 25
   3.2.2 Price list 27

IV SOFTWARE DEVELOPMENT 29
4.1 Introduction 29
4.2 Programming Strategy 29
4.3 The Compiler 32
4.4 PIC Programming Overview 32
4.5 Software and Hardware 32
4.6 PicBasic Pro Compiler 33
   4.6.1 ADCON 34
4.6.2 Pin and variable names 34
4.6.3 Ports and other registers 35
4.6.4 Pins 36
4.6.5 Comments 37
4.6.6 GOTO 37
4.6.7 High pin 37
4.6.8 Low pin 38
4.6.9 Input pin 39
4.6.10 If...then 40
4.6.11 Pause period 41
4.6.12 Math operators 42
4.7 Advantages of PicBasic Compiler 43
4.8 Microcode Studio 44
4.9 How to Use Microcode Studio 45
  4.9.1 Step1: writing code (the basic program) 45
  4.9.2 Step 2: using the compiler 50
    4.9.2.1 Target processor 50
    4.9.2.2 Compile “F9” 51
  4.9.3 Step 3: programming the PIC chip 51
4.10 IC-PROG Setting Procedure 52
  4.10.1 Selecting device 52
  4.10.2 Selecting programmer 53
  4.10.3 Option setting 54
  4.10.4 Oscillator setting 56
  4.10.5 Fuse setting 56
  4.10.6 Verify programming setting 56

V ANALYSIS AND FINDING 58
5.1 Introduction 58
5.2 Analysis 58
  5.2.1 Testing PIC16F877 59
    5.2.1.1 Project guidelines 59
    5.2.1.2 Part list 60
5.2.2 Testing running lights 62
  5.2.2.1 Project guidelines 63
  5.2.2.2 Part list 63
5.2.3 Testing 7 segment: dice 65
  5.2.3.1 Project guidelines 66
  5.2.3.2 Part list 66
5.2.4 Testing DC motor 68
  5.2.4.1 Project guidelines 69
  5.2.4.2 Part list 70

VI DISCUSSION AND CONCLUSION 72
6.1 Introduction 72
6.2 Discussion 72
6.3 Recommendation 73
  6.3.1 Detect smallest obstacles 73
  6.3.2 Display distance of obstacles 74
  6.3.3 Adjustable frequency 74
  6.3.4 Use more sensors 74
  6.3.5 Use servomotor 74
6.4 Conclusion 75

REFERENCES 75
APPENDIX 77
# LIST OF TABLES

<table>
<thead>
<tr>
<th>NO.</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Work Schedule for PSM 1</td>
<td>5</td>
</tr>
<tr>
<td>2.1</td>
<td>8-bit microcontrollers and their features</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>12-bit instruction core</td>
<td>11</td>
</tr>
<tr>
<td>2.3</td>
<td>14-bit instruction core</td>
<td>12</td>
</tr>
<tr>
<td>2.4</td>
<td>16-bit instruction core</td>
<td>13</td>
</tr>
<tr>
<td>2.5</td>
<td>Price list</td>
<td>27</td>
</tr>
<tr>
<td>5.1</td>
<td>Project guidelines for testing PIC16F877</td>
<td>59</td>
</tr>
<tr>
<td>5.2</td>
<td>Part list for testing PIC16F877</td>
<td>60</td>
</tr>
<tr>
<td>5.3</td>
<td>Project guidelines for testing running lights</td>
<td>63</td>
</tr>
<tr>
<td>5.4</td>
<td>Part list for testing running lights</td>
<td>63</td>
</tr>
<tr>
<td>5.5</td>
<td>Project guidelines for testing 7 segment: dice</td>
<td>66</td>
</tr>
<tr>
<td>5.6</td>
<td>Part list for testing 7 segment: dice</td>
<td>66</td>
</tr>
<tr>
<td>5.7</td>
<td>Project guidelines for testing DC motor</td>
<td>69</td>
</tr>
<tr>
<td>5.8</td>
<td>Part list for testing DC motor</td>
<td>70</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>NO.</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Project Workflow</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>PIC16F877 pins and functions</td>
<td>14</td>
</tr>
<tr>
<td>3.1</td>
<td>PIC16F877 microcontroller circuit</td>
<td>25</td>
</tr>
<tr>
<td>3.2</td>
<td>Microcontroller circuit on board</td>
<td>26</td>
</tr>
<tr>
<td>3.3</td>
<td>Oscillator</td>
<td>27</td>
</tr>
<tr>
<td>4.1</td>
<td>Programming flow chart</td>
<td>31</td>
</tr>
<tr>
<td>4.2</td>
<td>Microcode studio environment page</td>
<td>45</td>
</tr>
<tr>
<td>4.3</td>
<td>JDM programmer</td>
<td>52</td>
</tr>
<tr>
<td>4.4</td>
<td>IC-prog menu</td>
<td>53</td>
</tr>
<tr>
<td>4.5</td>
<td>Programmer setting menu</td>
<td>54</td>
</tr>
<tr>
<td>4.6</td>
<td>First option setting</td>
<td>55</td>
</tr>
<tr>
<td>4.7</td>
<td>Second option setting</td>
<td>55</td>
</tr>
<tr>
<td>4.8</td>
<td>Verify programming setting</td>
<td>57</td>
</tr>
<tr>
<td>5.1</td>
<td>Testing 16F877 circuit</td>
<td>61</td>
</tr>
<tr>
<td>5.2</td>
<td>Testing running lights circuit</td>
<td>65</td>
</tr>
<tr>
<td>5.3</td>
<td>7 segment circuit</td>
<td>67</td>
</tr>
<tr>
<td>5.4</td>
<td>Testing DC motor circuit</td>
<td>69</td>
</tr>
</tbody>
</table>
# LIST OF APPENDIX

<table>
<thead>
<tr>
<th>APPENDIX</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PIC16F877 DATASHEET</td>
<td>77</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The aim of developing this project, namely software development of search and retrieval, object detection and obstacles avoidance algorithm for a mobile robot is to build a mobile robot by using microcontroller PIC16F877 that capable of searching an object within three meters by itself without any external connection. This project is a simple robot and the tasks are limited but this is a starting for building a real robot which can be use in dangerous job such as put out fire and bomb.

In this project, BASIC language has been selected as the programming language and the program that I used is PicBasic PRO. PicBasic PRO is simple, easy to learn and construct. The most important reason is it can be programmed in modular forms and its compiler was small enough to fit on a diskette with ample room for programs as well.
This chapter will cover title and objectives, scopes, project workflow and the work schedule of this project.

1.2 TITLE AND OBJECTIVES

The Title of this project is “Software Development of Search and Retrieval, Object Detection and Obstacles Avoidance Algorithm for a Mobile Robot”. The main objective of this project is to develop a robot program and microcontroller circuit.

The main project objective can be defined in detail as below:

- To provide hands on skill in designs and programming.
- To enhance knowledge in computer interfacing.
- At the end of the project, we shall demonstrate on how the robot operates by using PIC16F877 microcontroller and the sensor.
- To study how to control the DC motor to make them move forward and backward by using PIC16F877 microcontroller output ports.
- To study the PIC16F877 microcontroller and how to program it with PIC programmer.
1.3 SCOPES

The scopes of this project are:

a. Mastering the software that is used in this project; such as MPLAB, Proteus and others.

b. Understanding the principles and theory of dc motor movement and how to program it with PicBasic PRO.

c. Understanding the hardware that is used in the project. The hardware included the PIC16F877 microcontroller.

1.4 PROJECT WORKFLOW

First of all I need to know the fundamental of this project, besides, a good procedure is needed to implement this project from beginning to the end. The workflow of my project is provided as shown in figure 1.1:
Figure 1.1: Project Workflow
<table>
<thead>
<tr>
<th>Aktiviti Projek</th>
<th>Project Activities</th>
<th>2004</th>
<th>2005</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Title selection</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Make a proposal of the project</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Literature review</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Draw flow chart</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Understanding software instructions, and how to execute the Hardware</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Write the programming</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Troubleshooting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Report</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

PIC microcontroller is a main element for this project because PIC can be used to control next elements. The type of PIC microcontroller used is PIC16F877. PIC microcontrollers are inexpensive, easy to purchase through various sources and the development tools for them are also inexpensive.

2.2 WHAT IS A MICROCONTROLLER?

A microcontroller, which is self sufficiency and cost effectiveness is use to control devices. The different between a microcontroller and microprocessor is a microcontroller contains all the memory including RAM (read access memory) and ROM (read only memory) and I/O interfaces needed, whereas a microprocessor requires additional chips to provide those necessary functions.
A microcontroller contains a CPU, memory for the program (ROM or Flash memory), memory for data (RAM), timers (customizable ones and watchdog timers), and I/O lines to communicate with external peripherals and complementary resources in a single integrated circuit. A microcontroller is different from a CPU chip because it is quite easy to make into a working computer, with a minimum of external support chips. To overcome this situation, the microcontroller will be placed in the device to control, hooked up to power and any information it needs.

A traditional microprocessor won't allow you to the task as it need to be handling by other chips. For example, some number of RAM memory chips must be added. The amount of memory provided is more flexible in the traditional approach, but at least a few external memory chips must be provided, and additionally requires that many connections must be made to pass the data back and forth to them.

A typical microcontroller usually will have a built in clock and memory such as EEPROM. To make it work, it needs software to control it and a timing crystal. Microcontrollers will also usually have a variety of input/output devices. For example, analog-to-digital converters, timers, UARTs, Serial Peripheral Interface and Controller Area Network.

Some modern microcontrollers include a built-in high-level programming language; BASIC is quite common for this.

Microcontrollers trade away speed and flexibility to gain ease of equipment design and low cost. There's only so much room on the chip to include functionality, so for every I/O device or memory increase the microcontroller includes, some other circuitry has to be removed. Finally, it must be mentioned that some microcontroller
architectures are available from many different vendors in so many varieties that they could rightly belong to a category of their own. [23]

2.3 WHY USE A MICROCONTROLLER?

Microcontroller is cheap and it’s ability to store and run unique programs make it popular. For example, we can program a microcontroller to make decision (perform functions) based on predetermined situations (I/O line logic) and selections. Besides, microcontroller also can perform mathematic and logic functions which allows it to mimic sophisticated logic and electronic circuit. A microcontroller also can be design like a neural circuit or a fuzzy logic controller.

There are hundreds types of microcontrollers on the market. Listed here are some of the popular 8-bit microcontroller and their features. These devices are the lowest cost representative devices from respective manufactures. [24]
<table>
<thead>
<tr>
<th>COMPANY</th>
<th>DEVICE</th>
<th>ON-CHIP MEMORY</th>
<th>OTHER FEATURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atmel</td>
<td>Attiny II</td>
<td>1-Kbyte flash</td>
<td>8-bit timer, analog comparator, watchdog, on-chip oscillators, one external interrupt.</td>
</tr>
<tr>
<td>Hitachi</td>
<td>H8/3640</td>
<td>8-Kbyte ROM 512 Byte RAM</td>
<td>Three, 8-bit timers, one 16-bit PWM timer, one watchdog, two SCI ports, eight 8-bit ADC, 32 KHz sub clock generator.</td>
</tr>
<tr>
<td>Microchip</td>
<td>PIC16CR54C</td>
<td>768-byte ROM 25-byte Ram</td>
<td>Twelve I/O pins, 8-bit timer, high current sink/source for direct LED drive, watchdog, and timer RC oscillator.</td>
</tr>
<tr>
<td>Motorola</td>
<td>68HC705KJ1</td>
<td>1240-byte OTP 64-byte RAM</td>
<td>15-stage multifunction timer, on-chip oscillator, low voltage reset, watchdog, keyboard, interrupt, high current I/O port.</td>
</tr>
<tr>
<td>Zilog</td>
<td>Z8E000</td>
<td>0.5-Kbyte OTP 32-byte RAM</td>
<td>One 16-bit timer, Watchdog, four hardware interrupts, 13 I/O pins.</td>
</tr>
</tbody>
</table>