A Rescue Mechanism for Borehole Accident using Robotics

S. Balasubramani[1] K. Sai Sandeep[2] K.Suprabath[3] K. Sai Dinesh[4]
[1]Assistant Professor [2,3,4]UG Student
[1,2,3,4]Department of Electronics & Communication Engineering
[1,2,3,4]R.M.D Engineering College, Tamilnadu 601206, India

Abstract— Many child deaths are reported in the past due to bore well accidents. During bore well accident an immediate rescue operation is required and it is quiet challenging to perform a rescue operation as the environment inside the bore well is highly unpredictable. Developing low cost robotic system and simple control will help rural people use it easily. In this paper, we propose a novel design of rescue robot which would enlarge as per the diameter of bore well and attach to the bore-well by adjusting to the diameter while travelling up or down the bore-well. It would contain two artificial arms, which would help in holding the baby with the visual help offered by camera and also aid in the survival of the child.

Keywords: Rescue Robot, Camera, Artificial arms

I. INTRODUCTION

In India, recently we have witnessed some of the most tragic but helpless incidents which touched us deeply and forced us to look after the matter seriously. As the statistics suggests in the consecutive years starting from 2006, still more than 30 deaths occurred while stuck in bore well.The most mournful fact in that figure is that 92% of that victim is under the age of 10. The children were playing around the bore well unaware of the fact that the bore well was waiting for them in the form of a death trap. After slipping in the rotten congested pitch black environment they were waiting for the help to come. But the lack of oxygen and deathly atmosphere has taken their life slowly before the rescue team can reach them. The incident of losing lives trapped in bore-well was highlighted in 2006 where a 5 year old child named Prince was rescued by Indian Army experts after a tough combat which lasted 49 hours. The boy showed tremendous survival instinct by remaining calm and being co-operative with survivors. Statistics reveal that not many kids were as lucky as Prince, many of them died; some received public attention, while many went unnoticed. Another incident in tamilnadu took place in the 2019 year where a child name sujith stuck in the pit hole and died for the lack of oxygen. We have tried to summarize the incidents in this concern. Now days we can see nearly everything which was once controlled by human being are being automated using machines and electronics circuits. An embedded system is a computer system designed to perform one or a few dedicated functions often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts. By contrast, a general-purpose computer, such as a personal computer (PC), is designed to be flexible and to meet a wide range of end-user needs. Embedded systems control many devices in common use today. Microcontrollers are commonly referred to as general purpose processors as they simply accept the inputs, process it and give the output. In contrast, a microcontroller not only accepts the data as inputs but also manipulates it, interfaces the data with various devices, controls the data and thus finally gives the required result. The main aim of this project is to design a rescue robot, can control with either computer with MATLAB or with joystick with the help of Zigbee technology, which can get into the borehole and save the child fallen in borehole. The controlling device of the whole system using Microcontroller

A. Objective

To enable a rescue mechanism for borehole accidents using robotics

II. LITERATURE SURVEY

A. Design of Automatic Measuring Robot for Steam Generator in Nuclear Power Plant

Abstract

In this paper, a kind of automatic measuring robot for steam generator is designed. The robot is created to satisfy the increasing demand of unmanned operation in nuclear power plant. To overcome the limited work space, a novel kind of mechanical structure is proposed. To accommodate large amount of measuring points, control system based on FPGA and AVR is established. The experiments are arranged to verify the function of the robotic system. The results show that the automatic measuring robot is reliable and accurate. More than that, the system has been accepted by China National Nuclear Corporation (C.N.N.C.), which stands for the high application value of robot system and new interests growing in nuclear energy field.

Author: Yang Yang Beijing Inst. of Technol., Beijing, China XingguangDuan

Beijing Inst. of Technol., Beijing, China Meng Li Beijing Inst. of Technol., Beijing, China

Published in: Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on 12-14 Dec. 2013

B. Omnidirectional Driving Gears and Their Input Mechanism with Passive Rollers

Abstract:

As ordinary dual-axis driving mechanisms in X-Y directions, for example, commercially available X-Y stages with ball screws are familiar. However, such driving mechanisms have two stages, namely both upper and lower linear actuators, the latter of which must generate sufficient thrust to carry large weights, including that of the upper actuator mechanism, which has hampered efforts to achieve suitably fast and smooth driving motion due to the inertial force effect. It is also difficult to achieve a small and slimline driving mechanism with such overlapping two-stage structure. In these ordinary two-stage driving mechanisms, the motion of the X-Y stage can be disturbed by the wires of the upper actuator. In this research, we have considered the abovementioned problems, and propose a new omnidirectional driving gear mechanism that enhances its driving area from the normal X-Y plane to convex and concave curved surfaces respectively, and even various combinations of both. The smoothness of basic omnidirectional motion and effectiveness of the driving method of this proposed omnidirectional driving gear mechanism have been confirmed with several experiments involving our setups.

Author: KenjiroTadakuma Osaka

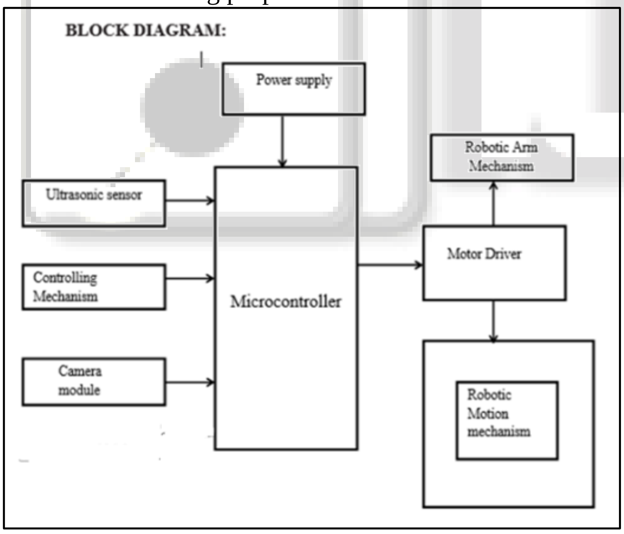
Published in: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on 7-12 Oct. 2015

III. PROPOSED SYSTEM

- In our project a proposed system would be a very dependable solution for such accidents.
- A robotic system can be used to locate the victim and also help to establish communication with the victim.
- In this paper, we would like to discuss the development of a robotic system.
- This system requires Ultrasonic sensor is used to get data of the distance between the robot wheel and the wall of the bore well.
- Camera module: The main function of camera module is to provide visual information to the operator.

A. *Block Diagram*

The block diagram shows the diagrammatic representation of our proposed system effectively for clear cut overview and for understanding purpose.



IV. HARDWARE REQUIREMENTS

- POWER SUPPLY
- MICROCONTROLLER
- ULTRASONIC SENSOR
- CAMERA
- SERVO MOTOR
- DC MOTOR
- BLUETOOTH
- MOTOR DRIVER

V. SOFTWARE REQUIREMENTS:

- EMBEDDED C
- MP LAB

A. *Description of Hardware and Software Requirements:*

1) *Power Supply:*

a) Transformer

The potential transformer will step down the power supply voltage (0-230V) to (0-6V) level. Then the secondary of the potential transformer will be connected to the precision rectifier, which is constructed with the help of op–amp. The advantages of using precision rectifier are it will give peak voltage output as DC, rest of the circuits will give only RMS output.

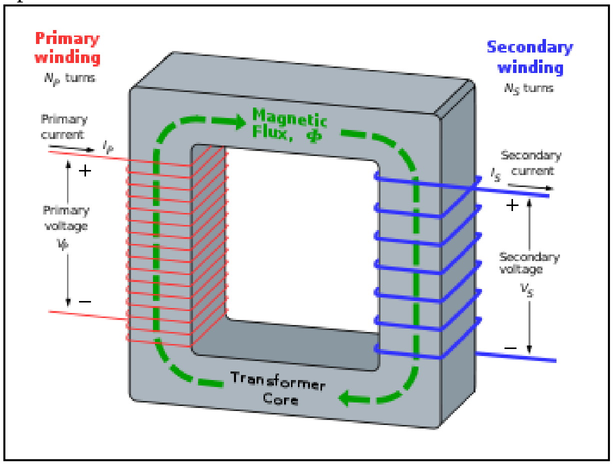


Fig. 2: Bridge rectifier

When four diodes are connected as shown in figure, the circuit is called as bridge rectifier. The input to the circuit is applied to the diagonally opposite corners of the network, and the output is taken from the remaining two corners.

Let us assume that the transformer is working properly and there is a positive potential, at point A and a negative potential at point B. the positive potential at point A will forward bias D3 and reverse bias D4.

The negative potential at point B will forward bias D1 and reverse D2. At this time D3 and D1 are forward biased and will allow current flow to pass through them; D4 and D2 are reverse biased and will block current flow.

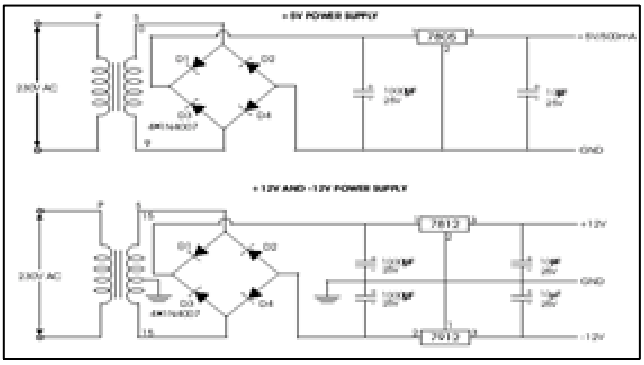
The path for current flow is from point B through D1, up through RL, through D3, through the secondary of the transformer back to point B. this path is indicated by the solid arrows. Waveforms (1) and (2) can be observed across D1 and D3.

One-half cycle later the polarity across the secondary of the transformer reverse, forward biasing D2 and D4 and reverse biasing D1 and D3.Current flow will now be from point A through D4, up through RL, through D2, through the secondary of T1, and back to point A. This path is indicated by the broken arrows. Waveforms (3) and (4) can be observed across D2 and D4. The current flow through RL is always in the same direction. In flowing through RL this current develops a voltage corresponding to that shown waveform (5). Since current flows through the load (RL) during both half cycles of the applied voltage, this bridge rectifier is a full-wave rectifier.

One advantage of a bridge rectifier over a conventional full-wave rectifier is that with a given transformer the bridge rectifier produces a voltage output that is nearly twice that of the conventional full-wave circuit.

2) IC voltage regulators

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustably set voltage. The regulators can be selected for operation with load currents from hundreds of milli amperes to tens of amperes, corresponding to power ratings from milli watts to tens of watts.



A fixed three-terminal voltage regulator has an unregulated dc input voltage, Vi, applied to one input terminal, a regulated dc output voltage, Vo, from a second terminal, with the third terminal connected to ground.

The series 78 regulators provide fixed positive regulated voltages from 5 to 24 volts. Similarly, the series 79 regulators provide fixed negative regulated voltages from 5 to 24 volts.

VI. MICROCONTROLLER

All the functions required on a single chip. A microcontroller differs from a microprocessor, which is a general-purpose chip that is used to create a multi-function computer or device and requires multiple chips to handle various tasks. A microcontroller is meant to be more self-contained and independent, and functions as a tiny, dedicated computer.

They are typically designed using CMOS (complementary metal oxide semiconductor) technology, an efficient fabrication technique that uses less power and is more immune to power spikes than other techniques. There are also multiple architectures A microcontroller is an integrated chip that is often part of an embedded system. The microcontroller includes a CPU, RAM, ROM, I/O ports, and timers like a standard computer, but because they are designed to execute only a single specific task to control a single system, they are much smaller and simplified so that they can include used, but the predominant architecture is CISC (Complex Instruction Set Computer), which allows the microcontroller to contain multiple control instructions that can be executed with a single macro instruction. Some use a RISC (Reduced Instruction Set Computer) architecture, which implements fewer instructions, but delivers greater simplicity and lower power consumption.

Early controllers were typically built from logic components and were usually quite large. Later, microprocessors were used, and controllers were able to fit onto a circuit board. Microcontrollers now place all of the needed components onto a single chip. Because they control a single function, some complex devices contain multiple microprocessors.

Microcontrollers have become common in many areas, and can be found in home appliances, computer equipment, and instrumentation. They are often used in automobiles, and have many industrial uses as well, and have become a central part of industrial robotics. Because they are usually used to control a single process and execute simple instructions, microcontrollers do not require significant processing power.

Products like CD players, hi-fi equipment, video games, washing machines, cookers and so on fit into this category. The communications market, automotive market, and the military share the rest of the application areas. Microcontrollers have traditionally been programmed using the assembly language of the target microcontroller. Different microcontrollers from different manufacturers have different assembly languages.

Assembly language consists of short mnemonic descriptions of the instruction sets. These mnemonics are difficult to remember and the programs developed for one microcontroller cannot be used for other types of microcontrollers. The most common complaint about microcontroller programming is that the assembly language is somewhat difficult to work with, especially during the development of complex projects.

The solution to this problem is to use high-level languages. This makes the programming a much simpler task and the programs are usually more readable, portable, and easier to maintain. There are various forms of BASIC and C compilers available for most microcontrollers. BASIC compilers are usually in the form of interpreters and the code produced is usually slow.

A. *Evolution of Microcontroller*

First, microcontrollers were developed in the mid-1970s. These were basically calculator-based processors with small ROM program memories, very limited RAM data memories, and a handful of input/output ports. As silicon technology developed, more powerful, 8-bit microcontrollers were produced. In addition to their improved instruction sets, these microcontrollers included on-chip counter/timers, interrupt facilities, and improved I/O handling. On-chip memory capacity was still small and was not adequate for many applications. One of the most significant developments at this time was the availability of on-chip ultraviolet erasable EPROM memory. This simplified the product development time considerably and, for the first time, also allowed the use of microcontrollers in low-volume applications.

B. *Introduction of 8051 Family*

The 8051 family was introduced in the early 1980s by Intel. Since its introduction, the 8051 has been one of the most popular microcontrollers and has been second-sourced by many manufacturers. The 8051 currently has many different versions and some types include on-chip analogue-to-digital converters, a considerably large size of program and data memories, pulse-width modulation on outputs, and flash memories that can be erased and reprogrammed by electrical signals. Microcontrollers have now moved into the 16-bit market. 16-bit micro-controllers are high-performance processors that and applications in real-time and compute intensive fields (e.g. in digital signal processing or real-time control). Some of the 16-bit microcontrollers include large amounts of program and data memories, multi-channel analogue-to-digital converters, a large number of I/O ports, several serial ports, high-speed arithmetic and logic operations, and a powerful instruction set with signal processing capabilities.

The simplest microcontroller architecture consists of a microprocessor, memory, and input/output. The microprocessor consists of a central processing unit (CPU) and the control unit (CU).The CPU is the brain of a microprocessor and is where all of the arithmetic and logical operations are performed. The control unit controls the internal operations of the microprocessor and sends control signals to other parts of the microprocessor to carry out the required instructions. Memory is an important part of a microcomputer system. Depending upon the application we can classify memories into two groups: program memory and data memory. Program memory stores all the program code. This memory is usually a read-only memory (ROM). Other types of memories, e.g. EPROM and PEROM flash memories are used for low-volume applications and also during program development. Data memory is a read/write memory (RAM).In complex applications where there may be need for large amounts of memory it is possible to interface external memory chips to most microcontrollers.

Input/ Output (I/O) ports allow external digital signals to be connected to the microcontroller. I/O ports are usually organized into groups of 8 bits and each group is given a name. For example, the 8051 microcontroller contains four8-bit I/O ports named P0, P1, P2, and P3. On some microcontrollers the direction of the I/O port lines are programmable so that different bits of a port can be programmed as inputs or outputs. Some microcontrollers (including the8051 family) provide bi-directional I/O ports. Each I/O port line of such microcontrollers can be used as inputs and outputs. Some microcontrollers provide `open-drain' outputs where the output transistors are left floating (e.g. port P0 of the 8051 family). External pull-up resistors are normally used with such output port lines.

C. 8051 Family:

The 8051 family is a popular, industry standard 8-bit single chip micro-computer (microcontroller) family, manufactured by various companies with many different capabilities. The basic standard device, which is the first member of the family, is the 8051, which is a 40-pin microcontroller. This basic device is now available in several configurations. The 80C51 is the low-power CMOS version of the family. The 8751 contains EPROM program memory, used mainly during development work. The 89C51 contains flash programmable and erasable memory (PEROM) where the program memory can be reprogrammed without erasing the chip with ultraviolet light. The 8052 is an enhanced member of the family which contains more RAM and also more timer/counters. There are many versions of the 40-pin family which contain on-chip analogue-to-digital converters, pulse-width modulators, and so on. At the lower end of the 8051 family we have the 20-pin microcontrollers which are code compatible with the 40-pin devices. The 20-pin devices have been manufactured for less complex applications where the I/O requirements are not very high and where less power is required (e.g. in portable applications). The AT89C1051 and AT89C2051 (manufactured by Atmel) are such microcontrollers, which are fully code compatible with the 8051 family and offer reduced power and less functionality.The 8051 is an 8-bit, low-power, high-performance microcontroller. There are a large number of devices in the 8051 family with similar architecture and each member of the family is downward compatible with each other. The basic 8051microcontroller has the following features:

- 4 Kbytes of program memory
- 256*8 RAM data memory
- 32 programmable I/O lines
- Two 16-bit timer/counters
- Six interrupt sources
- Programmable serial UART port
- External memory interface
- Standard 40-pin package

D. PIC16F877A

The term PIC, or Peripheral Interface Controller, is the name given by Microchip Technologies to its single – chip microcontrollers. PIC micros have grown to become the most widely used microcontrollers in the 8- bit microcontroller segment.

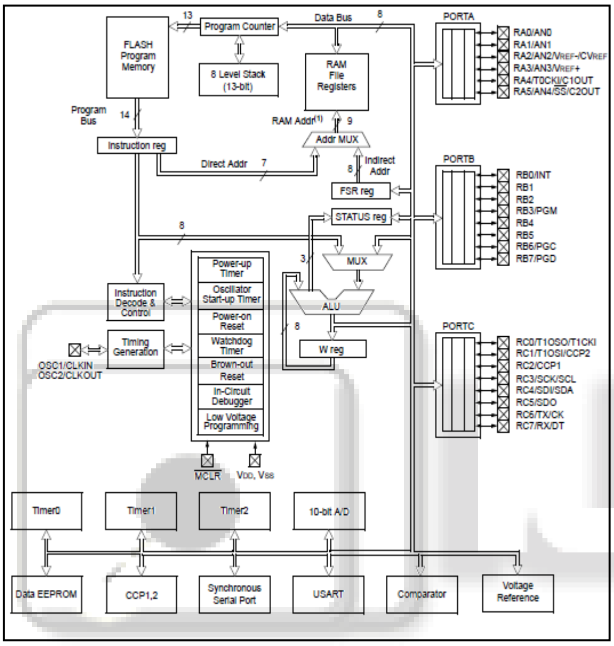
The PIC16F877A CMOS FLASH-based 8-bit microcontroller is upward compatible with the PIC16C5x, PIC12Cxxx and PIC16C7x devices. It features 200 ns instruction execution, 256 bytes of EEPROM data memory, self-programming, an ICD, 2 Comparators, 8 channels of 10-bit Analog-to-Digital (A/D) converter, 2 capture/compare/PWM functions, a synchronous serial port that can be configured as either 3-wire SPI or 2-wire I2C bus, a USART, and a Parallel Slave Port.

1) Special Microcontroller Features

- Flash Memory: 14.3 Kbytes (8192 words)
- Data SRAM: 368 bytes
- Data EEPROM: 256 bytes
- Self-reprogrammable under software control
- In-Circuit Serial Programming via two pins (5V)
- Watchdog Timer with on-chip RC oscillator
- Programmable code protection
- Power-saving Sleep mode
- In-Circuit Debug via two pins
- 10-bit, 8-channel A/D Converter
- Brown-Out Reset
- Analog Comparator module
- Peripheral Features
- 33 I/O pins; 5 I/O ports
- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler
- Can be incremented during Sleep via external crystal/clock

- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - 16-bit Capture input; max resolution 12.5 ns
  - 16-bit Compare; max resolution 200 ns
  - 10-bit PWM
- Synchronous Serial Port with two modes:
  - SPI Master
  - I2C Master and Slave
- USART/SCI with 9-bit address detection
- Parallel Slave Port (PSP)
  - 8 bits wide with external RD, WR and CS controls
- Brown-out detection circuitry for Brown-Out Reset

E. Block Diagram of PIC16F877A



The internal block diagram of PIC16F877 is shown in the figure. It contains 4-banks of register files such as Bank 0, Bank 1, Bank 2 and Bank 3 from 00h-07h, 80h-FFh, 100h-17Fh and 180h-1FFh respectively. And it is also having program FLASH memory, Data memory and Data EEPROM of 8K, 368 and 256 Bytes respectively.

1) Register File

The term register file in PIC terminology used to denote the locations than an instruction can access via an address. The register file consists of two components, they are

1) General purpose register file
2) Special purpose register file

1) General Purpose Register File

The general-purpose register file is another name for the microcontrollers RAM. Data can be written to each 8-bit location, updated and retrieved any number of times. All control registers are coming under the general purpose register file.

2) Special Purpose Register File

The special purpose register file contains input and output ports as well as the control registers used to establish each bit of port as either an input or output. It contains registers that provide the data input and data output to the variety of resources on the chip, such as the timers, the serial ports and the analog–to–digital converter.

2) Pin Configuration And Description

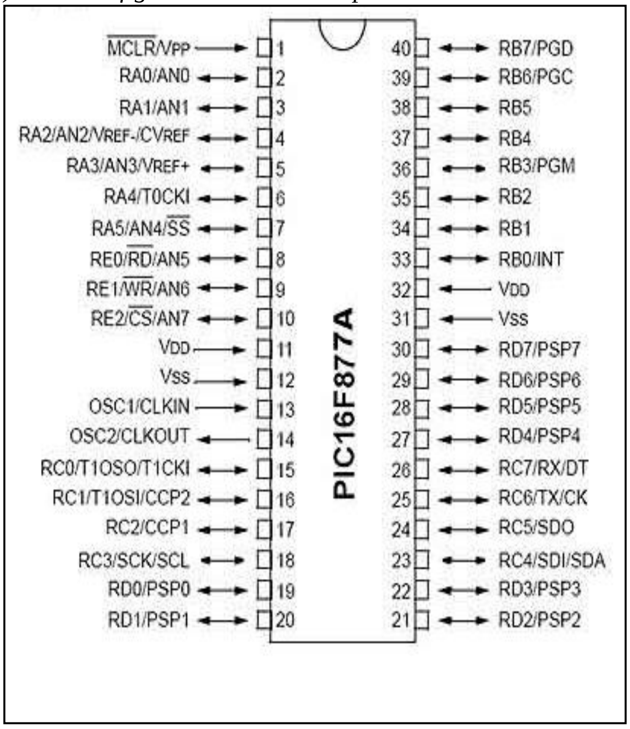


Fig. 4.2: Pin Diagram of PIC16F877A

The pins RB0-RB7, RC0-RC7, and RD0-RD7 are digital I/O pins. The pins CCP1 and CCP2, which share locations with RC1 and RC2, can be used for a PWM signal (see DC Motor tutorial). The pins AN0-AN7 are for analog I/O (see Photo resistor tutorial). TX and RX are for debugging I/O (see Output Messages to Computer tutorial). The remaining pins deal with power/ground, the clock signal, and programmer I/O.

A PIC is made of several "ports." Each port is designated with a letter, RB0-RB7 are a port. RC0-RC7 and RD0-RD7 are a port as well. RA0-RA5 and RE0-RE2 are also ports, but with fewer pins. Some of these pins have special purposes, but most can be used as basic input/output pins.

On the other hand, if you wanted to set RBO as an output pin, you could choose to make RB0 either be 5v, or 0v. This can be used, for example, to turn off or on a LED, or to turn off or on a motor.

F. Memory Organization

There are three memory blocks in each of the PIC16F87XA devices. The Program Memory and Data Memory have separate buses so that concurrent access.

G. Program Memory Organization

The PIC16F87XA devices have a 13-bit program counter capable of addressing an 8K word x 14 bit program memory space. The PIC16F876A/877A devices have 8K words x 14 bits of FLASH program memory, while PIC16F873A/874A devices have 4K words x 14 bits. Accessing a location above the physically implemented address will cause a wraparound. The RESET vector is at 0000h and the interrupt vector is at 0004h.

Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 (STATUS<6>) and RP0 (STATUS<5>) are the bank select bits.

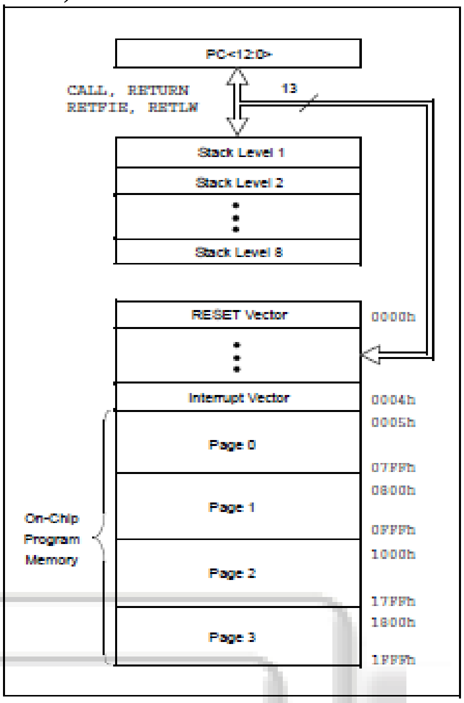


Fig. 4.3: Memory Organization of Pic16f877a

RP1:RP0	Bank
00	0
01	1
10	2
11	3

Table 4.2: Register and Bank Allocation

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain Special Function Registers. Some frequently used Special Function Registers from one bank may be mirrored in another bank for code reduction and quicker access.

NAME	FUNCTION	ADDRESS RAM ADDRESS
STATUS	Status register	03H,83H,103H
FSR	File select register	04H,84H,104H
PORTA	I/O latch	05H
PORTB	I/O latch	06H
PORTC	I/O latch	07H
PORTD	I/O latch	08H
PORTE	I/O latch	09H
INTCON	Interrupt control register	0BH,8BH,10BH
PIR1	Peripheral interrupt	0CH
RCSTA	Receive status and control register	18H
TXREG	Transmit register	19H
RCREG	Receive register	1AH
OPTION_REG	Optional register	81H
TRISA	I/O register	85H
TRISB	I/O register	86H
TRISC	I/O register	87H
TRISD	I/O register	88H
TRISE	I/O register	89H

Table 4.3: Register File Maps Used In Pic16f877

H. Embedded C Portability

By design, a number of properties in Embedded C are left implementation defined. This implies that the portability of Embedded C programs is not always guaranteed. Embedded C provides access to the performance features of DSPs. As not all processors are equal, not all Embedded C implementations can be equal For example, suppose an application requires 24-bit fixed-point arithmetic and an Embedded C implementation provides only 16 bits because that is the native size of the processor. When the algorithm is expressed in Embedded C, it will not produce outputs of the right precision.

In such a case, there is a mismatch between the requirements of the application and the capabilities of the processor. Under no circumstances, including the use of assembly, will the algorithm run efficiently on such a processor. Embedded C cannot overcome such discrepancies. Yet, Embedded C provides a great improvement in the portability and software engineering of embedded applications. Despite many differences between performance-specific processors, there is a remarkable similarity in the special-purpose features that they provide to speed up applications.

Writing C code with the low-level processor-specific support may at first appear to have many of the portability problems usually associated with assembly code. In the limited experience with porting applications that use Embedded C extensions, an automotive engine controller application (about 8000 lines of source) was ported from the eTPU, a 24-bit special-purpose processor, to a general-purpose 8-bit Freescale 68S08 with about a screen full of definitions put into a single header file. The porting process was much easier than expected. For example, variables that had been implemented on the processor registers were ported to unqualified memory in the general-purpose microprocessor by changing the definitions in the header definition and without any actual code modifications. The exercise was to identify the porting issues and it is clear that the performance of the special-purpose processor is significantly higher than the general-purpose target.

1) MPLAB IDE

1) Free integrated development environment (IDE) from Microchip to implement code for PICs
2) Latest version 8.60(recommended), In Lab 7.6
3) IDE and documentation (user guide) can be downloaded from the Microchip website
4) To open MPLAB IDE

5) Start→All Programs→Microchip→MPLAB IDE v8.60→MPLAB IDE

2) Project Creation

A project must be created for implementation

1) Specify your device
2) Create and edit your files
3) Compile and link your project
4) Program the device
5) Project→Project Wizard…
6) Begin project and specify device
7) Select Microchip MPASM Tool suite
8) Create New Project File
9) Add project files

3) Updating Source Code

- Modify .asm files
- Under 'Source Files' folder in project window
- Open and edit files to implement new functionality
- Additional files can be created and added

4) Building Projects

- Project→Build All (or Ctrl+F10)
- Output window indicates success or failure
- HEX file generated when project is built

5) Programming the Pic

Directly Downloading Hex files to the PIC Memory

- Used to transfer HEX file to the PIC and begin program execution
- Two methods (HEX file must be generated)
- MPLAB IDE
  - Programmer→Select Programmer→PICKit3
- PICkit3
  - Separate program
  - Start→All Programs→Microchip→PICkit 3

VII. CONCLUSION

Integrating features of all the hardware components used have been developed in it. Presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Secondly, using highly advanced IC‟s with the help of growing technology, the project has been successfully implemented. Thus the project has been successfully designed and tested.

REFERENCE

[1] R, Azuma, "A survey of augmented reality," Presence, vol. 6, pp. 355–385, 1997.

[2] F. Zhou, H. Duh, and M. Billinghurst, "Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR," In 7th IEEE/ACM International. Symposium on Mixed and Augmented Reality, Cambridge, UK, pp.193-202, 2008.

[3] V. Heun, S. Kasahara, and P. Maes, "Smarter Objects: Using AR technology to Program Physical Objects and their Interactions," In CHI '13 Extended Abstracts on Human Factors in Computing Systems, ACM, New York, USA, pp. 961-966, 2013.

[4] S. Kasahara, R. Niiyama, V. Heun, and H. Ishii, "exTouch: spatiallyaware embodied manipulation of actuated objects mediated by augmented reality," Proc. of the 7th Intl. Conf. on Tangible, Embedded and Embodied Interaction, ACM, New York, USA, pp. 223-228, 2013.

[5] J. Lee, J. I. Kim, J. Kim, and J. Y. Kwak, "A Unified Remote Console Based on Augmented Reality in a Home Network Environment," Consumer Electronics, Intl. Conf. on Digest of Technical Papers, pp. 1-2, 2007.

[6] M. Billinghurst, M. Hakkarainen, and C. Woodward, "Augmented assembly using a mobile phone," Procs.of the 7th International Conf. on Mobile and Ubiquitous Multimedia, ACM, New York, USA, pp.84-87, 2008.

[7] J. Quarles, S. Lampotang, I. Fischler, P. Fishwick, and B. Lok,."A Mixed Reality Approach for Merging Abstract and Concrete Knowledge," Procs.of the Virtual Reality Conf., pp. 27-34, 2008.

[8] G. Soros, H. Seichter, P. Rautek, and E. Groller, "Augmented visualization with natural feature tracking," In Procs.10th Intl. Conf. on Mobile and Ubiquitous Multimedia, New York, pp. 4-12, 2011.