IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 5, Ver. VI (Sep-Oct. 2014), PP 78-84
www.iosrjournals.org

Building Of Smart Systems Using Mechatronic Engineering: A Case Study Of 'Smart Door' System.

Jonathan O. Obande[1], Garba M. Rabiu[2] and John D. Galadima[3]
[1]*Department of Electrical/Electronic Engineering. Nasarawa State Polytechnic, Lafia. Nasarawa State, Nigeria.*
[2]*Department of Computer Science. Nasarawa State Polytechnic, Lafia. Nasarawa State, Nigeria.*
[3]*Department of Mechanical Engineering. Nasarawa State Polytechnic, Lafia. Nasarawa State, Nigeria.*

Abstract: *Often, Mechanical, Electrical and Software Engineers in many companies live and work from different locations. In some cases, they may be in the same building or the same office but live in different worlds, speaking different languages and therefore, cannot effectively communicate with one other when it comes to product design or problem solving; simply because they come from different backgrounds knowing very little about other disciplines. For instance, when the Mechanical Engineers design a system, they pass it over to the Electrical/Electronic Engineers to design and fit the control system and they in turn roll it over to the Software Engineers to write the control program. This serial and disjointed engineering practice results in producing and un-optimized product or solution. To overcome these difficulties, Mechatronics evolved as a trans-disciplinary approach to solving engineering problems based on open communication systems and concurrent practices, to design better engineering products. This paper describes the building of smart systems using mechatronics command set; the Flowcode programming software was used in coding instruction into the PIC Microprocessor 16F877A.*

Keywords*: Mechatronics, Flowcode, Microcontroller, Servo motor*

I. Introduction

The term `mechatronics' is coined out of two words; **mecha**nics and elec**tronics**. Integrated mechanical – electronic systems emerged from suitable combination of mechanics, electronics and control or information processing. Mechatronics in its fundamental form can be regarded as the fusion of mechanical and electrical disciplines in modern engineering processes. In the IEEE/ASME Transactions on mechatronics a preliminary definition is given as, "the synergetic integration of mechanical engineering with electronics and intelligent computer control in the design and manufacturing of industrial products and processes", [1] (see figure 1). This integration is between the computers (hardware) and the information driven function (software), resulting in the integrated systems. Their development involves finding an optimal balance between the basic mechanical structure, sensor and actuator implementation, automatic digital information processing and overall control.

II. Historical Overview

The word "mechatronics" was probably first created by a Japanese engineer; Kyura Oho in 1969. On or before 1900, pure mechanical systems like; the steam engines of 1860s, DC motors of 1870s, circular pumps and combustion engines of 1880s, AC motors of 1889, mechanical typewriters, and tool machines were all associated with the era of increasing electrical drives. By 1920, electro-mechanical systems with electrical drives like; relays, solenoids, hydraulic and pneumatic pumps, electric amplifiers, the PI-controllers of 1930s were associated with the era of increasing automatic control [2]. By 1935, mechanical systems with automatic control like; the transistors of 1948, thyristors of 1955, steam turbines and aircrafts were also associated with the era of increasing automatic control. Between 1955 and 1975, mechanical systems with analog electronic and sequential control like; the digital computers of 1955, process computers of 1971, real-time software of 1966, micro-computers of 1971, digital decentralized automations of 1975, and electronic controlled lifts were associated with the era of increasing automation with process computers and miniaturization. By 1985, mechatronic systems like; mobile robots, Computer Integrated Manufacturing (CIM), magnetic bearings, automotive control (ABS/ESP) were also associated with the era of increasing integration of processes and microcomputers [3].

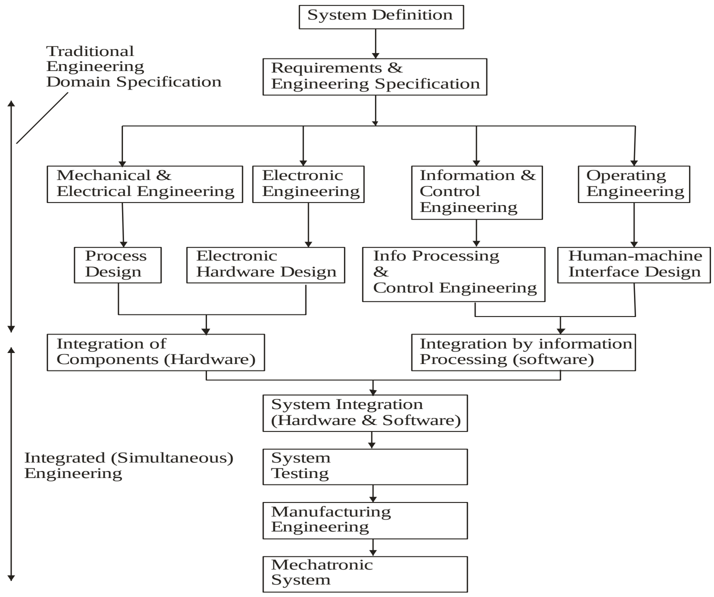

Fig. 1: Design procedure for Mechtronic systems.(Isremannn, 2005)

III. Design Procedures For Mechatronic Systems

The design of mechatronic systems requires a systematic development and utilization of modern software design tools. As with any design, mechatronic design is also an iterative procedure. Figure 2 shows that in addition to the traditional domain specific engineering, an integrated simultaneous (concurrent) engineering is required.

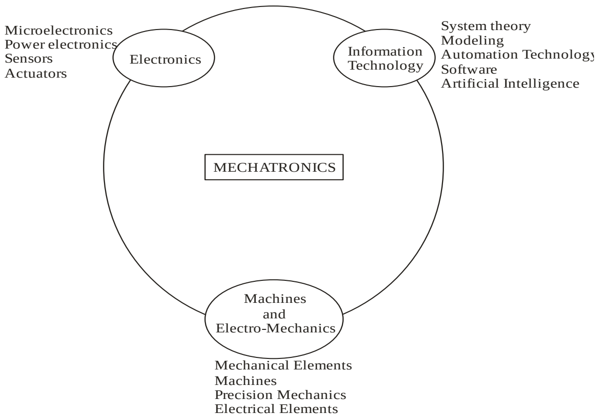

Fig 2 Mechatronics- synergetic integration of different disciplines (Isremann, 2005)

IV. Methodology

This project involves the building of a smart door system using mechatronics engineering principles. A typical mechatronics system picks up signals, processes them as an autonomous system by using sensors, microcontrollers/microprocessor and other electronic components.

4.1 Description of the Device

A smart door is a type of door which opens automatically if someone approaches a building entrance allowing easy access to the building. These doors are becoming popular in most public buildings as they eliminate the need for hiring a doorman to usher in visitors.

Fig 3 Block diagram of the Smart door

Fig 3 shows the working principles of the Smart door in block diagram form. The microcontroller gets input signal from the sensor and the power unit, processes the signal and gives out instruction to the actuator which is the output device, which in turn carries out the function of opening or closing the door.
The smart door is made up of the following sub circuits:

4.1.1 Transducers

A transducer is a device which measures a physical quantity and converts it into a signal which can be read by an observer [4]. The sensors used in this system are:

- ◊ Servo motor
- ◊ Sharp GP2Y0A02YK0F IR proximity sensor

4.1.2 Analog to Digital Converter

In the physical world, parameters such as distance, temperature, pressure, humidity, and velocity are analog signals. For a physical quantity to be converted into electrical signals, an analog to digital converter (ADC) is required. The ADC is an electronic circuit that converts continuous signals (analog) into discrete form (digital) so that the microcontroller can read the data. Analog to digital converters are the most widely used devices for data acquisition. Fig 4 describes the conversion of analog to digital signals in block diagram form.

Fig. 4: Getting data from the analog world

4.1.3 Microcontroller (pic16F877A)

The microcontroller is the heart of the proposed embedded system. It constantly monitors the digitized parameters of the various sensors and verifies them with the predefined threshold values. It checks if any corrective action is to be taken for the condition at that instant of time, and if such a situation arises, it activates the actuators to perform a controlled operation. Fig AA shows the pin configuration of the PIC 16F877A microcontroller.

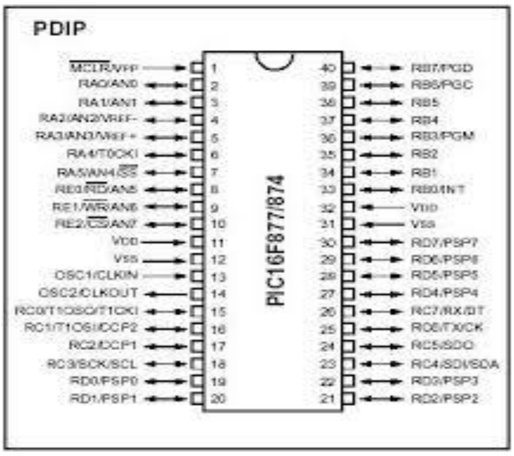

Fig 5: PIC 16F877A microcontroller. (Source: hobbyprojects.com)

4.1.4 Liquid Crystal Display

A liquid crystal display (LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector. Each pixel consists of a column of liquid crystal molecules suspended between two transparent electrodes, and two polarizing filters, the axes of polarity of which are perpendicular to each other. The LCD is used as an indicator to monitor the state of the door whether it is opened or closed.

4.1.5 Power Supply Connection

The power supply section consists of step down transformers of 230V primary to 9V and 12V secondary voltages for the +5V and +12V power supplies respectively**.**

4.2 Hardware

- ◊ Servo motor
- ◊ Sharp GP2Y0A02YK0F IR proximity sensor
- ◊ Character-based LCD (16X2)

- ◊ Connection wire
- ◊ Phase Plus PIC16F877A development board
- ◊ Potentiometer (optional)
- ◊ 10K resistor (optional)

4.2.1 Servo Motor

Servo motors are motors which use error sensing negative feedback for effective position control based on a modulated signal. It could either be a DC or AC motor combined with a position sensing device, such as a digital decoder and could be of linear or rotary type. The motor incorporates a DC motor, a gear-train; limit stops beyond which the shaft cannot turn, a potentiometer for position feedback, and an integrated circuit for position control. The motor retains the shaft angular position so long as the modulated signals exist on the input terminal. The angular position of the shaft is changed by altering the modulated signal.

There are different varieties and classes of servo motors. This project is based on the remote control (RC servo motor, typically used on radio-controlled models to provide mechanism actuation. The three terminals of the motor are +5V, GND and control input signal. The input signal terminals takes in a pulse-width modulated (PWM) signals and commands the position of the motor shaft. RC Servo motors are extremely useful in Mechatronics applications. Despite their size compared to standard brushed or brushless motors, they are extremely powerful. For the purpose of this work the HiTEC HC-422 was used in implementing the program.

4.2.2 Sharp GP2Y0A02YK0F IR proximity sensor

Sharp GP2Y0A02YK0F IR proximity sensor is a three terminal IR sensor module comprising an integrated combination of PSD (position sensitive detector), IRED (infrared emitting diode) and signal processing circuit. The sensor operates on the principles of triangulation where the location of a point is determined by measuring angles to it from known points at either ends of a fixed baseline. This device outputs a voltage which non-linearly corresponds to the detection distance. So this sensor can be used as a proximity sensor as well as distance measuring sensor. The three terminals of the sensors are output signal terminals, +5v and GND.

4.3 Software

4.3.1 Flowcode: Flowcode is a very high level language programming system for PICmicro microcontrollers based on flowcharts. Flowcode allows you to design complex PIC based robotics and control systems by simply drawing a flow chart of your desired program in a matter of minutes even without any prior programming skills.

Flowcode is built on a C compiler - C2C. This is a general purpose 8/16 bit compiler designed specifically for PICmicro devices. Flowcode generates a C code file from the flow chart that you create. This C-code is automatically compiled into assembler code by the C2C compiler and then translated into a Hex machine code file using Microchip's MPASM assembler.

Any third party PIC programmer can then be used to download the resulting Hex file into the target PIC's program memory.

If Flowcode is used with an optional training and development board then this whole operation of compiling assembling and downloading is carried out with one button, providing a totally seamless PICmicro development tool.

4.3.2 Device Database: Flowcode runs on Windows 98, 2000, NT, ME and XP and supports code generation for a wide range of PIC devices: PIC12C671 to PIC16F877A

***4.3.3 Peripheral Simulation**:* The Flowcode provides complete simulation for the CPU and on chip peripherals of most embedded devices.

4.3.4 Programmer

The programmer used is a powerful programmer for the *PIC* series of microcontrollers. Major parts of this programmer are Serial Port, Power Supply and Firmware microcontroller. Serial data is sent and received from 9 pin connector and converted to/from TTL logic/RS232 signal levels by MAX232 chip. A Male to Female serial port cable, connects to the 9 pin connector of hardware and the other side connects to the back of computer.

V. Implementation Procedure

The implementation of this work was done on a breadboard. The power supply was first derived from a bench power supply in the laboratory. (To confirm the workability of the circuits before the power supply stage was soldered).

Stage by stage testing was done according to the block representation on the breadboard, before soldering the components on the Vero board. The various circuits and stages were soldered in tandem to meet the desired workability of the hardware.

Three general steps were followed to appropriately select the control system [5]:
Step # 1: Identify measurable variables important to Production.
Step # 2: Investigate the control strategies using threshold sensors that directly affect actuation of devices.
Step # 3: Identify the software and the hardware to be used considering factors such as reliability, support, previous experiences with the equipment (successes and failures), and cost.

5.1 Construction and Testing

The motor input signal terminal and the sensor's signal terminal connect to pin D1 and A0 of the microcontroller. While their power terminals connect to +5V and GND. The LED's and node connect to pins D4 of the microcontroller and the cathode connects to GND. The listing below indicates how the LCD connects to the microcontroller.

- ◊ Data lines (D0-D3) connects to PORTB pins RB0-RB3
- ◊ Enable, read/write RS pins connects to B5, GND and B4 respectively

The LCD back light pins, A and K connects to V_{CC} via a 220 resistor and GND
Fig. 6 depicts the functional schematic diagram for the project.

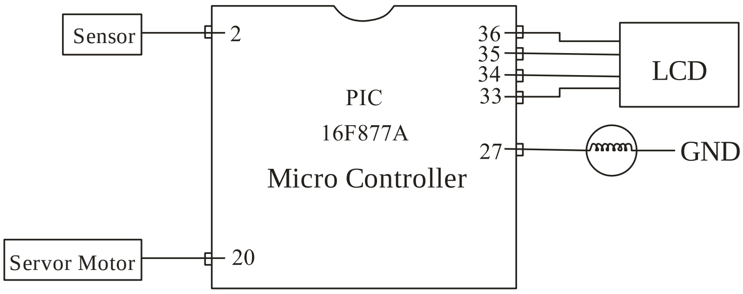

Fig.6: Functional schematic diagram of the project.

5.2 Program Description

This program implements an automatic swinging door opening system that can be installed at any building entrance. By default; the door is always closed, the LED turned off and the LCD displaying "door closed". However, if the system senses someone approaching the entrance, the door opens automatically, the LED turns off and the LCD displays "door opened". After the person must have passed, the system reverts to its default operation mode. For safety, the door cannot close as long as someone is within the preset opening range.

5.2.1 Flowcode Program for the Smart Door

- ◊ **Step 1:**
  - Launch **flowcode** and start a new program
- ◊ **Step 2:**
  - Add and **LCD** component.
  - Connect the **LCD** to **PORTB.**
- ◊ **Step 3:**
  - Initialize the LCD component for display.
- ◊ **Step 4:**
  - Add a Servo motor component.
- ◊ **Step 5:**
  - Select **one** servo on channel **zero**.
- ◊ **Step 6:**
  - Connect the servo motor to pin **D1** of the microcontroller.
- ◊ **Step 7:**
  - Servo needs to be activated before use and deactivated when not in use.
  - Therefore enable the servo motor on channel **zero.**
- ◊ **Step 8:**
  - Set the servo motor shaft to an initial position 0^o
- ◊ **Step 9:**
  - Add a **Loop** to make the program run repeatedly.
- ◊ **Step 10:**
  - Set the **LCD cursor** to column **two** and row **zero**
- ◊ **Step 11:**

- Any time the door is closed indicate on the LCD.
- Print the string constant "**door closed"** to the LCD.

◊ **Step 12:**
- Add an **ADC** device to represent the **IR** sensor (for simulation).
- **Note:** (flowcode does not have an **IR proximity** sensor in it components library).
- **IR** sensors are analogue devices.
- Connect the **ADC** device to pin **A0** (analogue channel zero) of the microcontroller.

◊ **Step 13:**
- Read as an **integer** variable at the sensor signal terminal.

◊ **Step 14:**
- Declare a variable of **integer** type and name it **ir sensor value** to store the value read from sensor signal terminal.

◊ **Step 15:**
- Return the variable **ir sensor value** to the main program.

◊ **Step 16:**
- Check if the variable **ir sensor value** is greater than 200
- If someone is within the door-define opening range, the value at the sensor signal terminal will be greater than 200ADC value.

◊ **Step 17:**
- Define a macro to handle the opening and closing of the door.
- Name the **macro** as **door opened.**

◊ **Step 18:**
- Edit the new macro to the following:
- Command the door to open.
- Command the **LCD** to display **door opened.**
- Command the **LED** attached to pin **D4** to turn **on**.

◊ **Step 19:**
- Set the servo motor to position 130^{o} at once. (i.e. swing the door open)

◊ **Step 20:**
- Add an **LED** and connect the **LED** to pin **D4** to indicate when the door opens.

◊ **Step 21:**
- Turn **on** the **LED** attached to pin D4 any time the door is opened.

◊ **Step 22:**
- Delete the message "**door closed"** on the **LCD** and prepare it to display "**door opened".**

◊ **Step 23:**
- Set the **LCD** display cursor again to **column two** and **row zero.**

◊ **Step 24:**
- Print **"door opened"** on the LCD.

◊ **Step 25:**
- Keep the door opened for **4 seconds.**

◊ **Step 26:**
- Gradually close the door when no one is within the door-defined opening range.

◊ **Step 27:**
- The door is now closed, so turn **off** the **LED**.

◊ **Step 28:**
- Check continuously for when the sensor's signal terminal will change from **High** to **Low.**

◊ **Step 29:**
- Check every **200 millisecond** to keep track of step **24.**

◊ **Step 30:**
- Simulate the program by clicking on the **Run** button and download it to the micro phase plus development board.

5.3 Testing

The physical realization of this work was carried out to achieve the conceived idea. Here the work will be seen not just on paper but also as a finished hardware. After carrying out all the paper design and analysis, the hardware was implemented and tested to ensure it's working ability and was finally constructed to meet desired specification

VI. Result And Discussion

The design result was achieved as the LCD displayed 'Door Closed' and the LED was OFF which is the design default condition when the door is closed, and also displayed 'Door Opened' and the LED was ON when the door was opened. Two major problems were encountered which has to do with how long the door will stay open or closed, but the problem was solved by introducing a time delay of 4seconds for the door to remain open, the other problem is that the door closes after the preset time of 4seconds irrespective of whether someone was within the door-defined opening range, this was overcome with the introduction of 200milliseconds check to verify if someone is within the door-defined opening range.

VII. Conclusion

A step-by-step approach in designing a smart door has shown that the system performance is quite reliable and accurate. If the system is installed in public buildings, offices and shopping malls will eliminate the need for hiring a doorman to usher in visitors. The highlight of the advantage of using mechatronics principles as highlighted in this paper includes;

- Flexibility in manufacturing, and optimum utilization of raw materials.
- High speed of operation
- Increased quality control (less wastage)
- Improved accuracy
- Safety assurance and so on.

References

[1] Isremann, R, Mechatronic Systems Fundamentals. Spring-verlag London Ltd. 2005
[2] Craig, J, Introduction to Robotics: Mechanics and Control, 2nd edn. (Reading: Addison-wesley, 1989)
[3] AIM IEEE/ASME Conference on Advanced Intelligent Mechatronics. Atlanta (1999), Como (2001), Kobe(2003)
[4.] SENSORS- The Journal of Applied Sensing Technology, Advanstar Communications Inc.
[5] Dorf, R.C., Bishop, R.H, Modern control systems, 9th edn. (Englewood Cliffs, NJ: Prentice-Hall,).
[6.] PIC microcontroller Data Sheet, Microchip.
[7.] http://www.mechatronicconcept/turning.com Date accessed: 09/08/2014
[8.] http://www.mechatronics/edu.com Date accessed: 09/08/2014
[9.] http://www.hobbyprojects.com/microcontroller/pic16f877a Date accessed: 10/02/2014