

# An Autonomous Robot for Controlling Fire in a Structured Environment

M.L.E. Fernando<sup>1</sup>, H. Ekanayake<sup>2</sup> and D.N. Ranasinghe<sup>3</sup>

University of Colombo School of Computing  
Colombo, Sri Lanka

<sup>1</sup>lasithf@zone24x7.com , {<sup>2</sup>hbe, <sup>3</sup>dnr}@ucsc.cmb.ac.lk

## Abstract

*This paper presents a practical application of autonomous robots - a robot that has the ability to locate a fire in a structured environment and extinguish it. The robot has built around the hybrid paradigm, where both the reactive and deliberative controllers are used. The robot's ecological niche is on top of an electrically wired grid of arena. The robot gets the fire alerts through this structured network and navigates to the fire point through its own decisions. In addition, a person sitting at a central monitoring station sees the entire process of fire fighting effort from a user interface updated by both the robot and the wired grid of arena. Other major concerns were preserving the efficiency and effectiveness at a higher degree.*

Keywords: Autonomous Robots, Fire Fighting, Hybrid Paradigm

## 1. Introduction

Fire is useful to man only when there is a good control of it. Therefore, it is important to stay away when there is an emergency fire. To prevent or minimize the risks of fire there should be sufficient precautionary measures, such as, immediate warning and notifying the threat. However, this needs allocating of huge money even though some measures are not practical enough. One solution may be to allocate someone to keep on alert to monitor emergency fire situations and take necessary steps in case of fire.

Usually, except in some cases, humans are sensitive to their own safety and as a result they would not come forward to take any risk to do any activity which can be harmful in emergency situations without a proper understanding or an advance planning. Further, in a case of an emergency fire, normally people get scared and they won't be able to deliberately respond to the situation with planned activities as the situation get changed dynamically.

Machineries come to play a major role in such dangerous and harmful situations, because they don't

have an immune system like human beings have. Therefore, machines can actively respond to emergency situations without caring much about the danger!

In the case of a robot, it is general that robots will do the activities that are dangerous to the people and as robots are machines, people do not watch over too much on them. They do not have a big overhead of work compared with a human, so the only difference is a robot has a special kind of work to perform.

Therefore a possibility of introducing a robot to monitor fire and take necessary actions when there is a critical situation it will be of great importance. In addition, people can feel free with a fire phobia as well.

This paper presents a model of an autonomous robot, a Robot that can handle things in its own without support of any other party and will start duties in an emergency fire situation.

So when an emergency fire exists in the structured arena, it will be tracked by the thermal sensor network installed in the arena itself and there after the coordinates of the fire point will be passed to the Personal Computer (PC) immediately.

PC will receive fire point coordinates and will send signals of fire point location to robot and also will send it to the Central Monitoring System.

After Robot receives the fire point signal it will slightly move ahead and get its location coordinates through the PC where PC receives robot location through the location sensor network installed in the arena. This signal will also be passed to the Central Monitoring System.

Now the Robot has received its location and then decides to move in the way to minimize the coordinate difference between fire point and the current location of the Robot. When Robot moves along the arena, it will update its current location and this continuation of communication process enables him to decide whether to move ahead, turn to the left/right...etc.

Finally when reaches to the fire location, Robot will focus that fire point through Robot's Eye and then a fire controlling signal will be generated.

The Central Monitoring System is just a monitoring system that users can view the arena plan and the robot's location dynamically. And when encountered a fire place the emergency buzzer will alarm and the safety paths can be illustrated easily to show exit points from the arena. Finally it will confirm that the Fire is controlled successfully.

Furthermore this paper furnishes the possible solutions for the robot's subsystems and how those subsystems act together in such emergency fire situations.

## 2. Hybrid Paradigm of the Robot

Robot paradigms are basically defined in terms of Sense, Plan and Act primitives. Robots are broadly categorized into 3 paradigms [BookAIRobotics] based on these primitives. They are,

- Hierarchical Paradigm
- Reactive Paradigm
- Hybrid Paradigm

Hybrid Paradigm is based on both the hierarchical and reactive paradigms, and came forward in 1990's and still it is the widely accepted model. In this hybrid paradigm, a robot first plans how to best decompose a task in to sub tasks and what are the suitable behaviors to accomplish each subtask. Then the behaviors start executing as for each the Reactive Paradigm. This is a Plan-Sense-Act type of organization. Here planning done in a separate phase and Sense and Act done together in the next phase. The figure 1 shows the Hybrid Paradigm scenario.

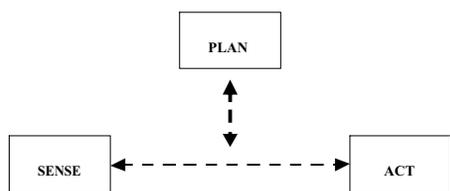


Figure 1: Hybrid Paradigm

Planning phase in the hybrid paradigm receives sensed data and constructs a task oriented global world model. Then sensed data routed to each behavior and again fed in to the planner. After getting to the behavior, it precedes a Sense-Act reactive paradigm. The table 1

shows the inputs and outputs of robot primitives in the hybrid paradigm.

<b>ROBOT PRIMITIVE</b>	<b>INPUT</b>	<b>OUTPUT</b>
PLAN	Sensed Information	Directives
SENSE-ACT (Directives to Behavior)	Sensor data	Actuator commands

Table 1: Robot primitives and inputs and outputs in the Hybrid Paradigm

According to the hybrid paradigm in machine automation, our robot has three subsystems:

- Sensory Subsystem
- Decision Making Subsystem
- Reaction Subsystem

## 3. Sensory Subsystem

Sensory subsystem consists of two main sensors, fire and light. Light sensor is for location sensing and turning point identification.

**Fire Sensing** can be done in two methods. Smoke sensing and thermal sensing. In the smoke sensing technique, disturbance to the light of the environment is considered. So here the smoke will reduced the intensity of light in the environment and that can be sense through a light intensity sensing. Some of sensors made based on Light Dependent Resistors (LDR) and it will indicate the falling of the pre-defined intensity below a defined critical value [WebElecClub].

Figure 2 shows the abstract scheme of the smoke sensing technique.

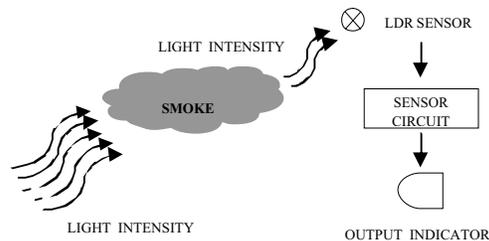


Figure 2: Abstract scheme of the smoke sensing technique.

The other method is the thermal sensing of the environment. Environment temperature will arise with a fire and this used to sense the fire by using a Thermistors. Thermistor is a temperature sensor that can sense temperature in a given range. Thermistor will indicate the arising of pre-defined temperature above a defined critical value [WebElecClub].

Figure 3 shows the abstract scheme of the thermal sensing technique.

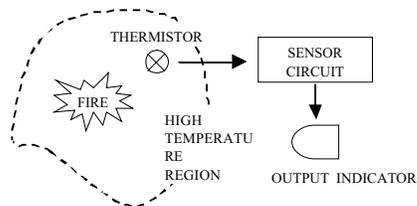


Figure 3: Abstract scheme of the thermal sensing technique.

Structure of the thermal sensing process of the robot is shown in figure 4. Temperature is rising up with a fire near the point B. Hence the thermal sensor will indicate that there is a fire near point B. Meanwhile a message indicating that there is a fire in point B will be passed to the PC immediately.

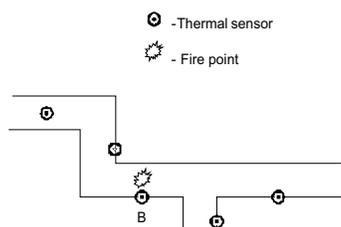


Figure 4: Structure of the Thermal sensing process

**Light Sensing** is more useful to find obstacles through the reflection. There are several types of light sensors such as Infra-red, LDR ...etc.

Infrared sensors are sensitive to the Infrared electromagnetic wave length. Therefore they will not respond to the pure visible light but can interfere with the visible light because these two regions are laid out connectively in the Electro Magnetic Spectrum.

The Infrared sensing technique consists of pair of an IR transmitter and IR receiver. In this robot IR sensors are used to detect turning points in the structured arena.

Figure 5 shows the abstract scheme of the Infrared sensing technique.

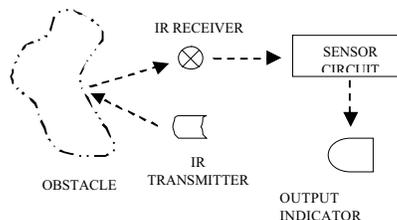


Figure 5: Abstract scheme of the Infrared sensing technique.

To identify the turning points, dead ends, such things IR transceivers are mounted as given in the figure 6.

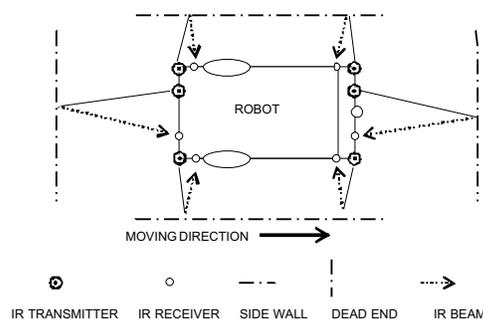


Figure 6. Mounting IR transceivers

The implementation for the Infra-Red module mounted in the robot is shown as a circle in the figure 7.



Figure 7: Infra-Red Sensor Module

LDR sensors are sensitive to the visible light. Absence and presence of falling light to the LDR surface will indicate whether there is an obstacle in front of the LDR.

The LDR sensing technique consists of pair of a light source pointed to the LDR and a LDR receiver. Figure 8 shows the abstract scheme of the LDR sensing technique.

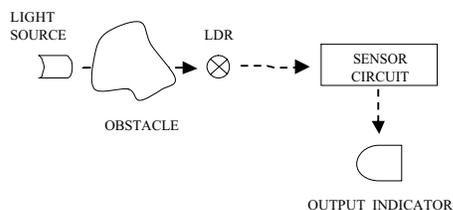


Figure 8: Abstract scheme of the LDR sensing technique.

Current location of the robot is most wanted parameter to step to the fire point. Robot can be placed anywhere in the arena. But in case of emergency robot will step along the corridor of its own place to find a location sensing point. Location sensing points are along the way of each and every corridor. Hence robot can find immediately his position after passing the first location sensor.

The structure of the location sensing of the robot is given in figure 9.

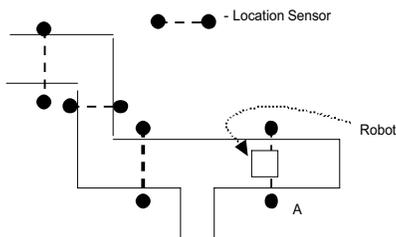


Figure 9: Structure of the Location sensing process

The robot drives through the point A light beam and now the output indicates the light beam is disturbed by the robot and hence the position can be evaluated by identifying the location sensor.

The implementation for the Thermistor and LDR module mounted in the arena is shown in figure 10.

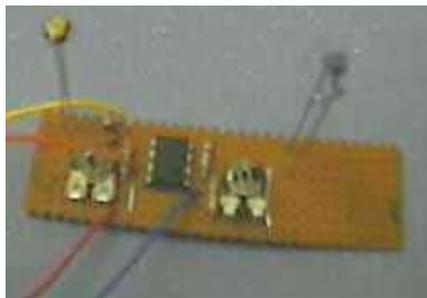


Figure 10: Thermistor + LDR Module

#### 4. Decision Making Subsystem

Decision making is very much interesting when talking about the Hybrid paradigm for a robot. That part is most important due to heavy logic combinations are willing to work in a way to perform better decisions. Here going to discuss about the way of doing Hybrid decision making and use of microcontroller, Interrupts as well as other hardware controllers.

Here used a PIC 16F877A microcontroller as the heart of the robot controller and this microcontroller have a RS-232 interface to communicate with serial data through RS-232 interface in the PC [BookPIC16F877A].

The figure 11 illustrates Microcontroller Interfacing Model in the robot.

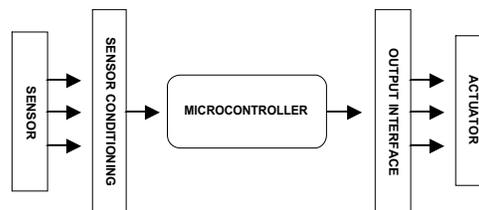


Figure 11: Microcontroller Interfacing Model

There are several sensors interacting with the microcontroller. So, how they are going to work with each other? There should be a proper controlling mechanism to do this. In the other hand, some actuators are connected to the microcontroller output.

Output from the thermal sensor network is the starting point of the fire controlling process. This fire point is taken to the inside of the microcontroller through an input port of the microcontroller. Now the initial point is there and has to get the location of the robot. Location sensor network will deliver the current location of the

robot through an input port of the microcontroller. After getting these two points at the beginning, a calculation will be done on behalf of the destination point. Thereafter robot get to know about the coordinate difference separately in x and y axis of the destination fire point. Now the robot is able to move according to the coordinates in the sense, to minimize both coordinate differences either in x or y axis at a time.

The robot starts to move along the building and continuously listen to the sensors to identify the turning points, dead ends as well as the current location. After reaching to a location, the microcontroller re-calculates the difference of coordinates and then decides where to go next. Finally after reaching to the destination microcontroller will turn on the robot eye which focus the actual fire point and then turn on the fire controlling signal to control the fire.

#### 4.1. RS 232 Communication

USART stands for Universal Synchronous Asynchronous Receiver Transmitter in the PIC 16F877A microcontroller and it can be configured as a Full-Duplex asynchronous system that can communicate with peripheral devices such as PC. The USART communication module needs a MAX232 Multi channel Driver IC as a voltage shifter from TTL/CMOS to RS232 [WebMax232].

This module is used to communicate between robot and the PC through RS 232 interface. The implementation of the microcontroller with RS232 compatible interface is shown in the figure 12.



Figure 12: RS 232 Compatible PIC 16F877A Demonstration Board

#### 4.2. Using Interrupts

There are two ways that a robot receives sensor inputs: Interrupts and Overheads. Both receive sensed signals from the controlling unit in different ways. Interrupts works in a way that when a signal comes to the microcontroller, it will identify that is an incoming signal from a sensor and immediately notify to the microcontroller about that received signal. While an interrupt, the microcontroller will not listen to the inputs. Interrupt handling process of microcontroller will do that part. Hence, there is no any waiting situation in the process until another input signal receives from the sensors. So the Robot interrupts are used to handle input signals to the microcontroller.

Overheads are also a way of receiving sensor outputs. But in overheads, there should be a listener to the input signals to the microcontroller and that listener should listen frequently to the inputs and that part should be done by the microcontroller software program. This frustrates the performance of the microcontroller program. And due to that, overheads are ignored and interrupts are used.

#### 4.3. Interrupt Handling Module

Interrupts from the sensors mounted on the robot body will catch from the Interrupt Handling Module and then search for which sensor outputs an interrupt. So this unit is basically consisting with some logic gate network. The schematic diagram is shown in figure 13. Here S1...S2 are sensor outputs.

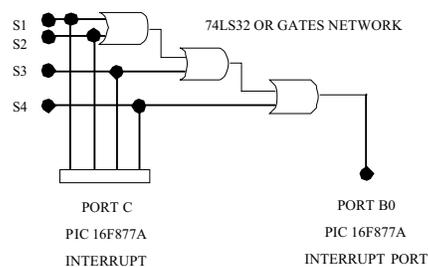


Figure 13: Interrupt Handling Module

This interrupt handling module implementation is shown in the figure 14.

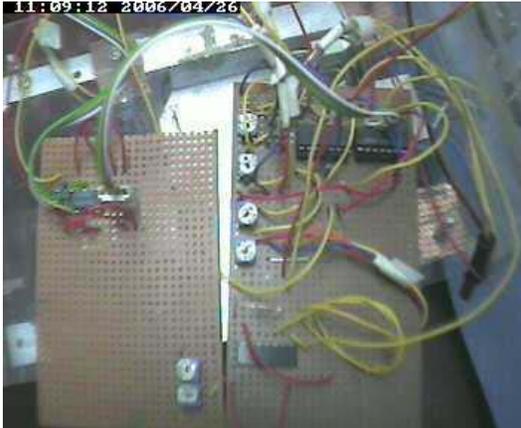


Figure 14: Interrupt Handling Module

## 5. Reaction Subsystem

After talking about the Hybrid sensing, decision making process the destination is the reaction part. Although sensing and decision making is completed, most interesting things happen here because probably the visible part is the reaction of robot and now the most interesting part is going to discuss.

This part covers more of the kinematics activities and due to that the base of this technique is a stepper motor. So following will discuss about the movements of the robot and how these are going to be work in real.

### 5.1. Use of Stepper Motors

Since there are two methods to drive a stepper motor; due to torque is proportional to the weight plus friction of the total system, it is most require provide a high torque by the motors. So the figure 15 Single stepping Bipolar Stepper Motor shows the two methods of drive stepper motors and the first one is control sequences for single stepping a bipolar stepper motor. Here positive voltages are denoted by blue bars, and negative by red bars and 2 separate coils denoted by 1a,1b and 2a,2b.

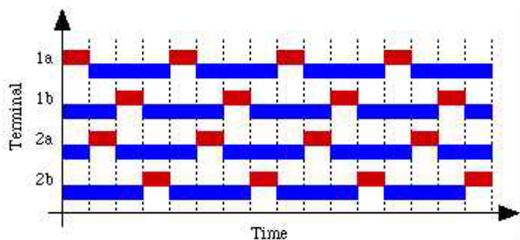


Figure 15: Single stepping Bipolar Stepper Motor

Regarding the above sequence, it only powers one winding at a time, so it provides lower torque at lower input power.

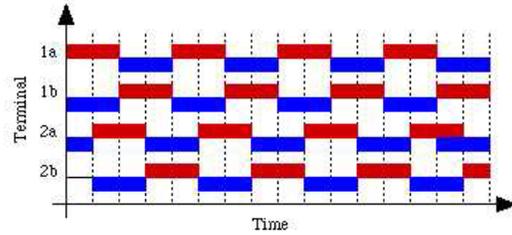


Figure 16 Dual stepping Bipolar Stepper Motor

According to the figure 16: Dual stepping Bipolar Stepper Motor, it powers two windings at a time and produces a torque about 40% greater than for the previous sequence while using twice as much power. [WebStepMtr]

### 5.2. Forward Movement

Forward motion is achieved by controlling two stepper motors in two opposite directions. That means to drive robot forward, both motors should rotate in opposite ways, clockwise and anticlockwise.

Here the most important thing is the delay between two steps. This delay is inversely proportional to the speed of the motor. The figure 17 shows how to control the movement of the stepper motor.

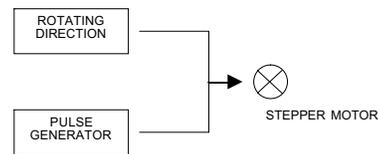


Figure 17: Stepper Motor Controlling Block Diagram

Here the PULSE GENERATOR is used to deliver the stream of pulses need to notify the delay and as well as the supply voltage to the stepper motor. So when consider the delay, here *Pulse Width Modulation* (PWM) is the way to controlling motors. This controlling plan is as in the figure 18.

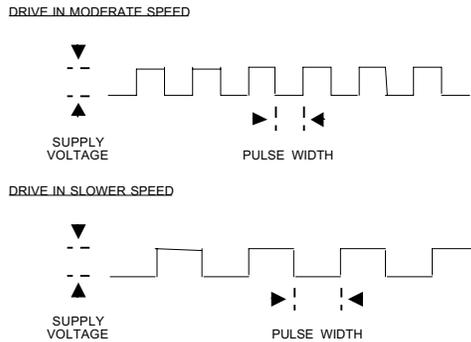


Figure 18: Pulse Width Modulation (PWM) Controlling process

The ROTATING DIRECTION is used to rotate the stepper motor in clockwise or anticlockwise. Here this part is done by sending a continuous pulse to the motor controlling board. Figure 19 illustrating the direction controlling process.

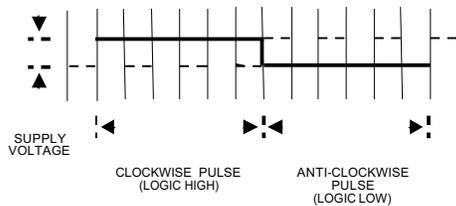


Figure 19: Motor Direction Controlling Process

### 5.3. Breaking

Breaking is done by controlling the above mentioned delay with a long time of period. The stepper motor wants a rapid varying pulse stream with limitations to the boundaries for the delay. So then, if the delay is below value to the lower boundary of the stepper motor required, it will not rotate further until the required PWM appears. The figure 20 shows how to break the stepper motor while rotating.

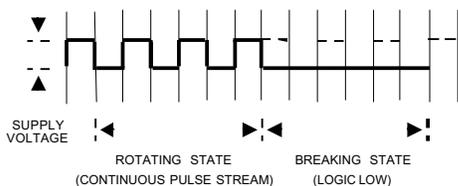


Figure 20: Breaking the Stepper Motor While Rotating

### 5.4. Turning Left/Right

As usually above states, turning process is deal with both stepper motors but there is a slight change in actions doing by each stepper motor. In this scenario, turning process is done by stopping one motor and rotates the other motor in a continuous manner until the turning is completed. This can be illustrated by a diagram much clearly and the figure 21 shows this scenario for the right side turning.

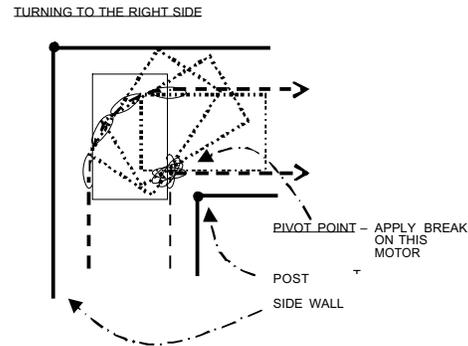


Figure 21: Robot Turning Scenario

### 5.5. Backtracking

Backtracking is used as a solution to the dead end situations. When there is a dead end in front of the robot, it will catch by the sensors mounted in the front of the robot and immediately pass the message to the microcontroller to do the backtracking.

Here LDR is used as the front sensor and this time microcontroller interest on the response to the light intensity to the face of the LDR. Intensity is because when the robot comes to a dead end, the intensity falling to the LDR is getting gradually a high value. So the figure 22 shows this circumstance.

$I_1$  and  $I_2$  represent the different intensities tracked on the way to the dead end. So in here  $I_1 < I_2$ .

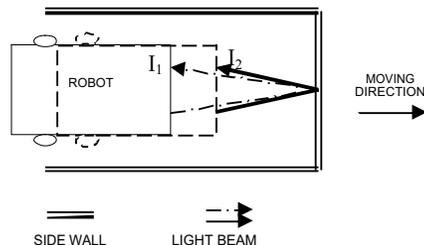


Figure 22: Dead End Tracking System

## 6. Movement Control Algorithm

Movement control algorithm is the most valuable point as it directly maps with the performance of the robot. Movement control algorithm can be divided into several parts. Moving along a corridor, decide and turning in a turning point, backtracking due to dead ends are the situations which the robot will have to face.

In the Movement control algorithm of the Robot is designed to get the maximum performance as well as quick response. To get the target result when moving through a corridor, acceleration of the robot is highly considered.

In the Robot acceleration should high when there are no any obstacles and should low when there are obstacles. Robot has an ability to drive safe in a maximum acceleration in possible places. When entering to a corridor robot will start with an average acceleration while it senses the LDR sensor in that corridor. After that robot will drive in a constant speed until there is a turning point or dead end.

At the beginning robot will receive the fire point through the PC's serial port and then robot moves slight ahead and get the current location. Then calculate the difference between fire point and the current location. This calculation is always done when the current location is changed and turning direction is defined according to that. Hence it will gain the performance of the robot.

In a turning point situation robot will turn to the way that he can minimize the coordinate difference. Priority is for x-axis coordinate when both axis are presence. If there is a possibility it will turn to the x-direction firstly to minimize the coordinate difference. Then look for y-axis. Turnings are  $90^0$  in angle and circular turnings rather than pivot turnings.

In backtracking robot will follow the array that has the coordinates of the points that robot passes after the last junction. According to that array values robot will backtrack until the junction is found. But here no rotation of the robot will happen. Only robot will reverse his moving direction with out reverse the robot body.

## 7. Arena Structure

An Arena is a wooden platform painted in black color that comprises of 6 rows and 6 columns in a matrix arrangement. Each lattice point in the arena matrix shall be placed a white plastic post and a white plastic wall is attached between two posts. The size of the post is 1.2cm (length) x 1.2cm (width) x 5.0cm (height) and the size of the wall is 16.8cm (length) x 1.2cm (width) x 5.0cm (height).

This Robot is running on this type of arena and that arena is pre-defined its walls and corridors to a given 2 dimensional plan. In this demonstration board location

sensors and fire sensors are mounted to a given structure. The figure 23 shows the arena structure.

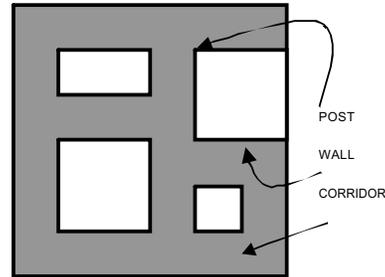


Figure 23: The Arena Structure

### 7.1. Use of Matrix Structure

Matrix structure is used to cabling the sensor network in the arena. And also this structure is able to figure out a static coordinate system of the arena. Hence sensors can update that coordinate system according to the robot's locations as well as the fire points. The figure 24 shows the matrix structure for the Robot. In a more practical way when talk about a building floor, this sensor network is an underground or overhead layer as other air conditioning, electricity, etc channels. So the thing is there should be a proper coordinate system defined to the building floor.

It is easy to add or remove any coordinate from the system with out doing much effort. If the floor is willing to expand, then only want to use an additional row and a column for the sensing network.

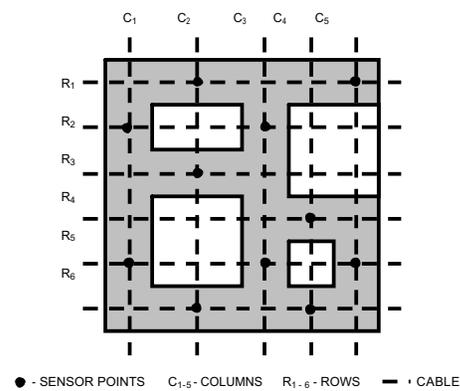


Figure 24: The Matrix Structure And Cabling Diagram

### 7.2. Identification of Coordinates

This is a most important part regarding the robot since the whole system depends on the arena location

points. That is to drive the robot itself he will listen to the arena coordinates through the PC's serial port. So now the PC port accessing part and data transmission media comes to play a major roll.

Basically in this robot parallel and serial communications are used to interface the data transmission between arena and PC as well as PC and robot respectively.

In the first phase of the data manipulation between arena and PC, Central Monitor will listen to the change of any column or row in the arena. This is done by connecting each column and each row to the Data and Control registers in the PC's parallel port. And inside the Central Monitor it will check the port values in 0X378 and 0x379 registers that is in Data and Control registers.

After that when the value is changed, the coordinate values for the both row and column will puts to the PC's serial port. And then robot will receive the data through his serial interface. Now robot will begin the calculations to find out where to turn.

## 8. Central Monitoring System:

Some screen shots are illustrated for the Central Monitoring System of the Fire Fighting Robot.

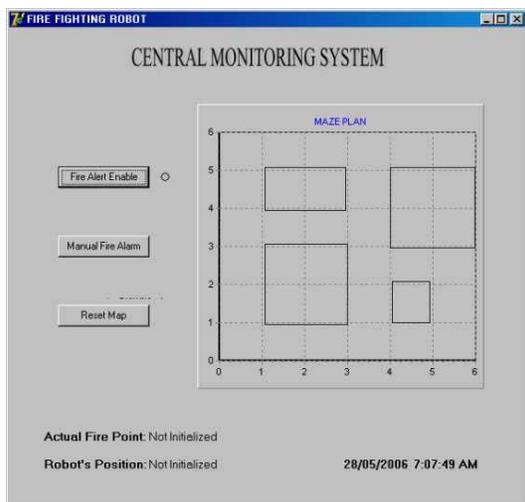


Figure 25 Initial Screen:  
Central Monitoring System

By clicking the “Fire Alert Enable” button, the system will track the Actual Fire Point and Robot's Location. This is the initial start of system as shown in figure 26.

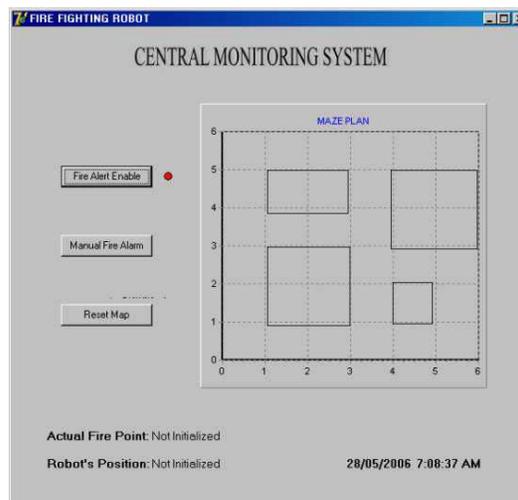


Figure 26: Enabling Fire Alert:  
Central Monitoring System

After clicking the “Fire Alert Enable” button, the system waits for a fire alert. If the system tracks an Actual Fire Point then it will be displayed in the monitoring system with Robot's current location as Figure 27.

The black spot is the Robot's position in the maze plan and red spot is the actual fire point. Now the buzzer will sound as an emergency state declared by the system.

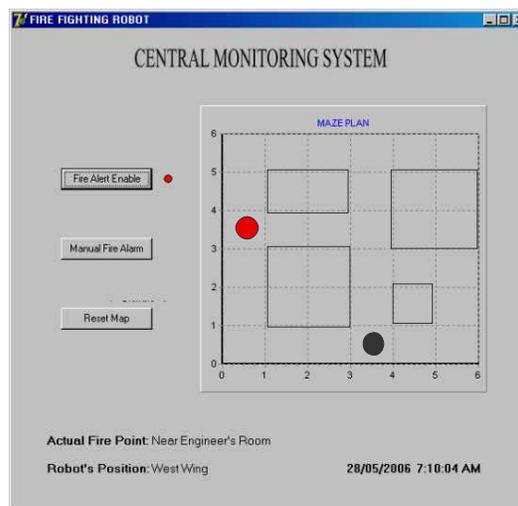


Figure 27 Tracking Robot's Movement and  
Actual Fire Point  
When the Fire Point is detected and Robot reaches to that point, the system will notify to the user as “Fire Successfully Controlled !” message.

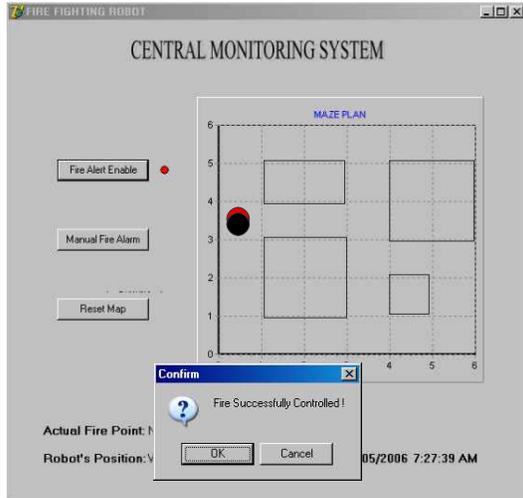


Figure 28: Fire Successfully Controlled!  
Message

## 9. Conclusions

This is basically a model of an autonomous robot for controlling fire in a structured environment. And this robot is able to track fire point through this structured network and will move along the arena by its own decisions according to the information given by its own sensors. Also this features to a monitoring personal who is able to monitor fire in this building through a user interface of the fire controlling process of Central Monitoring System. Following are some of ideas and views towards further development of this project.

Robots only sense limited environment like coordinates, dead ends, fire points and turning points. If the sensor network can be improved to cover a wide range of sensing with high resolution might result the effectiveness of the performance.

The matrix structure of the maze can be replace with the Radio Frequency Identification (RFID) module and then the location can be detected easily as well as extensions can be done with minimum effort. This RFID module consists with a RF receiver and an Identification card. The Identification card can configure the RFID receiver and then receiver can identify the ID through RF and then pass the corresponding data to the system.

User interfaces are able to visualize the map of the building and important location data channeled to the Central server frequently. So a mobile application can be programmed by using the coordinates receiving to the Central Monitoring System to generate warning signal using Short Message Service (SMS) to notify the staff

inside the building of the emergency situation. Further they can view safe exits from the map implemented in the mobile application.

## Acknowledgments

The authors would like to convey their sincere gratitude to Dr. Kasun De Zoysa for his valuable feedback, the researchers at the Robotics and Ubiquitous Systems laboratory and Wireless Ad-hoc and Sensor Networks laboratory - UCSC and Mr. Keerthi S. Goonatillake of the UCSC for providing necessary facilities and great support.

## References

[WebElecClub] The Electronics Club – J. HEWES 2005, <http://www.kpsec.freeuk.com/components/other.htm>

[WebStepMtr] <http://www.cs.uiowa.edu/~jones/step/types.html>

[WebMax232] <http://pdfserv.maxim-ic.com/en/ds/MAX220-MAX249.pdf>

[BookAIRobotics] MURPHY, R.R., “Introduction to AI Robotics”, PRENTICE HALL, 2000

[BookPIC16F877A] “PIC16F877A CMOS Flash Microcontroller Data Sheet”, MICROCHIP TECHNOLOGY INC, ©2001