

# Multi Robot Exploration System For Industrial Environment Parameter Analysing Using Labview

Dinu Mathew  
ME(Applied Electronics)  
Sathyabama University  
Chennai,India

**Abstract—** The Multi Robot System explore and collect various vital parameters from a sensitive or critically controlled industrial environment. The vital parameters taken from the critical environment in this application are temperature and flammable gas presence. The data acquired through robots are send to a central station where further processing and analyzing of acquired data is done. The data processing and analyzing is performed by a System design application software called LabVIEW. Virtual Representation of the industrial environment parameters can be achieved using LabVIEW. Data communication takes place through Zigbee wireless communication. Necessary control signals are also generated and sent back to the robots using LabVIEW. These robots can sense the obstructions in the path and avoid them to facilitate smooth voyage. A microcontroller, PIC16F877A is used to control the robots. IEEE 802.15.4 wireless transceiver is used for data communication and the hardware section interfacing with the LabVIEW is done through Virtual Instrumentation System Architecture(VISA).

**Keywords-** Multi Robot; Critically controlled; LabVIEW; Virtual Representation; Zigbee ;VISA

## I. INTRODUCTION

The research area of Multi robot systems consists of the study of algorithms for perception, cognition and actuation of systems composed of multiple robots. Until the early 1990s, mobile robotics focused mainly on the challenges of developing a single robot capable of robustly handling the physical environment. Industrial applications included multi robots, but in a preprogrammed sense of replication. In the early to mid -1990s, there was a bloom of multi robot systems inspired by the overall intelligent behaviour of large biological insect communities such as ants. Furthermore, the field of distributed artificial intelligence was also in great development. The multi-robot system can be defined as a robot group in which the robots can communicate with others and can cooperatively implement different kinds of tasks. Multi robots may achieve both more robust and more effective behaviour by accomplishing coordinated tasks that are not possible for single robots. The field of multi robot systems is now a large well established area of research that reached a high level of

maturity. The research includes a variety of engineering and scientific aspects of robotics.

Now a days the major safety problem faced by the Petrochemical or Nuclear Power plant Industry is that they lack precise and accurate sensing and monitoring of industrial environment parameters from any part of the industry. At present, in industries sensing and monitoring systems are present, but they have a lot of disadvantages or drawbacks which are of major concern. Sensor modules which are used for sensing purposes are fixed at a particular part of an industry. These sensing systems cover only a limited area of the industry. Thus in a critically controlled industrial environment any small variation in the concentration of the critical gases from any part of the industry has to be monitored and sensed. This can't be achieved with the sensing systems present in the industries right now. Another drawback is that separate monitoring and control units are required for each of the sensing systems which increases the complexity of the entire system. Monitoring and virtual representation of the entire industrial environment parameters cannot be done simultaneously at the central station.

In this paper a Multi Robot Exploration System(MRES) is designed that overcomes the above limitations and problems faced by the existing system and provides a better efficiency compared to the present system in the industries. Multi robots explore and monitor the industrial environment continuously and collect the critically controlled environmental parameters frequently whether there is any leakage of any flammable gas presence or any temperature variation from the normal level in the areas where human interference is required. Multi robots detect the obstructions using the obstruction sensors and move around the entire industry. Throughout its voyage, Robo cars sense the critical parameters using the specific sensors and the acquired data is sent back to the central station through wireless medium (Zigbee Technology). The data received at the central station is processed and analysed using LabVIEW technology. Use of LabVIEW helps us to virtually represent the entire industrial environment parameters using Control Knobs and Indicators. The Multi Robot System reduces the manpower required for monitoring and data acquisition process. It also reduces the risk of human beings being adversely affected by the imbalance in the critical environmental parameters like leakage of gases or temperature rise. Earlier Collision avoidance was a big issue in multi robot systems, many algorithms were proposed but a clear

solution to the coordination problem was not yet found out. With the use of TSOP 1738 Obstruction Sensor and programming of the microcontroller for obstacle avoidance, we can avoid this problem. User at the central station has full control over the On/Off states of the robots, thus reducing unnecessary power consumption in the idle state.

## II. DESIGN OVERVIEW

### A. Temperature Sensitivity Measurement

LM35 is used for sensing the temperature from a critically controlled environment. LM35 series are precision integrated circuit temperature sensors, whose output voltage is linearly proportional to the Celsius Temperature. The LM35 does not require any external calibration or trimming to provide typical accuracies of  $\pm 1/4^\circ\text{C}$  at room temperature and a  $\pm 3/4^\circ\text{C}$  over a full  $-55^\circ\text{C}$  to  $150^\circ\text{C}$  temperature range.

LM35 has 3 pins - Vcc, Gnd and Analog Out. For every  $1^\circ\text{C}$  rise, a 10mV analog voltage will be generated by the sensor. The analog voltage will be given as input to the PortA(A0) of the PIC Microcontroller. The ADC inbuilt in the microcontroller does the conversion of analog voltage to digital voltage before transmission. ADC used is a 8channel 10 bit one with Quantization Level  $2^n - 1$ .

$$\text{Resolution/Step Size} = \frac{V_{\text{Ref}}(1\text{V})}{2^n - 1} \quad (1)$$

As value of n increases, Resolution also increases.

For  $n=4$ ; Stepsize=67mV(Sensitivity) ie. ADC senses every 67mV analog voltage and converts into digital value. For  $1^\circ\text{C}$  change 10mV will be generated, similarly for  $6.7^\circ\text{C}$  change a 67mV analog voltage will be generated at the output of the LM35 sensor. To improve the detection capability of ADC we can increase the no of bits of ADC to 10.

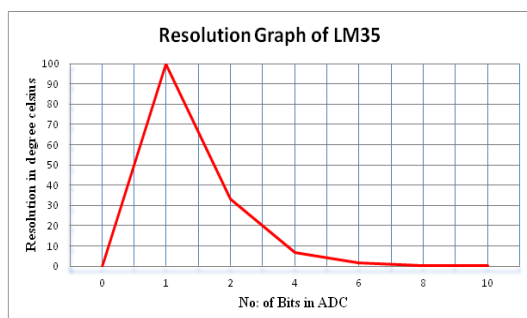


Figure 1. Resolution Graph of LM35

For  $n=10$ ; Stepsize=1mV(Sensitivity) ie. ADC senses every 1mV analog voltage and converts into digital value. For 10mV,  $1^\circ\text{C}$  change will be detected by ADC. similarly for 1mV,  $0.1^\circ\text{C}$  change will be detected by ADC. Fig 1 shows the variation in the resolution values of LM35 by varying the no of bits of ADC.

### B. Alcohol Sensor Calibration

The Alcohol Sensor used is MQ 303 which is suitable for detecting alcohol concentration in the atmosphere. It has high sensitivity and fast response time. The sensor has a detection range of about .05mg/L-10mg/L. The equation which converts analog output voltage to the mg/L gas concentration is

$$\text{Gas concentration} = \text{Analog Voltage(mV)} \times 0.002 \quad (2)$$

The gas concentration in an area or a location is usually specified in ppm(parts per million)

$$1\text{m}^3 = 1000\text{L}$$

$$\text{Ppm} = \frac{\text{mg/L} \times 24450}{\text{molecular weight of gas}} \quad (3)$$

From the ppm we can calculate the range of detection (how many square feet of area covered).

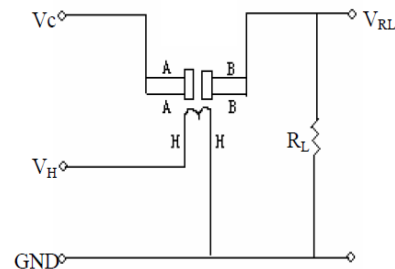


Figure 2. Test circuit of MQ 303

The above Figure is a basic test circuit of the sensor. The sensor needs to be put 2 voltage, heater voltage ( $V_H$ ) and test voltage ( $V_C$ ).  $V_H$  is used to supply certified working temperature to the sensor, while  $V_C$  is used to detect voltage ( $V_{RL}$ ) on load resistance ( $R_L$ ) which is in series with the sensor. The sensor has light polarity,  $V_C$  needs DC power.  $V_C$  and  $V_H$  could use the same power circuit with precondition to assure performance of the sensor. In order to make the

sensor with better performance, suitable RL value is needed

Sensitivity of the alcohol Sensor is specified in terms of its Sensor Resistance(Rs).

$$Rs = \frac{Vc}{(VRL-1)} \times RL \tag{4}$$

Where VRL denotes the voltage of load resistance and RL denotes the resistance of the load.

Resistance of sensor is usually (2KΩ-20KΩ) for 0.4mg/L gas concentration.

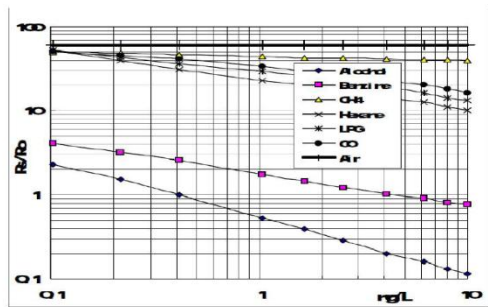


Figure 3. Sensitivity Characteristics

Fig. 3. shows the typical sensitivity characteristics of the MQ3, Ordinate means resistance ratio(Rs/Ro), abscissa is concentration of gases., Rs means resistance of sensor resistance in different gases, Ro means resistance of in 0.4mg/l alcohol. All test are under standard test conditions.

$$\text{Sensitivity} = \frac{Rs(\text{air})}{Rs(\text{alcohol})} \geq 5 \tag{5}$$

Where Rs(air) denotes the resistance of the sensor at various concentration of gas and Rs(alcohol) denotes the resistance of sensor in 0.4mg/L alcohol.

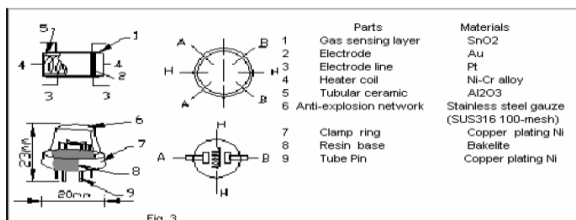


Figure 4. Structure and Configuration of MQ 303

Structure and configuration of MQ-3 gas sensor is shown as Fig. 3, sensor composed by micro AL2O3 ceramic tube, Tin Dioxide (SnO2) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-4 have 6

pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.

C. Zigbee Data Transmission

Secured wireless data transmission is achieved using Zigbee Module. Zigbee is a embedded RF module operating in the frequency of 2.4Ghz. Zigbee is a 20 pin module with only 4 pins being used- DIN, DOUT, VCC and GND. Zigbee module requires 2.1V-3.3V for its operation. Zigbee module has wireless antenna embedded in the antenna, which can be Yagi antenna or an omnidirectional antenna. The transmitting and receiving current required is 40mA. Zigbee module is interfaced with the Host device (PIC16F877A) through a logic level asynchronous serial port as shown below.

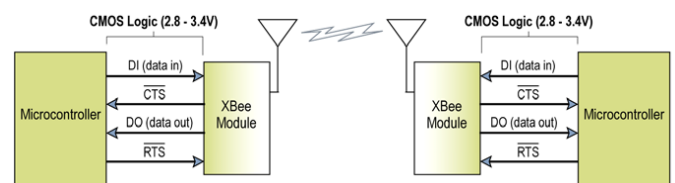


Figure 5. Zigbee interfacing with Microcontroller

In PIC 16F877A, Tx pin(25) is connected to DIN and DOUT from Zigbee is connected to Rx pin(26). Data enters the module UART through DIN(3) as an asynchronous serial signal. The signal should be idle high when no data is being transmitted. Each data byte consists of a start bit(low), 8 data bits and a stop bit(high). The following figure illustrates the serial bit pattern of data passing through module.

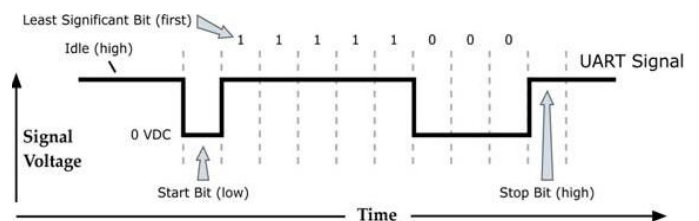


Figure 6. Serial Bit Pattern of Data in Zigbee

Serial Communication depend on the two UARTs(the microcontroller's and the RF module) to be configured with compatible settings(baud rate, parity,start bits, stop bits, data bits).The baud rate for serial data transmission is set as 9600bps.

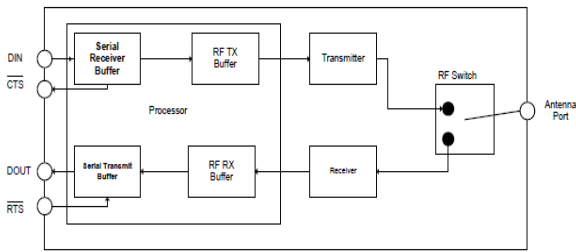


Figure 7. Data Flow Diagram

**D. Obstacle Detection Mechanism**

The device used for obstacle detection is the TSOP 1738 IR Sensor. Sensor keeps on transmitting modulated infrared light and when any object comes near, it is detected by the sensor by monitoring the reflected light from the object. The sensor has a detection range of upto 20cm. Operation Range varies according to the color of the object, light color has more range. The TSOP 1738 IR Sensor has 3 wires. Red wire is connected to the Positive supply. Analog output is taken out from the middle Brown wire. Black wire is connected to the Ground state. The output of the sensor is active low when object is sensed and active high when no object is sensed.

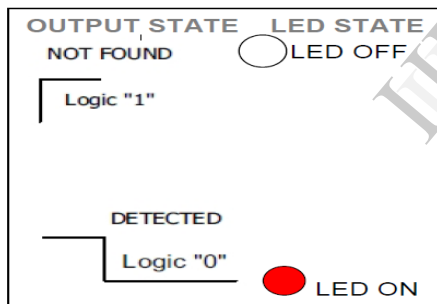


Figure 8. Logic Output State of Sensor

The main problem with the IR proximity sensor is the ambient light and surrounding sources of IR like the sun and the halogen lamps that can cause false triggering of the sensor due to emission of infrared light. To avoid this false detection, the solution is to send pulses of IR light at a certain frequency instead of a constant beam and build a receiver that would detect IR pulses of same exact frequency, cutting off all pulses of higher or lower frequency.

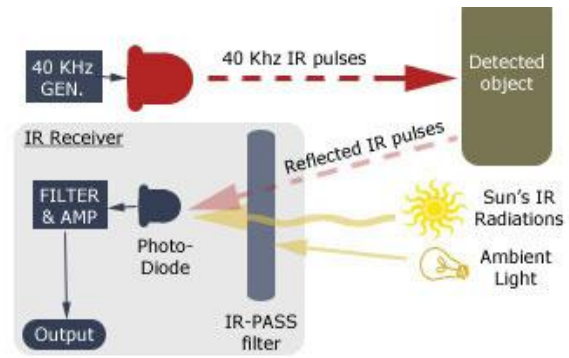


Figure 9. Obstacle Detection Mechanism

Instead of keeping just a BandPass filter to filter the signals, we use a common IR receiver Module TSOP 1738 that incorporates a receiver; an amplifier and a very reliable filter that rejects all the signals that are a couple of KHz far from the original central frequency. The central frequency is fixed in this module as 38KHz. IR Pass filter still detects a lot of noise due to other sources of IR light, so the signal received by the diode are fed to another stage composed of an active filter to select the 38KHz IR signals among others, amplify it and demodulate it providing a clear logic output (0 or 5V).

**III. APPLICATION SOFTWARE**

LabVIEW (Laboratory for Virtual Instrumentation Engineering Workbench) is a system design application platform and development environment for a visual programming language from National Instruments. LabVIEW is commonly used for data acquisition, instrument control and industrial automation on a variety of platforms including Microsoft Windows, various versions of UNIX, Linux and Mac OS X. The latest version of LabVIEW is LabVIEW 2012 release in August 2012.

The programming language used in LabVIEW, also referred to as G, is a data flow programming language. Execution is determined by the structure of a graphical block diagram on which the programmer connects different function nodes by drawing wires. These wires propagate variables and any node can execute as soon as all its input data become available. LabVIEW ties the creation of user interfaces (called front panels) into the development cycle. LabVIEW programs/subroutines are called virtual instruments (VIs). Each VI has three components: a block diagram, a front panel and a connector panel. The last is used to represent the VI in the block diagrams of other, calling VIs. Controls and indicators on the front panel allow an operator to input data into or extract data from a running virtual instrument.

**A. Front Panel Design**



The front panel is the user interface of a VI. Generally, the front panel is designed first, then design the block diagram to perform tasks on the inputs and outputs you create on the front panel. build the front panel with controls and indicators, which are the interactive input and output terminals of the VI, respectively. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. Controls simulate instrument input devices and supply data to the block diagram of the VI. Indicators simulate instrument output devices and display data the block diagram acquires or generates. Select Window » Show Controls Palette to display the Controls palette, then select controls and indicators from the Controls palette and place them on the front panel. Use property dialog boxes when you want to configure a front panel control or indicator through a dialog box that includes context help or when you want to set several properties at once for an object. Use shortcut menus to quickly configure common control and indicator properties. Options in the property dialog boxes and shortcut menus differ depending on the front panel object.

Some palettes contain a control and an indicator for the same type or class of object. You can change a control to an indicator by right-clicking the object and selecting Change to Indicator from the shortcut menu, and you can change an indicator to a control by right-clicking the object and selecting Change to Control from the shortcut menu. You can replace a front panel object with a different control or indicator. When you right-click an object and select Replace from the shortcut menu, a temporary Controls palette appears, even if the Controls palette is already open. Select a control or indicator from the temporary Controls palette to replace the current object on the front panel. Selecting Replace from the shortcut menu preserves as much information as possible about the original object, such as its name, description, default data, dataflow direction (control or indicator), color, size, and so on. Use the front panel controls and indicators located on the Controls palette to build the front panel. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. Controls simulate instrument input devices and supply data to the block diagram of the VI. Indicators simulate instrument output devices and display data the block diagram acquires or generates. Many front panel objects have a high-color, three-dimensional appearance. Set your monitor to display at least 16-bit color for optimal appearance of the objects. The 3D front panel objects also have corresponding low-color, two-dimensional objects. Use the 2D controls and indicators located on the Classic Controls palette to create VIs for 256-color and 16-color monitor settings. Select File » VI Properties and select Editor Options from the Category pull-down menu to change the style of control or indicator LabVIEW creates when you

right-click a terminal and select Create »Control or Create »Indicator from the shortcut menu.

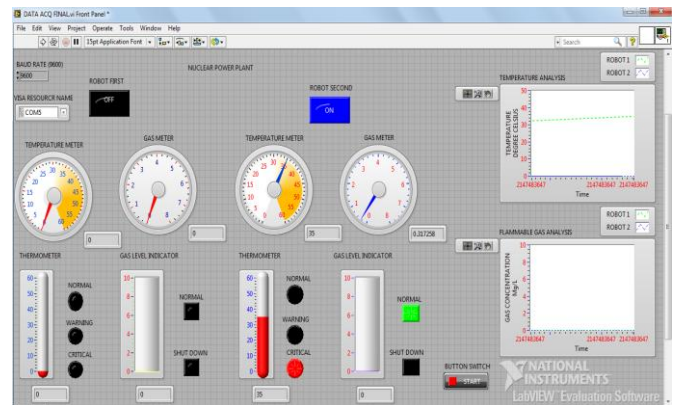


Figure 10. Front Panel design

Select Tools »Options and select Front Panel from the top pull-down menu to change the style of control or indicator LabVIEW creates in new VIs when you right-click a terminal and select Create »Control or Create » Indicator from the shortcut menu. Use the numeric controls and indicators located on the Numeric and Classic Numeric palettes to simulate slides, knobs, dials, and digital displays. The palette also includes color boxes and color ramps for setting color values and a time stamp for setting the time and date for the data. Use numeric controls and indicators to enter and display numeric data. The slide controls and indicators include vertical and horizontal slides, a tank, and a thermometer. If you drag the slider to a new position and the VI is running during the change, the control passes intermediate values to the VI, depending on how often the VI reads the control. Slide controls or indicators can display more than one value. Right-click the object and select Add Slider from the shortcut menu to add more sliders.

The rotary controls and indicators include knobs, dials, gauges, and meters. The rotary objects operate similarly to the slide controls and indicators. Change the value of a rotary control or indicator by moving the needles, by clicking a point of the rotary object, or by using the optional digital display. Rotary controls or indicators can display more than one value. Use graphs and charts to display data in a graphical form. Graphs and charts differ in the way they display and update data. VIs with graphs usually collect the data in an array and then plot the data to the graph, which is similar to a spreadsheet that first stores the data then generates a plot of it.

### B. Block Diagram Configuration

After you build the front panel, you add code using graphical representations of functions to control the

front panel objects. The block diagram contains this graphical source code. Front panel objects appear as terminals on the block diagram. Double-click a block diagram terminal to highlight the corresponding control or indicator on the front panel. Terminals are entry and exit ports that exchange information between the front panel and block diagram. Data you enter into the front panel controls enter the block diagram through the control terminals. During execution, the output data flow to the indicator terminals, where they exit the block diagram, reenter the front panel, and appear in front panel indicators. Objects on the block diagram include terminals, nodes, and functions. You build block diagrams by connecting the objects with wires. You can configure front panel controls or indicators to appear as icon or data type terminals on the block diagram. A terminal is any point to which you can attach a wire, other than to another wire.

The following nodes are used for configuring the Block Diagram of the above Front Panel

- VISA Resource Name

Denotes the serial port to which LabVIEW session is being connected. The input to the VISA Resource Name is the serial port name to which the external device or interface is connected.

- VISA Configure Serial Port

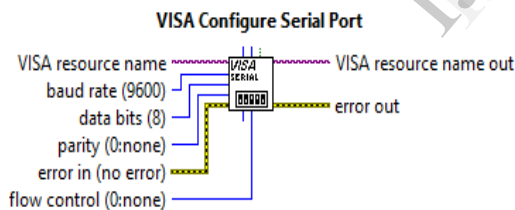


Figure 11. VISA Serial Port

Initialises the serial port specified by VISA resource name to the specified settings. Wire data to the VISA resource name input to determine the polymorphic instance to use or manually select the instance.

- VISA I/O Buffer

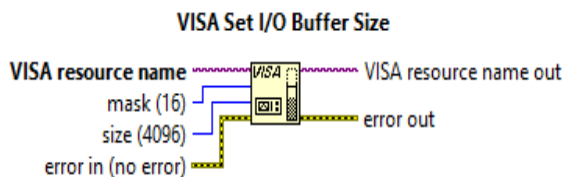


Figure 12. VISA I/O Buffer

Sets the size of I/O Buffer. Run the VISA configure serial Port VI first if size of the serial port buffer is set.

- VISA Read

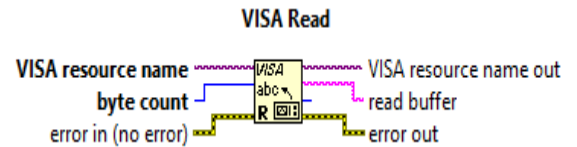


Figure 13. VISA Read

Reads the specified number of bytes from the device or interface specified by VISA Resource name and return the data in read buffer. Here the byte count is set as 11 bytes.

- Initialize Array

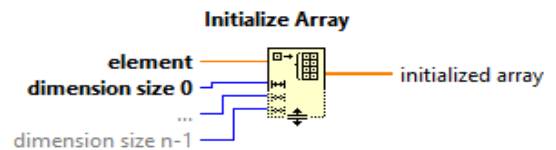


Figure 14. Array Initializing

Creates an 'n' dimensional array in which every element is initialized to the value of element. Array size is initialized to be 15.

- Index Array

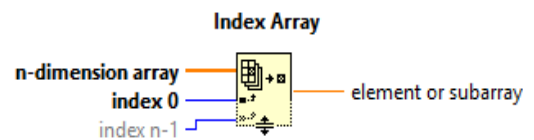


Figure 15. Array Index

Returns the element or subarray of n dimensional array at index. The index value is set as 11.

- MathScript Node

Executes LabVIEW MathScripts and the other text based scripts using the MathScript RT module engine. If a MathScript Node contains a warning glyph, LabVIEW operates with slower run time performance for the node. Script can be modified so that warning glyph can be removed and run time performance can be improved.

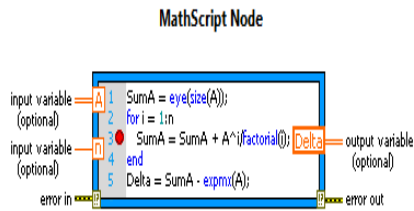


Figure 16. MathScript Node

- Decimal String to Number

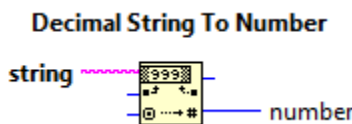


Figure 17. Decimal String to Number

Converts the numeric characters in string starting at offset, to a decimal integer and returns it in number.

- Case Structure

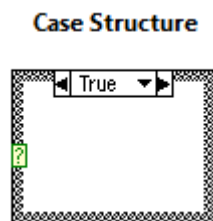


Figure 18. Case Structure

Has one or more subdiagrams exactly one of which executes when the structure executes. The value wired to the selector terminal determines which case to execute and can be Boolean, string, integer and enumerated type. Right click the structure Border to add or delete cases. Use the labelling tool to enter values in the case selector label and configure the values handled by each case.

Function icons on the Functions palette have pale yellow backgrounds and black foregrounds. Functions do not have front panels or block diagrams but do have connector panes. You cannot open nor edit a function. The Functions palette also includes the VIs that ship with LabVIEW. Use these VIs as subVIs when you build data acquisition, instrument control, communication, and other VIs. You transfer data among block diagram objects through wires. Each wire has a single data source, but you can wire it to many VIs and functions that read the data. You

must wire all required block diagram terminals. Otherwise, the VI is broken and will not run. Display the Context Help window to see which terminals a block diagram node requires. The labels of required terminals appear bold in the Context Help window.

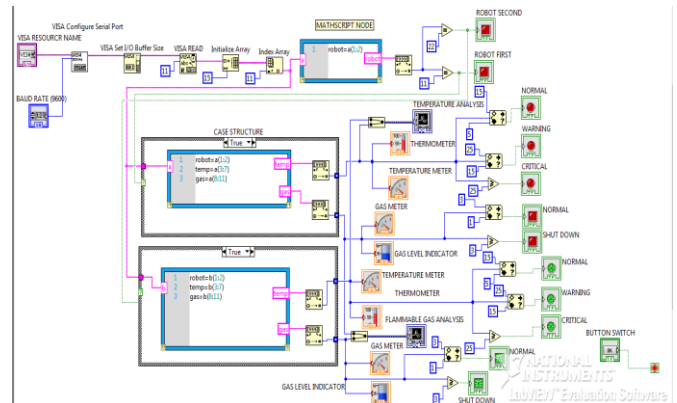


Figure 19. Block Diagram design

#### IV. HARDWARE DEVELOPMENT

##### A. Flow Diagram

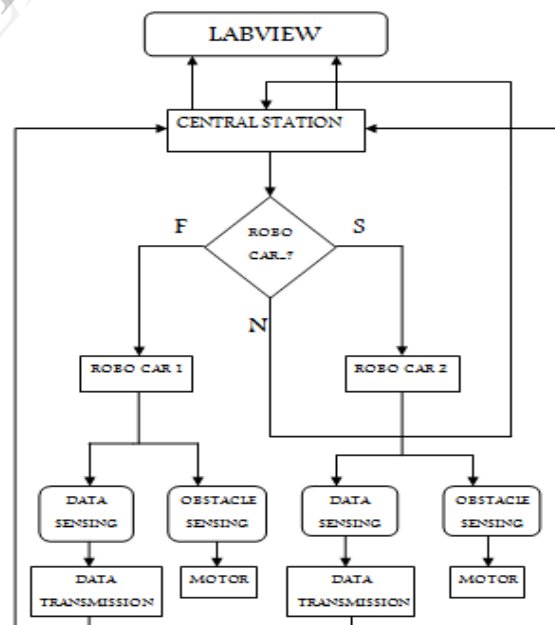


Figure 20. Flow Chart

The control of the Multi robot system is done by the user from the central station. A control switch serves this purpose. The switch sends a command to the robots through wireless medium. The corresponding robot gets activated and starts its voyage. It explores its path and detects its obstacles. Throughout its voyage data acquisition

operation is performed and data collected is transmitted back to the central station. Virtual Representation of the different environmental conditions at different locations in an industry can be done using LabVIEW.

### B. Block Diagram-RoboCar

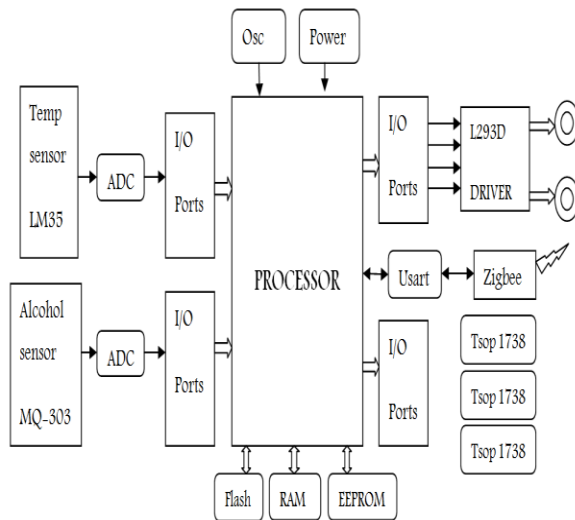


Figure 21. RoboCar Design

The Robo Car section can be considered as a Programmable Logic Controller part. It includes a microcontroller (PIC16F877A) with temperature and alcohol sensors (LM35 and MQ 303) interfaced with its I/O ports. IR obstruction Sensors are also connected to the microcontroller via I/O ports. Output of the IR sensor which is active is given as input to the microcontroller. There are 5 I/O ports with 33 I/O lines which can be programmed as input or output as per the user requirement.

Ports A,D are programmed as input ports and Ports B,C are programmed as output ports. Data transmission and reception takes place through Zigbee wireless communication interfaced with microcontroller through USART. DC motors are connected to the controller with the help of a motor driver IC (L293D). Current from the microcontroller is in microampere ( $\mu\text{A}$ ) range which is insufficient to drive the motor. With the help of this driver circuit, we can increase the current strength to milliamphere (mA) range.

### C. Block Diagram-Central Station

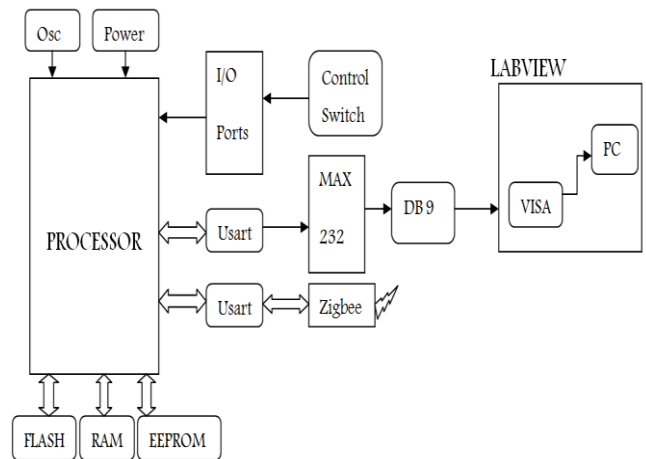


Figure 22. Central Station Design

The Central Station includes a microcontroller (PIC16F877A), a control switch, Zigbee wireless module, a MAX 232 converter and a DB9 female connector. The central station is interfaced with PC using a USB to Serial/RS232 converter which is a male connector at one end and a USB cable at the other end. Data received from the roboCar section is processed and transmitted to the PC using a MAX 232 converter.

MAX 232 converter is a voltage level converter which converts the microcontroller output voltage to the voltage level required by the PC to operate. The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept  $\pm 30\text{-V}$  inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels.

## V. SOFTWARE AND HARDWARE IMPLEMENTATION

### A. Softwares Used

- Embedded C is the programming language used for the PIC microcontroller.
- PIC C Compiler is used for the programming of the microcontroller.
- Proteus 7 is used for the design of PCB layout and simulation.
- LabVIEW is the system design Application Software used for the virtual representation of the environment.
- VISA is the interfacing software.
- WIN PIC 800 is used for burning the program into the IC.

### B. Hardware Implementation





Figure 23. Hardware Implementation

VI. EXPERIMENTAL RESULTS AND ANALYSIS

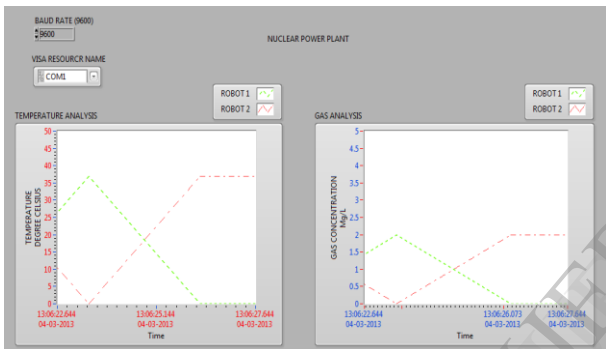


Figure 24. Temperature Analysis

Figure 24 shows the temperature and gas analysis of the two robocars moving in different parts of an industry. The green dotted lines indicate Robo Car 1 and the red dotted lines indicate Robo Car 2. The X axis denotes the current time values and Y axis denotes the temperature (in degrees) and gas concentration in mg/L.

TABLE I. OBSTACLE DETECTION ANALYSIS

OBSTACLE			OBSTACLE DETECTION ANALYSIS TABLE				ACCURACY RATE	
LEFT	FRONT	RIGHT	LEFT	RIGHT	DIRECTION	DAY	NIGHT	
N	N	N	ANTICLK	ANTICLK	REVERSE	100%	90%	
N	N	Y	CLK	STILL	RIGHT	90%	80%	
N	Y	N	CLK	CLK	FORWARD	100%	90%	
N	Y	Y	CLK	STILL	RIGHT	90%	80%	
Y	N	N	STILL	CLK	LEFT	90%	80%	
Y	N	Y	CLK	STILL	RIGHT	90%	80%	
Y	Y	N	STILL	CLK	LEFT	90%	80%	
Y	Y	Y	CLK	CLK	FORWARD	100%	90%	

TABLE I. shows the obstacle detection analysis of the TSOP1738 IR Obstruction Sensor. Three sensors will be placed in the robocar-front, left and right. When an

obstacle is detected, the output of the sensor will be active low. According to the outputs of the sensors, the motor will rotate in clockwise or anticlockwise directions. 90%-100% accuracy can be obtained when the experiment is conducted during day time whereas 80%-90% accuracy can be obtained during night.

TABLE II. TEMPERATURE AND GAS ANALYSIS RESULTS

TEMPERATURE AND FLAMMABLE GAS ANALYSIS FROM LABVIEW										
TAKEN ON 01-03-2013										
TIME	ROBO CAR 1(CLOSED ENVIRONMENT)				ROBO CAR 2(CLOSED ENVIRONMENT)					
	TEMPERATURE ANALYSIS	GAS ANALYSIS		TEMPERATURE ANALYSIS	GAS ANALYSIS		TEMPERATURE ANALYSIS	GAS ANALYSIS		
	VALUE	CONCLUSION	VALUE	CONCLUSION	VALUE	CONCLUSION	VALUE	CONCLUSION	VALUE	CONCLUSION
9.30 AM	26.32°C	NORMAL	.03mg/L	NORMAL	32.67°C	WARNING	0.56mg/L	NORMAL		
12.30PM	32.45°C	WARNING	1.58mg/L	SHUTDOWN	36.56°C	WARNING	3.34mg/L	SHUTDOWN		
3.00 PM	40.34°C	CRITICAL	3.46mg/L	NORMAL	43.78°C	CRITICAL	4.35mg/L	SHUTDOWN		
6.00 PM	27.36°C	NORMAL	.007mg/L	NORMAL	29.23°C	WARNING	1.46mg/L	SHUTDOWN		
8.00 PM	24.67°C	NORMAL	.323mg/L	NORMAL	28.45°C	WARNING	0.98mg/L	SHUTDOWN		

TABLE II gives the temperature and gas analysis results taken from the LabVIEW Front Panel on 01-03-2013. The analysis results taken at various time intervals are displayed in the table. The temperature and gas values are analysed and conclusions are made based on these values- normal, critical and shutdown.

VII. CONCLUSION

Multi Robot Exploration System is designed and it works according to the control given by the user regarding which robot should be operated at a particular instant of time. The Robots explore the location in which it is being installed and performs the data acquisition operation of the critical environmental parameters like temperature of that area and observes the leakage of any flammable gas. Multi robots explore the location and detects the obstacle presence using the IR obstruction sensors and corrects the path for its voyage. Coordination between the robots is also achieved using the sensors. The data acquired is sent to the central station from where it is given to the LabVIEW for data processing and analysing. We can have a complete virtual representation of the industrial environment. Interfacing of the Hardware with the LabVIEW is done using VISA.

We can extend this project work to a higher level, ie keeping the multi robot system has a platform we can have the image acquisition and its processing using LabVIEW. We can also interface many sensors to increase the efficiency of the system and thus we can have a complete control of the industry. Using Zigbee-Pro increases the data transmission range upto 1.5Km. Control of the robots can be also done using LabVIEW. We can also improve the Front Panel of

the LabVIEW using more functional palettes present in the latest version of LabVIEW 2012.

#### REFERENCES

- [1] Ding Chengjun,Liu Ximao,Duan Ping,"Development of Gas Leak Detection and Location system based on Wireless sensor Networks",2011 Third International Conference on Measuring Technology and Mechatronics Automation.
- [2] M.S.Tadpatrikar and J.G.Rana,"Automated Gas Leakage Detection System with Visual Data monitoring using Wireless Visual sensor",Proceedings of SPIT-IEEE Colloquium and International Conference,Mumbai,India.
- [3] Zongyao Wang, and Dongbing Gu,"Cooperative Target Tracking Control Of Multi Robots", IEEE Transactions on Industrial Electronics, Vol. 59, No. 8, August 2012
- [4] Baoding.Zhang,Shulan Gao.,"The Study Of Zigbee Technology's Application in Swarm Robotics System," IEEE International Conference on System of Systems Engineering.Vol 4,No.2,pp 43-45,2011.
- [5] Anjan Kumar Ray , Martin McGinnity , Laxmidhar Behera , Sonya Coleman , Mo Jamshidi," Coordinated Traffic Scheduling for Communicating Mobile Robots",Proceedings of 6th International Conference on System of Systems Engineering,USA,June,2011.
- [6] Emi Mathews, Yara Khaluf," Self-organized Cooperation in Swarm Robotics", 2011 14th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops.
- [7] Y. Hu, W. Zhao, and L. Wang, "Vision-based target tracking and collision avoidance for two autonomous robotic fish," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1401–1410, May 2009.
- [8] Xiong N.,He J.,Yang Y.,He Y.,Kim and T.,Lin C., "A Survey on Decentralized Flocking Schemes for a Set of Autonomous Mobile Robots," *Journal of Communications.* Vol 5, No. 1, pp. 31-38, 2010.
- [9] Barnett,Cox,C Cull,"Embedded C Programming and Microchip PIC"Thomson Publications,2007 .

IJERT