VISUAL PROGRAMMING LANGUAGE FOR EDUCATIONAL ROBOT

[1] PROF. JAGRUTI S. PATEL, [2] PROF. DHAVAL R. PATEL

[1]Asst.Professor And Academic Director, Department Of Electronics and Communication Engineering, Parul Institute Of Engineering And Technology, Waghodia, Vadodara

[2] Asst.Professor And Academic Director, Department Of Electronics and Communication Engineering, Universal College Of Engineering And Technology, Ahmedabad, Gujarat

jagujagruti@yahoo.com, dhavalpatel07ec@gmail.com

***ABSTRACT*:** *VPL (Visual Programming Language) is an embedded robotic environment creation which aims at helping novices or beginners to start programming with easy by simplifying complicated coding pattern into the simple script. We are going to develop a visual, simpler and smart environment for which one has no need of programming skills and electronic knowledge to build smart digital robots. User may visually implement the task, write simpler commands, and operate sensors to program robotic module according to their requirement.*

Keywords—Visual programming language, Educational robot, Control, Automation.

I: INTRODUCTION

Robots are a very helpful invention for human beings, but they still have to go a long way to get better results in the future. Robotics is one of the fastest growing segments of the industrial machine market and educational purposes. Driven mainly by advances in computer technology, older robots are quickly made obsolete by new models. As this is a recent most important problem in the field of robotics, various development and research is going on and some of them are already published.

II: CONCEPT

Educational Robots pertain to the use of robots of the same general characteristics as industrial robots, but low cost and safer to use by students.[5] Usually these robots come with curriculum that guides the students on how to program the robot, how to interface it to its environment (I.E. sensors, educational or industrial CNC machines, etc.), and how to maintain it. In addition the educational robots come with 3D simulation software, which enables the students to practice with a virtual robot and its environment.

III: GOAL

Any industry needs two engineers to work on robot, one is mechanical engineer and one is computer engineer. Computer engineer have the knowledge of programming language and mechanical engineer have the knowledge of all the drivers. But, to implement visual programming language for control the robotic motion, user can get more benefit. By implementing visual programming language for robot control, any user can work with on robot. So this robot can be used by any user. Here, It is not compulsory that user need the knowledge of any programming language. If user does not need the knowledge of programming language, then also any user can work on this robot. So there is no need of two engineers in any industries. Thus, only one mechanical engineer instead of two engineers is used in industry.The topic which is selected that "Visual Programming Language for Educational Robot" which is nothing, but an Autonomous Four Wheeled Robot which can move in any direction using visual language programming. The robot will be designed to follow all the instructions which are displayed in VB 6.0 page. Here we will control all the movement of robot. By using visual programming language, robot can control all movements through microcontroller. Here, suppose the robot is move up to 30 cm in forward direction, then after it turn on 90 degree left, and if any obstacle will come in the path of robot it sense the obstacle and that detection of obstacle will displayed on GUI in VB6.0.here all the data will be displayed on Graphical User Interface. The Robot is controlled by user friendly front end software developed in VB6.

IV:HARDWARE SECTION FOR ROBOTIC MODULE

In the hardware section, the sensors such as metal detector and Obstacle sensor continuously monitor the presence of a metal or an obstacle such as the wall and these data are sent to the Atmega16 AVR microcontroller. If there is a presence of any obstacle in the path of robot, the IR LED will detect that obstacle within some range. Here the object can be anything which has certain shape and size, the IR LED transmits the IR signal on to the object and the signal is reflected back from the surface of the object. The reflected signal is received by an IR receiver. The IR receiver can be a photodiode / phototransistor

IJIRECE

or a readymade module which decodes the signal. And that information is send to the microcontroller. The speed of the DC motor is controlled through the motor driver IC L293D which is connected with the AVR microcontroller. Speed Control of the motors is achieved using pulse width modulation technique. Prolific pl2303 is used for serial Communication interface between USART based serial port of microcontroller and USB port of personal computer. Visual basic based GUI gives the direction of robotic motion and makes it easy to store the path of robot. Visual based Graphical User Interface is sending and receiving the position of robotic motion and it is used for bi-directional wireless data communication.

V: COMMUNICATION BETWEEN PC AND MICROCONTROLLER

Prolific pl2303 is used for Communication interface between USART based serial port of microcontroller and USB port of computer. Here communication between these two entities is important for the information flow of transformation. In general the information transport system can be serial where only one communication line is available; this line is shared by all the bits sequentially. Here, serial communication between AVR microcontroller (ATmega16) and PC. Here data is transmitted from the controller using RS232 standard and displayed on the PC using Hyper Terminal. Visual basic based GUI gives the direction of robotic motion and makes it easy to store the path of robot. Visual based Graphical User Interface is sending and receiving the position of robotic motion and it is used for bi-directional wireless data communication.

VI:COMPONENT USED FOR ROBOTIC MODULE

An ATmega16 AVR Microcontroller is used for carrying out all the required computations and control. It has an in-built Analog to Digital converter. Hence an external ADC is not required for converting the analog speed input into digital value. In the following discussion, I will briefly discuss about the obstacle sensor and metal detector which I have used. An inexpensive 12V, 30 RPM DC motor and motor driver IC L293D is used for control the speed of motor, when robot is in motion. The system will get the response from the IC L293D and will display the speed of motor on the LCD. This speed is varying, when robot is in motion and user can see this speed on LCD display using Proteus software. Here 16*2 LCD is used for Display the path of robot means when robot will go in forward, left and right direction? And how many distance it will go in these directions? Also it will display the speed of motor in the path of robotic motion. DC motors are normally very easy to reverse. By simply changing the polarity of the DC input, the direction of the drive shaft reverses. This property makes DC motors very popular among enthusiast people involved in robotics. In most cases, DC geared motors are used. "H-Bridge" is derived from the actual shape of the switching circuit which controls the motion of the motor. It is also known as "Full Bridge". Usually H-bridge is preferred way of interfacing a DC motor. 12v lead acid battery is a dc battery with lead terminals and an acid, usually hydrochloric acid is used as an electrolyte in lead acid battery.

VII: VB 6.0 USED FOR REMOTE CONTROL UNIT

The VB serves as the front end software for sending the command to the target board through USART communication. This project is done to move the robot forward, left and right as per the command button selection in VB 6.0. That is nothing but the relays that will operate the forward, left and right functions.

VIII: PLANNING

The first thing to make a C programming for display the speed on Liquid Cristal Display through AVR-microcontroller and to control the speed of motor through AVR microcontroller. In VB 6.0 software, visual programming language for control the path of robot will be implementing here. Some software's like AVR studio 4, VB 6.0, Proteus and eagle-win-5.4.0 will be used in this chapter. How this software is used in this project that will also show over here. How program can be installing in this software? And how we can use this software? These all are discussed in this Paper.

IX: SOFTWARE IMPLEMENTATION-CREATING AN AVR STUDIO PROJECT

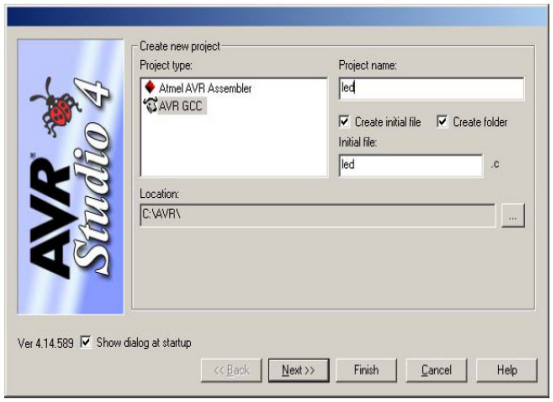


Fig 1. Entering project type, name and location

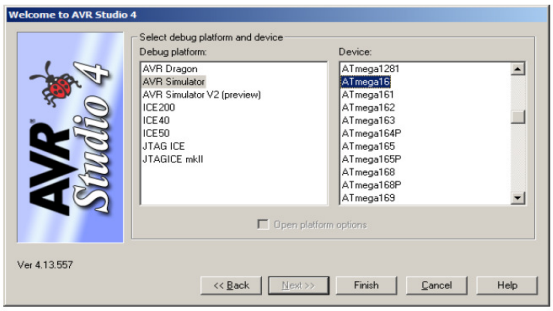


Fig 2. Selection of Debug platform and device window

IJKREC

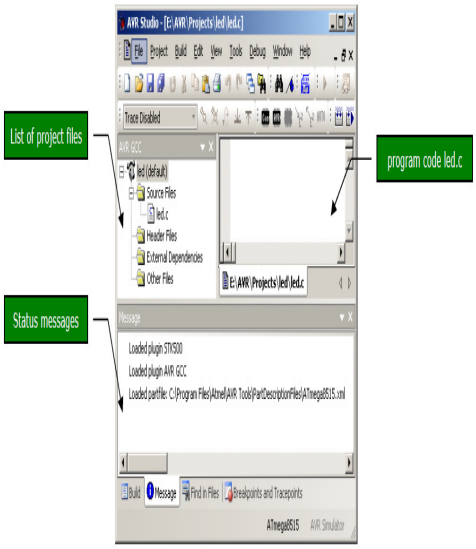


Fig 3. The AVR Studio with a project file open

X: COMPILING C CODE TO HEX FILE

Click menu Build or Rebuild All to compile the C code. If there is no error message, a file called led.hex will be produced. This file contains the machine code that is ready to be downloaded to the ATMEGA16 microcontroller. The file is stored in sub-folder '\default' of your project. Now, try entering Atmega-16 into the Keywords field.

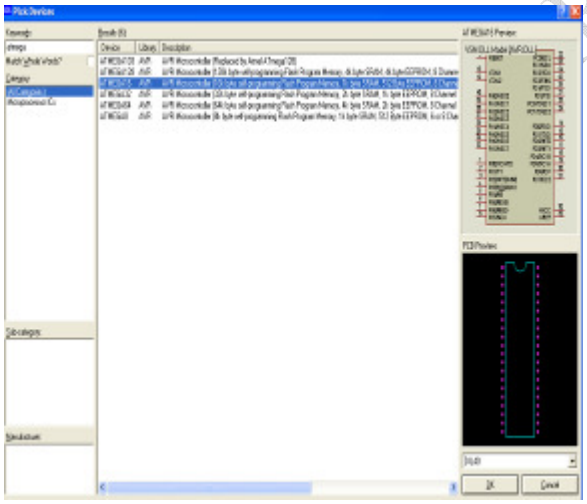


Fig 4.Entering Atmega-16 into the Keywords field.

Then press "ok "button to select that component.

We can customize the information displayed in the Library Browser's results list by right clicking the mouse on the results list. The context menu provides us with options to display Categories, Sub-Categories, Manufacturer and Library alongside each result. Finally, double click on the line in the results list to pick the 'Atmega-16' part into the design. The part will then appear in the Object Selector as shown below:

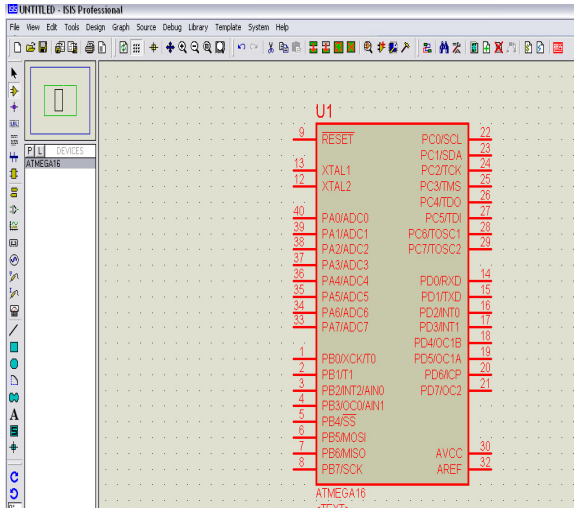


Fig 5.select and draw Atmega16 using Proteus software

We now have a grasp of how to select, place, edit and wiring procedure on the ISIS schematic editor. After connecting all the components, it looks like this, which is shown in below figure:

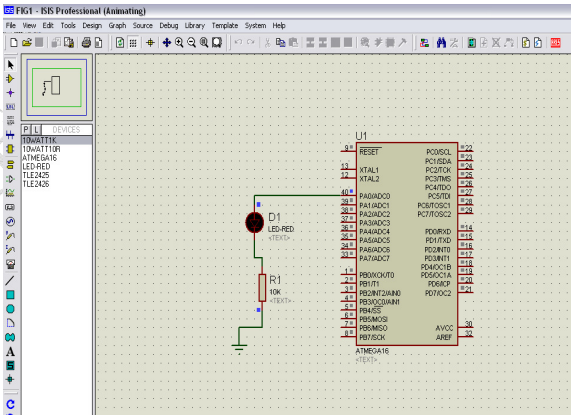


Fig 6.Connecting components with Atmega16

XI : LAYOUT EDITOR IN EAGLE SOFTWARE

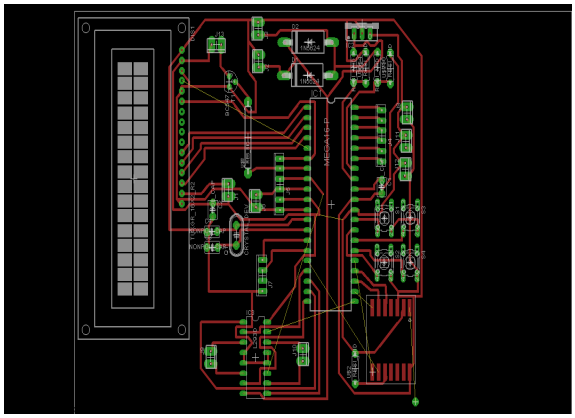


Fig 7. Layout Editor in Eagle

XII: VISUAL BASIC

Start Visual Basic by double- clicking the visual basic icon on desktop and after this the New Project dialog box appears, as shown in Fig. 8.

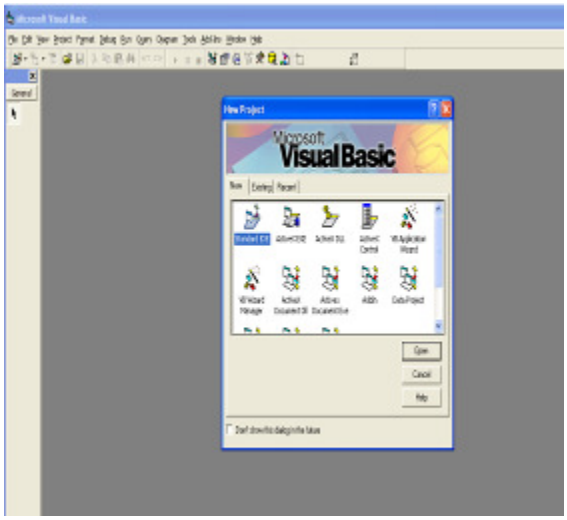


Fig 8.Creating a new Visual Basic project

Double-click the Standard EXE item in the New Project dialog box. Thereafter, using the Command Button tool in the Visual Basic toolbox, add a new command button to the main form. Next, in the Properties window, change the **Name** property of this button from **Command1** to as required in the program. Say we add a command ON and OFF as shown in Figure.

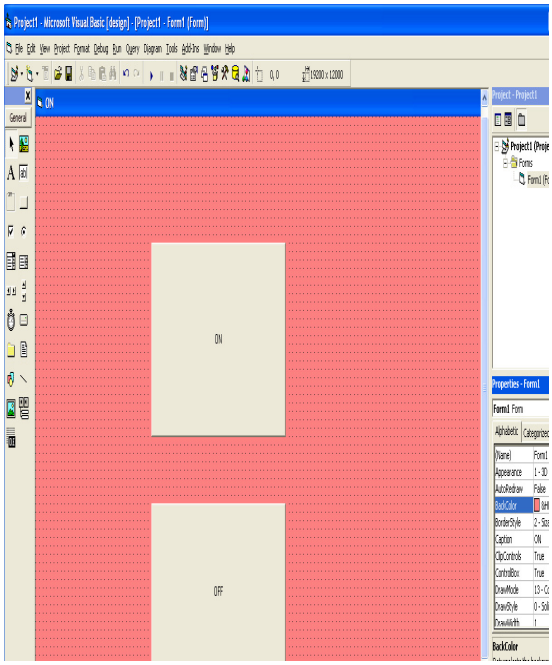


Fig 9.Object Window in VB

To start coding, double-click any button (ON or OFF), opening the code window, as shown in below Fig 10.

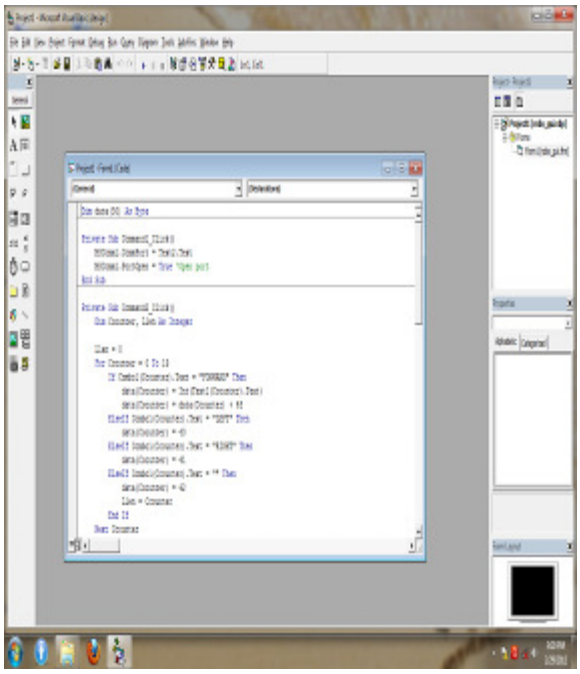


Fig 10.Code Window in VB

After the program is complete, run it, and the output window appears as shown in Fig. 11.

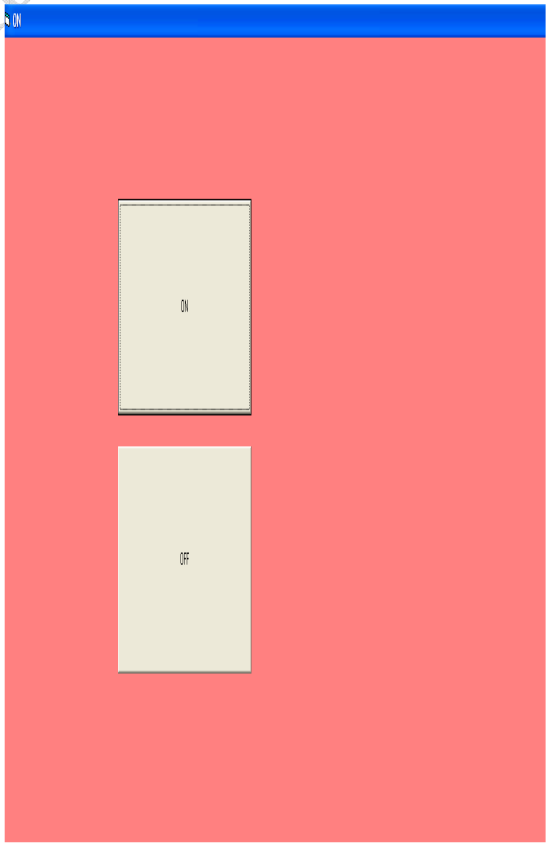
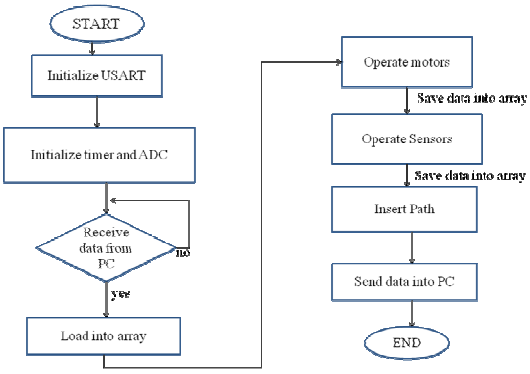


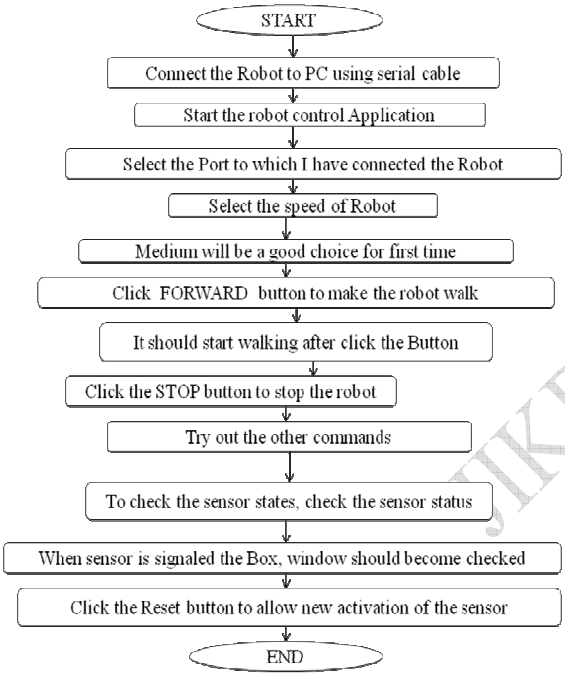
Fig 11.Window after running the code in VB

JKRECE

XIII: FLOWCHART FOR ROBOTIC MODULE



XIV: FLOWCHARTFOR PC COMMUNICATE WITH MICROCONTROLLER



XV: HARDWARE IMPLEMENTATION AND TROUBLESHOOTING

Fig 12.Hardware of robotic platform

In the above figure 12, connect two DC motor with 4 wheels means one motor for and make the wired connection with the 4 wheels in the robotic module.

Fig 13.PCB design

In figure 13, first of all prepare the PCB layout and then soldering of all this components are completed.

Fig 14.12V Lead acid battery

In the above figure 14, there is 12V lead acid battery,which gives the power supply to the interfacing circuitry, ATmega16 AVR micro controller and two DC motors.

Fig 15.Hardware of robotic module

Here, all the components are connected with PC and AVR microcontroller, which is shown in figure. In this figure, AVR microcontroller is connected with MAX 232 IC and also connected with motor driver IC L293D for analog to digital conversion. Now give the power supply to this robotic module. By using visual programming language, Graphical User Interface will be implement on VB6.0. So Robot can move in given direction, which is written in GUI page. Whatever information written on GUI, that all commands are coming to the AVR microcontroller through PC. After sending this command, LCD is displayed the path of robotic motion with given direction and given distance, that is shown in figure 15.

Fig 16. Path followed by the given robotic motion

After moving this robot in given direction and with given distance, finally, Robot will stop. So that is also displayed on LCD with 0, 0, which is shown in figure 16.

XVI: CONCLUSION

The objective of the topic to design four wheeled autonomous robot which changes the speed of robotic motion and move in given direction with the turn of angle. Also the sensors for obstacle detection and metal detection which are in the path of robot are correctly detected. These different sensors such as IR sensor for obstacle detection and Proximity sensor for Metal detection are satisfactory working. The program for the speed control of DC motor, program for Autonomous robot is completed in C language and also the Visual programming language for control the direction of robotic motion is prepared in VB software. Finally, the robot is successfully built and tested as specified by the objective. At last, it is concluded that four wheel autonomous robots is move in given path and that path will displayed on GUI. If any obstacle or metal will come in the path of robot, it returns back on that path. After that, again new path will be given to avoid this obstacle or metal.

REFRENCES

[1] Oberlander, Jon, Paul Brna, Richard Cox, Judith Good, "The GRIP Project, or... The Match-Mismatch Conjecture and Learning to Use Data-Flow Visual Programming Languages," Collaborative project between HCRC, The University of Edinburgh & CBLU, The University of Leeds, April 1997 to September 1999.

[2] Visual Language Research Bibliography, see the website http://www.cs.orst.edu/~burnett/vpl.html#V2A3.

[3] IEEE Symposium on Visual Languages (VL), see the website http://VisualLanguages.ksi.edu/.

[4] Najork. Marc, "Programming in Three Dimensions," Journal of Visual Languages and Computing 7(2):219-242, June 1996.

[5] Erwig. Martin,"Semantics of Visual Languages," 13th IEEE Symp. on Visual Languages (VL'97), 304-311, 1997.

[6] Burnett, Margaret, Marla Baker, "A Classification System for Visual Programming Languages," Journal of Visual Languages and Computing, September 1994, 287-300.

[7] Ibrahim, Bertrand, "Semiformal Visual Languages, Visual Programming at a Higher Level of Abstraction," World Multiconference on Systemics, Cybernetics and Informatics (SCI'99 and ISAS'99), Orlando, Florida, July 1999, 157-164.

[8] Barr, A., E. A. Feigenbaum, "General Problem Solver (GPS) ", developed by Newell, Shaw and Simon, in Handbook of Artificial Intelligence, Vol 1, Article II.D.2, 113-118, 1981.

[9] Schonlau, W., "MMS, a Modular Robotic System and Model-Based Control Architecture", SPIE Sensor Fusion and Decentralized Control in Robotic Systems, Boston, Massachusettes, September 1999.