International Journal of Computers, Communications & Control
Vol. I (2006), No. 2, pp. 73-83

Inteligent Line Follower Mini-Robot System

Román Osorio C., José A. Romero, Mario Peña C., Ismael López-Juárez

Abstract: This paper shows a prototype development of an intelligent line follower mini-robot system, the objective is to recognize, understand and modify the actual performance of the movements of the robot during its pathway by way of getting information in real time from different magnetic sensors implemented in the system and based in a V2X digital compass, microcontroller and odometric measurements. The paper shows as well, the system characterization of the V2X sensor (digital compas) and the cost-benefit of the prototype implementation and performance. The programming techniques and easy operation is detailed too.
Keywords: autonomous, mobil-robot, inteligent system

1 Introduction

It is shown in the article the design and implementation of an “autonomous intelligent line follower navigation system” which has been designed in the DISCA-SEA IIMAS-UNAM, it is used an inductive-magnetic based sensor to measure magnetic fields which shows the orientation and position of the pathway of the mobil-robot to get real-time navigation information. This efficient potential tool has been used since the 90‘s in industrial applications with a lot of success because of its good performance, small size and compatibility with different electronic systems. The follower system (FS) uses a continuous black line on a white background in its pathway during its travel trajectory, the line has to be 18 millimeters width in order to have an acceptable functioning. To analyze the inductive sensor, test experimentation was done within a roadway, capturing data where possible conflicts could be appears during the test, this data was used as inputs to the algorithm to achieve corrections on the decision speed and control movements. A microcontroller assigns the speeds to optimize the performance according to time movements in order to get a closer behavior as humans could do it without a vision system to catch its environment. The task of this work has been to study the V2X compass real performance to develop the navigation algorithms and implement them in a AT90S1200 Atmel microcontroller.

2 System Development

The elements involved during the physical design are:

2.1 Transmission

The mobil prototype was implemented with a transmission system conformed with two rear wheels, coupled to two direct current motors, each wheel however is independent functioning for gears and reduction box system.

In front of the prototype there is a “crazy wheel” system making possible to achieve free turns up to 360 degrees. Wheels are made of heavy duty plastic with a rugged polymer plastic made friction-avoid central line, this implementation allows heavy weight carry performance (between 0.8 and 1.5 kilograms) with minimal current consumption (typically 300 milliamps). Allowed movements are:

- Forward direction
- Backward direction

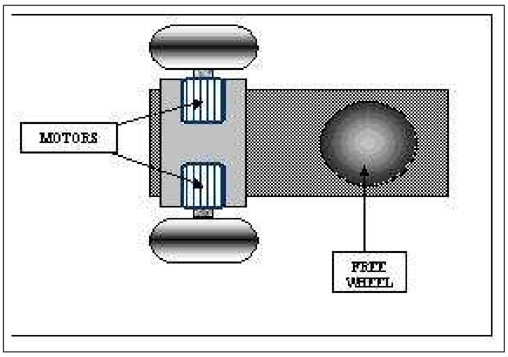


Figure 1: Transmission system and wheels location

- Left and Right turns
- 360 degrees rotation

2.2 Direct current motor control

Control for the two motors in the system is carried out by using the L293D integrated circuit H bridge, a microcontroller enable and disable the motor excitation elements using the internal H bridges in the circuit. The easy operation of the L293D circuit and considering the only one movement in a single direction, it it possible to us the following operation movements codification as shows in following table:

MOTOR 1		MOTOR 2		ACTION
IN 1	IN 2	IN 3	IN 4	
0	0	0	0	High "Z"
1	0	1	0	Rotate right
0	1	0	1	Rotate left

Table 1: L293D operation

And because of the direct control using pulse width modulation (PWM) to on/off and speed control for motors, we use the table 2 input/output information to enable the motors, as it is showed next:

INPUT	PIN ENABLE	OUT
1	1	1
0	1	0
1	0	High "Z"
0	0	High "Z"

Table 2: Motors enable input/output signal sequence

In order to get microcontrollers electrical protection as short circuit and high energy current peaks because of continuous motor switching activity, an isolated purpose interface was implemented between micrcontrollers and L293D circuit by using non-inverted buffers (74HC4050 high speed CMOS circuits) as it is showed in figure 2.

2.3 Line Follower Prototype Architecture

Figure 3 shows the line follower architecture, where microcontroller, sensors and motors interaction can be observed.

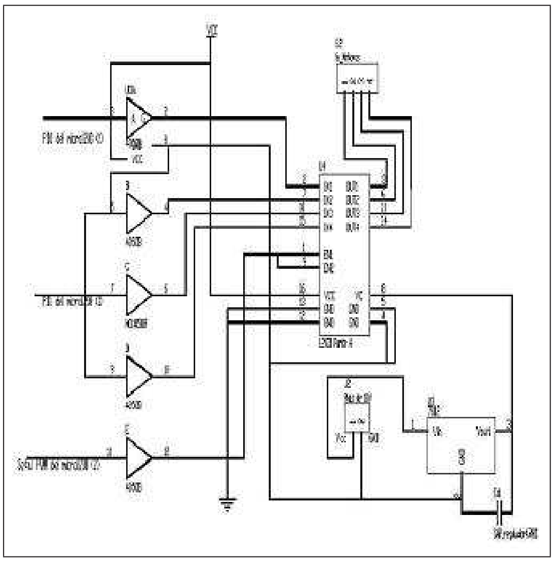


Figure 2: Motor isolated interface

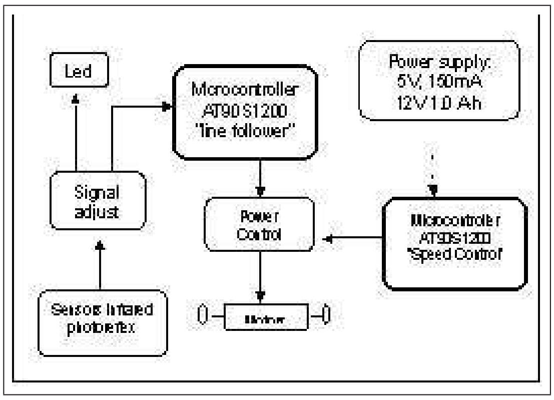


Figure 3: Prototype Architecture

2.4 Sensors Distribution

Sensor distribution within the system is showed in figures 4 and 5.

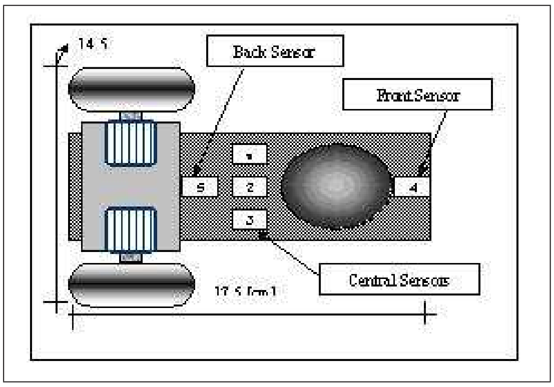


Figure 4: Sensors location within the prototype

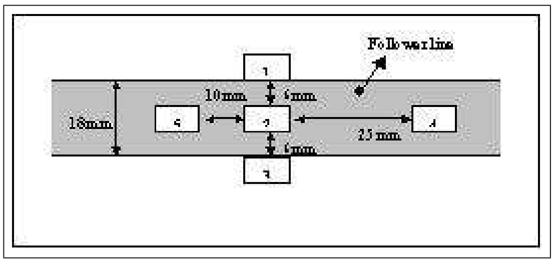


Figure 5: Distance distribution of sensors within the layout platform prototype

2.5 Line following programming description

To implement this section of prototype, a microcontroller AT90S1200 was used, because of its simplicity for programming purposes, the program achive five sensor monitoring and direct current motor control, using the available 32 registers and 2 communication ports of the chip. The microcontroller is good situated for such a task, even still can be expanded if a more complex implementation is required for some other application. First approximation in the design, showed the possibility of using only 3 central sensors for line follower control, but different experiments showed there was a problem when very close curves where in the robot pathway, so adding some more sensors fixed this problem and the line follower system showed good performance for all pathways in the experiments, methodology analysis for sensors and prototype reaction was done as it is showed in next section.

Line drawing analysis

Microcontroller achieves following analysis criteria:

- When two side-central lateral sensors (1 and 3) are out of pathway line and central sensor is on pathway line, the decision is the prototype is following a straight line drawing, then the 2 motors have to be activated instantly. (Figure 4 and 5).
- If prototype is following a line and the front sensor is not ON, there is no curve line and two motors must be still ON.
- If a curve line has not been detected but the right sensor (3) is within the line, the microcontroler activates the left motor to turn left until the signal sensor indicates is already out of the line, then the two motors are activated again at the same time to go fordward.
- If a curve line has not been detected but the left sensor (1) is within the line, the microcontroler activates the right motor to turn right until the signal sensor indicates is already out of the line, then the two motors are activated again at the same time to go fordward.
- When a curve line is detected (fornt sensor signal is activated), the operation process jumps to the algorithm for curve lines reaction.

Curve line analysis

When the front sensor in the prototype finds a curve line (front sensor 4 is ON) the microcontroller achieves next analysis:

- If central and side sensors are out of the line, and the middle sensor is within the line, the microcontroller will activate the right motor until the central right sensor is within the line, this action is named "zigzagder". When right sensor is already within the line, it jumps immediately to "zigzagizq" action.
- ZIGZAGIZQ: if the middle and right central sensors are within the line (left sensor does not care), the microcontroller must activate the left motor until the right sensor is out of the line drawing. When the action gets the right sensor out of the line, the operation jumps to the "zigzagder" action.
- ZIGZAGDER: the middle central sensor is within the line and the right sensor is out of the line, the microcontroller will activate the right motor until the right sensor is within the line. When this is done it jumps immediately to he "zigzagizq" action.

While a curve line is been followed, the front sensor is out of the line (figures 6B,6C and 6D) and the rear sensor is within the line in about first half of the curve line pathway,(figures 6D and 6E). This action guarantees the prototype been in a straight line again, this action is graphically explained in figure 6.

While mentioned actions "zigzagder" and "zigzagizq" are taken placed, at the same time the front sensor signal state is been continuosly monitored, the inf the fornt sensor is OFF (logical "0"), the algorithm jumps to verify the rear signal sensor, and having next two possible decisions:

1. If it is ON (logical "1") implies curve line still does not finish and then "zigzagder" and "zigzagizq" actions must still been taken place until rear sensor is OFF and then the straight line operation is guaranteed and takes place.
2. If it is OFF, implies the sensors are aligned and "zigzagder" and "zigzagizq" actions can still be operating and go to straight line operation mode.

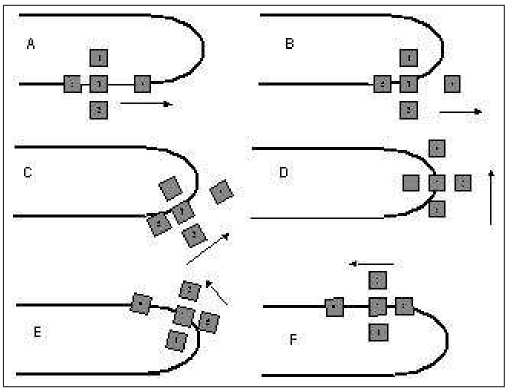


Figure 6: Curve line behavior of prototype sensors

Speed Control

This module in the system architecture allows to have proportional signals to achieve pulse width modulation control for motor speed. The input signal to change width pulse is acquired by way of a dip-switch and a microcontroller to enable and disable the H bridge in the L293D integrated circuit mentioned before.

This stage uses an Atmel AVR AT90S1200 microcontroller to achieve pulse width modulation, this chip was used because of its "cost-benefit" advantages as small size and cost, as well as, easy programming development facilities. Using a microcontroler gives the system flexibility and user capabilities to change frequency operation and optimize power consumption and performance of motors. Most microcontrollers are restricted to 4Khz. frequency to be used for PWM reducing motors performance.

One problem with microcontroller operation used to capture and process information, is the clock cycle lossed signaling as a result of speed changes, this generates a delay operation in the infrared sensors and magnetic-inductive sensor data acquisition signaling.

From experimental results, using different pathways and circuits, and by slope and curve analysis of line drawings used as the prototype road, 4 output driving signals were choosed to speed control of the mobil system. This signals are shown in figures 7, 8 and 9, in figures can be observed a 11.0 Khz. carrier frequency for PWM and ON/OFF timing.

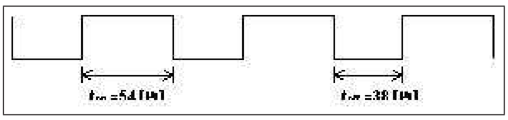


Figure 7: Speed 1 signal representation

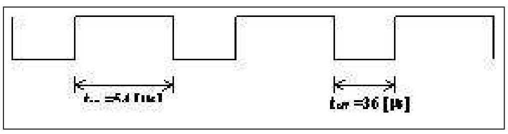


Figure 8: Speed 2 signal representation

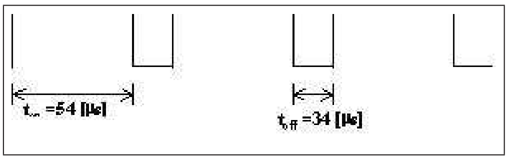


Figure 9: Speed 3 signal representation

2.6 V2X sensor implementation with mobil prototype

Capture and processing information

To capture data from line drawings, the V2X magnetic-inductive sensor was used. In the application an slave mode operation and a binary output format were used in order to make arithmetic operations easier for microcontroller programming, having directly angular data output within the range of 0 to 168ř in hexadecimal numerical representation. In this stage of system architecture, an AT90S8515 microcontroller was used to achieve following operations:

- Infrared sensor monitoring
- Data acquisition for V2X digital compass
- Data analysis to validation process
- Continuous turn recognition
- Data storage
- Data acquisition and comparative processing for continuous turn case operation.
- Data interpretation from mass storage.

V2X acquisition and interpretation programming method

First, data capture is taken place with troubled sections on road pathway, in order of having a best criteria for easier pathways sections as straight and small curve lines in a "learning and acknowledge" route in the pathway road. As it has mentioned before, to detect a line curve first the front infrared sensor numbered 4 (figure 4) is used, if the sensor is out of the line, the microcontroller capture an orientation data from digital compass and storage the information in flash memory. Every time this action takes place the microntroller actually capture two consecutive orientation data to get an acceptable data range to guarantee the data is correct (2 consecutive data has to be almost the same value) because the V2X sensor sometimes generates not-correct data because of prototype vibrations in the road pathway. Typical data from sensor in technical manual information is $\pm 1°$, but for the present design $\pm 3°$ were considered to achieve a robust acquisition.

After orientation of first point-set within the curve line has been detected, it has to be detected when the line finishes, this is done by using the rear sensor (5) (figure 4) as follows:

Because of the front sensor is out of the curve line (it gets a logical "1"), the microcontroller expects the infrared sensor number 4 gets inside the line again, when this is done, inmediatly the rear sensor (5) is monitored and following decisions are taken:

- Sensor is ON, prototype still inside the curve line.
- Sensor is OFF, it is guaranteed that prototype is aligned and curve line has finished.

Once the curve line is finished, the microcontroller acquires an orientation data and storage it in flash memory.

Orientation data storage method

To easy analysis and interpretation of captured data, 3 index registers "X,Y and Z" were used as follows:

1. Data list for orientation to indicate the beginning and end of curve lines, is stored using the X index register within the memory range \$62 to \$100 addresses.

2. Address \$60 is defined to storage the orientation data for the very beginning of the prototype pathway.

3. For every orientation data to indicate the end of every curve line, data is stored in memory beginning in address \$102.

4. In register "z" the "curve characterization" is stored, this data tells the system the size of the curve line (time duration) and is memorized beginning in address \$1A4.

The method is graphically illustrated in figure 10, when generated and stored data structure is showed.

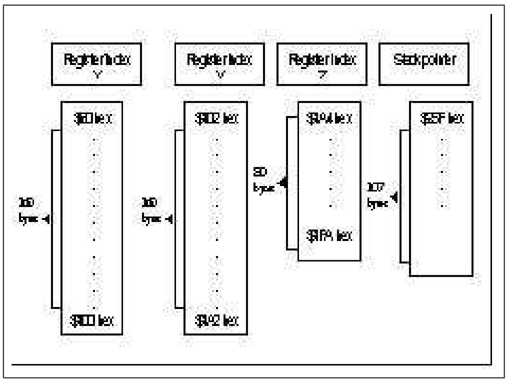


Figure 10: Generated data structure in Microcontroller

The used AT90S8515 microcontroller can storage 512 bytes of data memory, as it is shown in the figure the memory was sectioned in 4 parts to optimize data storage and access of information. Registers X and Y can storage 160 bytes, every curve is codified with 2 bytes so 60 curve lines can be stored which makes the system available to almost any type of curve lines within a line drawing road pathway real operation and with acceptable speed movements. In the other hand, register Z only storage 1 byte data which lets only 107 free bytes for stack pointer operation beginning in memory address \$25F.

Data Interpretation method

For curve characterization, the magnitude difference of initial and final orientation values are used (Z_1), if X_1 is the initial orientation value and Y_1 the final value:

$$Z_1 = |X_1 - Y_1|$$

Z_1 is the curve representation in degrees with a correspondence speed value, this action can be seen in figure 11.

To evaluate and relate the real behavior of the mobil with this values, experimentation was done with different curves types to approximate the real operation of the system. Three experimental speeds were used with different pathways curves and straight lines and the following results are showed in figure 12 and 12 for two different curves.

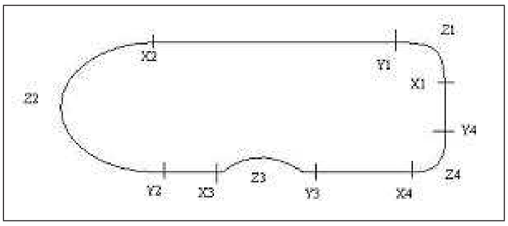


Figure 11: Orientation values X and Y, and difference value Z

3 Results

Curve case studies and speed reaction.

Figures from 12 to 17, show the different curve types case studies, to classify the difficult degree of road line drawing pathway, and so the times and behavior of the prototype in its travelling operation.

CURVE 1

Measurement (V2X)	Portion curve			
	1 +	2 *	3 +	4 *
Stable	341°	66°	75°	158°
Mobil	343°	56, 57°	65, 66°	158°

+ Sensor ON front (begin curve)
* Sensor OFF front (end curve)

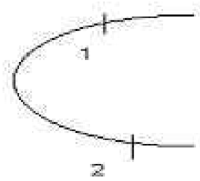
Figure 12: Curve 1 case

Curve case 1 with speed 1:
T_{14}= 4.29-5.44 [seg]
Curve case 1 with speed 2:
T_{14}=3.35-4.07[seg]

Times for travelling through pathway:
For this speeds, the sensor 2 is going inside the line, while sensors 1 and 3 are going out of the line, considering this values adequate for the curve type used.
Curve case 1 with speed 3:
T_{14}=3.18-3.45[seg]

This velocity shows a small time operation, however the mobil gets an unstable behavior because of is trying to follow the best way the curve line, and fast head movements are carried out in the mobil prototype (shaking head movements).

CURVE 2



Measurement (V2X)		
	1 +	2 *
Stable	340 °	170°
Mobil	344°	176, 177°

+ Sensor ON front (begin curve)
* Sensor OFF front (end curve)

Figure 13: Curve 2, case study

Time of travelling pathway:
with speed 1:

t12=3.75-4.31 [seg]

with speed 2:
t12=2.75-3.31 [seg]

With this speeds, sensor 2 is going inside the line and sensor 1 and 3 are going out the line.
With speed 3:
t12=2.31-2.80 [seg]

with this speed the mobil makes shaking head movements 3 times before reaches part 2 of the curve, and never gets a stable state.

Figures 14 show results for 10 different curves and conclusions about decision on its use operation.

CURVE 1					
Section Curve	Measure	Data to compare	Analysis	Range [°]	Speed assign
Section 1 at 2	Stable	341° –> 66°	360° - 341° = 19° 19° + 66° = 85°	73 - 74°	The suitable speed for this curve is the 2, for times and way to travel.
	Mobil	343° –> 56°	360° - 343° = 17° 17° + 56° = 73°		
		343° –> 57°	360° - 343° = 17° 17° + 57° = 74°		
Section 2 at 3	Stable	75° –> 66°	75° - 66° = 9°	8 - 10°	The suitable speed for this curve is the 3, because the range to assign is short
	Mobil	65° –> 56°	65° - 56° = 9°		
		65° –> 57°	65° - 57° = 8°		
		66° –> 56°	66° - 56° = 10°		
		66° –> 57°	66° - 57° = 9°		
Section 3 at 4	Stable	158° –> 75°	Speed assign 33°	92 - 93°	The suitable speed for this curve is the 2, for times and way to travel to assign is short
	Mobil	158° –> 66°	158° - 66° = 92°		
		158° –> 65°	158° - 65° = 93°		

CURVE 2					
Section Curve	Measure	Data to compare	Analysis	Range [°]	Speed assign
Section 1 to 2	Stable	340° –> 170°	360° - 340° = 20° 20° + 170° = 190°	192-193°	The suitable speed for this curve is the 1, for the very close curve (difficulty grade)
	Mobil	344° –> 176°	360° - 344° = 16° 16° + 176° = 192°		
		344° –> 177°	360° - 344° = 16° 16° + 177° = 193°		

Figure 14: Curve characterization format

4 Conclusions

Results and conclusions are showed in figure 16 and 15 for 10 different curves, fr values 0 to 17° speed number 3 was used, and the mobil makes an interpretation as to consider the pathway a straight line. For 18 to 109° speed used was number 2, within this range most curves are classified when experimental results were done and the mobil prototype get fast movements without getting unstable behavior.

For values bigger than 110° experiments shows and agree to use speed number 1, because in this range the mobil make high degree difficult turns like a "U" turn, so the speed has to have a small value in order to get stable behavior.

Range [°]	Assign speed
0 - 17	speed 3
18 - 109	2
110 - 200	1

Figure 15: Speed assigned value ranges

Section Curve	Range [°]	Assign Speed
CURVE 1		
1 - 2	73 - 74°	speed 2
2 - 3	8 -10°	3
3 - 4	92 - 93°	2
CURVE 2		
1 - 2	192 - 193°	2
CURVE 3		
1 - 2	86 - 88°	2
CURVE 4		
1 - 2	167°	1
CURVE 5		
1 - 2	42 - 48°	2
2 - 3	46 -56°	2
CURVE 6		
1 - 2	81 -86°	2
2 - 3	9 -14°	3
3 - 4	81 - 92°	2
4 - 5	68 - 108°	2
CURVE 7		
1 - 2	35 - 42°	2
2 - 3	188 - 199°	1
3 - 4	23 -29°	2
CURVE 8		
1 - 2	17 - 25°	2
2 - 3	45 - 55°	2
3 - 4	26 - 39°	2
CURVE 9		
1 - 2	86 -90°	2
2 - 3	79 - 85°	2
3 - 4	11 - 20°	3
4 - 5	69 - 73°	2
CURVE 10		
1 - 2	81 - 89°	2

Figure 16: Ten different curves characterization

References

[1] Anderson H Peter and Foster Vance, “Interfacing with A Precision Navigation Vector 2X Compass Module”, *Dept of Electrical Engieering*, Morgan State University, http://www.phanderson.com/printer/compass/compass.htm.

[2] Cormen, C. E. Leiserson, R. L. Rivest, The MIT Press, McGraw-Hill, 1990.

[3] Osorio R, Romero J, Bautista A., “Estudio de un Mini Robot seguidor Autónomo Inteligente utilizando un compas V2X.”, Facultad de Ingeniería UNAM.

[4] M. de Berg, M. van Kreveld, M. Overmars and O.Schwarzkopf., *Introduction to Algorithms, Computational Geometry: Algorithms and Applications.*, Springer-Verlag, Berlin, 1997.

[5] K. Mehlhorn, *Data Structures and Algorithms*, Springer-Verlag, 1984.

[6] R. Sedgewick, *Algorithms*.

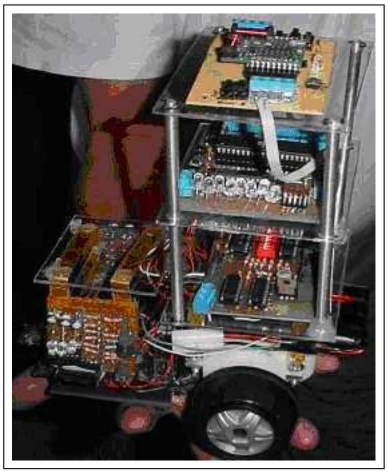
[7] Stefan Gotts, *Collision Queries using Oriented Bounding Boxes.* Tesis North Caroline University, U.S. 2000.

[8] Stolfi., *Oriented Projective Geometry: A Framework for Geometric Computations*, Academic Press, 1991.

[9] Thomas Möller et. al., *Fast, minimum storage Ray/Triangle Intersection.* Universidad Tecnológica de Chalmers.

[10] www.avrfreaks.com, *Microntroladores, Atmel*,10-septiembre-2001.

[11] www.atmel.com , septiembre-2001.



Román Osorio C., José A. Romero, Mario Peña C
Universidad Nacional Autónoma de México
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas
Departamento de Ingeniería de Sistemas Computacionales y Automatización
Sec. Electrónica y Automatización Apdo. Postal 20-726, México D.F.
E-mail: roman@servidor.unam.mx

Ismael López-Juárez
CIATEQ, A.C. Centro de Tecnología Avanzada
Grupo de Investigación en Mecatronica y Sistemas Inteligentes de Manufactura
Manantiales 23A. Parque Ind. B. Quintana, El Marques, Qro. CP 76256. MEXICO