Journal of Mechatronics, Automation and Identification Technology
Vol. 3, No.3. pp. 8 -15

The Importance of an Autonomous Transport that Avoid Collisions in an Industrial Field

Daniel Katz Bonello, Yuzo Iano, Umberto Bonello Neto

DECOM, University of Campinas, Campinas, Brazil
DECOM, University of Campinas, Campinas, Brazil
DECOM, University of Campinas, Campinas, Brazil
Daniel.eng@yandex.com, Yuzo@decom.fee.unicamp.br, Umbertobonello410@gmail.com

***Abstract*— This work refers to the study and development of Mechatronic's area, searching for an autonomous locomotion system that can be used in various industrial sectors without the risk of collisions on work. This project was based on research: bibliographical and of field of the Exploratorium type, using sampling. We obtained positive results with the applications of the sensors, microcontrollers, transistors and more.**

***Keywords*— Robotics, autonomous, microcontrollers, PIC**

I. INTRODUCTION

The project in question views the study and development of an autonomous robot that has the capacity to interact with the extern environment and choose the great trajectory for your locomotion to start off from the obtained data, being possible this process due the automation mechanisms implementation.

Is highlighted the importance of this kind of automation to the industrial sector, because this area is always looking for new alternatives to fasting the process and enhance the quality of your products, having in the confection of this robot with articulated rotation devices (wheels), an opportunity to found a load transportation system or even more of monitoring that solves your more variable practical problems (constant monitoring of inaccessible locations for humans, for example).

The scheme described along of this scientific work, even being a bit complex, allows the implementation a serial of changes that looking for adapt him to the final user objective with the finality of elimination some existing blocks.

A. Problem delimitation

Due to the vast field that the transportation and the management of loadings and materials have in the relation of human with the century XXI, the project was delimited to supply the more important questions inside the various sectors related to the transportation of small and medium charges and to the management of resources either in industries nor in organizations that need of autonomous machines of transportation. The questions are: supply the major agility and flexibility in the transportation of small and medium charges; replace the human in tasks that put him in life-risk; participate of production process, being attached to the robot components that can end-off the problem of dependence of humans in integral period in the realization of simple tasks.

II. LITERATURE REVIEW

Old researches define the autonomous robot as: "a robot able to locomote only through independent systems control".

One of the first autonomous robot of history was the "Shakey" robot of Research Institute of Stanford, in the decades of 60's and 70's, developed to interact with complex environments. Many other mobile autonomous robots was builded, for example: Cart Robot, to the studies of stereo vision; Pluto Robot, for problems studies and stabilization; Neptune Robot, moving through cameras and sound sensors.

"Terrorism, population old-age, changes in the weather, are some challenges of the world nowadays. But the intelligent machines will aid to solve those and other social problems and regarded to the nature. As before sayed Emilio Alvarez, this autonomous system really will be of extreme importance to the society in various factors and areas.

The new technology disponibilization allows the formation of new ideas in the areas of intelligent machines. "The intelligent systems to the autonomous mobilie robots" allows the simple behaviors like insets allowing beyond the hudge technology, a small size to realize simple tasks". Dr. Emilio Alvarez demonstrate that big intelligent technologies not is possibly only in hudge machinaries of enormous industries, but also in a tiny scale to specific tasks.

According various researches like Latombe (1996), Victorino (1998) and Andrade (2001), the transportation area will be one of the most affected with the autonomous system, supplying perfection and more safe trajectories.

This project consists to go ahead systems as one of those authors that have innovated in the decade.

III. METHODOLOGY OF RESEARCH

The work utilizes the bibliographical model type to have a theorical basis and contextualize the Exploratorium field research, in this case, the industrial field, where was utilized a sampling.

To the elaboration of this research the data collecting was happened through questionaries, interviews and observations. To start from the bibliographical material search present in the internet have founded the necessary instruments to the project realization, being that the observation and the knowledges obtained during the research was essentials to provide enhancements in the project.

With the aid of professors, practical tests, theorical e computational was possible organize the steps of necessary data to the computational system of project.

The research methodology is compared with the materials contained in books, magazines and internet to verify the reality of development of the project, as well as your possibilities.

IV. TECHNICAL DEVELOPMENT

A. *Introduction*

The project consists in a Robot that can recognize the extern environment and be capable to realize movements to scape from possible blocks. For this reason, the Project is divided in three main steps: Sensors, Microcontroller and Movement, represented in the Fig. 1.

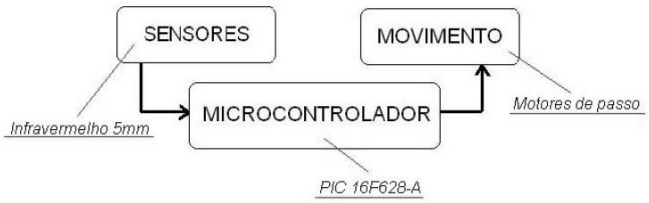


Fig. 1. Process

The sensors receive the information from extern environment sending to the microcontroller, that realize all the logic and repass the elements signal capable to generate movement and the tracking of Robot.

B. *Sensors*

The sensors elements work as a relation agent between the extern environment and the controller within all electrical section of Robot. The pair of infrared sensors are basically one LED (*light emissor diode*) that send infrared rays (emissor) and another led that receive the infrared ray (receptor) identifying and turning him in an electrical signal, passing through the circuit.

To have a correct recognition of extern environment, it is necessary that infrared ray emitted have a reflection in the obstacle and return to the system. This is realized through reflection principle according to the Fig. 2.

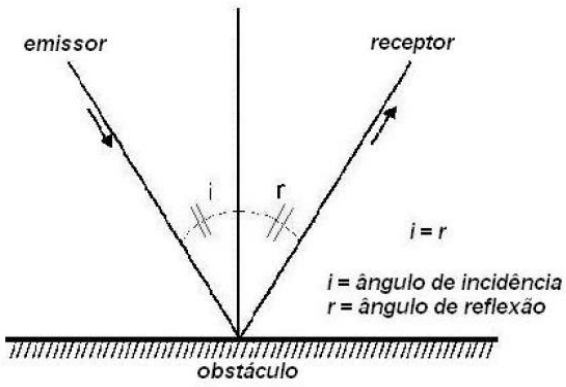


Fig. 2. Sensors of position

Where the infrared ray is emitted with a certain angle in relation to the vertical norm, founds an obstacle and reflects with the same angle in direction to the receptor. Due this reason, the sensor must be positioned in a correct manner in relation to the angles.

The sensor inside the circuit works as an indicator of 0 and 1 signals. While there are not obstacles, the tension will remain constant in 4,32 V (high logic level), sending 1 signal to the microcontroller, that requires to move forward normally. From near of an obstacle, the infrared begins to reflect, and the tension begins decrease to the circuit, till reach 0,2 V (high logical level), sending 0 signal to the microcontroller, that at this time gives the command to realize necessary scape movements. The Fig. 2. represents the relation between tension and distance from an object in the infrared sensors.

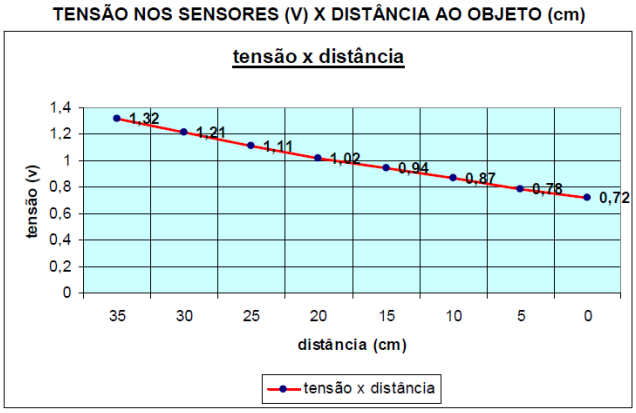


Fig. 3. Infrared sensor measurement

The Table 1 shows the values regarding to the Fig. 3.

TABLE I – VALUES FROM THE FIG. 3

Y (tension in sensors)	X (distance to the obstacle)
1,32	35
1,11	25
0,94	15
0,78	05
0,72	00

The Fig. 4. Indicates the tracking compared to tension.

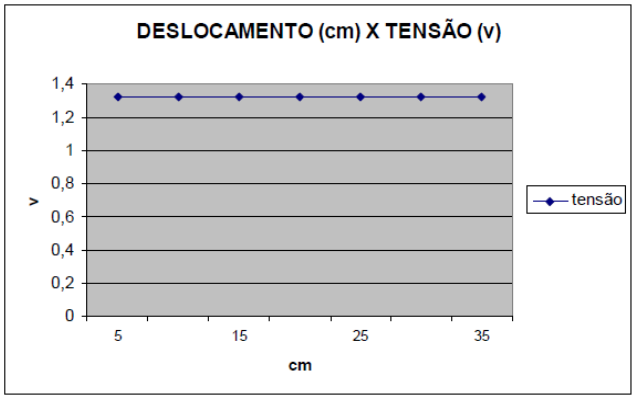


Fig. 4. Movement x Tension

While there are not obstacles, the tension will remain in 4,32 V (high logic level), sending 1 signal to the actuators. So, the Robot just only will go back to the inertia when the logical level is "low", sending 0 signal to the actuators. The Fig. 5. represents the infrared sensor positioned in the Robot.

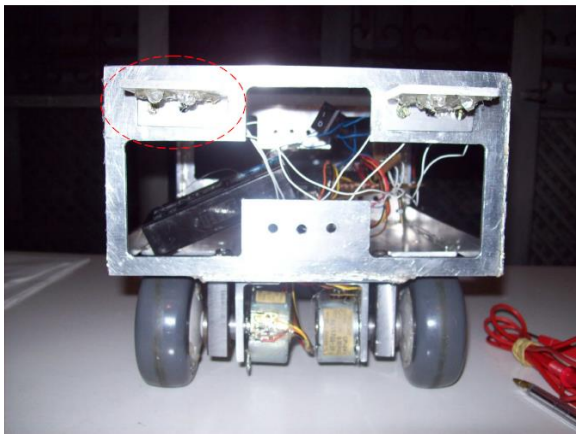


Fig. 5. Positioned sensors in Robot

The Fig. 6. represents a close up of infrared sensor.

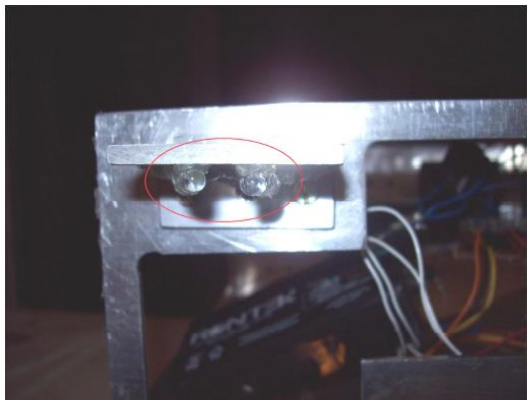


Fig. 6. Close up from the infrared sensor

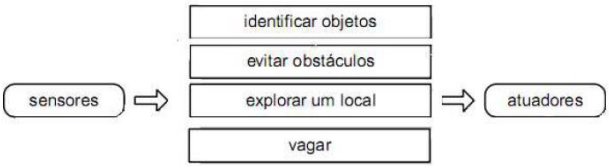
C. Controller

1) PIC 16F628A: By definition, a Controller is a device that works as the interface between system and human actions. There are various types of controllers, varying by necessity, quality, facility, size and other factors. Among the various types, was worked specially with the family PIC of microcontrollers. Specifying the PIC 16F628A, that represents the following advantages:

- Allows from 2048 words, while the majority represents only 1024;
- Have an internal oscillator;
- Reduced complexibility level;
- Apresents the CCP module (Capture, compare and PWM).

2) Programmation: This programmation system consists in the movement data processing that Robot will realize, and also give energetic autonomy and control; the locomotion is made through stepper motor based on 16F628A microcontroller. Where the main module (PIC) receive the signals from extern sensors and process them according to the implemented control strategy; in that way there is not necessary extern intervention, because the microcontroller takes the decisions from sended signals by the sensors, and inform to the stepper motor which will be the realized movement. To understand in a better way the program sintax we must generate the following knowledgement of basic working principle as shows the Fig. 7.

Fig. 7. Programmation process



Where the sensors makes a sweep from situated environment, while none object id detected the Robot must moving in straight line always when possible and keep to explore the location, but if some object is identified immediately, the Robot must realize an evasive movement and calculate a new route to keep moving on; in this case the actuators are two stepper motors that have the task to moving this autonomous Robot. Before the program development is essential the development of flow chart containing your functions, being in this way making easy the thinking line to initiate the programmation in C language due your simplicity, structure, great performance, and imperativity, making easy the development of any project.

The C language was developed in decade of 70's by Dennis Ritchie, while developing the UNIX operational system and have enthused other successor systems for example the language C++, that is a extension for C.

The flow chart consists in the beginning that is the header of the program, where we claim the PORT's A/B definitions. In the initialization of variables, we have to claim all the used variables, entrances and outputs: the initial processing address that acts directly in the programmation insert to the microcontroller.

Before the actuation of stepper motors, the system must verify and analyse the provided data by the sensors and the information interpreted by PIC face those received signals, to finally send the defined action to the stepper motors. The flow chart is just only a sketch of system that will be developed, the condition of traject that will be realized, the definitions and the objectives that must be reached in a fast and safety way, within of scientific patterns, must be highlighted in a comprehensive way for all those that have interest by studying the system, as shows the Fig. 8.

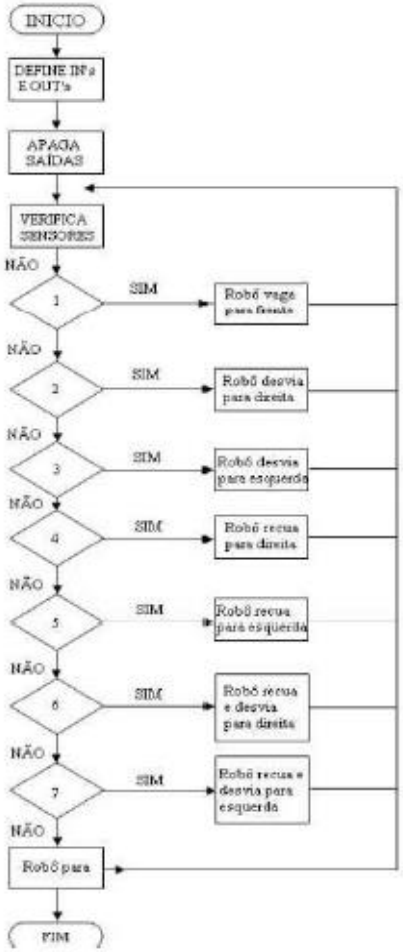


Fig. 8. Flow chart for programmation

Firstly is beginning with the program head, setting the utilized PIC in the memory storage, the processing address, and the Port's A/B. Following the C language patterns we have:

```
void main ()
{
trisa=1;
porta=0;
trisb=0;
portb=0;
```

Was defined the name and address of all variables that was used by the system, remembering that Port B (RB0 and RB7) was declared as output, because he will have the task to send the stimulated signals from PIC to the motors:

Stepper motor 1

```
#define portb.f0;
#define portb.f1;
#define portb.f2;
#define portb.f3;
```

Stepper motor 2

```
#define portb.f4;
#define portb.f5;
#define portb.f6;
#define portb.f7;
```

For that Robot keep activated while do not finishes the battery charge, was utilized the command *while(1)* that realize an infinite loop, even that the time of each routine finishes this command reset the system and reinitialize from each 20ms. The proposed logical for that the Robot move till found an obstacle, have by finality excites sequentially the windings of each stepper motor, in that way we can realize a clockwise direction or a counter-clockwise, depending only from the programmer; in this case the stepper motors due the clockwise movement to move forward. The algorithm showed in the Fig. 9. Indicates the 1st code.

```
while (1)
{

portb.F0=1;          portb.F0=0;
portb.F1=0;          portb.F1=0;
portb.F2=0;          portb.F2=1;
portb.F3=0;          portb.F3=0;
portb.F4=1;          portb.F4=0;
portb.F5=0;          portb.F5=0;
portb.F6=0;          portb.F6=1;
portb.F7=0;          portb.F7=0;
delay_ms(20);        delay_ms(20);

portb.F0=0;          portb.F0=0;
portb.F1=1;          portb.F1=0;
portb.F2=0;          portb.F2=0;
portb.F3=0;          portb.F3=1;
portb.F4=0;          portb.F4=0;
portb.F5=1;          portb.F5=0;
portb.F6=0;          portb.F6=0;
portb.F7=0;          portb.F7=1;
delay_ms(20);        delay_ms(20);
```

Fig. 9. 1st Algorithm utilized

Analysing this previous process was precepted that the execution of each routine has by duration 20ms, and

that *1* represents the alimentation of a winding regarding to RB (portb pin) that is located, and *0* is the absence of alimentation; was necessary four routines to form a movement cycle, once that those stepper motors have four windings, being feed in a sequential form with interruptions, because without interruptions the process would be turbulent and the activities would suffer a general black-out: not realize the clockwise movement neither counter-clockwise, pending to those two movements at the same time. The cited command just only is interrupted case a block is recognized, and in this way the PIC will do comparisons to how return and move in the same direction when possible, for that was developed more three situations of behaviour utilizing the command *if* to execute her in a precision way within a success probability of 99,99%, leaving 0,1% in function to the sensors by the actual status, if they are operating in a correct way.

The first step to operate the *if* command is measure through a multimeter the value of emmited signals by the sensors to PIC without the presence of an obstacle and with obstacle, respectively. If the signal is major than 1,32 V will be considered *1*, and if minor considered *0*. Remembering that *if* function works as a contact key to be activated in the necessary time to realize scape movement task. When we measure with a multimeter, we verify the presence of an obstacle when the signal was equal *1* and with block equal *0*. It means that the command will me called everytime that the received signal by the microcontroller is *0*, executing the route of scape referent to the position of the object located by the sensor.

The second step is develop each situation to start off from sensors, because there are connected to the PIC through Port A, being utilized the pin 17 (RA0) and the pin 1 (RA2); now we can tell that: "if the signal of two sensors was equal to *0*, the Robot stops and move to back, being that one stepper motor will do the clockwise movement and the other stepper motor will do the counter-clockwise movement. In that way, the robot move to back and direct himself to other direction at the same time, decreasing the formation of new cycles of control." It is interesting reset the whole Port B everytime that a routine is altered, was utilized the command *portb=0* bellow *if* as shows the 2nd Algorithm in the Fig. 10.

```
IF (porta.f0&porta.f2==0)

portb=0;
portb.F0=1;          portb.F0=0;
portb.F1=0;          portb.F1=0;
portb.F2=0;          portb.F2=1;
portb.F3=0;          portb.F3=0;
portb.F4=0;          portb.F4=0;
portb.F5=0;          portb.F5=1;
portb.F6=0;          portb.F6=0;
portb.F7=1;          portb.F7=0;
delay_ms(20);        delay_ms(20);

portb.F0=0;          portb.F0=0;
portb.F1=1;          portb.F1=0;
portb.F2=0;          portb.F2=0;
portb.F3=0;          portb.F3=1;
portb.F4=0;          portb.F4=1;
portb.F5=0;          portb.F5=0;
portb.F6=1;          portb.F6=0;
portb.F7=0;          portb.F7=0;
delay_ms(20);        delay_ms(20);
```

Fig. 10. 2nd Algorithm utilized

What we can conclude about the two routines that remains, is that makes the robot scape from right and to the left, the sintax of *if* command remains intact. We must only remove one of RA's because we will say: "if Port A F0 localize an object, the robot scape to the left"; this command serves just to the sensor of pin 17. The Fig. 11 shows the 3rd Algorithm utilized.

```
IF(porta.f0==0)

portb.F0=1;          portb.F0=0;
portb.F1=0;          portb.F1=0;
portb.F2=0;          portb.F2=1;
portb.F3=0;          portb.F3=0;
portb.F4=0;          portb.F4=0;
portb.F5=0;          portb.F5=1;
portb.F6=0;          portb.F6=0;
portb.F7=1;          portb.F7=0;
delay_ms(20);        delay_ms(20);

portb.F0=0;          portb.F0=0;
portb.F1=1;          portb.F1=0;
portb.F2=0;          portb.F2=0;
portb.F3=0;          portb.F3=1;
portb.F4=0;          portb.F4=1;
portb.F5=0;          portb.F5=0;
portb.F6=1;          portb.F6=0;
portb.F7=0;          portb.F7=0;
delay_ms(20);        delay_ms(20);
```

Fig. 11. 3rd Algorithm utilized

That is also acceptable for the pin 1 sensor, that will scape the Robot to the right when identify some object, but that is the ultimate process of the whole logical showed it is mandatory finalize this cycle within two keys } because it indicates that there is the end of the program and the *while* tie *1* must reinitialize the program and execute once again all the cycles, remembering that each one have your case to be executed, and that the cycle will be executed automatically case none block will be located (is that what makes the robot moving on). The Fig. 12 shows the 4th Algorithm to be used.

```
IF(porta.f2==0)
portb.F0=0;          |      portb.F0=0;
portb.F1=0;          |      portb.F1=1;
portb.F2=0;          |      portb.F2=0;
portb.F3=1;          |      portb.F3=0;
portb.F4=1;          |      portb.F4=0;
portb.F5=0;          |      portb.F5=0;
portb.F6=0;          |      portb.F6=1;
portb.F7=0;          |      portb.F7=0;
delay_ms(20);        |      delay_ms(20);

portb.F0=0;          |      portb.F0=1;
portb.F1=0;          |      portb.F1=0;
portb.F2=1;          |      portb.F2=0;
portb.F3=0;          |      portb.F3=0;
portb.F4=0;          |      portb.F4=0;
portb.F5=1;          |      portb.F5=0;
portb.F6=0;          |      portb.F6=0;
portb.F7=0;          |      portb.F7=1;
delay_ms(20);        |      delay_ms(20);
                     |
                     |      }
                     |      }
```

Fig. 12. 4th Algorithm used

3) Datasheet of 16F628A: The Fig. 13 indicates the datasheet of PIC 16F628A

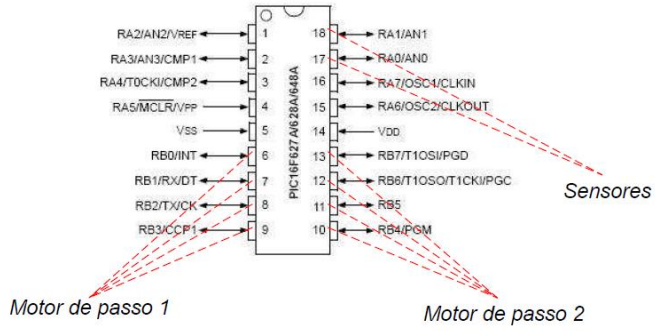


Fig. 13. Datasheet of microcontroller 16F628A

Fortunately through those steps was possible develop this program, where founded a lots of difficulties in some sectors, mainly in the Main Routine, where is contained all the probabilities of movements and how the stepper motors must react in some different situations. Also is valid the usage of manual of microcontroller PIC 16F628A, because him contains all the basic instructions about how must be utilized the systems inside of Assembler language without make unnecessary errors.

To develop this programmation was utilized as instrument of work the MPLAB software, a program responsible by the providing of a support to the programmers develops your own projects in Assembly. In that program is created any kind of program with different functions one of them inside of herself. The Fig. 14 shows the working area of MPLAB.

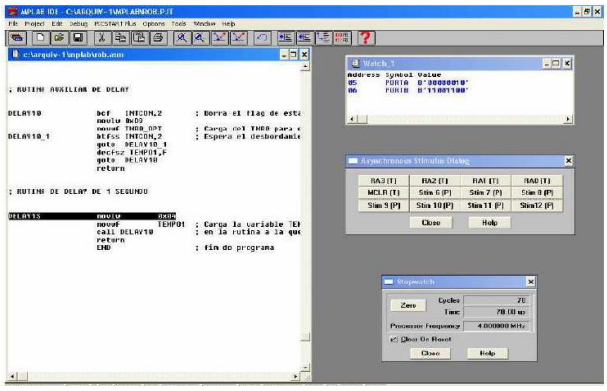


Fig. 14. MPLAB environment

The programmation was inserted, activating each port of PIC in a determined time, delimiting the action of the stepper motors and sensors while those goes to the direction of the object, that is the reason because the routines of programmation are of indeed importance to the whole success of the experience.

The programmation with the hexadecimal file generated in the MPLAB - -. HEX was inserted generating binary codes, in which are inserted in the PIC 16F84-A with a PIC recorder. In this process, was necessary utilize the program ICPROG that acts as intermediate between HEX files and PIC, for that can be possible the recording of those data in PIC. The Fig. 15. Appoints this process.

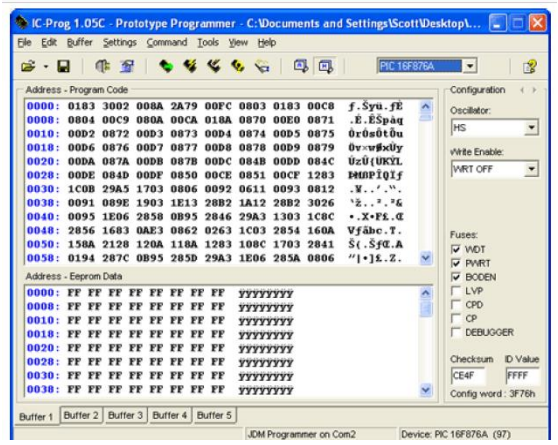


Fig. 15. PIC recording process

The PIC16F84-A was inserted through the SERIAL entrance in a microcomputer for that "machine codes" would be recorded in your memory storage of 1024 words, 68 bytes of RAM memory and 68 bytes of EEPROM storage.

D. Movement

After the extern recognition made by sensor and the logic realized by the controller, comes the signal to the

movement components. The selected stepper motors was about 10V, that in the next steps we will look better your working principle. Besides, the signal sent by the controller it's about 5V only, making that the stepper motor not works with all the necessary tension. For that was proposed the scheme indicated in the Fig. 16.

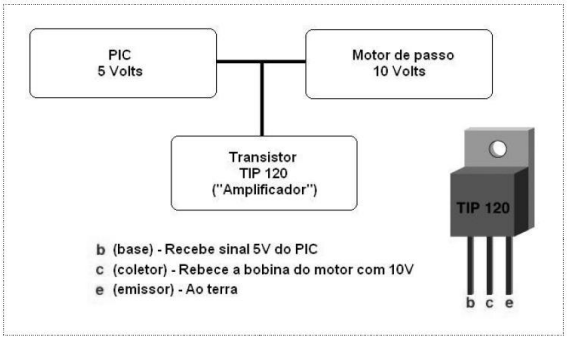


Fig. 16. Transistor demonstration

To the resolution of stepper motor was imposed a transistor (TIP 120), that will works as an "amplifier" and also as a key function, releasing the tension just only when receives the signal from controller.

A) Transistors: The transistor is the electronic component responsible to the amplify the current in the circuit region related to the output of the microcontroller through the flux control of current between your semiconductors materials, being this process essential to the stepper motor working.

In the utilization of this component is necessary to know the associated logic to the polarity of your elements, being those resumed in collector (receive the current), basis (controls the current) and emissor (responsible to the current supplying to the stepper motor).

There are two basic types of transistors, being those NPN or PNP. The main difference between him is the flux direction of current, being the direction of current in a NPN component inverse to the PNP component. It means that for a NPN entry in conduction we must send logic levels *1* to the basis, but for that a PNP we must send logic levels equal to *0*. The Fig. 17. looks at the difference between a PNP and NPN transistor.

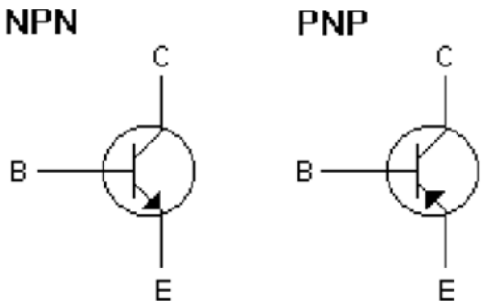


Fig. 17. Difference between NPN and PNP

The transistor represents the amplificatory and a key functions. In your amplificatory function, your objective is to caught the 5V signal and send him up to 10V to the stepper motor, as appoints the electrical circuit design in the Fig. 18.

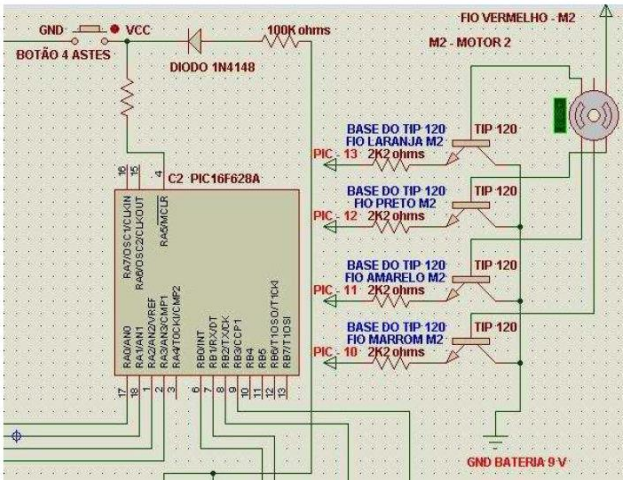


Fig. 18. Electrical circuit design

A very important term when we talk about "amplifier" of transistor is the gain of this component, it means the capacity of this transistor to increase the strength of a signal without promote significative alterations in him.

The second function of the transistor, key system, servers to select the correct pulses from stepper motor order and do not leave the tension of 10V "entries" directly in all the windings. Just when the transistor receive the signal from controller that will close the contact and release each pulse individually to the stepper motor. The transistor utilized in this project was the TIP 120, from NPN type.

B) Motors: When we desire exact movements, necessary to the scapes and perfect movements, is recommended utilize an electrical motor of stepper motor type. In this kind of motor, the rotation is controlled by a serie of electromagnetic fields, abled and disabled, where we say pulses. The Fig. 19. shows the electrical diagram.

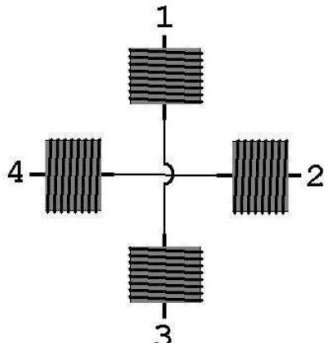


Fig. 19. Stepper motor representation

During this project, was analysed various other types of motors, and the more viable to the employment was the stepper motor. Because the capacity of angles calculation, exact dimensions and moving according to the escapes that the Robot presents.

V. CONCLUSIONS (USE STYLE JMAIT HEADING 1)

Was possible, to start off the elaboration of this project, obtain solutions to the questions related to comprehension of an efficient loader transportation system, being that to achieve the objectives resumed in the application of the concept of movement in function of automation and assure the major safety and agility in the process with a less expensive cost in comparison to a conventional system, was necessary to leave open an ample space to the modifications that would come to be realized with the intuit of to adapt the Robot in a serie of new functions, for example, the usage of cameras that allows the tracking of areas that provide risk to the humans, having in this flexibility offered the opportunity to expose our knowledgement acquired to the public utilization related to the industrial sector.

ACKNOWLEDGMENT

I am thankful by God,
By my family,
By my friends.

REFERENCES

[1] BOLTON, Bill. *Análise de circuitos elétricos*. Makron Books, 1st edition, São Paulo-SP, 1995.

[2] SEDRA, Smith. *Microeletrônica*. Makron Books, 4th edition, São Paulo-SP, 2000.

[3] SOUZA, David José de. *Desbravando o PIC 16F628 "Ampliando e Atualizando"*. Érica, 7th edition, Tatuapé-SP, 1971.

[4] ZANCO, Wagner Silva. *Microcontroladores PIC 16F628A "Uma abordagem prática e objetiva"*. Érica, 1st edition, Tatuapé-SP, 1969.