

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
CURSO DE ENGENHARIA DE COMPUTAÇÃO
MICROPROCESSADORES I**

**PROJETO INTEGRADO
Laut**

**CURITIBA
2009**

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
CURSO DE ENGENHARIA DE COMPUTAÇÃO
MICROPROCESSADORES I**

**Marcelo Fernandes
Ricardo Queiros
Rodrigo Kotlevski**

**PROJETO INTEGRADO
Laut**

**Projeto entregue ao professor Afonso
Ferreira Miguel, da disciplina de
Microprocessadores I, do curso de
Engenharia de Computação da Pontifícia
Universidade Católica do Paraná, com o
objetivo de demonstrar os conhecimentos
adquiridos durante o semestre e agregar a
eles novos, de maneira a aplicá-los de
maneira prática através de um projeto
desenvolvido pelos alunos.**

**CURITIBA
2009**

SUMÁRIO

Fundamentação Teórica.....	4
1. Introdução.....	5
1.1 Justificativas.....	5
1.2 Metodologia.....	6
1.3 As responsabilidades.....	6
2. Objetivos.....	7
3. Não está Incluso no Escopo deste Projeto.....	7
4. O Projeto.....	7
5. Os resultados esperados.....	8
6. Resultados obtidos.....	8
7. Materiais.....	8
8. Esquemático do Projeto.....	8
9. Conclusões.....	9
10. Anexos.....	9

FUNDAMENTAÇÃO TEÓRICA

ARDUINO

Arduino, por vezes traduzida ao português como Arduíno, é um computador físico baseado numa simples plataforma de hardware livre, projetada com um microcontrolador de placa única, com suporte de entrada/saída embutido e uma linguagem de programação padrão, na qual tem origem em Wiring, e é essencialmente C/C++. O objetivo do projeto é criar ferramentas que são acessíveis, com baixo custo, flexíveis e fáceis de se usar por artistas e amadores. Principalmente para aqueles que não teriam alcance aos controladores mais sofisticados e de ferramentas mais complicadas.

Pode ser usado para o desenvolvimento de independentes objetos interativos, ou ainda para ser conectado a um computador hospedeiro. Uma típica placa Arduino é composta por um controlador, algumas linhas de E/S digital e analógica, além de uma interface serial ou USB, para interligar-se ao hospedeiro, que é usado para o programar e o interagir em tempo real. Ele em si não possui qualquer recurso de rede, porém é comum combinar um ou mais Arduinos deste modo, usando extensões apropriadas chamadas de shield. A interface do hospedeiro é simples, podendo ser escrita em várias linguagens. A mais popular é a Processing, mas outras que podem comunicar-se com a conexão serial são: Max/MSP, Pure Data, SuperCollider, ActionScript e Java.

Atualmente, seu hardware é feito através de um microcontrolador Atmel AVR, sendo que este não é um requerimento formal e pode ser estendido se tanto ele quanto a ferramenta alternativa suportarem a linguagem Arduino e forem aceitas por seu projeto. Considerando esta característica, muitos projetos paralelos se inspiram em cópias modificadas com placas de expansões, e acabam recebendo seus próprios nomes.

Apesar do sistema poder ser montado pelo próprio usuário, os mantenedores atualmente possuem um serviço de venda do produto pré-montado, através deles próprios e também por distribuidores oficiais com pontos de venda mundiais.

O projeto iniciou-se na cidade de Ivrea, Itália, em 2005, com o intuito de interagir em projetos escolares de forma a ter um orçamento menor que outros sistemas de prototipagem disponíveis naquela época. Seu sucesso foi sinalizado com o recebimento de uma menção honrosa na categoria Comunidades Digitais em 2006, pela Prix Ars Electronica, além da marca de mais de 50.000 placas vendidas até outubro de 2008.

1. Introdução

A idéia consiste no desenvolvimento de um robô de limpeza automática que tem por objetivo a utilização de sensores para o desvio de obstáculos e motores para a movimentação.

A idéia foi elaborada pelo aluno Ricardo Queiroz e serão desenvolvidos pelos alunos Ricardo Queiroz, Rodrigo Kotlevski e Marcelo Fernandes do curso de Engenharia de Computação da Pontifícia Universidade Católica do Paraná (PUC-PR).

O nome do projeto surgiu pela união das palavras Limpeza (L) e automática (aut) formando assim a palavra Laut.

1.1 Justificativas

As vantagens do Laut estão na praticidade e utilidade e tem por objetivo o desenvolvimento de um robô de limpeza de chão automático.

Em meio à correria do dia a dia, serviços autônomos que tem por objetivo facilitar a vida das pessoas se tornam cada vez mais úteis.

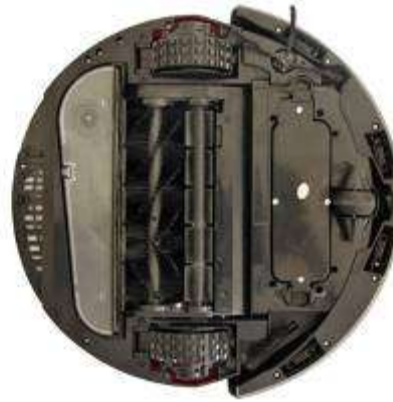
Uns dos exemplos similares existentes no mercado é o Aspirador de Pó Robo RC3000 Karcher e o iCleaner robô aspirador.



1. Aspirador de Pó Robo RC3000 Karcher



2. iCleaner visão superior



3. iCleaner visão frontal

O iCleaner e o RC3000 da Karcher consiste em um robô aspirador que circula pela casa aspirando a sujeira sozinho e possui sensores de desvio de obstáculos.

O Laut é um produto que tem as características de limpeza de uma vassoura mágica, mas que possui autonomia na limpeza devido a sua movimentação automática e detecção de obstáculos.

1.2 Metodologia

O projeto será desenvolvido usando como estrutura uma vassoura mágica sem cabo placas de circuitos impresso, motores, sensores, computador para programação, microcontrolador. Softwares: eagle para criação das placas de circuito impresso.

O “cérebro” da Laut e o microcontrolador Arduino, que interpreta os sinais recebidos dos sensores de proximidade, os motores são necessários para a movimentação da Laut. A ferramenta utilizada neste projeto foi à linguagem C/C++, em que e muito utilizada no curso.

Foi necessário dois servos motores para a movimentação e o torque da Laut. Além disso, foi necessário sensor de proximidade para desviar de obstáculos.

1.3 Responsabilidades

Na primeira fase do projeto, os integrantes foram responsáveis pela procura da vassoura mágica, pois ela seria a base do nosso projeto e pelo estudo do mesmo para saber um

pouco mais sobre como funciona a vassoura mágica em si e também o robô aspirador, onde foi a base para este projeto.

Na segunda etapa, foi proposto para os alunos Ricardo e Rodrigo a procura dos componentes eletrônicos que iríamos precisar para a parte do circuito elétrico do projeto.

Tendo em mãos todos os equipamentos necessários, a divisão de tarefas ficou: o aluno Ricardo ficou responsável pela estrutura, pela programação do projeto e pelos motores, o aluno Rodrigo ficou responsável pelo circuito elétrico e pelos sensores, e o aluno Marcelo ficou responsável pela documentação e auxílio do projeto.

A PUC tem por responsabilidade disponibilizar laboratório para apresentação do projeto e o auxílio com ferramentas e laboratórios.

Os professores têm por responsabilidades envolvidas o aprendizado e orientação que tornam possível o desenvolvimento de projetos acadêmicos.

2. Objetivo

O objetivo deste projeto foi confeccionar um robô autônomo capaz de desviar de obstáculos e limpar o chão.

Academicamente, haverá um aprimoramento no estudo e implementação de servos motores, no estudo de sensores e aplicação prática de um novo equipamento, onde ficou mais fácil para a conclusão deste projeto.

Todos os objetivos foram alcançados, mas durante a implementação do projeto houve alguns problemas com os circuitos elétricos, e os sensores de proximidades, mas com muito trabalho todos esses problemas foram resolvidos.

3. Não está Incluso no Escopo deste Projeto

- Este projeto não será capaz de detecção de queda de desnível do chão (exemplo: escada)

4. O projeto

O projeto consistirá em um robô autônomo de limpeza de chão que irá conter sensores que permitirão o desvio de obstáculos.

O projeto terá os módulos, movimentação que consistirá em motores que permitirão a locomoção do objeto, e o módulo de desvio de obstáculos que permitirão que o robô não pare de se locomover devido a objetos no caminho. O robô será autônomo e utilizará um

microcontrolador para que isso seja possível. A estrutura será usada de uma vassoura mágica sem cabo.

5. Os resultados esperados

Como resultados deste projeto, serão apresentados aos professores os seguintes itens / funcionalidades:

1. Circuito motor 1;
2. Circuito motor2;
3. Circuito do microcontrolador;
4. Programação motores;
5. Circuito sensor1;
6. Circuito sensor2;
7. Programação sensores;
8. Vídeo
9. Documentação do projeto dos itens acima.

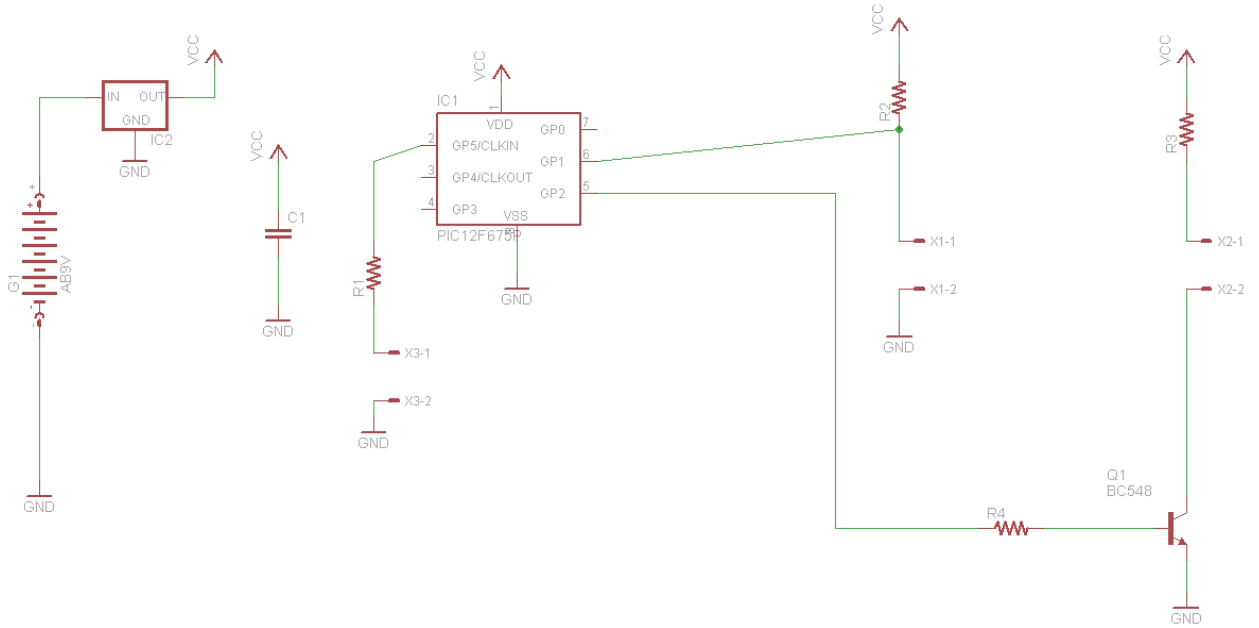
6. Resultados obtidos

- Circuitos dos motores e dos sensores funcionando.
- Programação funcionando para os motores e sensores
- Sensores de proximidade calculam a distancia de qualquer objeto evitando assim a colisão.
- Servo motores dando o torque necessário para que a Laut comece a limpar o chão.

7. Materiais

- Vassoura mágica
- Dois servos motores com torque de 6,5 kg
- Sensores de proximidade infravermelho com PIC 12F675
- Arduino

8. Esquemático do Projeto



9. Conclusão

O presente projeto contribuiu para o enriquecimento intelectual dos integrantes através do estudo de referências e técnicas utilizadas no campo de microprocessadores, controle de servo e utilização de uma nova tecnologia que é a do Arduino. Este projeto foi criado para facilitar na hora da limpeza de casa, onde num será necessário muito esforço físico e também deixar um tempo extra para resolver outras necessidades.

Houve alguns problemas ao decorrer da construção e implementação deste projeto, mas todos estes problemas foram resolvidos sem muita dificuldade.

Inicialmente o projeto seria feito com um microcontrolador AT89S52, mas com falta de tempo o professor Afonso demonstrou o hardware Arduino, onde poderíamos programar na linguagem C/C++.

10. Anexos



Arduino funcionando com o servo motor.



Estrutura montada junto com o circuito elétrico e os servos motores

```
#include <Servo.h>
```

```
Servo myservo1; // create servo object to control a servo
```

```
Servo myservo2; // a maximum of eight servo objects can be created
```

```
int pos=0;
```

```
const int sensorPin = 2;
```

```
int buttonState = 0; //estado atual do botão
```

```
void setup()
{
  myservo1.attach(9); // attaches the servo on pin 9 to the servo object
  myservo2.attach(11);
  pinMode(sensorPin , INPUT);
}
```

```
void loop()
{
  buttonState = digitalRead(sensorPin);

  for (pos=0;buttonState == LOW;pos++){

    buttonState = digitalRead(sensorPin);

    myservo1.write(45);
    myservo2.write(405);
    delay(15);
  }

  for(pos=0; pos != 70 ;pos++)
  {
    myservo1.write(45);
    myservo2.write(45);
    delay(15);
  }
}
```