



# Revista EaD & tecnologias digitais na educação

## ***Praxedes: Protótipo de Um Kit Educacional de Robótica Baseado na Plataforma Arduino***

Francisco Ioneiton da Silva, Daniel Scherer

Universidade Estadual da Paraíba (UEPB)

*neitonfsilva@gmail.com, professorscherer@gmail.com*

*Resumo. Este artigo descreve a criação de um protótipo de um robô móvel, com finalidade educacional utilizando a plataforma Arduino, tendo como principal finalidade simplificar o uso desta plataforma, eliminando a necessidade de conhecimentos complexos em determinados níveis de ensino como bem como de oferecer uma alternativa a os kits disponíveis no mercado com custos relativamente acessíveis.*

*Palavras-chave. Robótica Educacional, Arduino, Robótica de Baixo custo.*

*Abstract. This article describes the creation of a prototype of a mobile robot for educational purposes using the Arduino platform, with the primary purpose to simplify the use of this platform, eliminating the need for complex knowledge in certain levels of education as well as to offer an alternative to the kits available in the market, with relatively affordable costs.*

*Keyword. Educational Robotics, Arduino, Robotics Low Cost.*

### **I. Introdução**

A robótica tem se difundido e se destacado nos últimos anos principalmente por sua inserção em diferentes setores da vida humana. Começando pela fantasia na literatura e filmes, seguindo para aplicações reais tais como: robôs manipuladores na indústria; robôs usados em tarefas domésticas (PIRES, 2002); robôs enfermeiros (ex.: Robô RX desenvolvidos pela empresa norte americana McKesson (MCKESSON, 2012)); até robôs exploradores (ex.: programa Mars Exploration Rovers (MARSROVERS, 2012)).

Outra aplicação da robótica está no campo educacional. Segundo Siebra e Lino (2010) a robótica se apresenta como um recurso que pode auxiliar no processo de ensino/aprendizagem. Seymour Papert, um dos precursores da robótica educativa, ao criar a linguagem Logo buscou com ela proporcionar uma forma de ensino motivadora (Solomon e Papert, apud SOARES e BORGES, 2011). Papert, além da linguagem de programação LOGO, também criou a tartaruga robótica Yellow Turtle. A

tartaruga respondia a comandos de andar, girar, além de possibilitar deixar um rastro de caneta por onde passava (PEREIRA, 2008). Observa-se uma preocupação de Papert da relação entre LOGO e a Yellow Turtle, visando expandir as capacidades da linguagem LOGO e possibilitar a relação entre o abstrato da programação do LOGO com o concreto da ação da tartaruga. Na década de 1980, Papert ainda seria responsável pelo projeto LEGO-LOGO, que objetivava inserir sensores, motores e engrenagens aos blocos encaixáveis da empresa LEGO (PEREIRA, 2008), que por sua vez, foi a base para os Kits da linha de robótica educativa LEGO Mindstorms.

A utilização da robótica pode focar-se na montagem de dispositivos, programação e trabalhar conceitos ligados a matérias curriculares tais como física e matemática. Segundo Siebra e Lino (2010), proporciona o amadurecimento destes conceitos uma vez que os alunos precisam coloca-los em pratica. Trabalhando a parte de programação, a robótica torna-se uma ferramenta poderosa na construção do raciocínio lógico dos alunos, para Castilho (2002) a programação de robôs é apoiada pela necessidade dos alunos em formalizar uma solução para um problema por ele diagnosticado. Esta vertente de uso da robótica é muito útil para introdução de conceitos computacionais como criação de algoritmos, segundo Lopes e Fagundes, citados por Filho e Gonçalves (2008) estudos comprovam que atividades de programação, depuração e design levam ao enriquecimento dos esquemas cognitivos com novos esquemas de representação lógico-matemático.

Kits de robótica educacional são essenciais para o trabalho em sala de aula, estes podem ser encontrados no mercado. De acordo com (SILVA, F. I. e SCHERER, D. 2012) existe uma grande variedade de Kits produzidos por diferentes empresas que podem ser usados para fins educacionais (ex.: Lego Mindstorms (LEGO MINDSTORMS, 2012), Lynxmotion (LYNXMOTION, 2012), e Modelix (MODELIX, 2012)) como exposto por Silva, F. I. e Scherer, D. (2012) estes Kits podem ser compostos por:

Peças de hardware:

- Componentes estruturais: São as partes que compõem a estrutura física dos Kits, este conjunto engloba manipuladores (braços e garras), rodas, e bases de fixação para outras partes.
- Componentes eletrônicos: Possibilitam a adição de recursos de movimento (motores) e permitem a interação do robô com o meio (dispositivos de som, dispositivos visuais e sensores).
- Unidade programável: Os microcontroladores estão inseridos nesta categoria, esta é a parte central de um robô, que permite incorporar um programa a ser executado.

Software:

- Linguagem de programação textual: Este tipo de linguagem abrange tanto as línguas tradicionais (Delphi, Java ou C), como linguagens proprietárias, como a utilizada nos produtos Arduino (ARDUINO, 2012).
- Linguagem de programação gráfica: Linguagens cuja construção de programas, baseia-se em arrastar e soltar os ícones e outros elementos gráficos, como exemplo temos a interface do Lego Mindstorms (LEGO MINDSTORMS, 2012).

Material de Apoio:

- Material de apoio pedagógico: Material de apoio ao professor, tradicionalmente contendo exemplos de projetos que podem ser desenvolvido com cada Kit.
- Manual do Usuário: Material tradicional que mostra a relação de peças disponíveis no Kit, bem como as instruções para instalação e manipulação do software.
- Documentação técnica: Material que abrange de dados técnicos, geralmente destinado a permitir a construção de componentes extras ou inserir implementações mais avançada como a criação de bibliotecas.

Além dos Kits, existem plataformas de hardware (Arduino (ARDUINO, 2012) e Maxwell Bohr (MAXWELL BOHR, 2012)) que podem ser utilizadas nos trabalhos com robótica educacional. Estas plataformas são alternativas de custo acessível, pois possuem unidade de programação básica e um padrão para a adição de outros componentes eletrônicos que possibilita o uso de materiais de sucata. Entretanto, essas plataformas exigem maiores conhecimentos de eletrônica (SILVA, F. I. e SCHERER, D. 2012).

O trabalho aqui exposto é a construção do protótipo de Kit Educacional Praxedes. Baseado na plataforma Arduino, utilizando-se de componentes eletrônicos compatíveis com a placa e de fácil aquisição no mercado, buscou-se simplificar a aplicação dessa plataforma nos trabalhos de robótica educacional. Para a programação do Kit, utilizou-se a linguagem nativa do Arduino.

As seguintes etapas foram adotadas no processo de desenvolvimento do protótipo: definição das funcionalidades do robô; escolha dos componentes e projeto do chassi; aquisição dos componentes; montagem do Kit e testes.

Na próxima seção tem-se uma pequena contextualização acerca do tema deste trabalho. Na seção 3 são apresentados os componentes que fazem parte do Kit Praxedes. Na seção 4 são apresentados os testes realizados. Na seção 5 são apresentados os custos e a lista total de componentes necessários para o Kit Praxedes. Após, tem-se a seção 6 onde são discutidos os resultados encontrados, seguidos dos trabalhos futuros.

## 2. Fundamentação Teórica

A plataforma Arduino não foi concebida para robótica educacional, portanto, não oferece oficialmente um Kit desenvolvido para essa natureza, porém no mercado nacional, são comercializados kits para Arduino compostos por componentes eletrônicos e separados por níveis que vão do iniciante como o Kit disponível no site da empresa Multilogica (Figura 1) (MULTILOGICA-SHOP, 2013), até o nível avançado. Estes kits não permitem, desenvolver atividades com robótica em sala devido a complexidade, da exigência de conhecimentos em eletrônica básica, embora reduzidos em sua composição estes produtos podem chegar a custar até R\$ 199,99 (ROBOCORE, 2013).

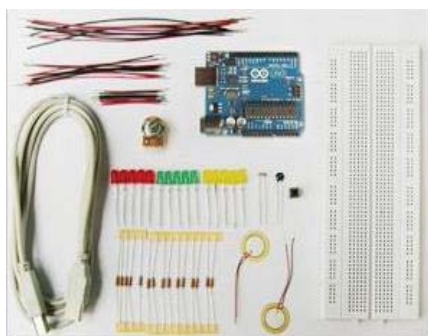


Figura 1: Kits comercial Arduino disponível no mercado

Fonte: <http://www.multilogicashop.com>

## 2.1. Praxedes Protótipo de Robô educacional

Inicialmente, o protótipo Praxedes contemplara duas funcionalidades básicas; seguidor de trilha e desviar de obstáculos, sendo possível desenvolver atividades com foco na abordagem resolução de labirintos entre outros problemas como competições de Robôs de resgate.

Optou-se pela plataforma Arduino pela sua robustez e versatilidade, além de ter uma gama de componentes que podem ser agregados facilmente à placa e ampliando suas capacidades. A versão da placa utilizada neste trabalho foi a Arduino Duemilanove (Figura 2<sup>a</sup>). A construção do Kit baseia-se em um robô móvel composto por duas rodas frontais de borracha (Figura 2b), uma terceira roda traseira (roda “boba”) modelo ball caster da Pololu (Figura 2c) dois motores de corrente contínua com caixa de redução, que são controlados por meio de uma ponte H, um sensor ultrassônico modelo HC-SR04 e um sensor de luz que permite o robô seguir uma linha de cor escura em uma superfície clara, estes componentes foram conectados por meio de uma mini Protoboard (Figura 2d) fixada na frente do Chassi superior .

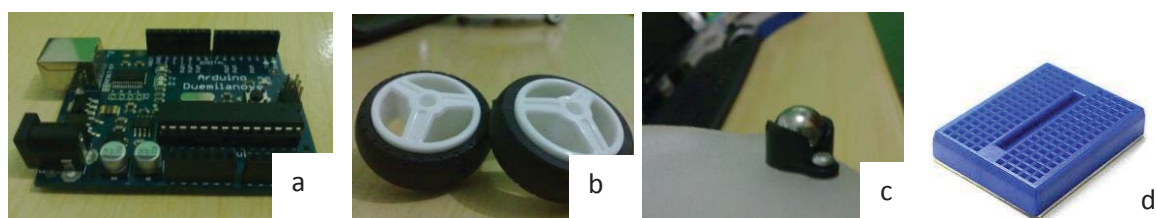


Figura 2: Arduino Duemilanove (a), Rodas (b), terceira roda (c) e mini protoboard (d)

Fonte: Figuras (a, b e c) Autoria própria, figura mini protoboard <http://elecbreaks.com>

## 3. Componentes e Montagem do Kit

### 3.1 Motores de corrente contínua

Os motores adotados no projeto são motores do tipo DC ou motores de corrente contínua, que como sua nomenclatura sugere funcionam com fluxo de corrente elétrica contínua. Estes motores se baseiam em princípios como atração e repulsão

de polos magnéticos, além de fluxo magnético e de indução de tensão elétrica<sup>3</sup>. Jones et al apud Gioppo et al (2009) afirma que este tipo de motor é comum em robôs móveis, devido a alimentação destes robôs se darem por meio de baterias.

Motores do tipo DC, como descrito por Gioppo et al (2009), giram a uma velocidade muito alta mas fornecem um torque baixo. Para inverter esta relação, estes motores devem ser ligados a uma caixa de redução, que consiste em um conjunto de engrenagens combinadas a fim de aumentar o torque do motor e reduzir sua velocidade de giro. Alguns motores DC são vendidos com caixas de redução acopladas. Os motores escolhidos para este projeto foram os motores duplos da Tamiya (Figura 3) que possuem caixa de redução.



Figura 3: Kit motores com caixa de redução

Fonte: Autoria Própria

Este conjunto de motores, dependendo de sua montagem, pode operar de forma conjunta ou separadamente, cada um dos motores opera com uma tensão de 3 volts e sua corrente pode variar de 150mA a 2100mA (ROBOCORE, 2012).

### 3.2. Ponte H

Para o acionamento e controle dos motores DC a partir de sinais gerados por um microcontrolador, utiliza-se um circuito conhecido como ponte H (Figura 4a) para controlar o sentido da corrente que passa pelos motores. Invertendo-se a corrente inverte-se com isso o sentido da rotação dos motores. Este controle é feito por meio de quatro chaves posicionadas ao lado do motor, esta configuração lembra o formato da letra H, daí então o nome do circuito (GIOPPO et al, 2009).

A ponte H também atua como circuito protetor para o microcontrolador, visto que as portas digitais do Arduino fornecem até 40 mA (ARDUINO, 2012), enquanto motores DC consomem valores superiores a este. Os motores adotados neste projeto chegam a consumir 2100 mA. Se ligados direto a uma porta digital do Arduino um motor DC pode danificar a placa (MCROBERTS, 2011). Para solucionar este impasse há a necessidade de se adicionar alimentação externa para os motores, fornecida por meio de uma bateria de 9 volts.

A ponte H pode ser construída utilizando-se chaves, reles ou transistores (MAXWELL BOHR, 2012). Porém, a fim de simplificar a montagem do circuito da ponte H, utilizamos neste projeto o CI L293D (Figura 4b). Este CI atua como uma ponte H dupla. No lugar de chaves, os CI's controladores de motores como o L293D utilizam transistores para inverter o sentido da corrente (MCROBERTS, 2011). O esquema da montagem dos motores utilizando o CI L293D encontra-se na figura 4 (c).

<sup>3</sup> <http://nilsonmori.blogspot.com.br/2011/05/principio-basico-de-funcionamento-de-um.html>

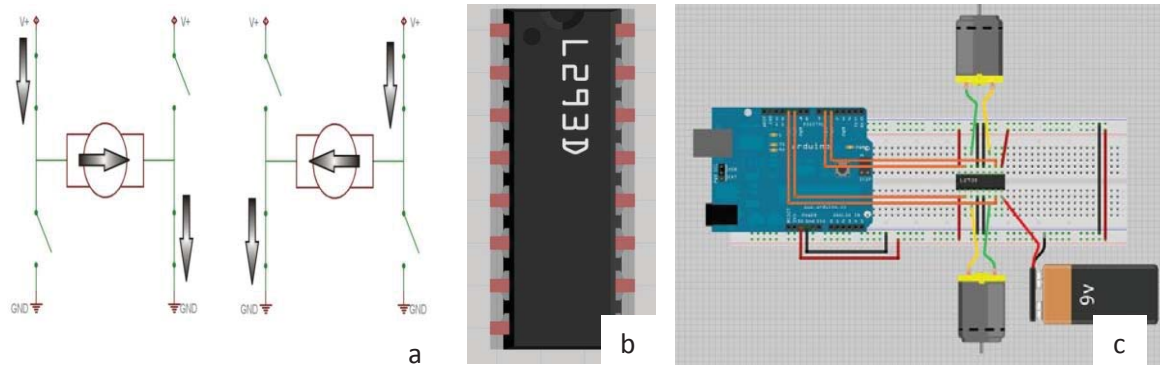


Figura 4: Esquema de uma ponte H(a) (PATSKO, 2006 apud GIPPO et al, 2009),

CI L293D (b), montagem da ponte H e motores (c)

### 3.3. Sensor Ultrassônico HC-SR04

O sensor ultrassônico modelo HC-SR04 (Figura 5a), permite a detecção de objetos a distancia mínima de 2 cm e máxima de até 4 metros, com precisão de até 3mm (SATATIS TRONICS, 2012).

O HC-SR04 possui 4 (quatro) pinos: Ground (Terra), VCC (5 volts), Trigger e Echo. A detecção de objetos se dá por meio do envio de um sinal ultrassônico que é refletido no objeto e retorna ao sensor. Este sinal, quando captado, permite calcular a distancia até o objeto a partir do tempo de ida e volta do sinal (FRISTEC, 2012) (Figura 5b).

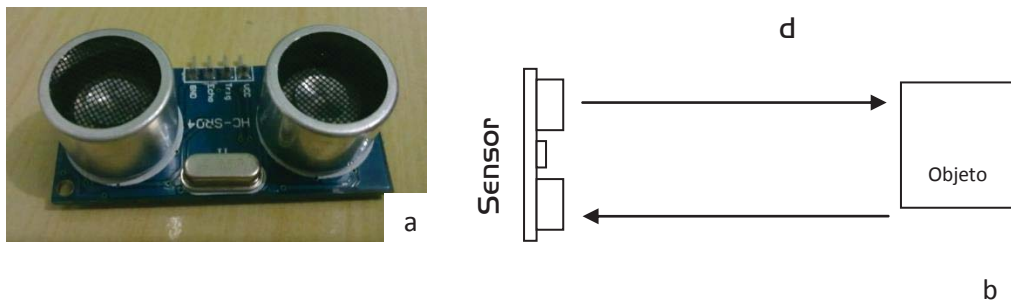


Figura 5: Sensor Ultrassônico HC-SR04 (a), Esquema de funcionamento do sensor (b)

Fonte:(a) Autoria Própria, (b) Adaptado de Fristec (2012)

A distancia é calculada levando-se em conta a velocidade do sinal ultrassônico que é de aproximadamente 340 m/s no ar (SATATIS TRONICS, 2012). Quando o sinal emitido encontra o objeto temos então à distância (d) para este objeto, porém a medição só é feita quando o sinal retorna ao sensor. Sendo assim, duas vezes a distancia (2d), que corresponde ao tempo de ida e volta do sinal ao sensor (FRISTEC, 2012). Desta forma o calculo da distancia obedece a Equação 1.

$$\text{Velocidade} = \frac{\text{Distância}}{\text{Tempo}} \rightarrow v = \frac{2d}{t} \rightarrow d = \frac{v \cdot t}{2}$$

Equação I

Fonte: <http://fristec.blogspot.com.br/2011/01/14-aplicacao-sensor-de-distancia-hc.html>

O sensor possui uma biblioteca desenvolvida para o Arduino com a finalidade de facilitar a sua utilização, porém o ambiente de programação do Arduino apresentou problemas com a biblioteca do sensor. Sendo assim, os cálculos para obtenção da distancia para objetos foram implementados sem o auxílio deste recurso.

Com base na documentação técnica<sup>4</sup> do sensor, observou-se que para se realizar a leitura de uma determinada distancia, o pino Trigger deve ser posto em nível alto (receber 5 volts) por 10 milissegundos, e colocado em nível baixo (0 volts) logo em seguida, com isso 8 pulsos de 40kHz são emitidos. Quando o sinal é retornado, o pino Echo gera um sinal de nível alto (5 volts) proporcional a distancia do sensor em relação ao objeto. Podendo-se então, utilizar o tempo que o pino Echo permaneceu em nível alto e fazer uso da Equação I, onde a Distancia seria igual à Velocidade do sinal ultrassônico (340 m/s) multiplicado pelo tempo que o pino Echo esteve em nível alto, dividido por 2 (dois), que representa a distancia de ida e volta do sinal ao sensor. Com essas informações foi possível implementar uma função para uso do sensor.

A função deste sensor é detectar possíveis obstáculos no ambiente. Baseado nos dados obtidos com o HC-SR04 e com a devida programação, o robô deve tomar decisões em relação a sua trajetória baseadas na existência ou não de obstáculos a distancias menores que 4 cm durante o percurso.

### 3.4. Sensor de Luz

Sensores de luz são utilizados em vários projetos na robótica, uma vez que este tipo de sensor é muito útil para construção de robôs seguidores de linha. Um desafio clássico da robótica consiste em completar um circuito seguindo uma linha preta em um fundo branco ou vice versa (SIEBRA e LINO, 2010).

Para este protótipo optou-se pela construção de um sensor de luz utilizando componentes eletrônicos simples como LED's, Resistores e LDR's.

O componente chave deste sensor é o LDR (Light Dependent Resistor) – resistor dependente de luz ou fotoresistor. O LDR possui dois terminais e cada um desses se conecta com um eletrodo. Entre os eletrodos está o fotocondutor (MCROBERTS, 2011, p. 117). Quando a luz atinge o fotocondutor o LDR perde sua resistência, com isso mais corrente fluirá entre os eletrodos (MCROBERTS, 2011, p. 117). Utilizando-se as portas analógicas do Arduino para obter o valor oriundo de um LDR, teremos um valor entre 0 (zero) e 1024 (proporcional a luz que incide no LDR). A partir destas premissas e de acordo com dados de Arduino By Myself (2012) podemos determinar valores de luminosidade preestabelecidos (Tabela I).

<sup>4</sup> <http://www.satistronics.com/myfiles/file/Module/About%20UltrasonicModule.pdf>

Tabela I: Valores de Luminosidade de um LDR

Intervalo	Grau de Luminosidade
0 - 9	Escuro
10 - 199	Penumbra
200 - 499	Iluminado
500 - 799	Brilhante
800 - 1023	Muito Brilhante

Fonte: Adaptado de:

<http://arduinobymyself.blogspot.com.br/2012/03/sensoreamento-de-luz-com-ldr-parte-2.html>

Para a montagem do sensor, os LED's foram posicionados atrás dos LDR's (Figura 7a) de maneira que a luz do LED, ao ser refletida em uma superfície branca e captada por um LDR, permita a passagem de corrente entre os eletrodos. Quando o LED incide sobre a linha escura sua luz não é refletida totalmente e o LDR passa a capturar menos luminosidade, criando com isso mais resistência a passagem da corrente elétrica. Após alguns testes, os componentes foram soldados em uma placa universal (Figura 7b) para fixação no chassi do robô. Utilizou-se na construção deste sensor dois LED's de auto brilho, dois LDR's, dois resistores de 100 ohms para os LED's e dois resistores de 620 ohms para os LDR's.

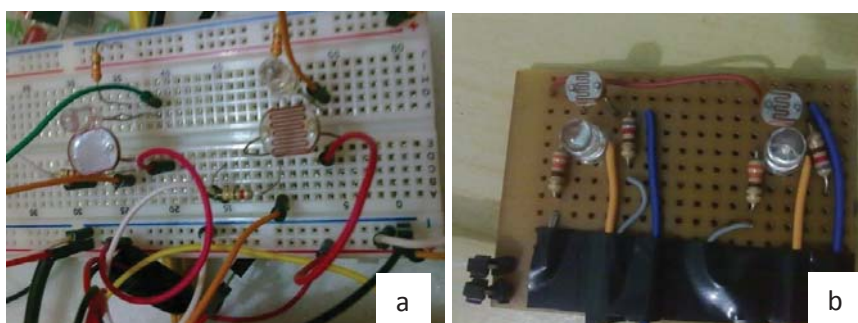


Figura 7: Montagem do sensor em uma Protoboard (a), Sensor soldado em uma placa universal para fixação no chassi (b).

A implementação da captação da linha por parte do sensor e o ajuste da trajetória do robô foi realizada utilizando-se as portas analógicas do Arduino que, a partir do valor de luminosidade captado, aplica-se uma manobra de correção do curso do robô a fim de mantê-lo na trilha escura (Figura 8).



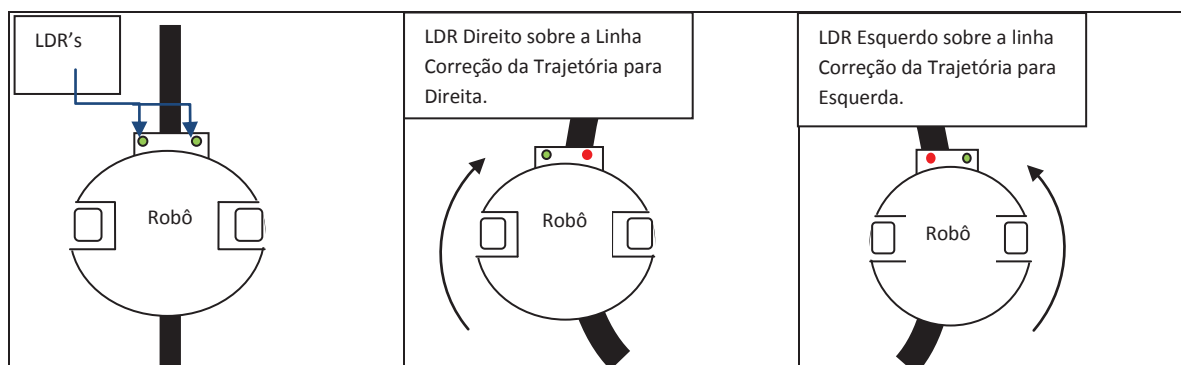


Figura 8: Correção da trajetória do robô

### 3.5. Chassi

A estrutura do Kit deveria ser rígida e leve a fim de fornecer estabilidade aos componentes e não prejudicar o desempenho do robô. Uma estrutura mais pesada acarretaria em um consumo maior de energia.

Para este projeto utilizou-se duas superfícies feitas em acrílico, utilizadas para cortar carne. Cortadas em forma circular, com diâmetro aproximado de 12 cm e ambas com uma abertura de 3,4 cm para acomodar as rodas. Na base inferior do chassi estão fixados os motores, as rodas de borracha e a terceira roda; além de duas baterias de 9 volts e o sensor de linha fixado na parte frontal do chassi. A base superior acomoda na parte frontal a mini protoboard com o circuito da ponte H e o sensor ultrassônico. Na parte posterior encontra-se a placa Arduino, fixada por meio de uma faixa de velcro. As duas bases são fixadas por meio de quatro parafusos, completando assim a montagem do Kit (Figura 9).

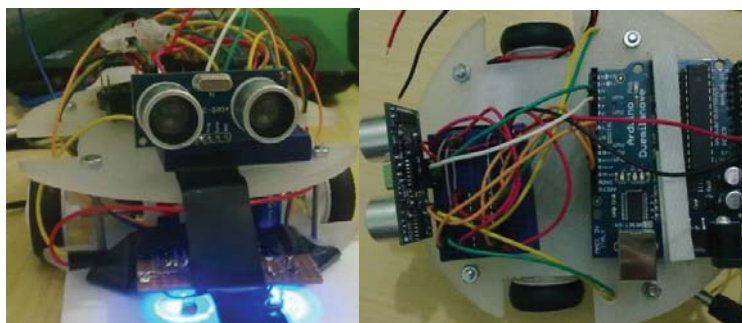


Figura 9: Kit montado

Fonte: Autoria própria

### 3.6. Software

Utilizou-se para a programação do robô a linguagem Arduino. Trata-se de uma DSL (Domain Specific Language – Linguagem de Domínio Específico) escrita em Java e desenvolvida tendo por base as linguagens wiring e processing (ARDUÍNO, 2012). Possui uma sintaxe muito semelhante ao C e a programação do Arduino pode ser dividida em três partes básicas (Multilogica 2012):

- Estrutura: onde se tem as duas funções básicas: *setup()*, usada para inicializar as variáveis e configurar os pinos digitais e analógicos da placa (esta função é executada apenas uma vez quando o Arduino é inicializado); e *loop()*, que executa as instruções em repetição contínua até que a

placa não esteja mais alimentada. Além disto, tem-se outras estruturas de controle: if, for, etc.;

- Valores: que podem ser variáveis e constantes, tais como INPUT e OUTPUT;
- Funções: tais como pinMode, digitalWrite, funções utilizadas para alterar/obter um valor em uma determinada porta do Arduino, funções matemáticas, de tempo, entre outras.

Além da IDE Arduino, outras soluções de software desenvolvidas para programação icônica (ex.: Minibloq<sup>5</sup> e Ardublock<sup>6</sup>) foram testadas, porém descartadas por não se mostrarem eficazes para o projeto.

#### 4. Testes Realizados

Os testes efetuados com o protótipo buscaram avaliar o funcionamento dos sensores, motores e o comportamento do Kit com um todo. O Robô foi testado em um pequeno circuito oval com aproximadamente 60 x 45 cm (Figura 10).

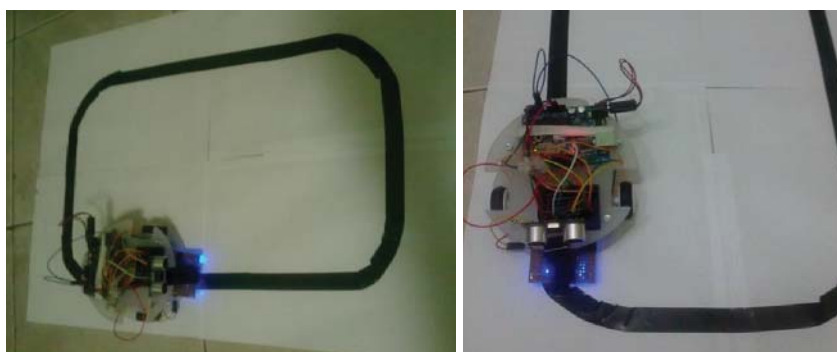


Figura 10: Circuito oval

O Robô teria que percorrer o pequeno circuito, evitando alguns obstáculos colocados ao longo da trilha. Assim que encontrasse um obstáculo, o robô sairia da trilha para contornar o objeto e voltaria mais a diante. O algoritmo para execução desta tarefa está ilustrado no diagrama de blocos (Figura 11).

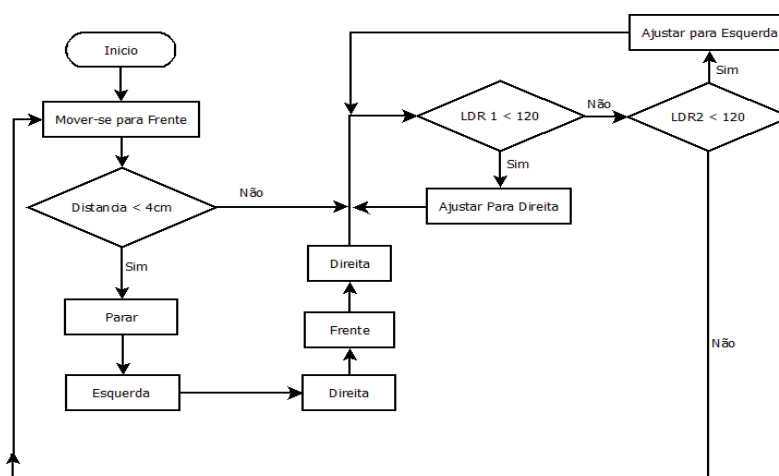


Figura 11: Diagrama de Blocos

<sup>5</sup> <http://minibloq.net/>

<sup>6</sup> <http://blog.ardublock.com/>

O robô se comportou bem durante os testes, com a execução do algoritmo ocorrendo conforme planejado, porém devido a distancia mínima estabelecida para detecção de um objeto e a altura do sensor em relação à superfície, alguns objetos de pequeno porte não foram devidamente reconhecidos pela programação.

## 5. Custos para Montagem do Kit Praxedes

O Kit construído neste trabalho teve sua montagem concebida a partir de materiais diversos: sucata (rodas de um brinquedo), materiais domésticos comuns (Tabuas de carne) e componentes eletrônicos disponíveis para a plataforma Arduino, resultando em uma plataforma de custos acessíveis (Tabela 2).

Tabela 2: Custos para Montagem do Kit

Componente	Quantidade	Custo
Arduino Duemilanove	IX	R\$ 75,00
Sensor HC-SRO4	IX	R\$ 35,00
Motor duplo Tamiya	IX	R\$ 45,00
Terceira Roda Pololu	IX	R\$ 13,00
Rodas de borracha	2X	-
Baterias de 9 Volts	2X	R\$ 19,80
Mini protoboard	IX	R\$ 10,00
CI L293D	IX	R\$ 10,00
Placas de Acrílico (Tabuas de Carne)	2X	R\$ 5,00
LDR	2X	R\$ 3,00
LED	2X	R\$ 2,00
Resistores	4X	R\$ 2,00
Placa Universal	IX	R\$ 5,00
<b>Total</b>		<b>R\$ 224,80</b>

Fonte: Autoria própria

## 6. Conclusões

Este trabalho mostrou a construção do protótipo Praxedes, um robô móvel baseado na plataforma Arduino. A construção do Kit utilizou conceitos de eletrônica, tais como a montagem de uma ponte H e utilização de sensores com a placa Arduino. Este Kit possui utilização variada, dependendo de como se darão as atividades de-

envolvidas com ele. Inicialmente, pode ser utilizado para solucionar labirintos, bem como a plataforma pode ser adaptada para competições, tais como de resgate.

O Kit se comportou relativamente bem em relação à integração dos componentes porém notou-se um consumo ligeiramente alto em relação as baterias, estando portanto estável para validação prática com alunos.

## 6.1. Trabalhos Futuros

As seguintes atividades são propostas para trabalhos futuros:

- Validação do Kit com usuários, no contexto do ensino fundamental;
- Desenvolver material de suporte pedagógico para o Kit;
- Desenvolver solução de software que facilite a programação do Kit Arduino por alunos não versados em programação textual.

## Referencias

ARDUINO. Disponível em: <<http://www.arduino.cc>>. Acesso em: 8 Fev. 2012.

ARDUINOBYMYSELF. Disponível em: <<http://arduinobymyself.blogspot.com.br/2012/03/sensoreamento-de-luz-com-ldr-parte-2.html>>. Acesso em: 23 Out. 2012.

CASTILHO, M. I. Robótica Educacional Com que Objetivos?, Porto Alegre, RS: 2002, Originalmente apresentado como Monografia de pós Graduação, Universidade Federal do Rio Grande do Sul - UFRGS, 2002.

FILHO, D. A. M.; GONÇALVES, P. C. Robótica Educacional de Baixo Custo: Uma realidade para as Escolas Brasileiras, in, XXVIII Congresso da Sociedade Brasileira de Computação, Workshop sobre Informática na Escola – WIE, Belém do Pará, PA: 2008.

FRISTEC. Disponível em: <<http://fristec.blogspot.com.br/2011/01/14-aplicacao-sensor-de-distancia-hc.html>>. Acesso: 21 Jul. 2012.

GIOPPO, L. L., HIGASKINO, M. M. K., COSTA, R. F., MEIRA, W. H. T. Robô Seguidor de Linha. Curitiba, PR: 2009, Originalmente apresentado como monografia a unidade curricular Oficina de integração II, Universidade Tecnológica Federal do Paraná, 2009.

LEGO MINDSTORMS. Disponível em: <<http://www.mindstorms.lego.com>> Acesso em: 10 Mar. 2012.

LYNXMOTION. Disponível em: <<http://www.lynxmotion.com>>. Acesso em: 16 Abr. 2012.

MARS EXPLORATION ROVERS. Disponível em: <<http://marsrovers.jpl.nasa.gov/home/ind ex.html>>. Acesso em: 10 Abr. 2012.

MAXWELL BOHR, Disponível em: <<http://www.maxwellbohr.com.br>>. Acesso em: 17 Jan. 2012.

MCROBERTS, M. Arduino Básico. 1ª Edição, São Paulo, Editora Novatec, 2011.

MCKESSON. Disponível em: <<http://www.mckesson.com>>. Acesso em: 04 Abr. 2012.

MODELIX, Disponível em: <<http://www.modelix.com.br>>. Acesso em: 15 Jan. 2012.

MULTILOGICA-SHOP. Disponível em: <<http://www.multilogica-shop.com>>. Acesso em: 10 Out. 2012.

PEREIRA, R. B. Robótica Educativa Como Recurso Didático No Ensino de Física no Ensino Médio. Cárceres, MT: 2008, Originalmente apresentado como Monografia de Graduação, Universidade do Estado de Mato Grosso – UEMT, 2008.

PIRES, J. N. Robótica: Das Maquinas Gregas a Moderna Robótica Industrial. Jornal Publico, Coimbra, Portugal. 8 Jun. 2002.

ROBOCORE. Disponível em: <<http://www.robocore.net>>. Acesso em: 20 de Jan. 2012.

SATATIS TRONICS. Disponível em: <<http://www.satistronics.com/myfiles/file/module/About%20UltrasonicModule.pdf>>. Acesso em: 21 Jul. 2012.

SIEBRA, C. A.; LINO, N. C. Q. An Experimental Study on the Use of Robotics as an Educational Tool. In: XXI Simpósio Brasileiro de Informática na Educação, 2010.

SILVA, F. I. e SCHERER, D.. A Study About Materials For Use In Educational Robotics. In: IX Latin American Robotics Simposium - I Simposio Brasileiro de Robotica - III Workshop de Robotica Educacional, 2012, Fortaleza-CE. Lars/Sbr 2012, 2012.

SOARES, R. F., BORGES, M. A. F. Robótica: aprendizado em informática de forma lúdica. In: XXXI Congresso da sociedade brasileira de computação - WEI - XIX Workshop sobre Educação em Computação, 2011, Natal, RN. XXXI Congresso da sociedade brasileira de computação. Natal, RN: UFRN, 2011. v. I. p. 1516-1519.