

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA
SOUZA**

ETEC PROFESSOR ARMANDO BAYEUX DA SILVA

Desenvolvimento de Sistemas

Bruno Antonello

GARRA AUTOMATIZADA SOBRE RODAS (G.A.R):

Robótica Didática

Rio Claro

2024

Bruno Antonello

**GARRA AUTOMATIZADA SOBRE RODAS (G.A.R):
Robótica Didática**

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Desenvolvimento de Sistemas da Etec Professor Armando Bayeux da Silva, orientado pelo Prof. Esp. Valdeci Ançanello, como requisito parcial para obtenção do título de técnico em Desenvolvimento de Sistemas.

Rio Claro

2024

AGRADECIMENTOS

Agradeço aos meus pais pela criação, desenvolvimento e apoio, tanto neste projeto quanto na minha trajetória de vida. E em especial ao meu pai, Marcio, por me orientar na construção do projeto e contribuir financeiramente para a conclusão do mesmo.

Meus agradecimentos, também, a todos os professores que me auxiliaram, orientaram e apoiaram.

“Esse é o principal ponto da tecnologia. Por um lado, ela cria um apetite por imortalidade e, por outro, ameaça extinção universal. Tecnologia é a luxúria removida da natureza.”

Don DeLillo

RESUMO

Este trabalho tem o intuito de registrar o desenvolvimento de um projeto sobre uma garra robótica acoplada à um carrinho de controle remoto via bluetooth denominado Garra Automatizada sobre Rodas (G.A.R), com foco de aprendizagem e entretenimento, contendo instruções didáticas. Sendo necessário a conexão com o aplicativo para controlá-la. Busca-se uma didática com o uso da robótica, incentivando, principalmente, o público infanto-juvenil.

Palavras-chave: Aprendizagem; Conexão; Desenvolvimento; Entretenimento; Robótica.

ABSTRACT

This work aims to record the development of a project about a robotic claw coupled to a remote-controlled cart via Bluetooth called Automated Gripper on Wheels (G.A.R), with a focus on learning and entertainment, containing didactic instructions. It is necessary to connect to the application to control it. Didactics are sought using robotics, mainly encouraging children and young people.

Keywords: Learning; Connection; Development; Entertainment; Robotics.

Lista de ilustrações

Figura 1– Resistor esquema elétrico.....	18
Figura 2– LED	20
Figura 3 - Sensor ultrassônico HC-SR04	20
Figura 4–Servos motores	21
Figura 5 - Buzzer	22
Figura 6 - Motor shield L293D	23
Figura 7- Módulo Bluetooth HC-05.....	24
Figura 8 - Logo Arduino IDE	27
Figura 10 - Tela inicial App Inventor.....	28
Figura 9 - Tela inicial Arduino IDE	28
Figura 11 - Tela inicial arquivo .APK	29
Figura 12 - Comunicação serial	30
Figura 13 - Comandos edição	31
Figura 14 - Logo GAR_bluetooth	33
Figura 15–Protótipo G.A.R 31/03.....	38
Figura 16 - Protótipo G.A.R 13/06	39
Figura 17 - Protótipo G.A.R 14/08	39
Figura 18 - Protótipo final G.A.R 24/10	40
Figura 19 - Esquema	45

Lista de tabelas

Tabela 1 - Tabela resistores	19
Tabela 2– ComandosGAR_bluetooth	32
Tabela 3 – Custos	36

Lista de abreviaturas e siglas

Android Package (APK)

Associação Brasileira de Normas Técnicas (ABNT)

Corrente Contínua (DC)

Garra Automatizada sobre Rodas (G.A.R)

Gigahertz (GHz)

Ground (GND)

Integrated Development Enviroment (IDE)

Internet of Things (IoT)

Light-Emiting Diode (LED)

Miliampere (mA)

Personal Computer (PC)

Pulse Width Modulation (PWM)

Recepção (RX)

Trabalho de Conclusão de Curso (TCC)

Transmissão (TX)

Universal Serial Bus (USB)

Volts (V)

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Motivação	12
1.2	Objetivos	12
2	REFERENCIAL TEÓRICO	13
2.1	Sistemas Embarcados	13
2.1.1	Aplicações	13
2.2	Robótica	14
2.2.1	Robótica educacional	14
2.2.2	Robótica na indústria e no mercado	14
2.3	Arduino	16
2.4	Ambiente de Desenvolvimento Integrado	17
2.5	Componentes eletrônicos	18
2.5.1	Resistores	18
2.5.2	LEDs	19
2.5.3	Sensores ultrassônicos	20
2.5.4	Servos motores	21
2.5.5	Buzzers	22
2.5.6	Motor shield	23
2.5.7	Módulo Bluetooth	23
2.6	Bluetooth	25
2.6.1	História	25
2.6.2	Tecnologia	25
3	DESENVOLVIMENTO	26
3.1	Entenda o que é G.A.R	26
3.2	Programas utilizados	27
3.2.1	Arduino IDE	27
3.2.2	App inventor	28
3.3	Controle remoto	30
3.4	Hardware	34
3.4.1	Servos motores	35
3.5	Custos	36
3.6	Problemas enfrentados	37
3.7	Protótipos	38

4	INTENÇÕES FUTURAS	41
5	CONCLUSÃO	42
	REFERÊNCIAS	43
	APÊNDICE A – Esquema	45

1 INTRODUÇÃO

O Trabalho de Conclusão de Curso visa documentar todo o projeto, com âmbito para a obtenção do título de técnico em desenvolvimento de sistemas, que envolve a robótica e a didática, com o projeto denominado de G.A.R: garra robótica acoplada a um carrinho de controle remoto via bluetooth. O aplicativo para comunicação serial foi criado pelo autor. Destaca-se a escrita da documentação, além da organização e explicação detalhada, seguindo as normas da ABNT, para que o leitor compreenda.

Para melhor entendimento, as subseções do referencial teórico estão separadas em: Sistemas Embarcados, Robótica, Arduino, Ambiente de Desenvolvimento Integrado, Componentes eletrônicos e Bluetooth. Procura-se esclarecer o pensamento e pretensões futuras do autor com o projeto.

1.1 Motivação

O mercado de trabalho está cada vez mais necessitando por profissionais técnicos nas mais diversas áreas, por conta da evolução tecnológica, que mudaram e mudam as vidas das pessoas na sociedade.

Por isso, a principal motivação do autor do projeto é o interesse pela didática da tecnologia, com entretenimento e diversão, além da praticidade de controle, com resultados rápidos e precisos.

1.2 Objetivos

O objetivo central do trabalho é construir uma garra robótica que será acoplada a um carrinho de controle remoto via Bluetooth. O controle remoto será conectado a um aplicativo, desenvolvido pelo autor, possuindo 15 comandos no total. Contudo, por o Arduino ter *software*¹ e *hardware*² livres, pode-se modificar seu código ou ligações (*hardware*), e, também, se divertir com o brinquedo, promovendo o *eduentretenimento* (educação com entretenimento).

¹ Parte de codificação; Programa de computador.

² Parte física; Componentes eletrônicos.

2 REFERENCIAL TEÓRICO

2.1 Sistemas Embarcados

No cenário contemporâneo, os sistemas embarcados fazem parte do cotidiano de todo o mundo. Sistemas embarcados são sistemas computacionais, conjunto de hardware e software, com a tarefa de controlar sistemas maiores. Projetados para serem simples e de baixo custo, e são compostos por conjuntos limitados de componentes, microcontroladores, sensores e atuadores. Com a função de controlar e/ou monitorar um processo (CHASE, Otavio; ALMEIDA, F.).

2.1.1 Aplicações

As aplicações destes se encontram em automóveis, dispositivos em geral, sistemas de segurança, aquisição de dados, entre outros.

No caso do projeto, utilizará um sistema embarcado para o desenvolvimento e montagem, buscando-se o êxito.

2.2 Robótica

A robótica é a ciência que estuda as tecnologias associadas a concepção e construção de robôs, sendo um ramo educacional e tecnológico. Robôs são mecanismos automáticos que utilizam de circuitos integrados para realizarem atividades e movimentos humanos. Segundo o *blog.eletrogate.com*, o termo robô deriva da palavra *robota*, de sua originalidade eslava significa “trabalho forçado”.

2.2.1 Robótica educacional

Para Flávio Rodrigues Campos, mesmo com inúmeros instrumentos tecnológicos, atualmente ainda nota-se a limitação na realidade escolar. Tal se deve pela compreensão e análise da maneira que essa é abordada. Os avanços da tecnologia são evidentes, e a tendência será sempre esta.

Logo, é um dever a capacidade de entendimento, principalmente, de forma divertida. O crescente interesse pela área da robótica é um fato, infere-se disto que, é imprescindível uma proposta pedagógica plausível.

A prática na educação é efetuada tecnicamente, no ensino profissionalizante. Esta abordagem pode ser ampliada para crianças e jovens, com aspectos relevantes que despertem o interesse, como brinquedos robóticos.

2.2.2 Robótica na indústria e no mercado

Inovações tecnológicas se fazem presentes na história da humanidade, estas fornecem conhecimento e aperfeiçoamento das habilidades adquiridas até então.

Industrialmente, desde a invenção do motor a vapor, de James Watt, em 1776, possibilitou-se as revoluções industriais, resultando na tecnologia contemporânea, que sempre busca a maior eficiência e desempenho, com menores custos.

Com isso, as indústrias buscam cada vez mais se destacar no mercado, com a melhoria contínua em seus processos, buscando agregar a robótica em linhas de produção. A evolução é constante.

O mercado de robótica é intenso, pelo fato de que a substituição da mão de obra humana pela robótica se torna mais viável, em questão de produtividade, eficiência, custos e eficácia. Isto é a quarta revolução industrial.

Segundo a Época Negócios (2017), dados da Federação Internacional de Robótica (IFR, em inglês) na Coreia do Sul os números de robôs para cada 10 mil operários ultrapassam a 500 no geral, e nos países como Singapura, Japão e Alemanha a média é superior a 300 para cada 10 mil operários, na China os números equivalem a mais de 40, com a previsão de que até 2025 cheguem a margem de 150. No Brasil a avaliação é de que há 10 robôs para cada 10 mil operários (FENERICK, Jessica Aparecida; VOLANTE, Carlos Rodrigo).

2.3 Arduino

O Arduino é uma plataforma de prototipagem eletrônica responsável por processar as informações a serem implementadas através de um ambiente de desenvolvimento (IDE). Ele foi criado em 2005, na Itália, por um grupo de cinco pesquisadores: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis. Seu surgimento deve-se pela vontade de Massimo para disciplinar alunos a terem noções básicas sobre eletrônica e programação (SILVA, Victor Augusto Pimenta).

Existem incontáveis versões da placa Arduino, em sua grande maioria são as placas clones, sendo estas não originais da marca (MCROBERTS, Michael). As placas, em geral, possuem pinos RX, TX, digitais, analógicos, tensão e GND. Os pinos digitais podem ser configurados como entrada ou saída de dados, com Input ou Output, respectivamente, assumindo valores de *HIGH* (1) ou *LOW* (0). Portanto, os pinos analógicos são apenas de entrada de dados, variando de 0 a 1023.

Ademais, os pinos PWM, que fazem pinos digitais agirem similares aos analógicos, com uma variação de 0 a 255, e são eles: pinos 3, 4, 6, 9, 10 e 11 (Arduino Uno). Nas placas de Arduino, os pinos PWM são representados pelo símbolo (~) ao lado do número do pino.

2.4 Ambiente de Desenvolvimento Integrado

De acordo com o *site locaweb.com.br*, um Ambiente de Desenvolvimento Integrado (IDE) é um programa de computador que reúne ferramentas para o desenvolvimento de softwares, com objetivo de agilizar e facilitar o processo. Ele possui editor de código, depurador, compilador/interpretador, gerenciador de projetos, ambiente de execução (emulador), controle de versão integrado, integração com ferramentas externas.

Segundo o *site redhat.com*, existem separações entre as IDEs, e diversos fatores influenciam na escolha da ferramenta. Podem-se citar alguns desses tópicos decisivos como compatibilidade de linguagens de programação, compatibilidade de sistemas operacionais, funcionalidades de automação, desempenho do sistema, *plug-in*³ e extensões. Logo, existem IDEs utilizadas apenas para desenvolvimento mobile, como por exemplo, o App Inventor.

A capacidade de criação é limitada à imaginação, sendo possível construir diversos tipos de projetos, desde softwares com as mais variadas funcionalidades até jogos, aplicativos de mensagens e demais categorias.

³ Plugar, em português. É um programa de computador que adiciona ferramentas a outros programas maiores, lhe proporcionando funcionalidades únicas.

2.5 Componentes eletrônicos

Os componentes eletrônicos são cruciais para o funcionamento de circuitos, que estão inseridos em dispositivos. O seu uso vai de limitar corrente, alterar polaridade, acender luzes até mesmo com sensores que medem a umidade do ar, distância e temperatura, por exemplo.

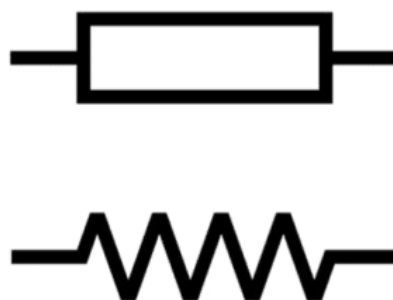
Estes possuem dois tipos, componentes eletrônicos digitais e componentes eletrônicos analógicos. Segundo o *site victorvision.com.br*, em circuitos digitais os dados são representados apenas como 1 ou 0, no caso, verdadeiro ou falso, respectivamente. Esta característica se encaixa no método booleano, de George Boole. Entretanto, em circuitos analógicos essas informações podem ser representadas com variações de valores, e possui sinal contínuo. Ou seja, a principal diferença está na representação e processamento de dados.

Utilizou-se diversos componentes eletrônicos para efetuar o desenvolvimento até sua montagem. Incluindo: resistores, LEDs, sensor ultrassônico, servos motores, módulo Bluetooth, buzzer, shield entre outros. Os detalhes e aplicações de cada componentes serão explicados a seguir.

2.5.1 Resistores

De acordo com o *site makerhero.com*, os resistores têm a função de limitar ou controlar a corrente elétrica em circuitos. Com a aplicação do efeito Joule, transformando a energia elétrica em térmica. Representado pela letra R, tem sua aplicação na primeira e segunda lei de Ohm. Sua representação em esquema elétrico está na figura 1.

Figura 1– Resistor esquema elétrico



FONTE: <https://ajpeleetroinfo.com.br/wp-content/uploads/2019/04/eletr%C3%B4nica-2-300x300.png>

Os resistores possuem faixas coloridas que indicam seus valores, em ohms. A tabela 1 representa como identificar a resistência pelas faixas coloridas.

Tabela 1 - Tabela resistores

Cor	1ª faixa	2ª faixa	Multiplicador	Tolerância
Preto	0	0	x 1Ω	
Marrom	1	1	x 10Ω	±1%
Vermelho	2	2	x 100Ω	±2%
Laranja	3	3	x 1kΩ	
Amarelo	4	4	x 10kΩ	
Verde	5	5	x 100kΩ	±0,5%
Azul	6	6	x 1MΩ	±0,25%
Violeta	7	7	x 10MΩ	±0,1%
Cinza	8	8		±0,05%
Branco	9	9		
Dourado			x 0,1Ω	±5%
Prateado			x 0,01Ω	±10%

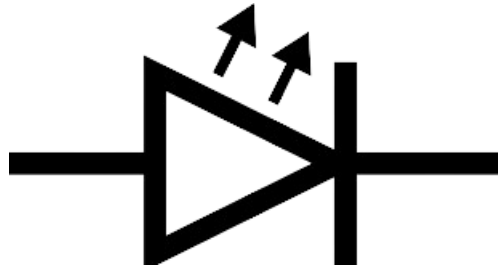
FONTE: https://evolutec.ind.br/wp-content/uploads/2016/07/Post-004-27-07-2016_01.jpg

2.5.2 LEDs

Segundo o *site tecnoblog.net*, a estrutura básica de um LED (*Light Emitting-Diode*/Diodo Emissor de Luz) é composta por dois terminais semicondutores chamados ânodo e cátodo. Esses componentes ficam dentro de cápsulas com formatos e tamanhos variáveis. O LED gera luz a partir do princípio da eletroluminescência, que ocorre quando os terminais são submetidos a uma corrente elétrica.

De acordo com a figura 2, o esquema elétrico do LED torna visível estes semicondutores, bem como suas ligações.

Figura 2– LED



FONTE: https://www.clker.com/cliparts/6/f/5/7/1197090733366797408vermeil_IEC_LED_Symbol.svg.hi.png

2.5.3 Sensores ultrassônicos

De acordo com o *blog elektraautomacao.com.br*, o sensor ultrassônico é um dispositivo que emite ondas sonoras de alta frequência, que se propagam pelo ar até encontrar alguma barreira, como por exemplo, uma parede. É possível identificar a distância desta barreira, que é calculada pelo tempo que esta onda é interceptada pelo objeto. Isso ocorre através do TX (transmite a onda - *trig*), e do RX (recebe a onda - *echo*), calculando a diferença de tempo entre a transmissão e recepção.

Uma aplicação de um sensor ultrassônico é no projeto, analisando a distância frontal entre o carrinho e algum obstáculo, e com determinada proximidade, é realizado a parada dos motores, e eventualmente das rodas. A figura 3 mostra um sensor ultrassônico HC-SR04, o mesmo modelo utilizado no projeto.

Figura 3 - Sensor ultrassônico HC-SR04



FONTE: https://images.tcdn.com.br/img/img_prod/590985/sensor_ultrassonico_de_distancia_hc_sr04_modulo_arduino_580_1_20201213200010.png

2.5.4 Servos motores

Segundo o *blog nepin.com.br*, o servo motor é indispensável nos recentes modelos de equipamentos industriais, bem como em diversas áreas da robótica. Utilizados como atuadores, dispositivos que realizam algum movimento, obter maior controle e precisão em processos.

Com foco nas áreas da robótica, os motores servos são dispositivos de movimento rotativo, sendo monitorados por ângulos (180°) ou velocidade e direção (360°). Os servos motores mais comuns na utilização de protótipos (como por exemplo, garras robóticas, esteiras ou persianas automatizadas) são os micros servos 9g SG90, MG90S e MG996R.

Subentende-se do que foi dito anteriormente que, nem todos os modelos são adequados para protótipos, pois cada um possui suas peculiaridades, como torque, tensão, tamanho, entre outros fatores. A figura 4 ilustra alguns dos modelos de motores servos.

Figura 4—Servos motores



FONTE: https://curtocircuito.com.br/pub/media/wysiwyg/blog/Introducao_ao_Servo_motor/1_-_Tipos_de_Servo_Motor.jpg

2.5.5 Buzzers

De acordo com o *blog autocorerobotica.blog.br*, buzzers são componentes eletrônicos capazes de gerar frequências sonoras. Muito utilizados em aparelhos que possuem alarmes, como micro-ondas. Este componente gera frequências que estão na faixa de 1 a 7 kHz, sendo um som extremamente agudo. Sendo assim, a frequência da voz humana gira em torno dos 250 Hz, no máximo, percebendo assim uma enorme diferença. Ele possui duas variações, buzzers passivos e buzzers ativos. A figura 5 apresenta um buzzer.

Figura 5 - Buzzer



FONTE: <https://i0.wp.com/www.blogdarobotica.com/wp-content/uploads/2020/10/image.png>

2.5.5.1 Buzzers passivos

Buzzers passivos são capazes de gerar diferentes sons de acordo com a frequência determinada pela programação. É possível criar músicas com ele, já que permite a modificação de frequência.

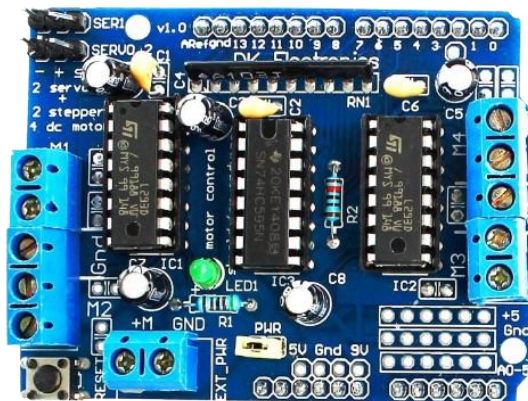
2.5.5.2 Buzzers ativos

Buzzers ativos emitem ondas sonoras do oscilador interno, que produz o som ao ser alimentado. Diferente do buzzer passivo, ele não possui frequência, logo não é possível modificar os sons produzidos por este.

2.5.6 Motor shield

Segundo o *site curtocircuito.com.br*, Motor Shield L293D é um drive que permite controlar até 4 motores DC, 2 Servos Motores ou 2 Motores de Passo através de uma Placa Arduino, suportando até 16V e 600mA. As aplicações são dependentes de placas de micro controladores, e são, principalmente, usadas em protótipos de carrinhos de controle remoto, no controle dos motores DC que possibilitam o movimento. A figura 6 ilustra um Motor Shield, modelo L293D.

Figura 6 - Motor shield L293D



FONTE: <https://www.usinainfo.com.br/blog/wp-content/uploads/2019/11/motor-shield-l293D-scaled.jpg>

2.5.7 Módulo Bluetooth

De acordo com o *site usinainfo.com.br*, o Módulo Bluetooth possibilita transmitir e receber dados através da comunicação *wireless*⁴. Este módulo é utilizado para comunicação sem fio, como por exemplo, com a porta *serial* do Arduino. É possível enviar dados com alguns comandos, e com a respectiva leitura, a execução de comandos via código.

⁴ Sem fio, em português. Permite a conexão entre dispositivos à distância.

Possui diversas aplicações com a placa do Arduino, com a execução de funções por comandos a curtas distâncias, realizando a comunicação com outros equipamentos. As versões criadas são: módulos HC-05, HC-06, HC-08, entre outros. A figura 7 retrata um módulo Bluetooth HC-05, o mesmo usado no projeto.

Figura 7- Módulo Bluetooth HC-05



FONTE: <https://cdn.awsli.com.br/600x450/573/573703/produto/26660540/12138b75a8.jpg>

2.6 Bluetooth

2.6.1 História

No ano de 1994 a empresa de telecomunicações Ericsson começou um estudo a respeito da viabilidade de uma interface rádio de baixo custo em dispositivos celulares e acessórios. O objetivo disto era eliminar os cabos de conexão em celulares e PCs (PUY, Inigo; HEINDL, Eduard).

Em 1997 Ericsson se aproxima de fabricantes de dispositivos portáteis para aumentados interesses. Em fevereiro de 1998, cinco empresas fundaram um Grupo de Interesse Especial (SIG), sendo estas: Ericsson, Nokia, IBM, Toshiba e Intel. Em maio do mesmo ano, o consórcio Bluetooth foi anunciado ao público, suas intenções eram estabelecer um dispositivo padrão e *software* que o controle. Com este anúncio, houve a adoção da tecnologia por várias outras empresas (PUY, Inigo; HEINDL, Eduard).

2.6.2 Tecnologia

O Bluetooth é uma tecnologia de conexão do tipo *wireless* de curto alcance. O endereço do dispositivo Bluetooth é de 48 bits⁵. Cada dispositivo deve estar equipado com um microchip transceptor na frequência 2,4 GHz (BLUETOOTH, S. I. G.).

Os estados do dispositivo são: *Standby*⁶, investigar, varredura, conectar, manter conexão (BLUETOOTH, S. I. G.). É possível conectar fones de ouvidos, caixas de som, *mouses*, *smartphones*, e outros dispositivos que possuem esta tecnologia acoplada.

⁵ Dígito binário, em português. Menor unidade da informação que pode ser armazenado ou transmitido. Assume apenas dois valores, verdadeiro (1) ou falso (0).

⁶ Em espera, em português. Aguarda a conexão.

3 DESENVOLVIMENTO

Procura-se documentar todas as etapas do projeto G.A.R, desde a construção, montagem, controles, conexão, funcionamento e problemas enfrentados.

Busca-se coerência por parte do leitor, então este tópico foi separado em setecapítulos. Sendo eles: Entenda o que é G.A.R; Programas utilizados; Controle remoto; Hardware; Custos; Problemas enfrentados; Protótipos.

3.1 Entenda o que é G.A.R

G.A.R (Garra Automatizada sobre Rodas), trata-se de uma garra robótica acoplada a um carrinho de controle remoto via Bluetooth, tendo uma mistura de *hardware* e *software*.

O controle remoto para comandar a garra e o carrinho será por um aplicativo mobile, desenvolvido pelo autor.

A sua finalidade é aprendizagem e entretenimento das pessoas interessadas por robótica e automação, além de curiosos.

3.2 Programas utilizados

Utilizou-se para a codificação do projeto duas IDE: Arduino IDE, para a construção do código embutido no hardware da G.A.R, ou seja, o desenvolvimento do *software*, e o *App Inventor*, utilizado para desenvolver o aplicativo *mobile*⁷, de controle remoto para controlar o carrinho, a garra robótica, faróis e sons. Este funciona como uma comunicação *serial* com o Arduino, sendo também, personalizável, com a inserção de comandos por meio do aplicativo.

3.2.1 Arduino IDE

O ambiente de desenvolvimento integrado do Arduino é o Arduino IDE, um *software* gratuito introduzido pela Arduino.cc, que possibilita a codificação para este microcontrolador. Utiliza-se as linguagens de programação C e C++. Disponível em <https://www.arduino.cc/en/software>. Sua logo está na figura 8.

Figura 8 - Logo Arduino IDE



FONTE: <https://store-images.microsoft.com/image/apps.14931.13510798887551775.c10ff9d-ead5-4a16-aa9c-4e638842c0f4.178c4604-4aa3-4864-b2e0-ee3ad9ceb7ba?h=464>

As suas principais ferramentas são: Edição de código, verificação, compilação, abrir, salvar, *monitor serial*⁸, *sketch*⁹, inclusão de bibliotecas, ajuda, ferramentas e seleção de *board*¹⁰ (FEZARI, Mohamed; AL DAHOUD, Ali).

⁷ Móvel, em português. Refere-se a telefones celulares.

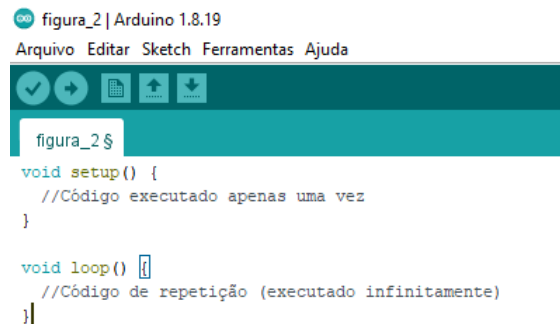
⁸ Local onde é mostrado a comunicação do Arduino com o computador.

⁹ Esboço, em português. É o código do Arduino.

¹⁰ Modelo da placa do Arduino.

A figura 9 mostra a tela inicial do Arduino IDE.

Figura 9 - Tela inicial Arduino IDE



FONTE: Arquivo original do autor.

3.2.2 App inventor

O App Inventor é um *software* gratuito para desenvolvimento *mobile* (iOS ou Android), sua linguagem de programação não é específica, pois se utiliza blocos de instruções como operadores lógicos para o desenvolvimento de aplicativos para celulares (WOLBER, David et al). O *software* pode ser utilizado em aplicações com IoT (*Internet das Coisas*), e está disponível em <https://appinventor.mit.edu>.

O aplicativo desenvolvido foi denominado de GAR_bluetooth. Seu uso será no controle remoto do circuito, sendo mais especificado no próximo tópico. A figura 10 mostra a tela inicial de desenvolvimento do aplicativo.

Figura 10 - Tela inicial App Inventor



FONTE: Arquivo original do autor.

A figura 11 exibe a tela do controle remoto no aplicativo, com os respectivos botões.

Figura 11 - Tela inicial arquivo .APK



FONTE: Arquivo original do autor.

3.3 Controle remoto

O controle remoto é via Bluetooth, e permite controlar direção do carrinho, direção da garra robótica, abertura e fechamento da garra, faróis frontais e buzina. Isto é possível graças à comunicação serial do módulo Bluetooth (HC-05) com o Arduino, no caso, utilizaram-se letras para esta conexão. A figura 12 mostra um trecho do código da comunicação serial.

Figura 12 - Comunicação serial

```
while (bluetooth.available()) {
  comando = bluetooth.read();
  distancia = sensorDistancia.dist(); //Sempre lê a distância (sensor)

  // Executa o comando correspondente
  switch (comando) {
    case 'A': //Atras
      avancar_tras();
      break;
    case 'B': //Baixo
      elevar_baixo();
      break;
    case 'C': //Cima
      elevar_cima();
      break;
    case 'D': //Direita
      Dir();
      break;
    case 'E': //Esquerda
      Esq();
      break;
    case 'F': //Frente
      avancar_frente();
      break;
    case 'G': //Garra
      Garra();
      break;
    case 'H': //Buzina
```

FONTE: Arquivo original do autor.

Uma breve explicação do trecho de código acima: o comando *while*¹¹ sendo *true*¹², em haver uma conexão com o módulo Bluetooth, permite a execução do bloco de código dentro dele. A variável “comando” é a leitura da comunicação recebida pelo Bluetooth. Já a variável “distância” é a leitura da distância usando o sensor ultrassônico HC-sr04, sendo um comando da biblioteca “HCSR04.h”,

¹¹ Enquanto, em português. Em programação, enquanto certa condição for avaliada com verdadeira, o bloco de código dentro dele é executado.

¹² Verdadeiro(a), em português. Em programação, é um valor Booleano, representado por 1.

aplicado ao carrinho. O *switch*¹³ analisa os casos de valores recebidos pelo módulo Bluetooth, se certo comando foi lido pela variável “comando”, ele executa a função correspondente. As funções, em programação, são caracterizadas por terem o final com parênteses, em alguns casos, há exigência de parâmetros, sendo variáveis, necessárias para a execução desejada e otimização do código.

Os comandos do aplicativo são personalizáveis, havendo a possibilidade de aplicação para os mais diversos projetos de controle com comunicação serial por bluetooth, portanto para G.A.R há um padrão, já definido ao início do aplicativo, com as letras correspondentes para o funcionamento ideal. Para melhor entendimento sobre os comandos, a figura 13 mostra a personalização de comandos do aplicativo.

Figura 13 - Comandos edição


















FONTE: Arquivo original do autor

¹³ Interruptor, em português. Em programação, tem a mesma funcionalidade do comando “if”, analisando casos, se forem verdadeiros, executa o bloco de código dentro do mesmo.

A tabela 2 ilustra as funcionalidades com as respectivas letras e botões do aplicativo, facilitando a compreensão na hora de controlar.

Tabela 2– ComandosGAR_bluetooth

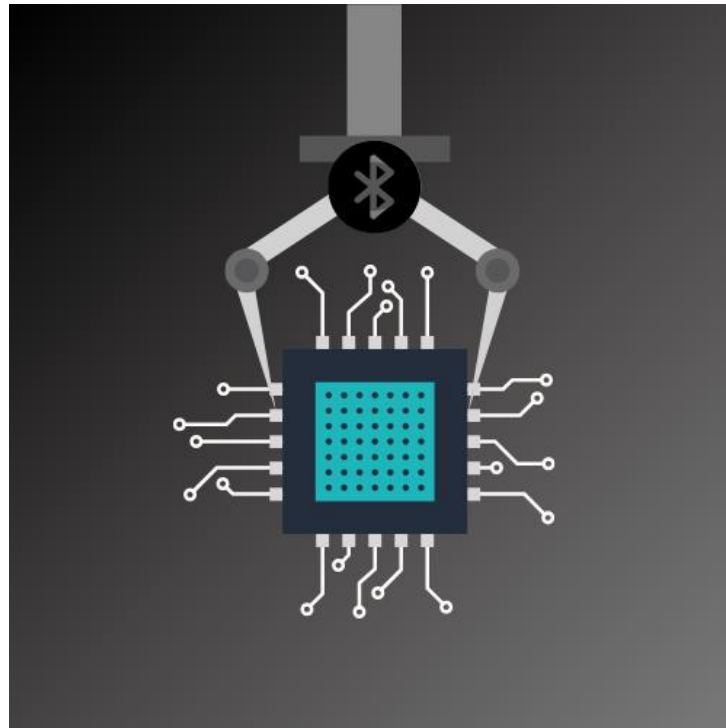
Símbolo	Função	Letra Padrão
	Sobe a garra robótica	C
	Base garra para direita	D
	Base garra para esquerda	E
	Desce a garra robótica	B
	Avança a garra robótica	F
	Avança a garra robótica	F
	Recua a garra robótica	A
	Recua a garra robótica	A
	Frente (carrinho)	f
	Direita (carrinho)	d
	Esquerda (carrinho)	e
	Atrás/Ré (carrinho)	a
	Faróis (dianteiros)	I
	Buzina do carro	s
	Abre e fecha a garra	G
Botão solto	Quando solta qualquer botão	P

FONTE: Arquivo original do autor.

Com isto apresentado, ao apertar algum botão do controle remoto (figura 11) enviará o texto inserido nele, personalizável (figura 13), assim o módulo Bluetooth envia para o monitor serial esse texto, e caso for igual ao da programação, ele executará o comando, realizando o que está no código.

O aplicativo está disponível apenas para pessoas autorizadas a terem o arquivo .APK, assim, funcionando com êxito. A figura 14 representa a logo do aplicativo G.A.R.

Figura 14 - Logo GAR_bluetooth



FONTE: Arquivo original do autor.

3.4 Hardware

O *hardware* é a parte física do projeto, no caso, o carrinho e a garra robótica. Utilizou-se o Arduino Uno, que possui 14 pinos digitais e 6 pinos analógicos, sendo o suficiente para conclusão do mesmo. A escolha deste modelo se deve pela praticidade e satisfação, em relação à quantidade de pinos e portabilidade, além do Arduino possuir *software* e *hardware* livres, ou seja, facilmente modificáveis.

Houve a necessidade do uso de componentes eletrônicos, para seu funcionamento adequado. Listados:

- 1x Arduino Uno;
- 2x LEDs;
- Resistores;
- Fios;
- 1x Sensor ultrassônico HC-sr04;
- 1x Módulo Bluetooth HC-05;
- 1x Motor MG90S (360°);
- 3x Motores MG90S (180°);
- 4x Motores DC (carrinho);
- Baterias;
- 1x Motor Shield;
- 1x Buzzer (passivo).

No item **(3.5 Custos)** há detalhes de quantidades e valores dos materiais, em formato de planilha do Excel.

Com o controle em mãos, a manipulação dos motores é possível. Isto ocorre de acordo com as ligações com os pinos de sinais do Arduino, comunicação com o módulo Bluetooth, e claro, com a programação correta.

O funcionamento do carrinho e da garra robótica deve-se pela movimentação dos motores, que se movem de acordo com a direção orientada pelo controle remoto, e a velocidade é definida no código. Utilizou-se, também, LEDs para os

faróis dianteiros, bem como um buzzer para a buzina. Ambos possuem comandos próprios para sua execução pelo controle remoto.

Em geral, o uso de resistores deve ser efetuado para contenção da corrente elétrica, impedindo danos a componentes. Os conectores jumpers foram essenciais para realizar as ligações, tanto no Arduino quanto na protoboard, possibilitando a formação do circuito responsável pelo G.A.R. A protoboard foi necessária para a ligação de alguns componentes, utilizando conectores jumpers, sendo utilizada para a fase de testes e para o protótipo inicial.

A fonte de energia principal são pilhas e baterias, alimentando com 5V e 9V a placa do Arduino Uno e a motor shield, respectivamente.

3.4.1 Servos motores

Os servos motores utilizados foram essenciais para o controle preciso de posição, velocidade e torque da garra robótica, que podem ser ajustados de acordo com o controle remoto. A garra robótica utilizou quatro servos motores MG90s, um de 360° para a base ter uma rotação sem limitação, e três 180° para elevar, avançar e abrir ou fechar a garra.

Estipularam-se limites de movimentação para a garra robótica, nos aspectos de fechá-la e abri-la, subi-la, abaixá-la, avançá-la e recuá-la. Devido aos servos motores serem de 180° foi possível regulá-los por ângulos. Já o servo motor de 360° é regulado por velocidade e direção, da seguinte forma: valores de 0 a 89 fazem o servo motor rodar para o sentido horário, e quanto mais perto de zero, mais veloz. De 91 a 180 fazem-no rodar para o sentido anti-horário, e quanto mais próximo de 180, mais veloz. O valor 90 faz o servo motor ficar estático, parando-o quando necessário.

3.5 Custos

O projeto teve gastos apenas voltados ao hardware, para a compra dos componentes e materiais. Na codificação não houve a necessidade de gastos financeiros com ferramentas de desenvolvimento ou planos de serviço, pois são gratuitos. A tabela 3 mostra os detalhes dos componentes utilizados, bem como os gastos.

Tabela 3 – Custos

COMPONENTE	UNIDADE	VALOR (UN)	TOTAL
Buzzer (passivo)	1	R\$ 16,04	R\$ 16,04
Carrinho de brinquedo	1	R\$ 9,99	R\$ 9,99
Garra robótica	1	R\$ 139,99	R\$ 139,99
Jumper	48	R\$ 0,50	R\$ 24,00
Módulo Bluetooth	1	R\$ 43,90	R\$ 43,90
Módulo de carregamento de bateria	1	R\$ 4,80	R\$ 4,80
Motor DC	4	R\$ 14,97	R\$ 59,88
Motor MG90s (180°)	3	R\$ 20,90	R\$ 62,70
Motor MG90s (360°)	1	R\$ 17,75	R\$ 17,75
Motor Shield	1	R\$ 29,95	R\$ 29,95
Placa Arduino Uno	1	R\$ 44,65	R\$ 44,65
Protoboard	1	R\$ 8,00	R\$ 8,00
Resistor	3	R\$ 0,15	R\$ 0,45
Sensor ultrassônico (HC-SR04)	1	R\$ 23,45	R\$ 23,45
			R\$ 485,55

FONTE: Arquivo original do autor.

3.6 Problemas enfrentados

Ao decorrer do desenvolvimento, o autor deparou-se com alguns problemas no projeto, que serão explicados neste tópico. Felizmente todas as dificuldades foram resolvidas e superadas antes da entrega.

Um destes problemas deve-se ao Bluetooth. Mesmo com a ciência de que esta tecnologia é definida por ser de curto alcance, enfrentou-se uma pequena dificuldade em relação ao controle e distância, perdendo facilmente a conexão com o módulo Bluetooth em longas distâncias, portando utilizou-se uma antena para ampliar a conexão, solucionando de maneira rápida e eficiente.

Outra dificuldade foi com o ajuste do código da garra robótica com o motor de rotação contínua, pois na fase de teste utilizaram-se servos motores 9g 180°, que logo apresentou problemas, devido ao baixo torque e resistência, precisando de uma modificação urgente para os servos motores MG90s 180° e 360°.

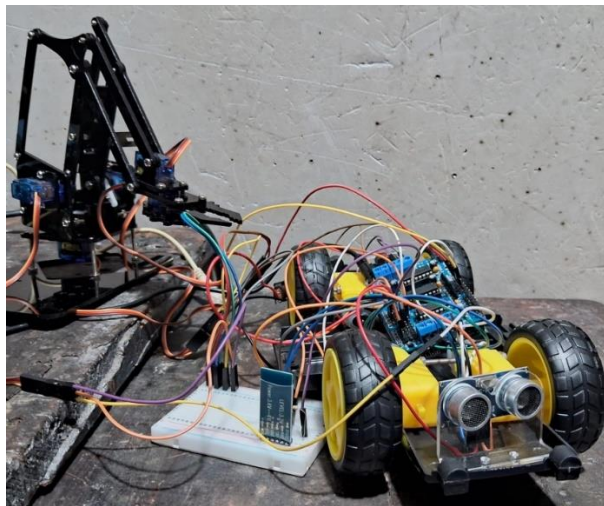
A principal dificuldade enfrentada foi a compatibilidade das bibliotecas (Servo.h – servos motores e AFMotor.h – motores DC) dos motores DC do carrinho com o da garra robótica, pelo fato de serem códigos prontos e de fácil uso, com funções predefinidas, portanto, utiliza-se alguns pinos como portas de saída iguais em ambas, ocasionando em conflito no controle e na disponibilidade de portas. A solução foi adaptar os pinos via código.

3.7 Protótipos

Com intuito de registrar todas as etapas de desenvolvimento do projeto, este tópico busca documentar todas as versões ou protótipos da G.A.R, com suas respectivas datas, alterações realizadas e fotos.

A figura 15 mostra a primeira versão da G.A.R, finalizada em 31/03/2024. Essa versão é um pilar fundamental para a conclusão e finalização do projeto. Em resumo, o aplicativo G.A.R_bluetooth já estabelece uma conexão com o protótipo, e exerce sua função de maneira eficiente. Algo a se ressaltar é que, a garra utilizada é diferente do protótipo final.

Figura 15–Protótipo G.A.R 31/03



FONTE: Arquivo original do autor.

A segunda versão da G.A.R foi finalizada em 13/06/2024. Essa etapa do projeto foi crucial para a compreensão da montagem, além de ajustes na programação. Houve a substituição da garra robótica por outra mais resistente, além da troca dos servos motores 9g por MG90s. O servo motor da base é o único de 360°, possibilitando uma rotação completa, já os demais são de 180°.

A figura 16 exibe a segunda versão do projeto G.A.R.

Figura 16 - Protótipo G.A.R 13/06



FONTE: Arquivo original do autor.

O protótipo fundamental está representado na figura 17, sendo a penúltima versão do projeto G.A.R. Através do controle remoto é possível controlar todas as funcionalidades com êxito. Dentro do carrinho há a placa do Arduino, juntamente com o módulo bluetooth, motor shield, buzzer, entre outros componentes e fios. A bateria está na cabine do motorista, fixada no topo. Em sua dianteira foram colocados o sensor ultrassônico e os LEDs (faróis). Atrás do carrinho há um botão liga/desliga do circuito, bem como um conector para o carregamento da bateria.

Figura 17 - Protótipo G.A.R 14/08



FONTE: Arquivo original do autor.

A figura 18 apresenta a versão final dos testes, com as seguintes adições: antena para aumento da propagação do sinal, contribuindo para controle do Bluetooth em distâncias maiores; peso na parte superior (cubo mágico), com intuito de evitar derrapagem do carrinho (centro de massa); acessórios, como adesivo e olhos móveis, ilustrando o sentimento de infância. Bem como alteração na lógica de alguns comandos, como as de direção para os lados direito e esquerdo, acionando agora todas as rodas (caso seja direita, as rodas da direita giram para trás, e as da esquerda giram para frente, sendo o oposto para esquerda), já que antes acionavam apenas as rodas do lado desejado, rodando-as para trás.

Figura 18 - Protótipo final G.A.R 24/10



FONTE: Arquivo original do autor.

4 INTENÇÕES FUTURAS

No cenário tecnológico atual, os jovens têm cada vez mais interesse e curiosidade pelas maravilhas da tecnologia e inovação. Com isso em mente, o autor pretende dar continuidade ao seu projeto, visando parcerias.

A G.A.R se destaca entre brinquedos eletrônicos, tendo também seu propósito como um diferencial: o aprendizado. Uma vez que haja um manual de instruções em mãos ou orientação adequada, qualquer pessoa pode fazer modificações para aprender e se divertir, sendo um produto adaptável.

Para seu objetivo principal ser concluído não há especificamente a necessidade de um profissional em robótica ou automação, apenas a força de vontade para aprender. E como recompensa desses estudos, o brinquedo proporcionará o entretenimento com suas diversas funções personalizáveis.

5 CONCLUSÃO

O registro e documentação do projeto tornam-se imprescindível a partir do momento em que se busca uma organização, bem como o entendimento de terceiros.

Houve diversos momentos em que o progresso do projeto estava reduzido, devido ao foco do autor em outras questões acadêmicas, mas com foco e vontade foram superados.

Em relação ao projeto, tudo aquilo almejado foi realizado com êxito, alguns aspectos com maiores dificuldades que outros, foi necessário determinação para continuar e aprimoramentos até o protótipo final. Foram meses dedicados do tempo do autor investidos nele, portanto o aprendizado e a experiência são eternos, e a certeza de que tudo aquilo prometido foi cumprido é gratificante.

Seu término foi possível graças ao apoio de docentes e familiares, tanto na colaboração e execução, quanto financeiramente, além da montagem, sendo lógico que a ausência de algum destes impactaria diretamente no produto final.

REFERÊNCIAS

CHASE, Otavio; ALMEIDA, F. Sistemas embarcados. **Mídia Eletrônica. Página na internet:**<[www. sbajovem. org/chase](http://www.sabajovem.org/chase)>, **capturado em**, v. 10, n. 11, p. 13, 2007.

<https://blog.eletrogate.com/o-que-e-robotica-conceito-historia-e-evolucao>. Acessado em 23/11/2024 às 12:31.

SILVA, Alzira Ferreira da. RoboEduc: uma metodologia de aprendizado com robótica educacional. 2009.

CAMPOS, Flavio Rodrigues. **A robótica para uso educacional**. Editora Senac São Paulo, 2019.

FENERICK, Jessica Aparecida; VOLANTE, Carlos Rodrigo. A evolução das indústrias, os benefícios da automação e as perspectivas do mercado da robótica no Brasil e no mundo. **Revista Interface Tecnológica**, v. 17, n. 1, p. 734-745, 2020.

SILVA, Victor Augusto Pimenta. **Robótica educacional para ensino stem: uma aplicação baseada em problemas**.

<https://repositorio.ifmg.edu.br/server/api/core/bitstreams/ecd62077-34bf-427c-a78a-95d0a482b953/content>. Instituto Federal De Educação, Ciência e Tecnologia de Minas Gerais, p. 17, 2024. Acessado em 23/11/2024 às 13:05.

MCROBERTS, Michael. **Arduino básico**. NovatecEditora, 2018.

<https://www.redhat.com/pt-br/topics/middleware/what-is-ide>. Acessado em 14/05/2024 às 08:41.

<https://www.locaweb.com.br/blog/temas/codigo-aberto/ide-o-que-e-um-ambiente-de-desenvolvimento-integrado>. Acessado em 09/04/2024 às 21:03.

<https://victorvision.com.br/blog/eletronica-digital/#:~:text=A%20principal%20diferen%C3%A7a%20entre%20elas,representar%20uma%20variedade%20de%20valores>. Acessado em 11/04/2024 às 09:52.

<https://www.makerhero.com/guia/componentes-eletronicos/resistor>. Acessado em 23/11/2024 às 14:24.

<https://tecnoblog.net/responde/o-que-e-led>. Acessado em 16/05/2024 às 10:23.

<https://tipotemporario.com.br/elektra/blog/voce-sabe-o-que-e-e-como-funciona-o-sensor-ultrassonico>. Acessado 16/05/2024 às 10:31.

<https://www.nepin.com.br/blog/solucoes-industriais/servomotor-o-que-e-e-como-funciona>. Acessado em 18/05/2024 às 10:37.

<https://autocorerobotica.blog.br/conhecendo-a-fundo-o-buzzer>. Acessado em 18/05/2024 às 20:21.

<https://curtocircuito.com.br/motor-shield-l293d-driver-ponte-h.html>. Acessado em 18/05/2024 às 20:49.

<https://www.usinainfo.com.br/modulo-bluetooth-arduino-528>. Acessado em 23/11/2024 às 15:11.

PUY, Inigo; HEINDL, Eduard. Bluetooth. **Hochschule Furtwangen University**, p. 1-20, 2008.

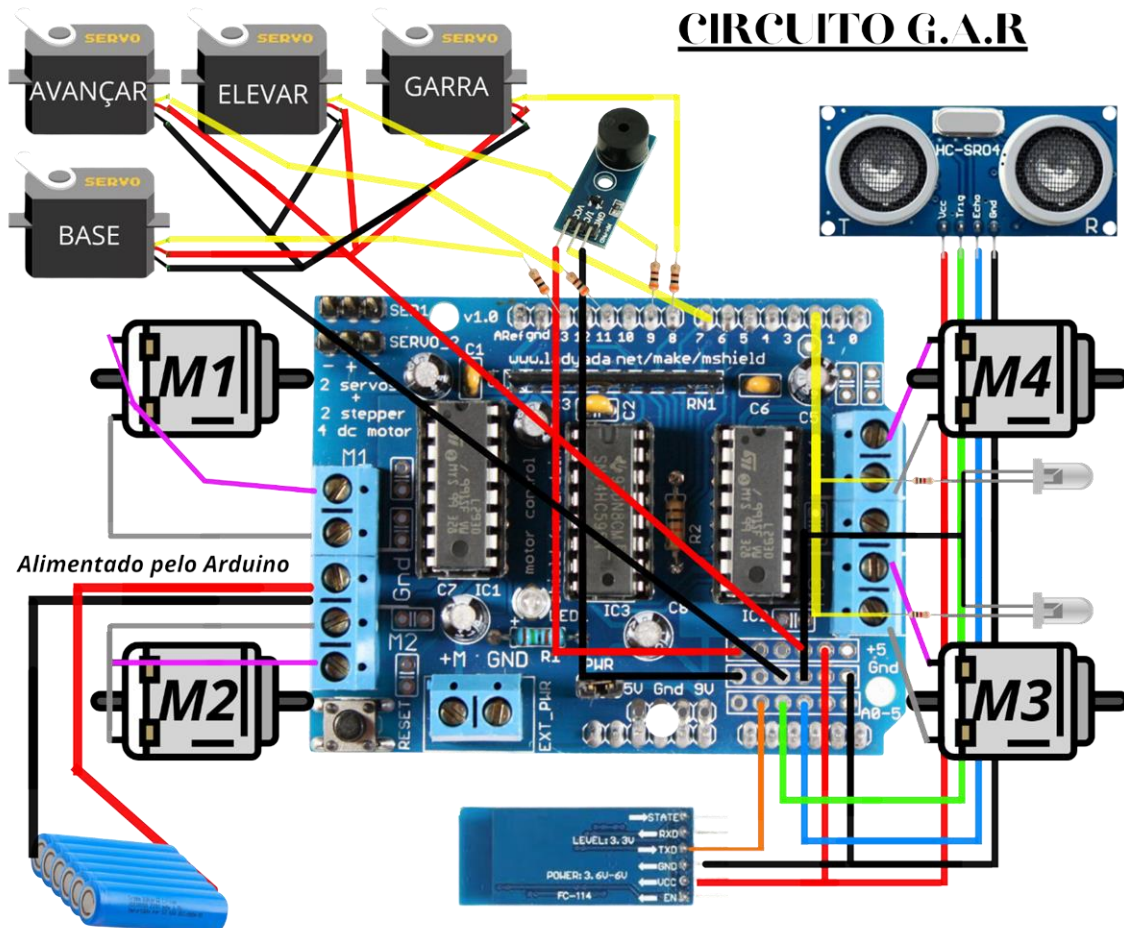
BLUETOOTH, S. I. G. Bluetooth technology. **Accessed: Apr**, 2020.

FEZARI, Mohamed; AL DAHOUD, Ali. Integrated development environment "IDE" for Arduino. **WSN applications**, v. 11, p. 1-12, 2018.

WOLBER, David et al. **App inventor**. " O'Reilly Media, Inc.", 2011.

APÊNDICE A – Esquema

Figura 19 - Esquema



FONTE: Arquivo original do autor.