

UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA

DEPARTAMENTO DE TECNOLOGIA

MI - PROJETO DE CIRCUITOS DIGITAIS

**JEP: ROBÔ DOMÉSTICO DA UEFS**

Enzo Cauã da S. Barbosa, Jamile Letícia C. da Silva, Pedro Lucas F. de Souza

**Tutor:** João Bosco Gertrudes e Wild Freitas da Silva Santos

**RESUMO**

Este relatório objetiva descrever o desenvolvimento de um robô doméstico em circuitos digitais em *Verilog Estrutural* no software Quartus e implementação em circuito integrado CPLD (Complex Programmable Logic Device) pertencente à família MAX II. O documento é dotado de referencial teórico acerca dos conceitos utilizados para o seu desenvolvimento, conceituando as portas lógicas e sua representação em álgebra booleana. Há descrição das soluções pensadas para atender aos requisitos do problema tal como a análise do circuito em nível geral e de grupos de portas lógicas para cada situação específica de movimento ou ausência de movimento do robô, além de estados de ligado/desligado e nível crítica de bateria. Essa descrição centra-se na definição das lógicas pertinentes aos grupos e as razões de sua utilização, a fim de otimizar o máximo possível dos requisitos solicitados no problema. Ademais, o documento define também as limitações do robô a nível lógico para determinadas situações.

## 1 INTRODUÇÃO

Ao longo da história, os seres humanos sempre buscaram formas e técnicas para otimizar e automatizar atividades recorrentes na vida cotidiana. Um exemplo histórico é a revolução agrícola, ocorrida no período neolítico, marcado pelo domínio do fogo pelos seres humanos e aperfeiçoamento das ferramentas desenvolvidas até então (SOUZA, 2024). Na era contemporânea, a revolução industrial foi o salto responsável por otimizar e automatizar ainda mais a execução de tarefas, delegando-as às máquinas mecânicas surgidas em meados dos séculos XVIII e XIX (CASTRO, 2024). Assim sendo, no início do século XXI, a revolução industrial entra em sua quarta fase<sup>1</sup>, marcada pelo advento do desenvolvimento digital e inovações como robôs, inteligência artificial e internet das coisas (JUNIOR, 2024).

Dessa forma, há intensa integração entre as esferas físicas, biológicas e digitais no mundo contemporâneo. Essa integração é tão forte e ocorre de tal modo que as tecnologias de otimização e automatização de tarefas têm adentrado aos lares, onde as máquinas digitais desempenham os mais diversos papéis dentro das casas, como assistência pessoal de inteligência artificial,

---

<sup>1</sup> Não há unanimidade entre acadêmicos sobre a quarta revolução industrial, mas é um artifício linguístico útil para destacar o surgimento da era digital.

eletrodomésticos inteligentes e robôs domésticos. Este último, por exemplo, exercendo tarefas de limpeza do ambiente doméstico.

Nesse sentido, os autores deste relatório foram convocados, por meio da disciplina projeto de circuitos digitais (TEC498), para projetar, em circuito digital, um protótipo de robô doméstico em hardware reconfigurável. As especificações do protótipo descrevem que este deve ser dotado de sensores infravermelhos para realizar análise do ambiente e detectar aproximações de objetos; além de rodas, movidas por motores, para locomoção. Ademais, o protótipo deve possuir um sensor para determinar criticidade da carga de bateria e botão de ligar/desligar, além de estabelecer lógica adequada para evitar colisões e retornar *Feedback* ao usuário em situações de erro ou descarga de bateria.

Para além disso, este relatório é estruturado em introdução, metodologia, que inclui referencial teórico e desenvolvimento, resultado e discussões, conclusão e referências. Tal estruturação objetiva explicar, de modo sucinto, os recursos e conceitos utilizados para o desenvolvimento do protótipo.

## 2 METODOLOGIA

### 2.0.1 REFERENCIAL TEÓRICO

O protótipo é desenvolvido em sistemas digitais, isto é, utiliza-se níveis de tensão elétrica que determinam o estado lógico de operações. Esses estados são representados de forma binária, com utilização de uma base numérica dotada de dois símbolos: 0 e 1, sendo estes adotados, respectivamente, como falso e verdadeiro. Esses valores são chamados de níveis lógicos e relacionam-se com níveis de tensão em um circuito digital de modo que o 0 represente nível de tensão baixo; e 1, alto (TOCCI, 2011).

Nesse sentido, em termos abstratos, os níveis lógicos podem relacionar-se entre si para produzir saídas determinadas por operações lógicas, tal como a lógica clássica (IEZZI, 2013). Em circuito digital, essas operações são realizadas por portas lógicas, responsáveis por produzir uma saída lógica a partir da combinação operacional das entradas, como ilustra a figura 1.

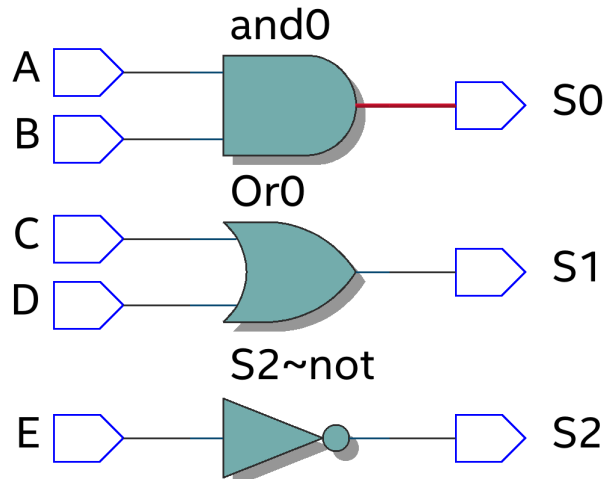


Figura 1 – Portas lógicas básicas. Fonte: Própria. 2024

Observa-se que, nas portas lógicas nomeadas *and0* e *Or0*, são dadas duas variáveis de entrada que, combinadas, produzem uma determinada saída dependente do tipo de operação realizada pela lógica da porta. A outra porta é nomeada *S2 not* e nela se é dada uma única entrada e produzida uma única saída. Cada uma dessas portas realiza operações e produzem saídas que dependem das relações de níveis de tensão entre suas variáveis de entrada (TOCCI, 2011). Conclui-se, portanto, que é possível contabilizar possíveis entradas para obter diversas saídas a depender das portas lógicas utilizadas. Como as entradas, independente da quantidade, podem assumir apenas dois níveis de tensão e, considerando uma quantidade  $n$  de variáveis, a quantidade de possíveis combinações para produção de determinadas saídas é dada por  $Q(n) = 2^n$ , sendo  $Q(n)$  a função de quantidade de combinações e saídas para os níveis lógicos das variáveis de entrada (TOCCI, 2011). Para quantificar e avaliar as saídas das combinações é possível utilizar uma tabela verdade, exemplificada na figura 2, que quantifica as possíveis entradas e as possíveis saídas dado um circuito lógico.

As operações realizadas pelas portas lógicas são equivalentes às operações proposicionais da lógica clássica. As portas *And*, *Or* e *Not* são conhecidas como conjunção, disjunção e negação (IEZZI, 2013).

AND		
A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

OR		
C	D	S
0	0	0
0	1	1
1	0	1
1	1	1

NOT	
A	S
0	1
1	0

Figura 2 – Tabela verdade das operações lógicas básicas. Fonte: Própria. 2024

A operação de conjunção dada pela porta *And*, como se observa na figura 2, cujas entradas estão em azul, produz saída, em verde, somente quando os níveis de tensão de ambas as variáveis de entrada estão em nível alto. Essa operação pode ser generalizada para  $n$  entradas (TOCCI, 2011). A operação de disjunção, representada pela porta *Or*, diferentemente da conjunção, produz nível alto de tensão em sua saída se pelo menos uma das variáveis estiverem em nível de tensão alto, como se pode observar na figura 2. Por fim, a negação simplesmente inverte o nível de tensão na saída: se o nível de tensão é alto, a saída produz saída baixa; se for baixo, a saída produz alto.

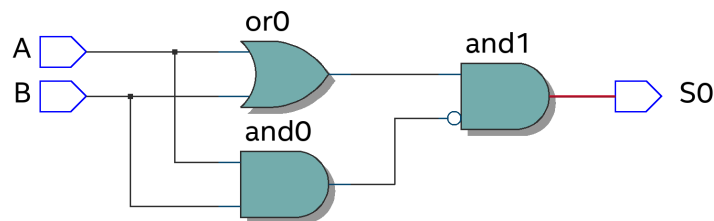


Figura 3 – Porta XOR criada pela combinação das portas lógicas básicas. Fonte: Própria. 2024

Essas operações podem ser combinadas para produzir novas portas lógicas. Por exemplo, pode-se combinar as portas lógicas básicas para gerar a porta de *Ou Exclusivo*, ou *XOR*, como ilustra a figura 3<sup>1</sup>. Portanto, qualquer operação lógica é dada pela combinação dos operadores lógicos básicos (TOCCI, 2011).

<sup>1</sup> O círculo no final da porta *and1* é outra simbologia para porta inversora.

Além do aspecto visual, os circuitos lógicos podem ser descritos em notação algébrica, sendo esta dotada de certas propriedades fundamentadas por teoremas. Essa notação é denominada de *Álgebra Booleana*. Na notação dessa álgebra, as portas lógicas são representados por símbolos mostrados na tabela 1 (TOCCI, 2011).

Porta lógica	Símbolo	Combinação
AND	$\cdot$	$A \cdot B$
OR	$+$	$A + B$
NOT	$-$	$\bar{A}$

Tabela 1 – Tabela de símbolos booleanos para portas lógicas.

$$S = (A + B) \cdot (\overline{A \cdot B}) \quad (2.1)$$

Dessa maneira, a saída do circuito mostrado na figura 3 pode ser expresso como mostrado na equação 2.1, que é um exemplo de expressão de circuito lógico em álgebra booleana.

Propriedade	Notação
Comutativa(+)	$x + y = y + x$
Comutativa( $\cdot$ )	$x \cdot y = y \cdot x$
Associativa(+)	$x + (y + z) = (x + y) + z = x + y + z$
Associativa( $\cdot$ )	$x \cdot (y \cdot z) = (x \cdot y) \cdot z = x \cdot y \cdot z$
Distributiva(a)	$x \cdot (y + z) = x \cdot y + x \cdot z$
Distributiva(b)	$(w + x) \cdot (y + z) = w \cdot y + x \cdot y + w \cdot z + x \cdot z$

Tabela 2 – Tabela de propriedades da álgebra booleana.

A álgebra booleana é dotada de certas propriedades. Tomando  $x$ ,  $y$  e  $z$  como variáveis de entrada, a tabela 2 mostra as propriedades algébricas das operações lógicas em álgebra booleana (TOCCI, 2011). Desse modo, qualquer circuito lógico pode ter sua saída expressa em álgebra booleana e pode ser simplificada pelas propriedades apresentadas, gerando assim um circuito lógico mais conciso que demande de menos portas lógicas para produzir as saídas requisitadas (TOCCI, 2011).

Ademais, os recursos visuais e algébricos supracitados foram utilizados para o desenvolvimento do protótipo a nível lógico, sendo os principais protagonistas do desenvolvimento do produto.

## 2.0.2 DESENVOLVIMENTO

O protótipo foi desenvolvido em circuito integrado CPLD (Complex Programmable Logic Device) pertencente à família MAX II, modelo EPM240T100C5N, disponível no Laboratório

de Eletrônica Digital e Sistemas (LEDS) da Universidade Estadual de Feira de Santana (DIAS, 2024). A placa é denominada CPLD-LEDS. O protótipo é projetado em linguagem de descrição de hardware *Verilog Estrutural* no software Quartus (Intel, 2024). O *Verilog Estrutural* é adotado mundialmente como linguagem de descrição de hardware e possui dependências para declarar portas lógicas e permitir associações entre entradas, fios e saídas (TOCCI, 2011).

O desenvolvimento do protótipo baseou-se na combinação lógica das portas a partir das entradas requisitadas no problema. Tais entradas simulam os sensores do robô, que direcionam sua movimentação, o botão de ligar/desligar e o nível da bateria. Dessa maneira, o protótipo foi projetado para receber 6 entradas de variáveis, combinadas logicamente para compor as possíveis saídas discutidas nas sessões tutoriais.

As saídas foram pensadas para simular situações-chave em que o robô poderia encontrar-se, como movimentação contínua com todos os sensores desligados, permitindo um trajeto mais livre, ou com pelo menos um de seus sensores acionados, que são logicamente computados para produzir saídas específicas da movimentação do robô no ambiente. As saídas também sinalizam quais rodas estão acionadas, além do sistema simbólico de *feedback* para o usuário.

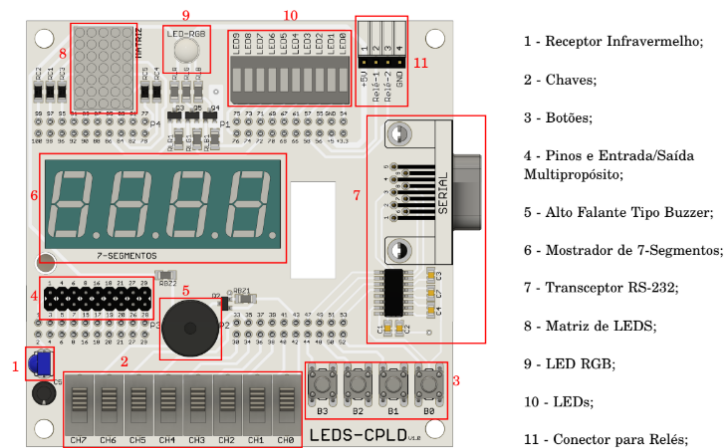


Figura 4 – Visão geral da placa CPLD-LEDS. Fonte: DIAS. 2024.

Nesse sentido, os principais componentes eletrônicos dispostos na placa CPLD-LEDS, mostrado na figura 4, foram os chaves (2), os LEDs (10), a matriz de LEDs (10) e o LED RGB (9).

Chaves	Entrada
CH0	Sensor traseiro
CH1	Sensor frontal
CH2	Sensor lateral esquerdo
CH3	Sensor lateral direito
CH4	Nível de bateria
CH5	Botão de ligar/desligar

Tabela 3 – Tabela de correspondência entre canais e entrada.

As chaves são responsáveis por simular as entradas do robô, onde as possíveis combinações dos estados das chaves (0 ou 1) determinam saídas diferentes. As chaves são pinadas conforme os dados da tabela 3. Pela quantidade de entradas, considera-se 64 possibilidades de saída. Dessas, 32 possibilidades representam o estado em que o botão de ligar/desligar está em estado 0 e em lógica conjuntiva com os demais componentes lógicos, isto é, o protótipo não realiza simulações de movimentação. Além disso, a bateria apresenta lógica semelhante, em lógica conjuntiva com os demais componentes do circuito. Isso significa que o nível de bateria e o botão de ligar/desligar são hierarquicamente superiores em relação ao restante do circuito. Nesse sentido, o robô inicia seu movimento até que haja três condições: botão de ligar/desligar em 0; nível de bateria em estado 1, indicando bateria em estágio crítico, e ausência de movimento do robô. As sinalizações para as duas últimas situações supracitadas estão descritas na figura 5.

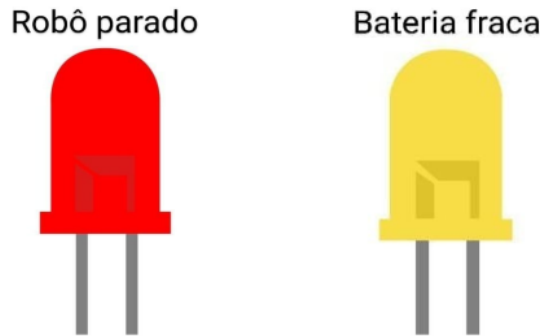


Figura 5 – Sinalização da bateria e ausência de movimento do robô. Fonte: Própria. 2024

Para as questões de movimento, o protótipo sempre segue a direção frontal caso nenhum dos sensores esteja ativado. O robô só adota outra direção caso os sensor frontal esteja ativado, adotando uma direção lateral para a direita, que é a direção padrão. O robô só adota movimentação para a esquerda caso o sensor da direita esteja ativado. Para ver as sinalizações de movimentação, veja o anexo A. Se todos os sensores estão ativados, o robô fica parado. Além disso, o robô rotaciona para a direita caso os sensores laterais e frontal estejam ativados.

A direção do robô relaciona-se com os motores ativados. Há três cenários: caso o robô esteja em movimento em direção frontal, ambos os motores, direito e esquerdo, são ativados; caso vire para a direita, o motor esquerdo é ativado enquanto o da direita é desativado; e, por fim, caso vire para a esquerda, ativa-se o motor direito e desativa o esquerdo. As sinalizações do movimento estão descritos no anexo A deste documento.

### 3 RESULTADOS E DISCUSSÕES

#### 3.0.1 ASPECTO TÉCNICO

A nível de desenvolvimento técnico, o código em linguagem de descrição de hardware foi construído em um único módulo de código, responsável por declarar as entradas e saídas, relacionadas por declarações de portas lógicas. Tal módulo pode ser separado em grupos de portas lógicas. Esses grupos são as portas de conjunção de sensores; as direcionadoras, que alteram a direção do robô; as de verificação das possíveis direções; as verificadoras de bateria; as acionadoras das rodas; e, por fim, as acionadores de sinalizações.

O grupo de conjunção de sensores é essencialmente construída pela lógica conjuntiva, que agem conjuntivamente para determinar a direção lateral do robô, as saídas dessas portas determinam a nova direção do robô através das portas direcionadoras.

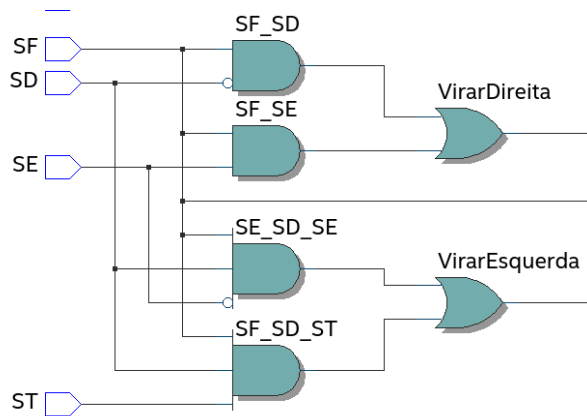


Figura 6 – Grupos de conjunção de sensores e direcionadores. Fonte: Própria. 2024

Considerando a direção frontal como movimento padrão em caso de estado 0 em todos os sensores, o movimento segue tal direção até que os demais sensores mudam para o estado 1. Como mostrado na figura 6, o robô só adota direção lateral direita em duas situações: a primeira situação é sensor frontal (*SF*) esteja em estado 1; e o sensor lateral direito (*SD*) esteja em estado 0<sup>1</sup>. Outra situação é quando há conjunção entre os sensores frontais e lateral esquerdo (*SE*), como mostrado na figura 6.

Em caso de direcionamento para a esquerda, a lógica conjuntiva é pertinente e, analogamente, só ocorre em duas condições. Na primeira, o robô só vira para a esquerda caso os sensores

<sup>1</sup> Note que a negação do estado 0 é 1, satisfazendo a condição da porta lógica e alterando a direção.



frontais e lateral direito estejam estado 1; e o esquerdo, em estado 0. Na segunda, observa-se a relação conjuntiva dos sensores frontais, lateral direito e traseiro ( $ST$ ), onde nessa situação apenas o sensor da esquerda está em estado 0.

Nota-se que as direcionadoras obedecem à lógica disjuntiva, pois suas entradas são essencialmente as saídas das portas conjuntivas dos sensores. Essa lógica é essencial para representar o estado de movimento para as laterais independente de qual da situação supracitada foi adotada, como demonstra a figura 6. Essencialmente, a conjunção de sensores e direcionadores obedecem as expressões do tipo  $(A \cdot B)$  e  $(C + D)$ , sendo  $A$ ,  $B$ ,  $C$  e  $D$  entradas abstratas.

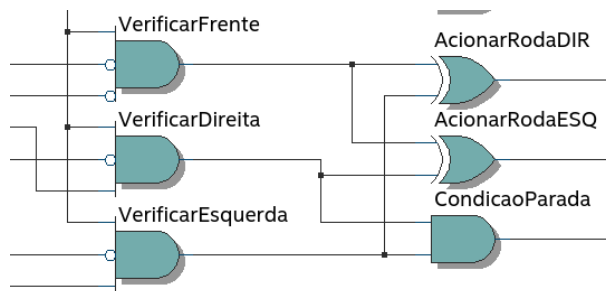


Figura 7 – Grupos de verificadores e acionadores das rodas. Fonte: Própria. 2024

O grupo de verificadores descritos, mostrados mais à direita na figura 7, é responsável essencialmente por determinar o estado dos motores e a condição de parada. Essa verificação é essencial para determinar quais motores serão acionados ou se o robô entrará em uma condição de parada. Observa-se que os verificadores possuem essencialmente três entradas em lógica conjuntiva. Há duas entradas comuns nos verificadores: a primeira é a negação do estado da bateria, isso é importante porque o robô só aciona seus motores caso o nível de bateria não esteja em nível crítico, isto é, em estado 1. A outra é o estado de ligado/desligado do robô, que vai determinar o acionamento das rodas. Nesse sentido, a figura 8 referencia os verificadores de bateria e de estado ligado/desligado, que são lógicas hierarquicamente superiores, visto que são elas que definem os estados de outros grupos para realização do movimento do robô.

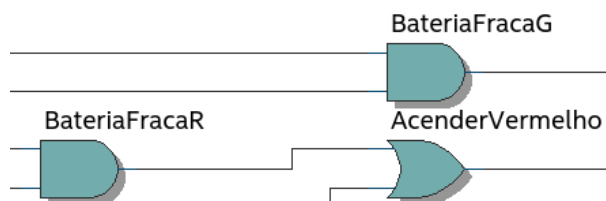


Figura 8 – Verificadores da bateria e entrada ligado/desligado. Fonte: Própria. 2024

Os acionadores de sinalizações, descritos na figura 9, são conjunções dos acionadores das rodas. Nota-se que estes são determinados em lógica de *OU EXCLUSIVO*, porque isso garante uma rotação para a direita apenas se ao menos um exclusivamente esteja em estado 1 numa situação de dualidade de possibilidade de rota. Os acionadores das rodas passam pelos acionadores

de sinalizações, que possuem a entrada comum da negação da condição de parada. Isso porque, caso esteja em estado 1, ele emite diretamente a sinalização de parada, não representando as sinalizações de outros movimentos, já que os acionadores de sinalizações são fundamentalmente conjunções. Para mais detalhes, o circuito completo está no anexo B.

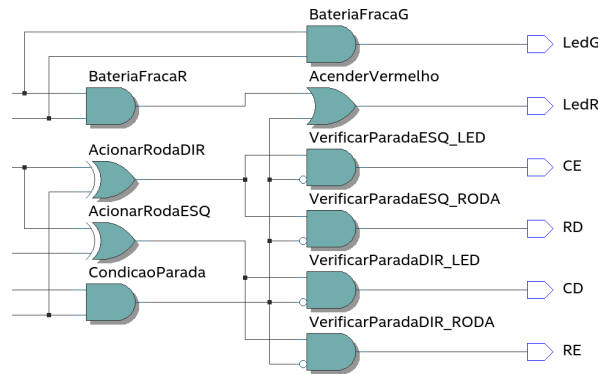


Figura 9 – Acionadores de sinalizações. Fonte: Própria. 2024

## 4 CONCLUSÃO

Em função do que foi relatado e considerando sua estrutura lógica, o produto mantém-se funcional visto que atende aos requisitos solicitados, como movimentações para quatro direções e estabelecimento de condições para garantir a relação usuário-máquina a fim de tornar o produto interativo e intuitivo para o uso nos lares.

A solução proposta mostra-se eficaz, porém é válido apontar algumas limitações e não demonstra eficiência em sua construção para situações mais complexas, como espaços com escadas e definição de trajetos de modo a tornar a limpeza do ambiente mais eficiente, como computar e armazenar as informações de espaços já limpos pelo robô a fim de evitar que ele passe por esses espaços, assim proporcionando uma eficiência maior de limpeza no espaço.

## REFERÊNCIAS

- CASTRO, L. L. de. *Revolução Industrial*. 2024. Toda Matéria. Disponível em: <<https://www.todamateria.com.br/revolucao-industrial/>>. Acesso em: 13 de Setembro de 2024. Citado na página 1.
- DIAS, A. M. *Kit de desenvolvimento LEDS-CPLD*. [S.l.], 2024. Disponível em: <[https://drive.google.com/file/d/1mqm-I2mHjwPsXzaypq\\_S3-uXIpvc42o5/view](https://drive.google.com/file/d/1mqm-I2mHjwPsXzaypq_S3-uXIpvc42o5/view)>. Acesso em: 13 de Setembro de 2024. Citado na página 6.
- IEZZI, G. *Fundamentos da Matemática Elementar. Ed 9ª*. [S.l.]: Atual, 2013. v. 1. Citado 2 vezes nas páginas 2 e 3.
- Intel. *Software de projeto Quartus Prime*. [S.l.], 2024. Disponível em: <<https://www.intel.com.br/content/www/br/pt/products/details/fpga/development-tools/quartus-prime.html>>. Acesso em: 13 de Setembro de 2024. Citado na página 6.
- JUNIOR, J. M. F. *Quarta Revolução Industrial*. 2024. Brasil Escola. Disponível em: <<https://brasilecola.uol.com.br/historiag/quarta-revolucao-industrial.htm>>. Acesso em: 13 de Setembro de 2024. Citado na página 1.
- SOUZA, T. *Revolução Agrícola*. 2024. Toda Matéria. Disponível em: <<https://www.todamateria.com.br/revolucao-agricola/>>. Acesso em: 13 de Setembro de 2024. Citado na página 1.
- TOCCI, R. J. *Sistemas Digitais - Princípios e Aplicações. Ed 11ª*. [S.l.]: Pearson Prentice Hall, 2011. Citado 5 vezes nas páginas 2, 3, 4, 5 e 6.

## **Anexos**

A ANEXO

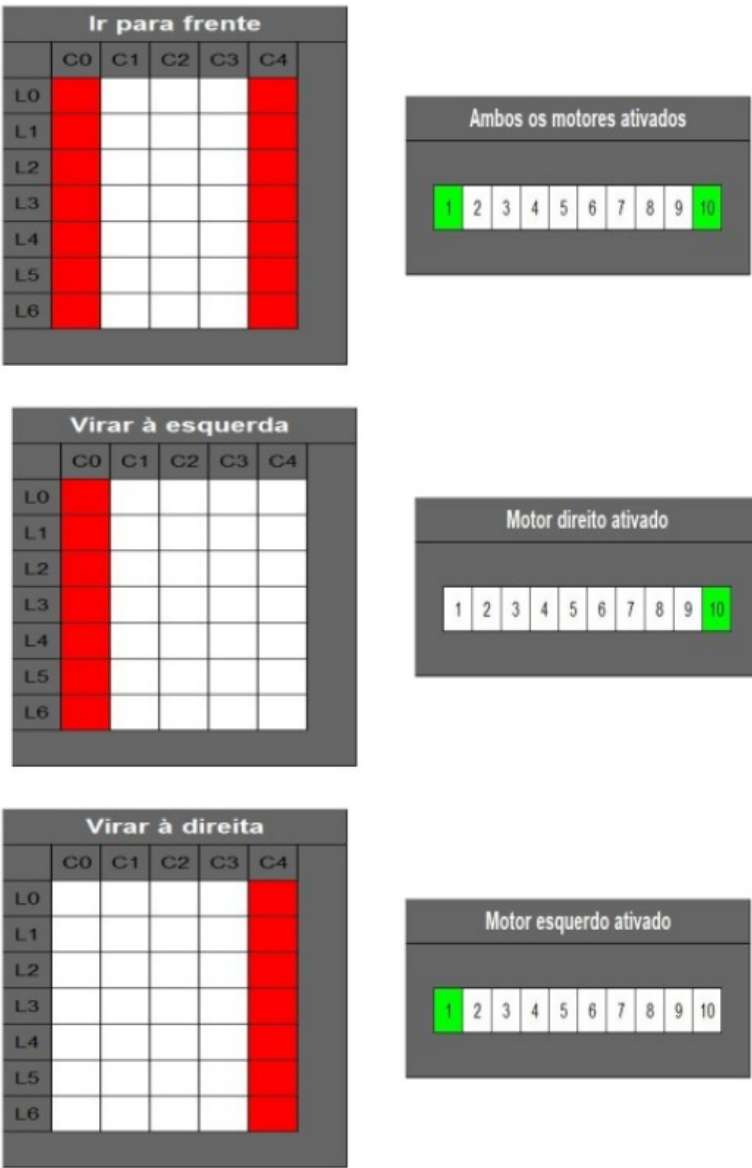


Figura 10 – Sinalizações de movimentação. Fonte: Própria. 2024

## B ANEXO

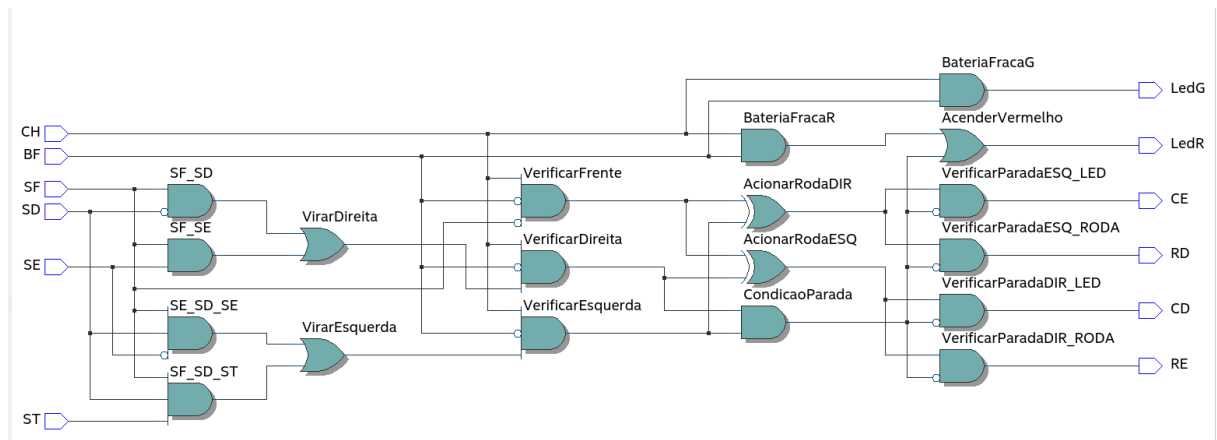


Figura 11 – Circuito lógico completo do robô. Fonte: Própria. 2024