

ROBOT BOOK



CRESCE
NDO



SUMÁRIO

I.SOBRE A MTECH.....	3
II.ROBO CAPITECH.....	4
III.SOBRE A TEMPORADA.....	5
IV.ESTRATÉGIAS.....	6
V.CHASSI.....	7
VI.INTAKE.....	8
VII.SHOOTER	9
VIII.MECÂNICA.....	10
IX.ELÉTRICA(PROGRAMAÇÃO)...	11
X.ELÉTRICA(AUTÔNOMO).....	12
XI.INTEGRANTES	13
XI. LINK TREE	14

INTRODUÇÃO

SOBRE A MTECH...

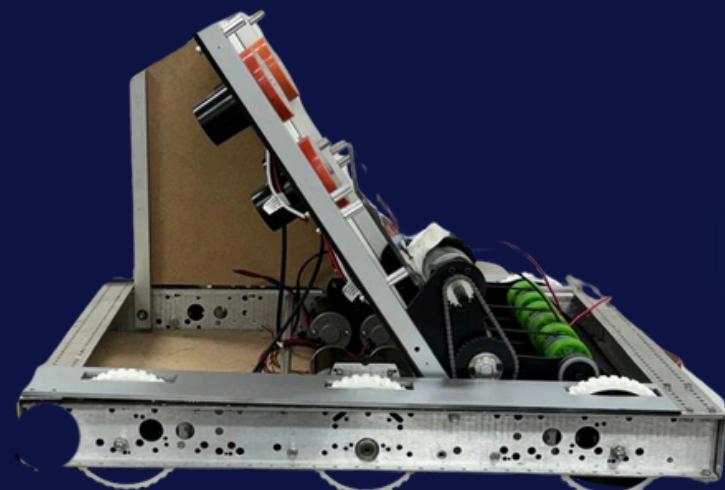
Na MTECH, equipe da FIRST ROBOTICS COMPETITION (FRC), priorizamos coesão, colaboração, e organização. Cada membro contribui com suas habilidades para alcançar objetivos comuns. Nós, valorizamos a comunicação eficaz interna e externamente. Liderança forte e habilidades de resolução de problemas são essenciais para superar desafios técnicos e estratégicos. Trabalho em equipe, criatividade e perseverança impulsionam nosso sucesso na tão aclamada FRC.



CAPITECH

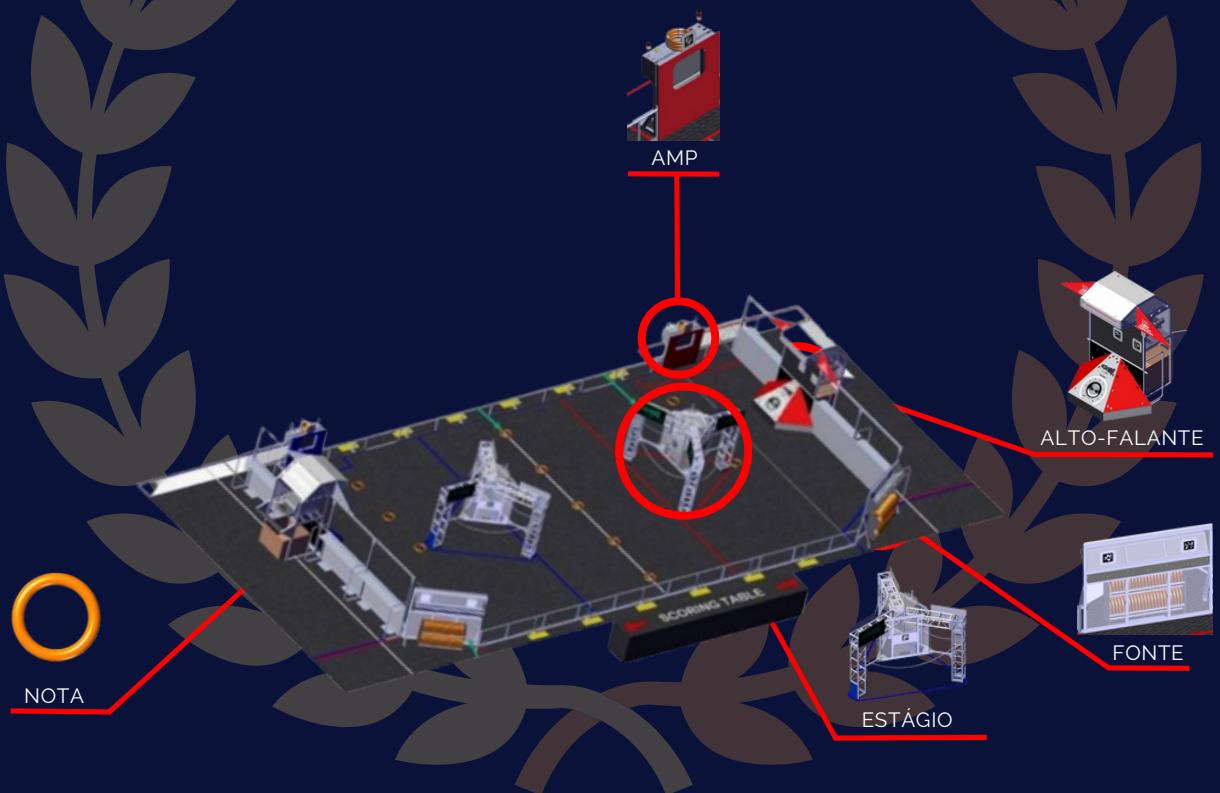
ORIGEM OU RAZÃO OU POR TRÁS DO NOME “CAPITECH”

O nome “ CAPITECH ” é uma combinação da palavra “ capivara ” e “ technology ” . A escolha desse nome é uma referência à capivara, que é o mascote da esquipe MTECH #9603, combinada com o termo “ tech ”, indicando o foco em tecnologia e inovação por parte da equipe.



TEMPORADA CRESCENDOSM

Sobre a competição deste ano, o tema central é música. Os participantes devem utilizar equipamentos como alto-falantes, amplificadores e o palco para apresentar suas performances e conquistar pontos.



ESTRATÉGIAS

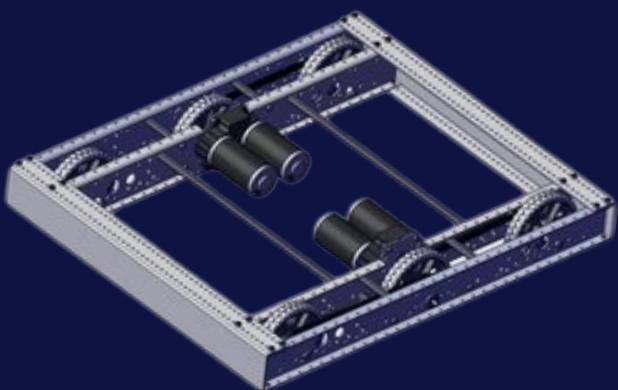
Em alguns jogos, os robôs da equipe da MTECH podem ser designadas para proteger áreas específicas do campo ou para evitar que os oponentes pontuem. Para alcançar esse objetivo, nossa equipe implementa estratégias defensivas que podem incluir bloqueio de caminhos vitais ou interferências direta com os robôs adversários, garantindo assim uma vantagem competitiva durante as partidas.

OBJETIVOS DA MTECH NO JOGO:

- Defender e proteger alvos
- Colaborar com outros robôs
- Mover-se com eficiência e precisão
- Adaptar-se a diferentes situações

CHASSI

Escolhemos o Chassi (AM14U5) Long como a base estratégica e robusta para nosso robô na equipe TuiuTech. Com suas dimensões expandidas, oferece uma plataforma sólida e estável para nossa estratégia de jogo. Aproveitamos o tamanho e a robustez do nosso robô para dificultar o avanço da outra aliança, limitando sua liberdade de movimento pelo campo. Este chassis é conhecido por sua resistência e durabilidade, garantindo que nosso robô possa lidar com as exigências e desafios da competição. Com o Chassi (AM14U5) Long, estamos preparados para enfrentar qualquer desafio no campo com confiança e determinação.



IN TAKE

DESING DA CAPITECH

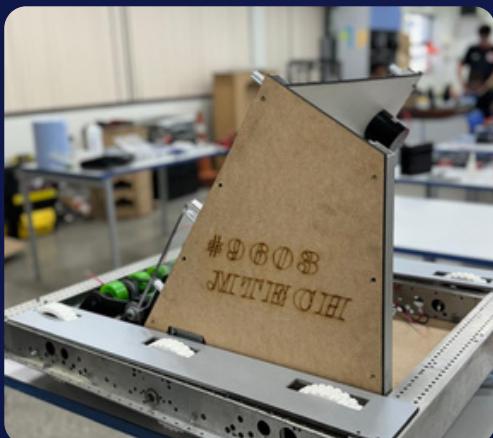
Como uma equipe rookie, projetamos um robô capaz de coletar uma peça de jogo do chão e lançá-la no alvo designado. Aprimoramos o conceito do Kitbot, otimizando a estrutura e os sistemas de coleta e lançamento. Nosso robô também é eficaz em funções defensivas, dificultando as estratégias dos oponentes. Fizemos ajustes precisos nas dimensões e parâmetros de desempenho para maximizar sua eficiência tanto em tarefas ofensivas quanto defensivas.

INFORMAÇÕES DO ROBÔ

- Altura 64cm
- Perímetro 302,1 cm
- Peso aprox. 50kg

SHOOTER

Nosso shooter, alimentado pelo intake e ajustado para pontuar no speaker, possui uma estrutura robusta com base de metalon. Utilizamos uma chapa traseira de MDF de 6mm e uma chapa frontal de policarbonato para equilibrar rigidez e absorção de impacto. Com quatro motores NEO, o sistema garante durabilidade, força e estabilidade, essenciais para disparos precisos e consistentes. O design inteligente e componentes de alta qualidade asseguram eficiência em todas as partidas.



MECÂNICA DESING DA CAPITECH

Como uma equipe rookie, projetamos um robô capaz de coletar uma peça de jogo do chão e lançá-la no alvo designado. Aprimoramos o conceito do Kitbot, otimizando a estrutura e os sistemas de coleta e lançamento. Nossa robô também é eficaz em funções defensivas, dificultando as estratégias dos oponentes. Fizemos ajustes precisos nas dimensões e parâmetros de desempenho para maximizar sua eficiência tanto em tarefas ofensivas quanto defensivas.

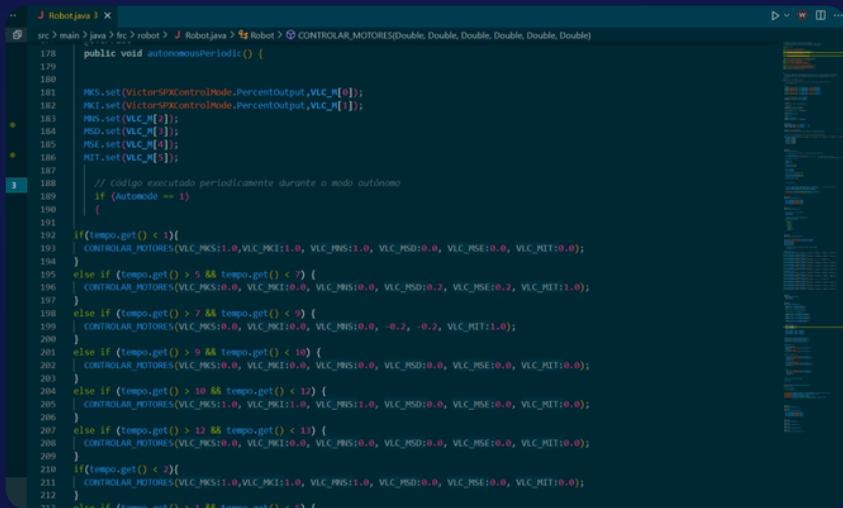
INFORMAÇÕES DO ROBÔ

- Altura 64cm
- Perímetro 302,1 cm
- Peso aprox. 50kg

ELETRÔNICA

PROGRAMAÇÃO

A linguagem Java é usada pela nossa equipe para programar o robô, aproveitando sua versatilidade e robustez. A sintaxe clara e o ambiente de desenvolvimento bem estabelecido do Java facilitam a criação e manutenção de códigos e sistemas complexos. Nossa equipe pode criar soluções complexas e confiáveis para controlar o robô com precisão com Java, que tem recursos avançados de orientação a objetos e uma grande biblioteca de classes. Essa linguagem melhora o desempenho e facilita a integração com outros sistemas e tecnologias.

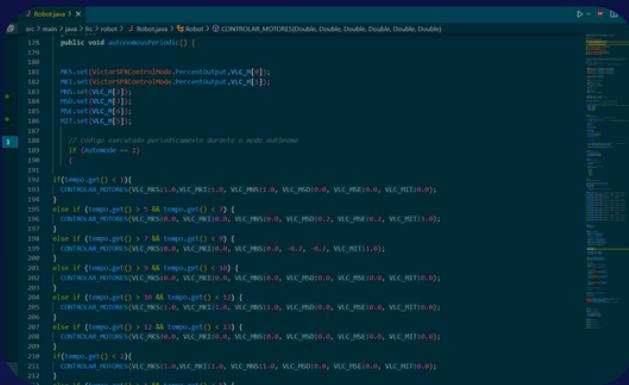


```
 178 public void autonomousPeriodic() {
179
180     MKS.set(VictorSPControlMode.PercentOutput,VLC_M[0]);
181     MKS.set(VictorSPControlMode.PercentOutput,VLC_M[1]);
182     MKS.set(VLC_M[2]);
183     MKS.set(VLC_M[3]);
184     MKS.set(VLC_M[4]);
185     MKS.set(VLC_M[5]);
186
187     // Código executado periodicamente durante o modo autônomo
188     if (Automode == 1) {
189
190
191     if(tempo.get() < 1){
192         CONTROLAR_MOTORES(VLC_MKS:1.0, VLC_MK:1.0, VLC_MS:1.0, VLC_MSD:0.0, VLC_MSE:0.0, VLC_MIT:0.0);
193     }
194     else if (tempo.get() > 1 && tempo.get() < 2) {
195         CONTROLAR_MOTORES(VLC_MKS:0.0, VLC_MK:0.0, VLC_MS:0.0, VLC_MSD:0.2, VLC_MSE:0.2, VLC_MIT:1.0);
196     }
197     else if (tempo.get() > 2 && tempo.get() < 3) {
198         CONTROLAR_MOTORES(VLC_MKS:0.0, VLC_MK:0.0, VLC_MS:0.0, -0.2, -0.2, VLC_MIT:1.0);
199     }
200     else if (tempo.get() > 3 && tempo.get() < 10) {
201         CONTROLAR_MOTORES(VLC_MKS:0.0, VLC_MK:0.0, VLC_MS:0.0, VLC_MSD:0.0, VLC_MSE:0.0, VLC_MIT:0.0);
202     }
203     else if (tempo.get() > 10 && tempo.get() < 12) {
204         CONTROLAR_MOTORES(VLC_MKS:1.0, VLC_MK:1.0, VLC_MS:1.0, VLC_MSD:0.0, VLC_MSE:0.0, VLC_MIT:0.0);
205     }
206     else if (tempo.get() > 12 && tempo.get() < 13) {
207         CONTROLAR_MOTORES(VLC_MKS:0.0, VLC_MK:0.0, VLC_MS:0.0, VLC_MSD:0.0, VLC_MSE:0.0, VLC_MIT:0.0);
208     }
209     if(tempo.get() < 2){
210         CONTROLAR_MOTORES(VLC_MKS:1.0, VLC_MK:1.0, VLC_MS:1.0, VLC_MSD:0.0, VLC_MSE:0.0, VLC_MIT:0.0);
211     }
212     else if (tempo.get() < 1 && tempo.get() < 3) {
```

ELÉTRICA

MODO AUTÔNOMO

O robô na forma autônoma por meio de uma série de instruções programadas por tempo, sem controle manual. O código fornecido configura minuciosamente o robô para este modo de operação.



```
1 // No 2 main > java -D2 robot > J Robot.java > Robot > CONTROLAR_MOTORES(double, Double, Double, Double, Double, Double)
2
3 public void autonomousInit() {
4
5     double[] pos = {0, 0, 0, 0, 0, 0};
6
7     pos[0] = VLC_0.set(VLC_0.getPos());
8     pos[1] = VLC_1.set(VLC_1.getPos());
9     pos[2] = VLC_2.set(VLC_2.getPos());
10    pos[3] = VLC_3.set(VLC_3.getPos());
11    pos[4] = VLC_4.set(VLC_4.getPos());
12    pos[5] = VLC_5.set(VLC_5.getPos());
13
14    pos[0] = VLC_0.set(VLC_0.getPos());
15    pos[1] = VLC_1.set(VLC_1.getPos());
16    pos[2] = VLC_2.set(VLC_2.getPos());
17    pos[3] = VLC_3.set(VLC_3.getPos());
18    pos[4] = VLC_4.set(VLC_4.getPos());
19    pos[5] = VLC_5.set(VLC_5.getPos());
20
21    // Códigos executados periodicamente durante o modo autônomo
22    if (autonomia == 1) {
23
24        if (tempo.get() < 1) {
25            CONTROLAR_MOTORES(VLC_PMS1(1,0), VLC_PMS1(1,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_MET1(0,0));
26
27            if (tempo.get() < 5 && tempo.get() > 0) {
28                CONTROLAR_MOTORES(VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_MET1(0,0));
29            }
30            if (tempo.get() > 7 && tempo.get() < 9) {
31                CONTROLAR_MOTORES(VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), -0,2, -0,2, VLC_MET1(0,0));
32            }
33            if (tempo.get() > 9 && tempo.get() < 10) {
34                CONTROLAR_MOTORES(VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_MET1(0,0));
35            }
36            else if (tempo.get() > 10 && tempo.get() < 12) {
37                CONTROLAR_MOTORES(VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_MET1(0,0));
38            }
39            else if (tempo.get() > 12 && tempo.get() < 13) {
40                CONTROLAR_MOTORES(VLC_PMS1(1,0), VLC_PMS1(1,0), VLC_PMS1(0,0), VLC_PMS1(0,0), VLC_MET1(0,0));
41            }
42        }
43    }
44}
```

Método "autonomousInit" é chamado quando o modo autônomo começa. Essa técnica reinicia um cronômetro e escolhe o melhor modo de operação usando o que foi selecionado no painel de controle do dashboard. O robô fará uma variedade de movimentos dependendo da escolha.

O método "autonomousPeriodic" é chamado repetidamente para controlar os motores do robô durante a execução autônoma. O cronômetro é usado para determinar qual comando deve ser enviado no início do modo autônomo.

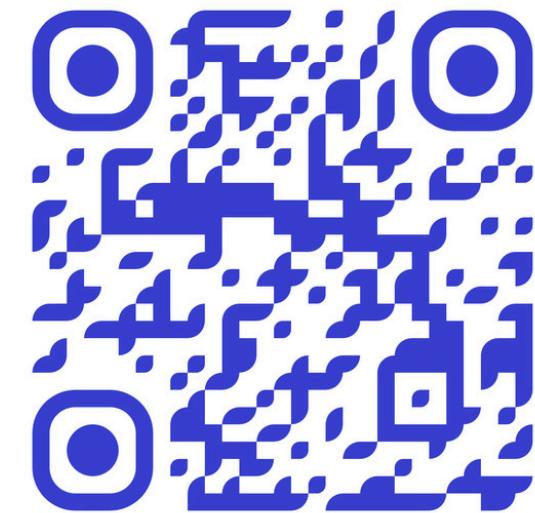
INTEGRANTES



SESI

SENAI

CONHEÇA MAIS



A EQUIPE MTECH!