

# Motores

- [Conexão elétrica](#)
- [Programação](#)

# Conexão elétrica

---

A presente seção tem como objetivo explicar a conexão elétrica dos motores, siga-a até o final caso queira movimentar o seu.

## Cabos

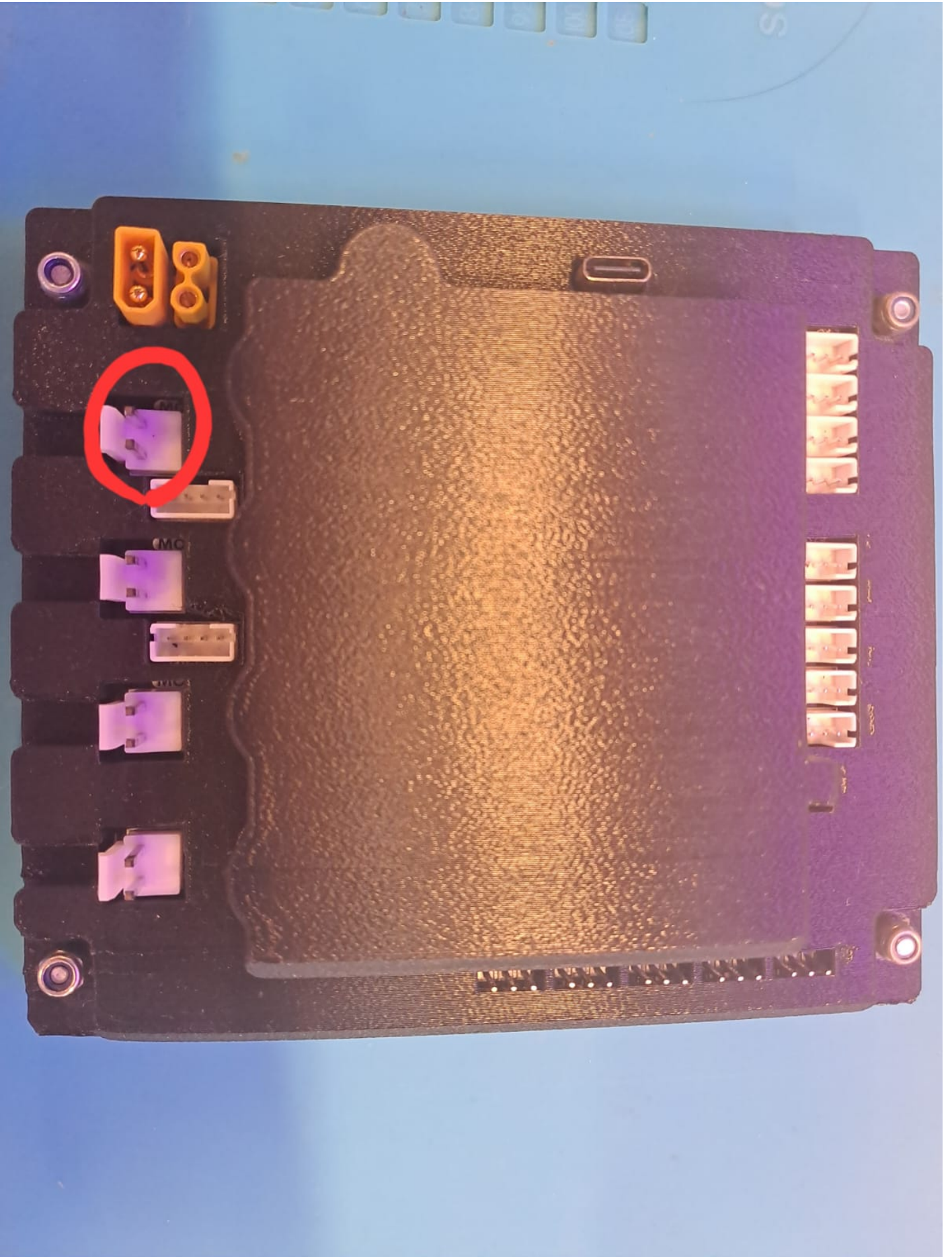
A conexão elétrica dos motores é feita utilizando cabos com terminais **JST-VH** de 2 pinos.



Na parte de trás dos motores disponíveis existirá um terminal oposto que se encaixa no contato do cabo descrito.



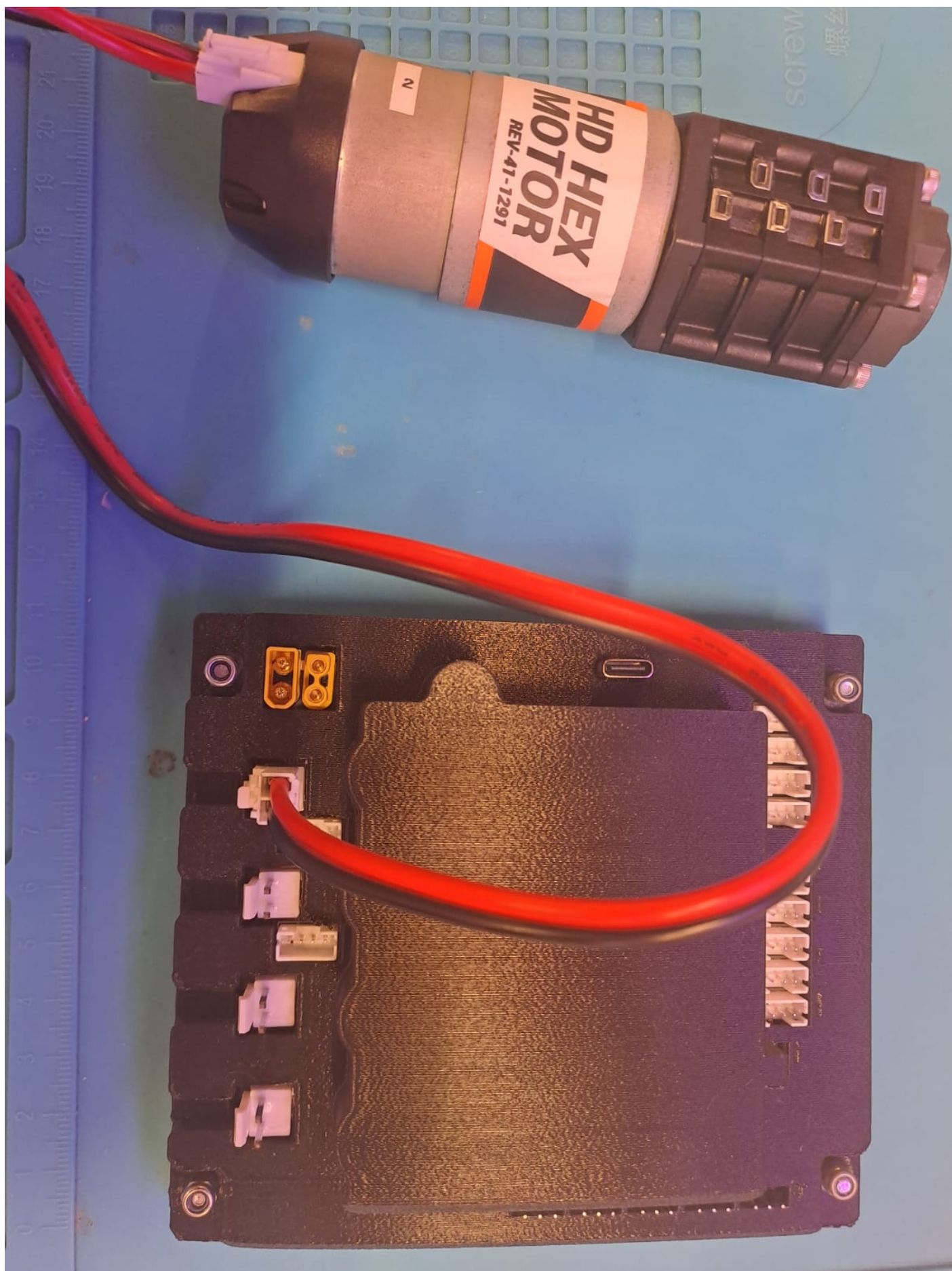
As portas correspondentes na placa EasySTEAM são as 4 que possuem esse mesmo contato, como mostrado abaixo:



O terminal circulado indica a porta 1, então, de cima para baixo temos ainda, porta 2, 3 e 4. Essa diferenciação é importante quando o código for efetuado.

A última parte é apenas realizar a conexão do cabo com os dois terminais, como indicado:





As seguintes recomendações são feitas para aumentar a vida útil desses contatos JST-VH.



## Alimentação

Os motores têm tensão elétrica de 12V, portanto, é necessário conectar a placa à uma bateria externa para movimentá-los. Isso é feito por meio dos conectores Xt30 presentes na placa - dois deles. Como segue:



# Programação

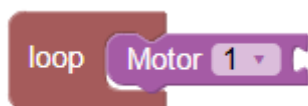
---

O motor em si possui 2 blocos que podem ser utilizados - sendo um deles o próprio motor.

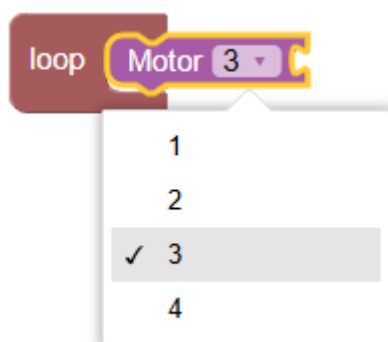
## Objeto de motor

O bloco de motor é utilizado para indicar com qual porta estaremos trabalhando, por isso é preciso escolher ela do lado do objeto, como indicado.

Motor configurado para porta 1:



Motor configurado para porta 3 (com outras opções disponíveis):



## Bloco de potência

O bloco que faz com que o motor se movimente é o bloco **setPower**, o qual recebe como parâmetro um valor de **-1.0** a **1.0**. Ele funciona em combinação com o objeto de motor. Como demonstra a imagem abaixo:



Entretanto, como explicado no parágrafo anterior, é preciso passar um valor numérico para esse bloco, isso pode ser feito de dois modos. Passando uma constante por meio da categoria “*Math*” ou



utilizando um valor dinâmico do controle - que será melhor explicado em outra seção.

## Valor Constante

Como supracitado, é possível passar um valor numérico constante para o bloco **setPower**, como mostra:



Como pode ser um valor de **-1.0** a **1.0** (valores maiores ou menores que os limites superiores e inferiores, respectivamente, serão cortados para 1.0, assim, -1.2 vira -1.0 e 1.2 vira 1.0), é possível colocar um valor fracionário e negativo, como:



Deve ser observado que o valor 1 tende a rotacionar o motor na velocidade máxima (100%), enquanto que um valor de 0.5 faz com que o motor gire na metade da velocidade total (50%). Caso seja colocado um valor negativo, a rotação se inverte.

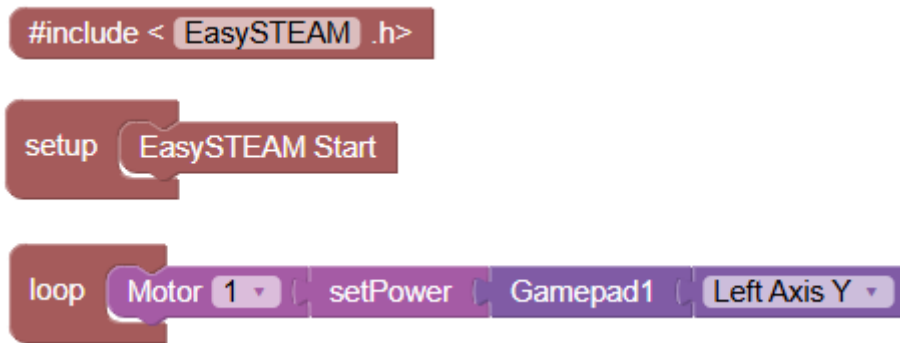
## Valor dinâmico

Para essa seção leia os trechos sobre **controle** e **conceitos iniciais da biblioteca**.

Como explicado na seção de conceitos iniciais da biblioteca, o bloco de loop opera repetidamente durante a execução do programa, é possível utilizar isso para atualizar periodicamente a saída do motor - visto que a cada momento é definido um valor novo para a porta. Para fazer isso, utilizamos o objeto de controle (**Gamepad** na interface), como explicado anteriormente, os valores dos eixos dos analógicos variam de **-1.0** a **1.0**, portanto, é totalmente viável colocar a o valor de saída de um dos eixos dos analógicos diretamente na entrada do bloco **setPower**. Como demonstrado abaixo:



Dessa forma, um código que rotacione um motor baseado no valor do eixo Y do analógico esquerdo do controle tem o seguinte formato:



## Execução

Com a montagem e a programação feita já é possível controlar o motor. Veja a seção de [Driver Station](#) para entender como conectar o computador ao EasySTEAM de forma wireless.