****

**Avy Chueke Pinto Rodolpho Filgueiras**

**Gabriel Vaz Nicolas Fonteyne**

**Pedro Casella Raphael Bomeisel**

**Luca Facciolo**

**Professores: Silvio Szafir, Fabio Ferraz, Francisco Lourenço**

**Projeto Mecatrônico - Sistemas Eletrônicos e Microprocessadores**

**-**

**Documentação Final do Projeto Mecatrônico - Hardware**

**São Paulo**

**02 de junho de 2017**

Sumário

[INTRODUÇÃO 3](#_Toc484265274)

[MOTORES 3](#_Toc484265275)

[SHIELD DE CONTROLE DO MOTOR 4](#_Toc484265276)

[PLACA DE CONTROLE 5](#_Toc484265277)

[DRIVERS 5](#_Toc484265278)

[DISPLAY DE VISUALIZAÇÃO 6](#_Toc484265279)

[I2C 7](#_Toc484265280)

[FIM DE CURSO 7](#_Toc484265281)

# INTRODUÇÃO

O objetivo deste relatório é detalhar de maneira sucinta os componentes usados na construção da máquina Pick & Place criada pelo nosso grupo durante o quinto semestre de engenharia do Insper. O objetivo da máquina era identificar a cor e diâmetro de um tarugo de polímero e de acordo com o que foi identificado, mover este tarugo para um lugar ou para outro.

Deve-se observar que neste documento não serão detalhados os sensores de distancia e de detecção de cor utilizados; isto estará documentado em outro arquivo.

# MOTORES

Usamos três motores de passo neste projeto. Escolhemos estes motores por alguns motivos específicos.

O primeiro deles é o controle da posição dos eixos de acordo com o número de passos fornecidos ao motor. O primeiro fator foi decisivo na nossa escolha pois conseguimos controlar a posição de cada eixo de maneira bastante acurada sem a necessidade de nenhum sensor, diminuindo os custos do projeto.

O segundo fator é a facilidade de controlar estes motores via software. Por terem um funcionamento baseado no eletromagnetismo, digitalmente seu acionamento se torna simples.

O terceiro e último fator é o custo benefício do motor. Por um preço relativamente baixo ele consegue nos entregar um bom torque e velocidade além de uma alta durabilidade além de ser mais simples de se trabalhar, se compararmos ele a outros tipos de motores como o DC.

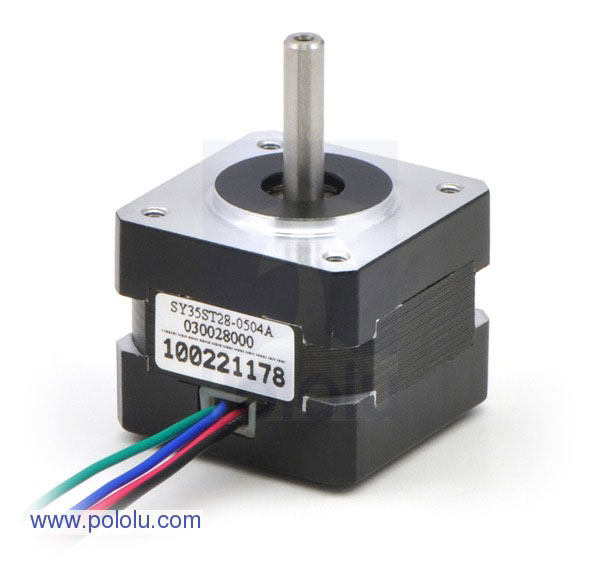


Figura 1 - Motor de passo

# SHIELD DE CONTROLE DO MOTOR

Usamos como shield de controle para os motores um CNC shield V3. Ele geralmente é usado para a construção de impressoras 3d caseiras ou maquinas CNC caseiras. Este shield comporta 4 motores, e necessita de um driver para cada motor. O atrativo deste shield foi a facilidade de comunicação com o motor além de nos possibilitar a definição de muitos requisitos técnicos via hardware. Por exemplo, via hardware, é possível definir o tipo de passo que o motor dará (1/2,1/4,1/8 ou 1/16 de passo), é possível duplicar um eixo e usar dois motores neste eixo e muitas outras funcionalidades que não foram exploradas.

Seu princípio de funcionamento é simples. Basicamente cada motor é conectado a este shield por meio de dois fios: Duas saídas digitais. Uma delas é responsável por definir a direção em que o motor rotacionará enquanto a outra é responsável por ditar a velocidade em que o motor girará. Por meio de pulsos dados a esta segunda saída digital o motor dá um passo. Exemplificando o que foi falado, cada vez que o sinal muda de valor lógico o motor se mexe um pouco, então na prática a definição da velocidade é dada pelo inverso do tempo de espera entre a troca dos sinais lógicos. Quanto menos tempo de espera programado via software, maior a velocidade do motor.

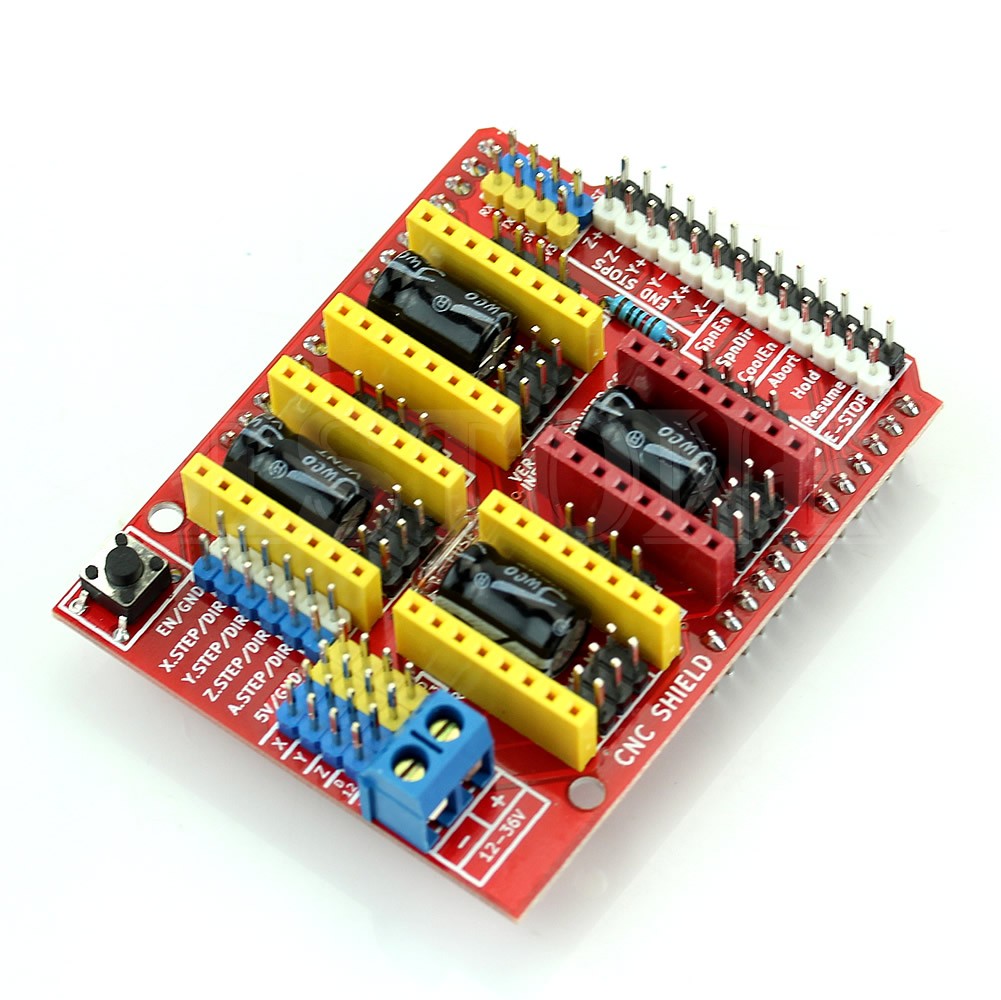


Figura 2 - CNC SHIELD

# PLACA DE CONTROLE

A placa usada como controlador foi o núcleo L053R8 da STM. Usamos esta placa pois ela tem um processador muito potente para um microcontrolador, de 32MHz. Além disso contamos também com uma memória de 64Kbytes para os nossos programas, 2Kbytes de dados EPPROM e 8KBytes.

Por fim a placa conta com muitos periféricos e muitas entradas digitais, necessárias para nosso projeto devido ao seu tamanho e quantidade de componentes.

Superestimamos a placa necessária para o projeto para não haver nenhuma limitação de velocidade, e tamanho de arquivos por parte do microcontrolador. Entretanto, seria possível desenvolver o mesmo projeto com uma placa inferior a esta.

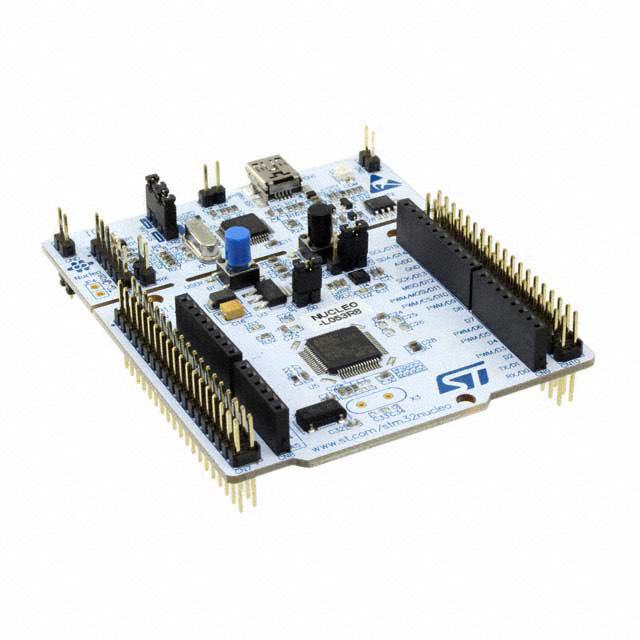


Figura 3 - Núcleo L053R8

# DRIVERS

Neste projeto usamos três drivers allegro a4988. Este driver é interessante pois tem proteção contra sobre correntes além de nos permitir usá-lo com meio, um quarto, um oitavo e um dezesseis avos de passo. Usamos neste projeto meio passo visando um torque maior. Na prática isto foi observado, o meio passo permitiu uma velocidade maior com um torque suficientemente grande para rotacionar todos os eixos de maneira silenciosa e constante.

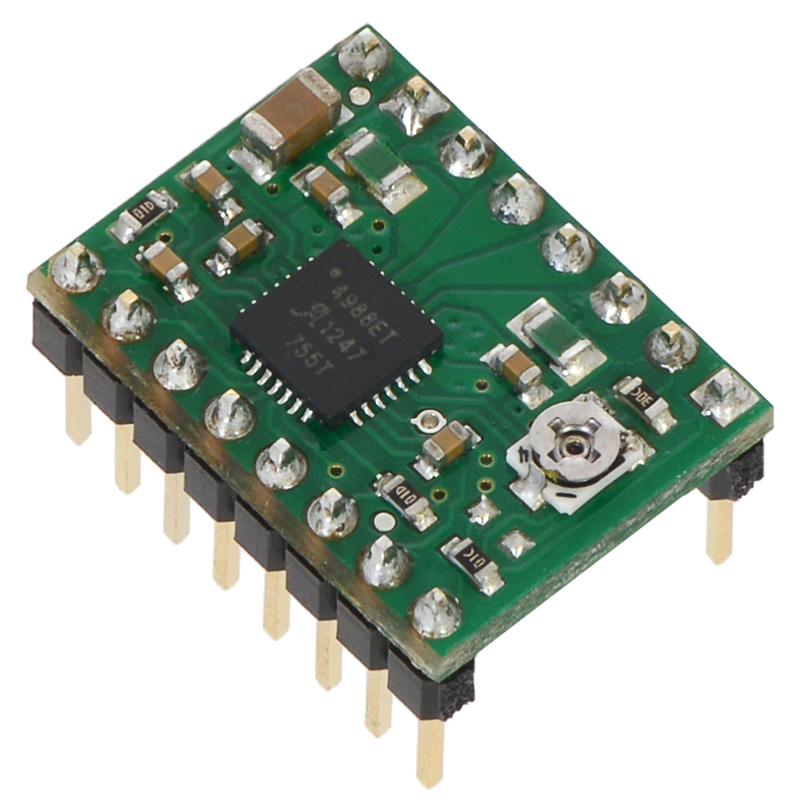


Figura 4 - Allegro a4988

Este driver nos possibilita modificar a corrente fornecida aos motores, o que é interessante pois não nos deixa limitados a correntes fixas, flexibilizando nosso projeto.

# DISPLAY DE VISUALIZAÇÃO

Para a construção do menu usamos um lcd20x4. Basicamente isto significa que temos 20 pixels por linha e temos 4 linhas.

Optamos por usar um display de visualização simples pois nosso objetivo primário era a funcionalidade e não a estética. Este display é simples de ser usado e não apresentou nenhum problema em sua implementação. Obviamente por haver poucos pixels não é possível desenvolver um menu muito elaborado. Entretanto, o que foi desenvolvido atende os requisitos e é intuitivo o suficiente para um usuário leigo usar.

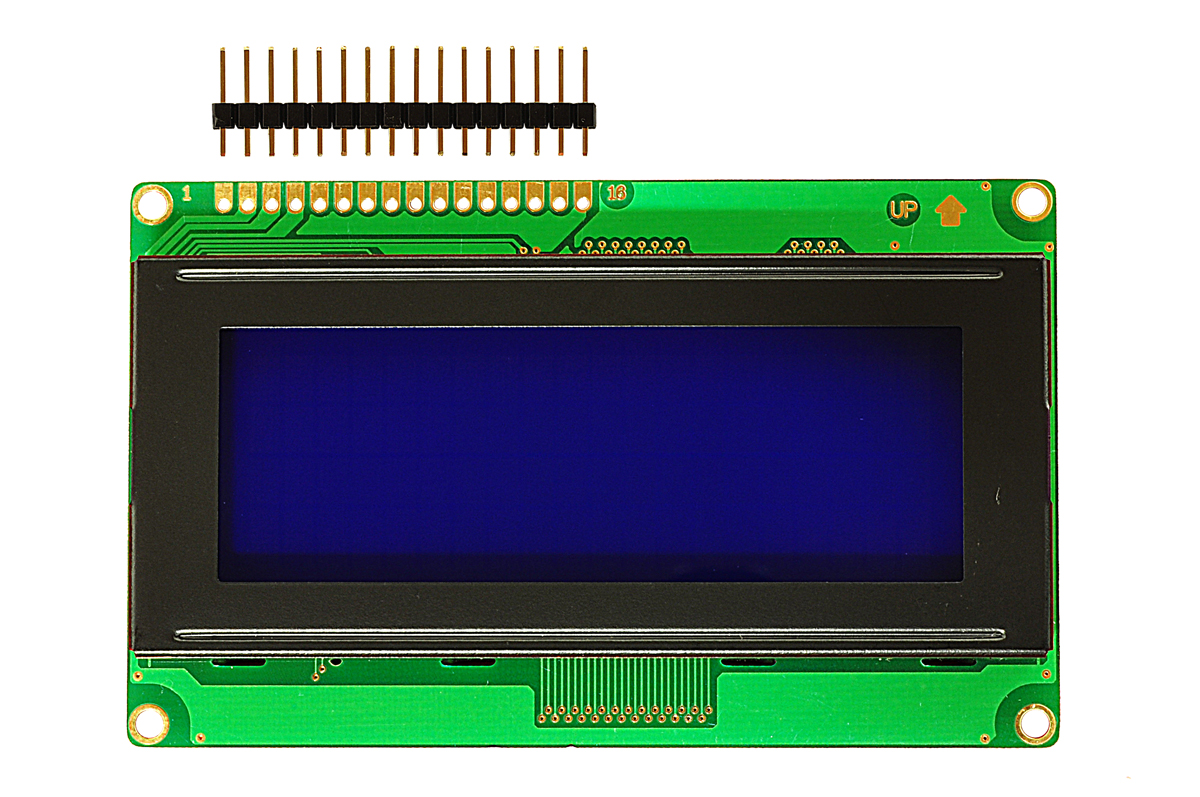


Figura 5 - DISPLAY 20X4

# I2C

Usamos um conector I2C para conectar nosso lcd na placa de controle do projeto. Usamos este módulo pois com ele conseguimos reduzir o número de fios de conexão entre o lcd e a placa de 20 para 2.

Isto é ótimo pois economizamos espaço em nossa placa além de organizar nosso projeto, fisicamente falando. Basicamente este é o benefício deste componente.

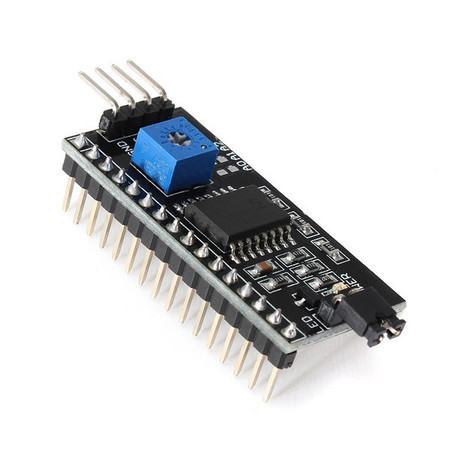


Figura 6 - MÓDULO I2C

# FIM DE CURSO

Para a detecção do fim do curso de cada eixo colocamos em cada eixo 2 sensores de fim de curso. Basicamente seu funcionamento é muito simples. Cada fim de curso funciona como um botão que dependendo de como você o conecta ao ser pressionado ele retorna um valor de 0 ou 1. Deste modo, é possível detectar via software se o fim de curso foi acionado ou não e caso ele tenha sido acionado alguma ação pode ser tomada.



Figura 7 - Fim de curso