



Disciplina de Microcontroladores

Projeto Final: ex_machina

Tiago Leite Brito

Bagé / 2018

projeto proposto

porque ex_machina?

projeto proposto

Deus **ex machina** é uma expressão latina com origens gregas ἀπὸ μηχανῆς θεός (apò mēkhanēs theós), que significa literalmente "Deus surgido da máquina", e é utilizada para indicar uma **solução inesperada, improvável e mirabolante para terminar uma obra** ficcional.

projeto proposto

Visão

Tem como visão projetar um dispositivo conceito que seja modular, adaptável e autômato, que sirva de modelo para desenvolver máquinas destinadas a criar produtos personalizados para os mais diversos fins.

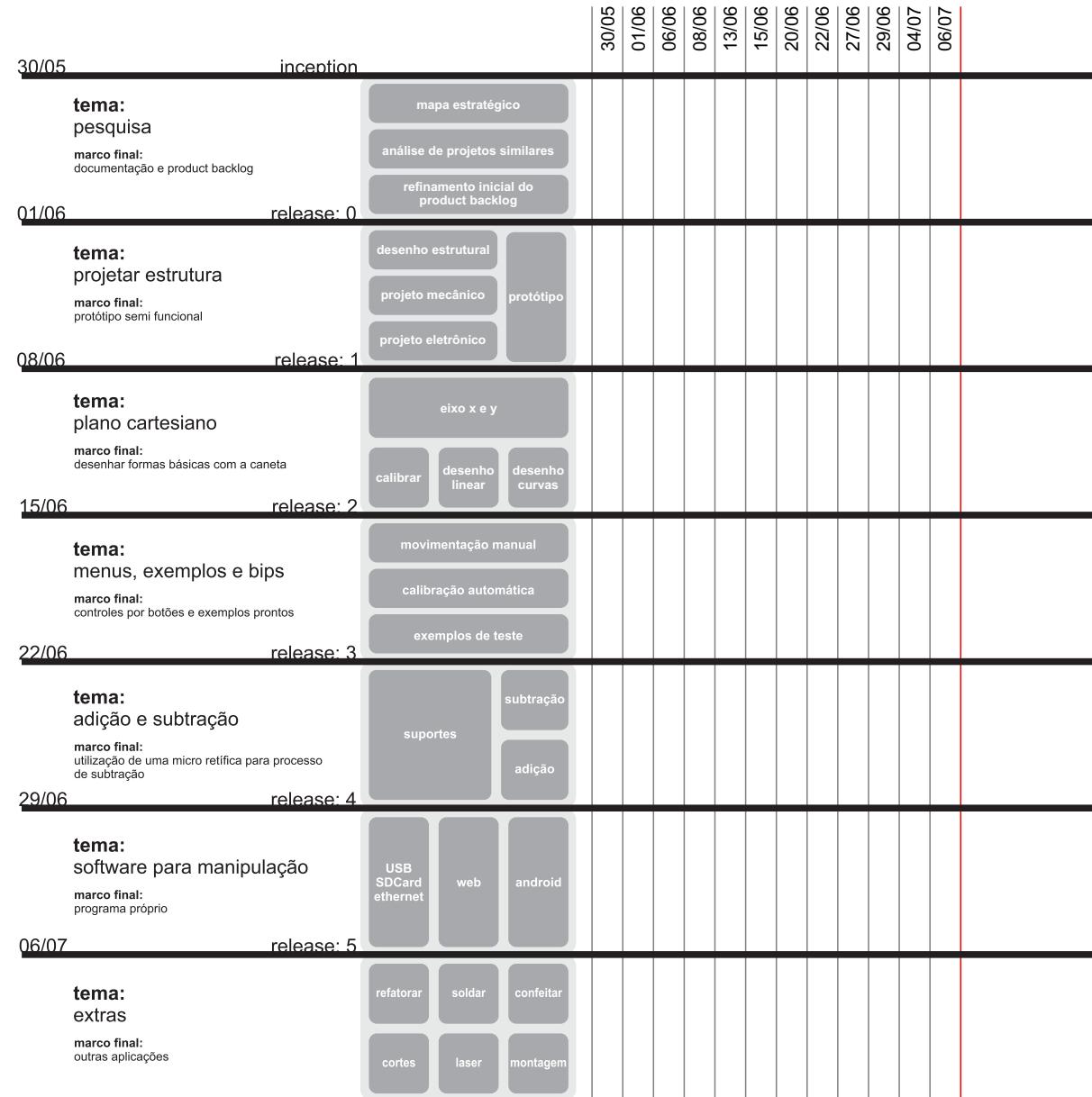
Missão

Tem como missão projetar de forma sistematizada e contínua, um modelo funcional de dispositivo para a criação e manipulação de produtos através de processos de adição e subtração em dois ou três eixos.

cronograma

definindo o product backlog

cronograma



cronograma

30/05	inception		22/06	release: 3	
		<p>tema: pesquisa</p> <p>marco final: documentação e product backlog</p>		<p>tema: adição e subtração</p> <p>marco final: utilização de uma micro retífica para processo de subtração</p>	
01/06	release: 0	<p>mapa estratégico</p> <p>análise de projetos similares</p> <p>refinamento inicial do product backlog</p>		<p>suportes</p> <p>subtração</p> <p>adição</p>	
			29/06	release: 4	
		<p>tema: projetar estrutura</p> <p>marco final: protótipo semi funcional</p>		<p>tema: software para manipulação</p> <p>marco final: programa próprio</p>	
08/06	release: 1	<p>desenho estrutural</p> <p>projeto mecânico</p> <p>projeto eletrônico</p> <p>protótipo</p>		<p>USB</p> <p>SDCard</p> <p>ethernet</p> <p>web</p> <p>android</p>	
			06/07	release: 5	
15/06	release: 2	<p>eixo x e y</p> <p>calibrar</p> <p>desenho linear</p> <p>desenho curvas</p>		<p>tema: extras</p> <p>marco final: outras aplicações</p>	
				<p>refatorar</p> <p>soldar</p> <p>confeitar</p> <p>cortes</p> <p>laser</p> <p>montagem</p>	
		<p>movimentação manual</p> <p>calibração automática</p> <p>exemplos de teste</p>			

inception

alinhamento da proposta

inception

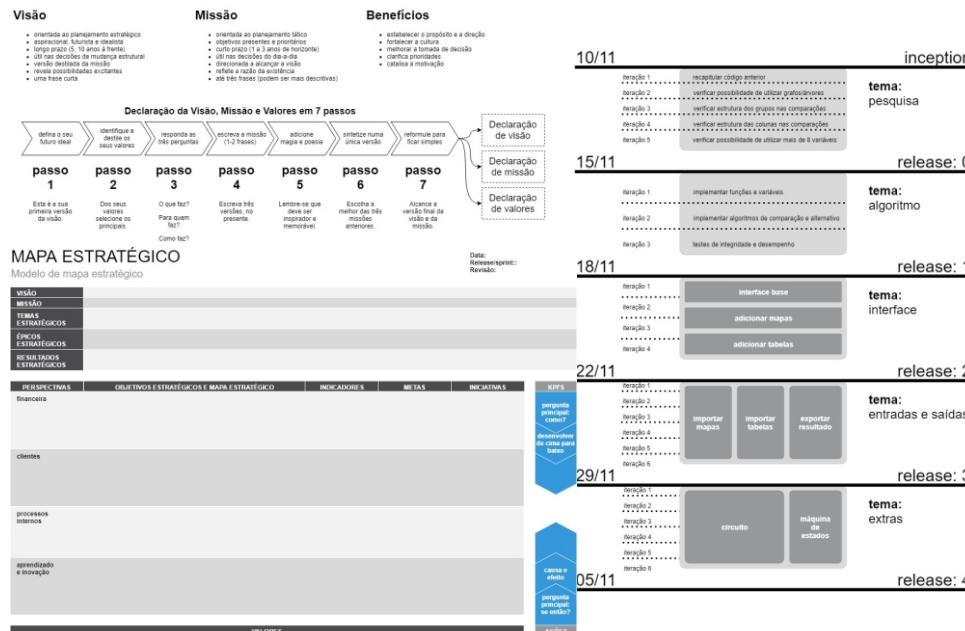
O CICLO DE IMPLEMENTAÇÃO DO MÉTODO EM NOVE ETAPAS

Integração de indicadores de desempenho ao Scrum



Missão, Visão e Valores

Integração de indicadores de desempenho ao Scrum

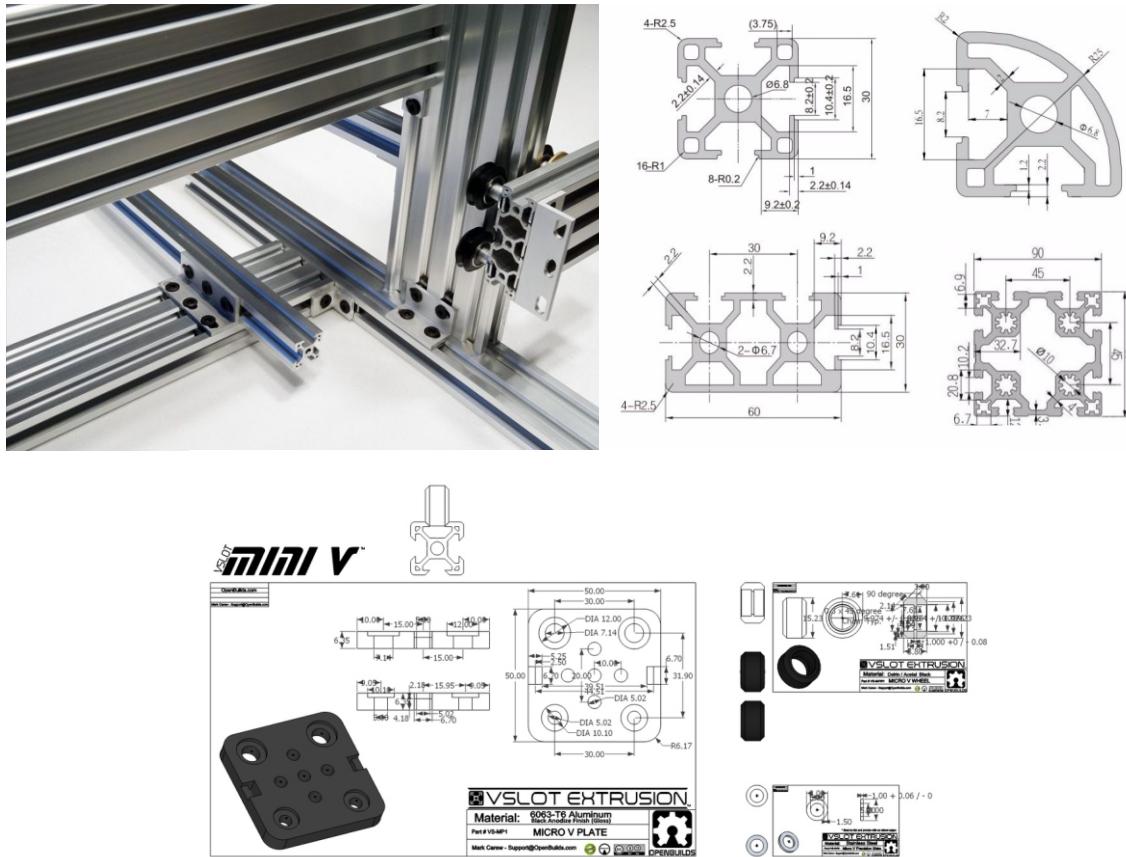


bscscrum

projetar estrutura

criando um protótipo semi funcional

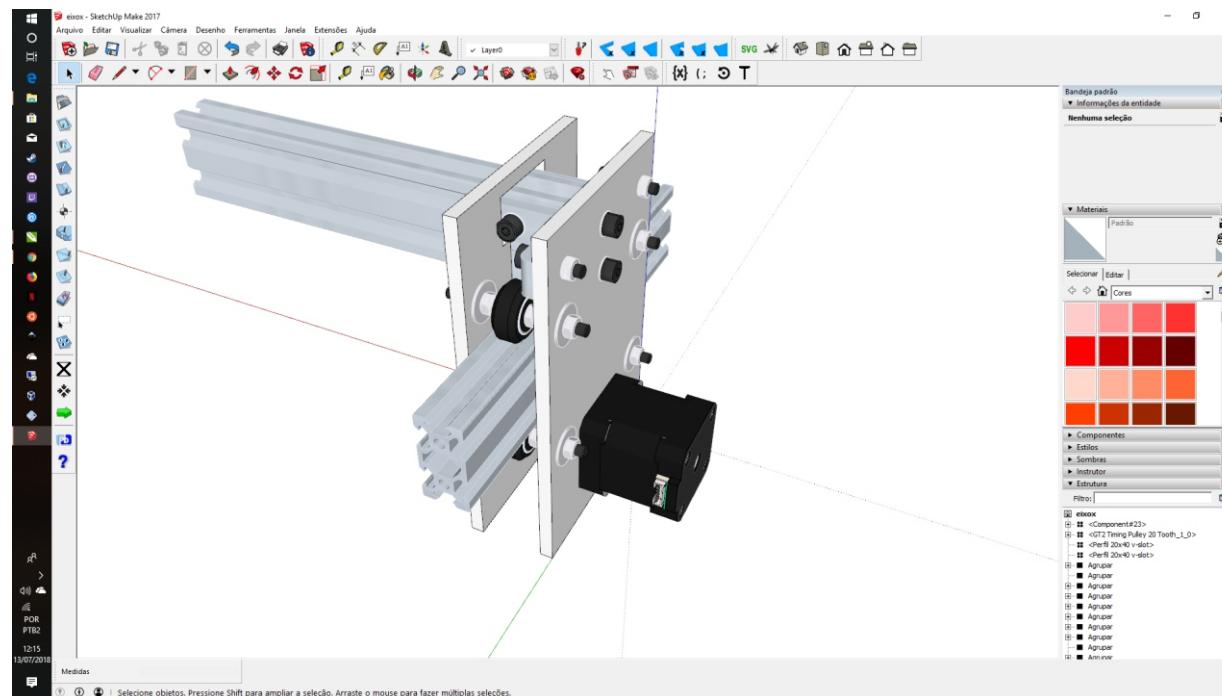
projetar estrutura



vslot system

desenho estrutural

projetar estrutura



vslot system

desenho estrutural

projetar estrutura

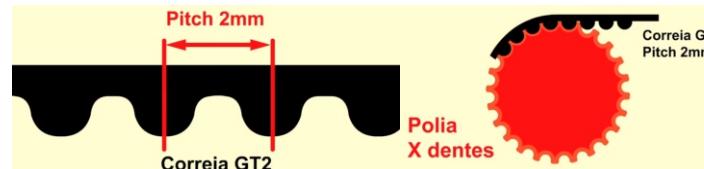
calculando step/mm para fusos

micropassos x 1 revolução do motor

passo do fuso

$$\frac{16 \times 200}{1} = 3200 \text{ steps/mm} \quad \text{resolução} = 0.0003125 \text{ mm}$$

calculando step/mm para correias

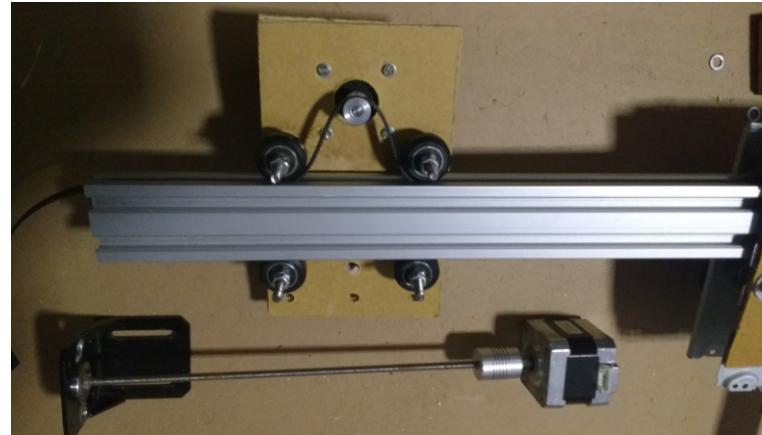


micropassos x 1 revolução do motor

pitch da correia x número dentes polia

$$\frac{16 \times 200}{2 \times 16} = 100 \text{ steps/mm} \quad \text{resolução} = 0.01 \text{ mm}$$

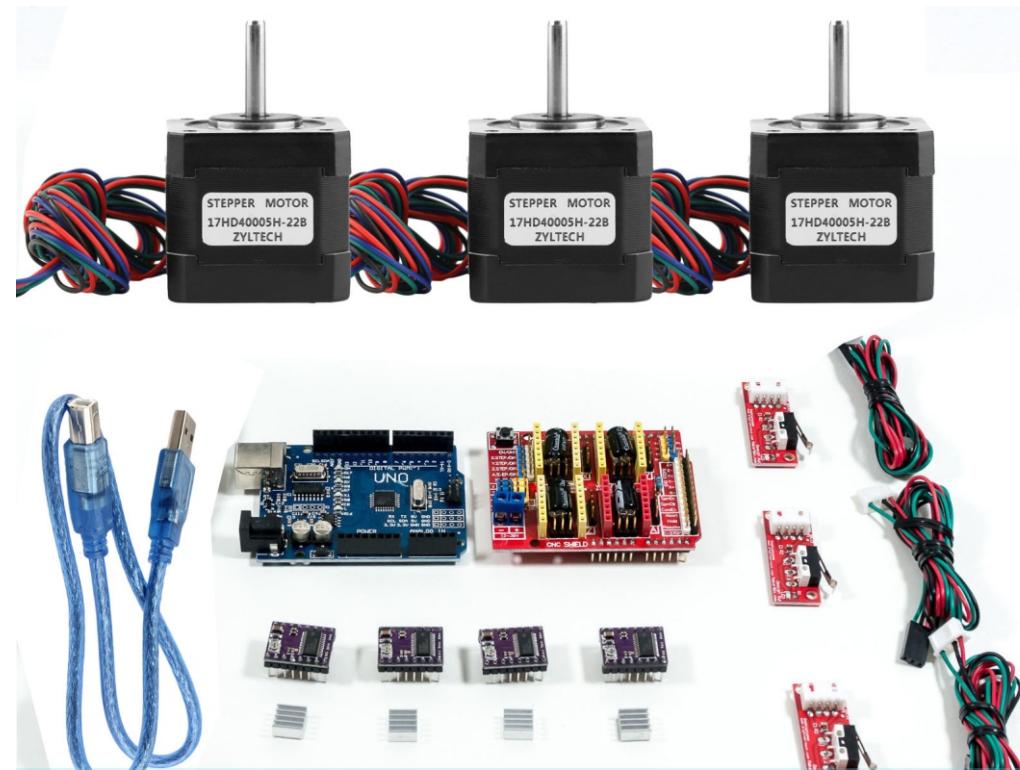
17HS4401										
2 Phase Hybrid Stepper Motor 17HS series-Size 42mm(1.8 degree)										
Wiring Diagram:										
Electrical Specifications:										
Series Model	Step Angle	Motor Length	Rated Current (A)	Phase Resistance (ohm)	Phase Inductance (mH)	Holding Torque (kg cm Max)	Diameter (mm)	Rotor Inductance (mH)	Lead Wire (m)	Motor Length (mm)
17HS4408	1.8	28	0.6	8	10	12	1.6	34	4	150
17HS4410	1.8	34	1.3	2.6	22	28	1.6	34	4	200
17HS4410	1.8	34	1.7	1.0	1.8	28	1.6	34	4	200
17HS4410	1.8	34	0.4	30	35	28	1.6	34	4	200
17HS4410	1.8	34	0.4	30	18	21	1.6	34	6	200
17HS4410	1.8	34	0.4	30	20	24	1.6	34	6	200
17HS4410	1.8	40	1.7	1.0	2.8	40	2.0	34	6	280
17HS4410	1.8	40	1.3	2.6	5.0	40	2.2	34	4	280
17HS4410	1.8	40	1.2	3.2	2.8	28	2.2	34	6	280
17HS4410	1.8	40	0.4	30	28	28	2.2	34	6	280
17HS4410	1.8	40	1.7	1.8	1.8	32	2.2	34	6	280
17HS4410	1.8	48	1.3	3.2	5.5	52	2.6	68	4	300
17HS4410	1.8	48	2.3	1.2	1.6	46	2.6	68	4	300
17HS4410	1.8	48	0.4	30	38	34	2.6	68	6	300
Note: Dimensions are subject to change according to customer's requirements.										Motor Length:
Dimensions: Unit/mm										
Model										17HS2000: 26 mm 17HS3000: 34 mm 16HS4000: 40 mm 16HS5000: 48 mm



correias, fusos e motores

desenho mecânico

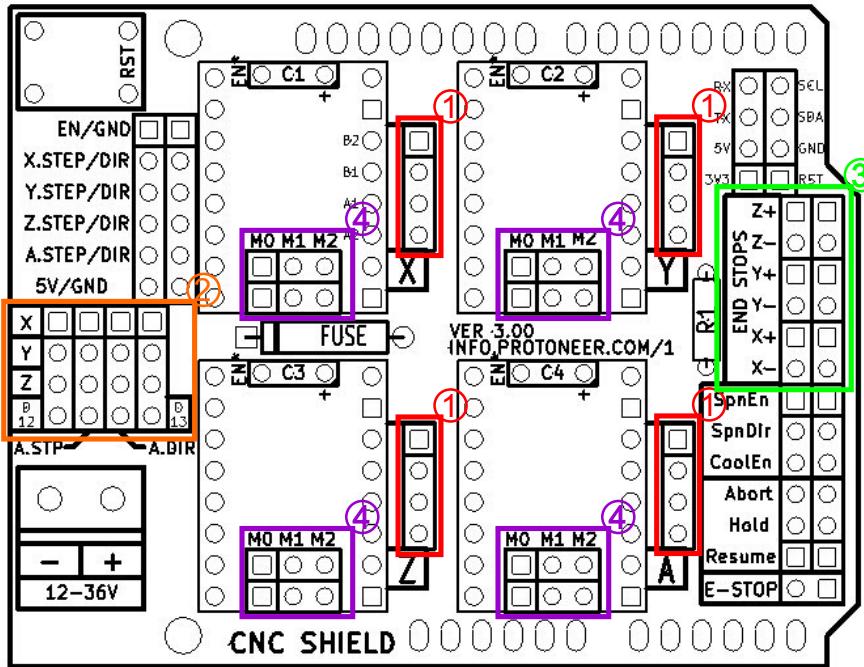
projetar estrutura



arduino, cnc shield e complementos

desenho eletrônico

projetar estrutura



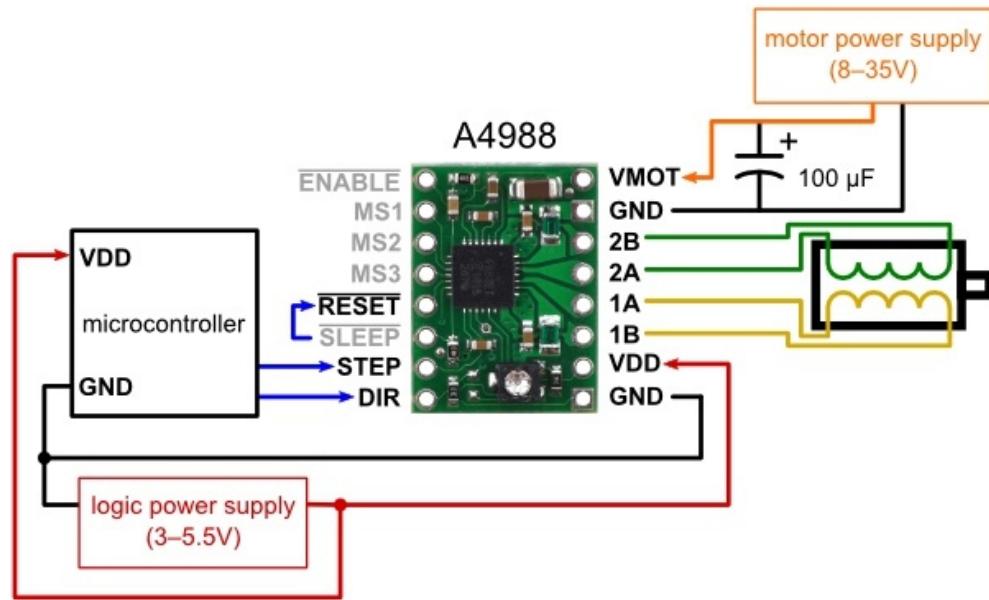
- 1 - motores
- 2 - fim de curso
- 3 - eixo a
- 4 - configuração de micro stepping

M0	M1	M2	resolução
low	low	low	fullstep
high	low	low	1/2 step
low	high	low	1/4 step
high	high	low	1/8 step
high	high	high	1/16 step

arduino, cnc shield e complementos

desenho eletrônico

projetar estrutura



$$V_{ref} = 8 * I_{max} * r_{cs}$$

tensão de referência

corrente máxima de corrente por bobina do motor

resistor do sensor do driver

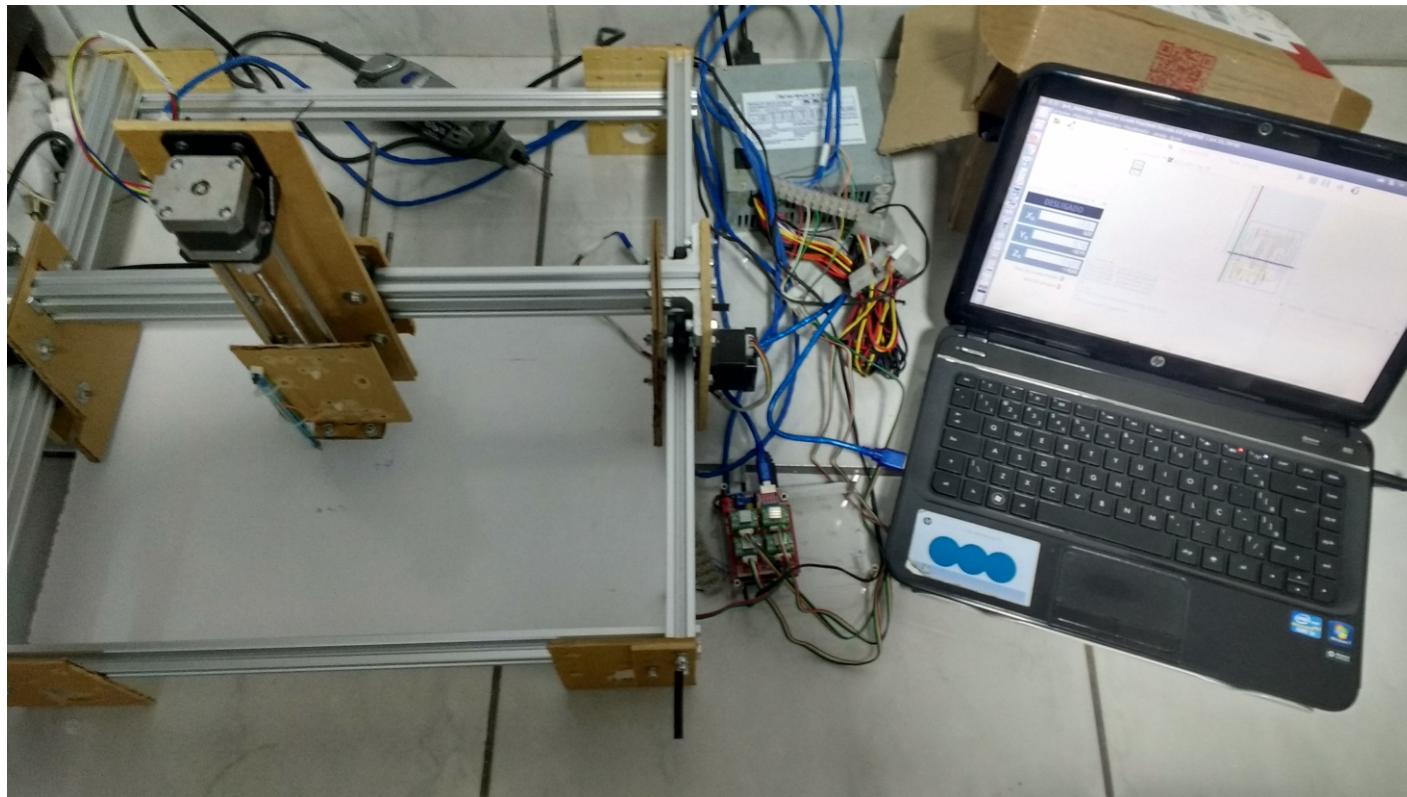
$$V_{ref} = 8 * 1,7A * 0,1\text{ohm}$$

$$V_{ref} = 1,36 \text{ V}$$

arduino, cnc shield e complementos

desenho eletrônico

projetar estrutura



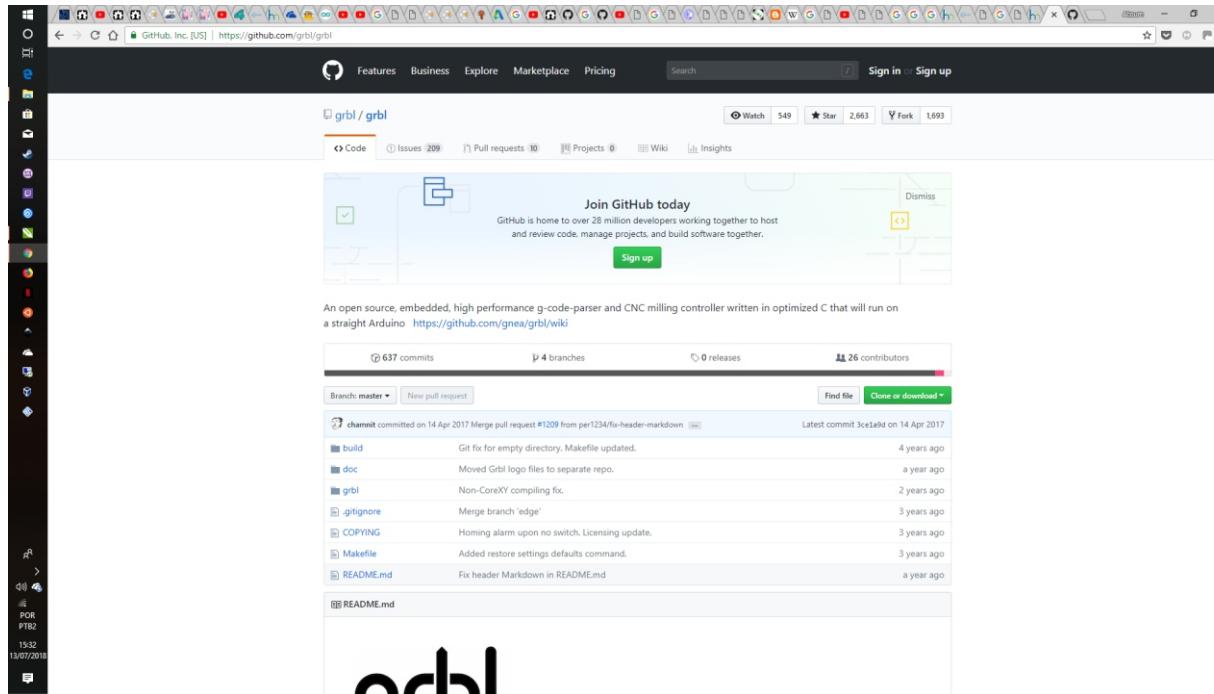
estrutura semi-funcional

protótipo

plano cartesiano

ajustando o hardware com o software

plano cartesiano



grbl

firmware open-source

plano cartesiano

```
cpu_map_atmega328p.h - Visual Studio Code
File Edit Selection View Go Debug Tasks Help

C cpu_map_atmega328p.h ...
25 #ifndef GRBL_PLATFORM
26 #error "cpu_map already defined: GRBL_PLATFORM=" GRBL_PLATFORM
27 #endif
28
29
30 #define GRBL_PLATFORM "Atmega328p"
31
32 // Define serial port pins and interrupt vectors.
33 #define SERIAL_RX USART_RX_vect
34 #define SERIAL_UDRE USART_UDRE_vect
35
36 // Define step pulse output pins. NOTE: All step bit pins must be on the same port.
37 #define STEP_DDR DDRD
38 #define STEP_PORT PORTD
39 #define X_STEP_BIT 2 // Uno Digital Pin 2
40 #define Y_STEP_BIT 3 // Uno Digital Pin 3
41 #define Z_STEP_BIT 4 // Uno Digital Pin 4
42 #define STEP_MASK ((1<<X_STEP_BIT)|(1<<Y_STEP_BIT)|(1<<Z_STEP_BIT)) // All step bits
43
44 // Define step direction output pins. NOTE: All direction pins must be on the same port.
45 #define DIRECTION_DDR DDRD
46 #define DIRECTION_PORT PORTD
47 #define X_DIRECTION_BIT 5 // Uno Digital Pin 5
48 #define Y_DIRECTION_BIT 6 // Uno Digital Pin 6
49 #define Z_DIRECTION_BIT 7 // Uno Digital Pin 7
50 #define DIRECTION_MASK ((1<<X_DIRECTION_BIT)|(1<<Y_DIRECTION_BIT)|(1<<Z_DIRECTION_BIT)) // All direction bits
51
52 // Define stepper driver enable/disable output pin.
53 #define STEPPERS_DISABLE_DDR DORB
54 #define STEPPERS_DISABLE_PORT PORTB
55 #define STEPPERS_DISABLE_BIT 0 // Uno Digital Pin 8
56 #define STEPPERS_DISABLE_MASK (1<<STEPPERS_DISABLE_BIT)
57
58 // Define homing/hard limit switch input pins and limit interrupt vectors.
59 // NOTE: All limit bit pins must be on the same port, but not on a port with other input pins
60 #define LIMIT_DDR DORB
61 #define LIMIT_PIN PINB
62 #define LIMIT_PORT PORTB
63 #define X_LIMIT_BIT 1 // Uno Digital Pin 9
64 #define Y_LIMIT_BIT 2 // Uno Digital Pin 10
65 #ifdef VARIABLE_SPINDLE // Z Limit pin and spindle enabled swapped to access hardware PWM on Pin 11
66 #define Z_LIMIT_BIT 4 // Uno Digital Pin 12
67 #else
68 #define Z_LIMIT_BIT 3 // Uno Digital Pin 11
69 #endif
70 #define LIMIT_MASK ((1<<X_LIMIT_BIT)|(1<<Y_LIMIT_BIT)|(1<<Z_LIMIT_BIT)) // All limit bits
71 #define LIMIT_INT PCIE0 // Pin change interrupt enable pin
72 #define LIMIT_INT_vect PCINT0_vect
73 #define LIMIT_PCMASK PCMSK0 // Pin change interrupt register
74
75 // Define spindle enable and spindle direction output pins.
76 #define SPINDLE_ENABLE_DDR DORB
77 #define SPINDLE_ENABLE_PORT PORTB
78 // Z Limit pin and spindle PWMenable pin swapped to access hardware PWM on Pin 11.
79 #ifdef VARIABLE_SPINDLE
80 #ifndef USE_SPINDLE_DIR_AS_ENABLE_PIN
81 // If enabled, spindle direction pin now used as spindle enable, while PWM remains on D11
82 #define SPINDLE_ENABLE_BIT 5 // Uno Digital Pin 13 (NOTE: D13 can't be pulled-high in
83 #else
84 #define SPINDLE_ENABLE_BIT 3 // Uno Digital Pin 11
85 #endif
86 #else
87 #define SPINDLE_ENABLE_BIT 4 // Uno Digital Pin 12
88 #endif
89 #ifndef USE_SPINDLE_DIR_AS_ENABLE_PIN
90 #define SPINDLE_DIRECTION_DDR DORB
91 #define SPINDLE_DIRECTION_PORT PORTB
92 #define SPINDLE_DIRECTION_BIT 5 // Uno Digital Pin 13 (NOTE: D13 can't be pulled-high in
93 #endif
94
95 // Define flood and mist coolant enable output pins.
96 // NOTE: Uno analog pins 4 and 5 are reserved for an 12c interface, and may be installed at
97 // a later date if flash and memory space allows.
98 #define COOLANT_FLOOD_DDR DORC
99 #define COOLANT_FLOOD_PORT PORTC
100 #define COOLANT_FLOOD_BIT 3 // Uno Analog Pin 3
101 #ifdef ENABLE_M7 // Mist coolant disabled by default. See config.h to enable/disable.
102 #define COOLANT_MIST_DDR DORC
103 #define COOLANT_MIST_PORT PORTC
104 #define COOLANT_MIST_BIT 4 // Uno Analog Pin 4
105 #endif
106
107 // Define user-control controls (cycle start, reset, feed hold) input pins.
108 // NOTE: All CONTROLS pins must be on the same port and not on a port with other input pins (1
109 #define CONTROL_DDR DDRC
110 #define CONTROL_PIN PINC
111 #define CONTROL_PORT PORTC
112 #define RESET_BIT 0 // Uno Analog Pin 0
113 #define FEED_HOLD_BIT 1 // Uno Analog Pin 1
114 #define CYCLE_START_BIT 2 // Uno Analog Pin 2
115 #define SAFETY_DOOR_BIT 1 // Uno Analog Pin 1 NOTE: Safety door is shared with feed hold. E
116 #define CONTROL_INT PCIE1 // Pin change interrupt enable pin
117 #define CONTROL_INT_vect PCINT1_vect
118 #define CONTROL_PCMASK PCMSK1 // Pin change interrupt register
119 #define CONTROL_MASK ((1<<RESET_BIT)|(1<<FEED_HOLD_BIT)|(1<<CYCLE_START_BIT)|(1<<SAFETY_DOOR_B
120 #define CONTROL_INVERT_MASK CONTROL_MASK // May be re-defined to only invert certain control p
121
122 // Define probe switch input pin.
123 #define PROBE_DDR DDRC
124 #define PROBE_PIN PINC
125 #define PROBE_PORT PORTC
In 1, Col 1 Spaces:2 UTF-8 LF Arduino Win32
```

grbl

firmware open-source

plano cartesiano

Configuração	Valor	Descrição
\$0	10	Step pulse time
\$1	25	Step idle delay
\$2	0	Step pulse invert
\$3	0	Step direction invert
\$4	0	Invert step enable pin
\$5	0	Invert limit pins
\$6	0	Invert probe pin
\$10	3	Status report options
\$11	0.010	Junction deviation
\$12	0.002	Arc tolerance
\$13	0	Report in inches
\$20	0	Soft limits enable
\$21	1	Hard limits enable
\$22	0	Homing cycle enable
\$23	0	Homing direction invert
\$24	25.000	Homing locate feed rate
\$25	500.000	Homing search seek rate
\$26	250	Homing switch debounce delay
\$27	1.000	Homing switch pull-off distance
\$100	100.000	X-axis travel resolution
\$101	100.000	Y-axis travel resolution
\$102	1000.000	Z-axis travel resolution
\$110	500.000	X-axis maximum rate
\$111	500.000	Y-axis maximum rate
\$112	500.000	Z-axis maximum rate
\$120	10.000	X-axis acceleration
\$121	10.000	Y-axis acceleration
\$122	10.000	Z-axis acceleration
\$130	200.000	X-axis maximum travel
\$131	200.000	Y-axis maximum travel
\$132	200.000	Z-axis maximum travel

grbl

firmware open-source

plano cartesiano

N## G## X## Y## Z## F## S## T## M##

N: Line number

G: Motion

X: Horizontal position

Y: Vertical position

Z: Depth

F: Feed rate

S: Spindle speed

T: Tool selection

M: Miscellaneous functions

I and J: Incremental center of an arc

R: Radius of an arc

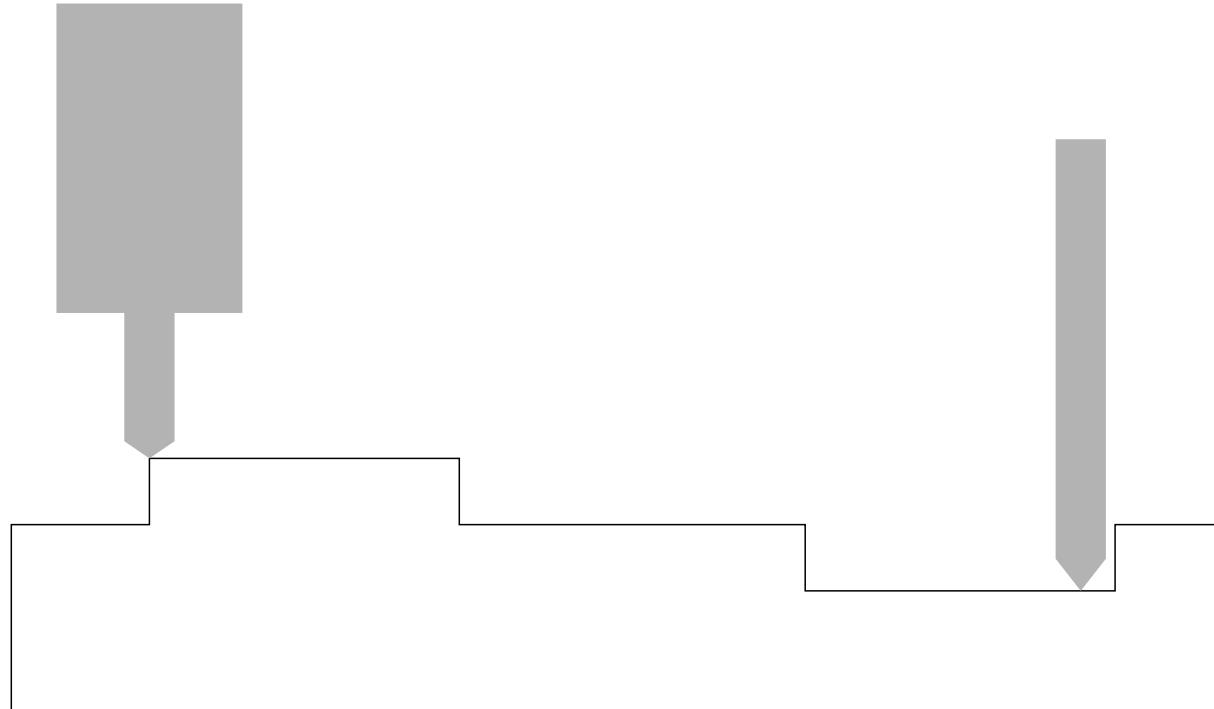
g code

linguagem para cnc

adição e subtração

como são produzidos os produtos finais

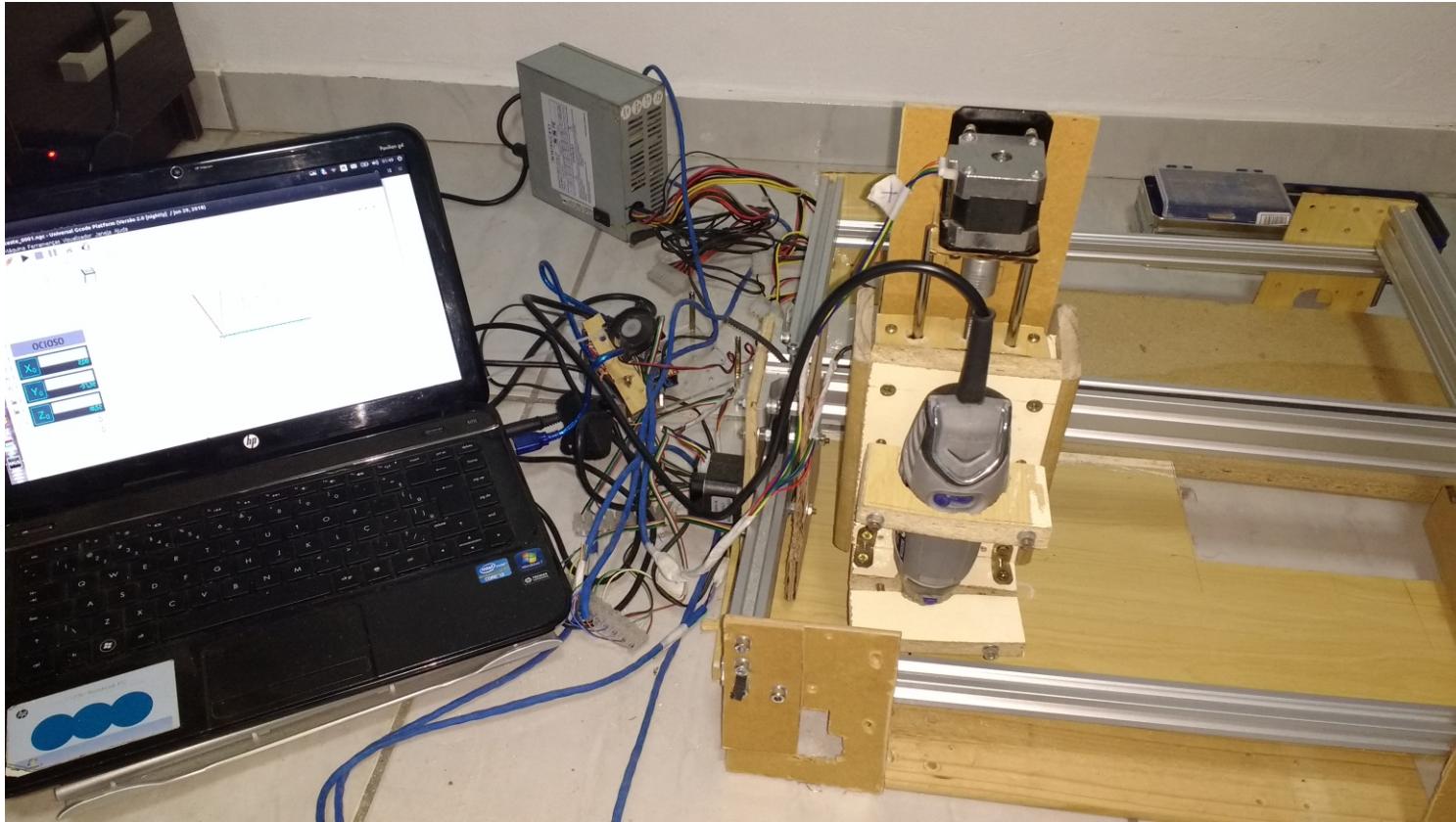
adição e subtração



adição e subtração

processos propostos

adição e subtração



microretífica

subtração

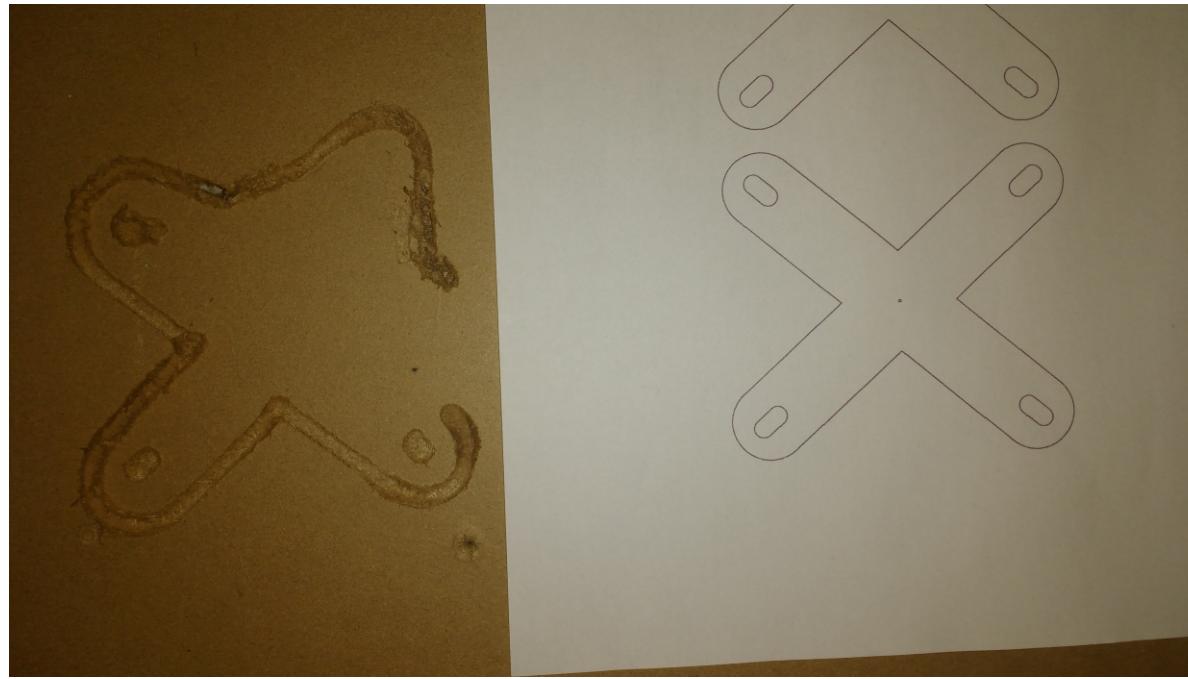
adição e subtração



resultados da sprint

subtração

adição e subtração



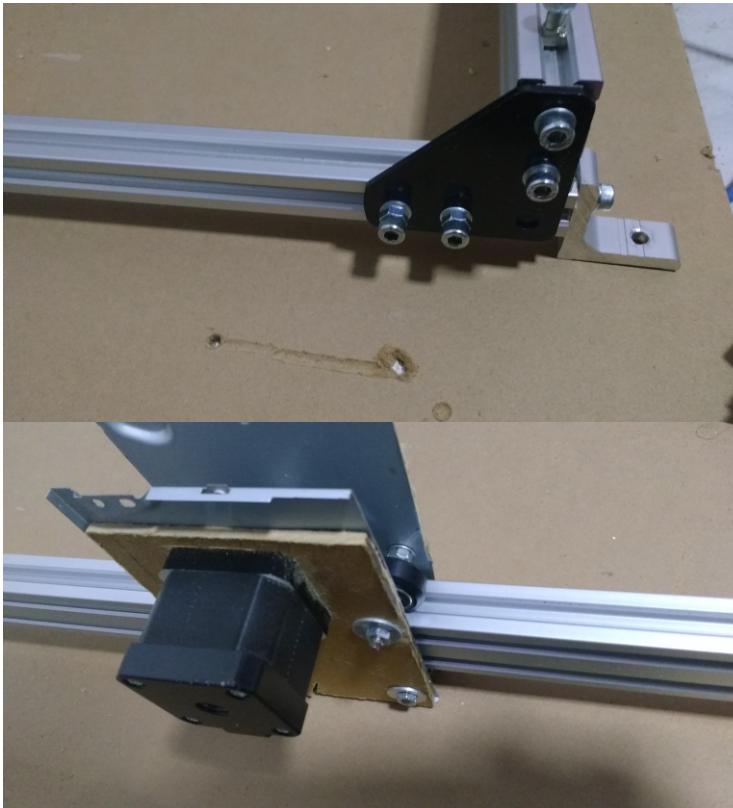
resultados da sprint

subtração

revisão do projeto

revisitando as adversidades

revisão do projeto



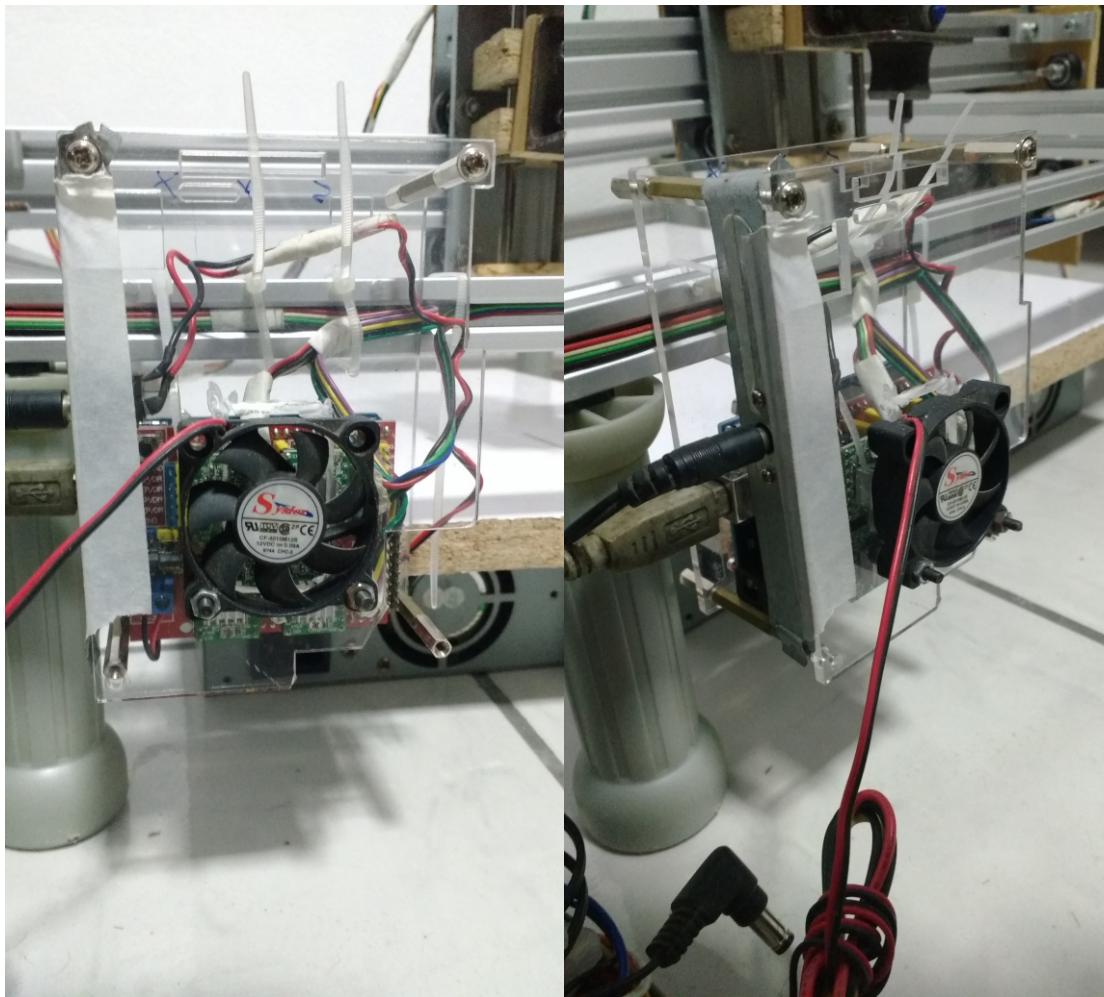
reforço do estrutura

revisão do projeto



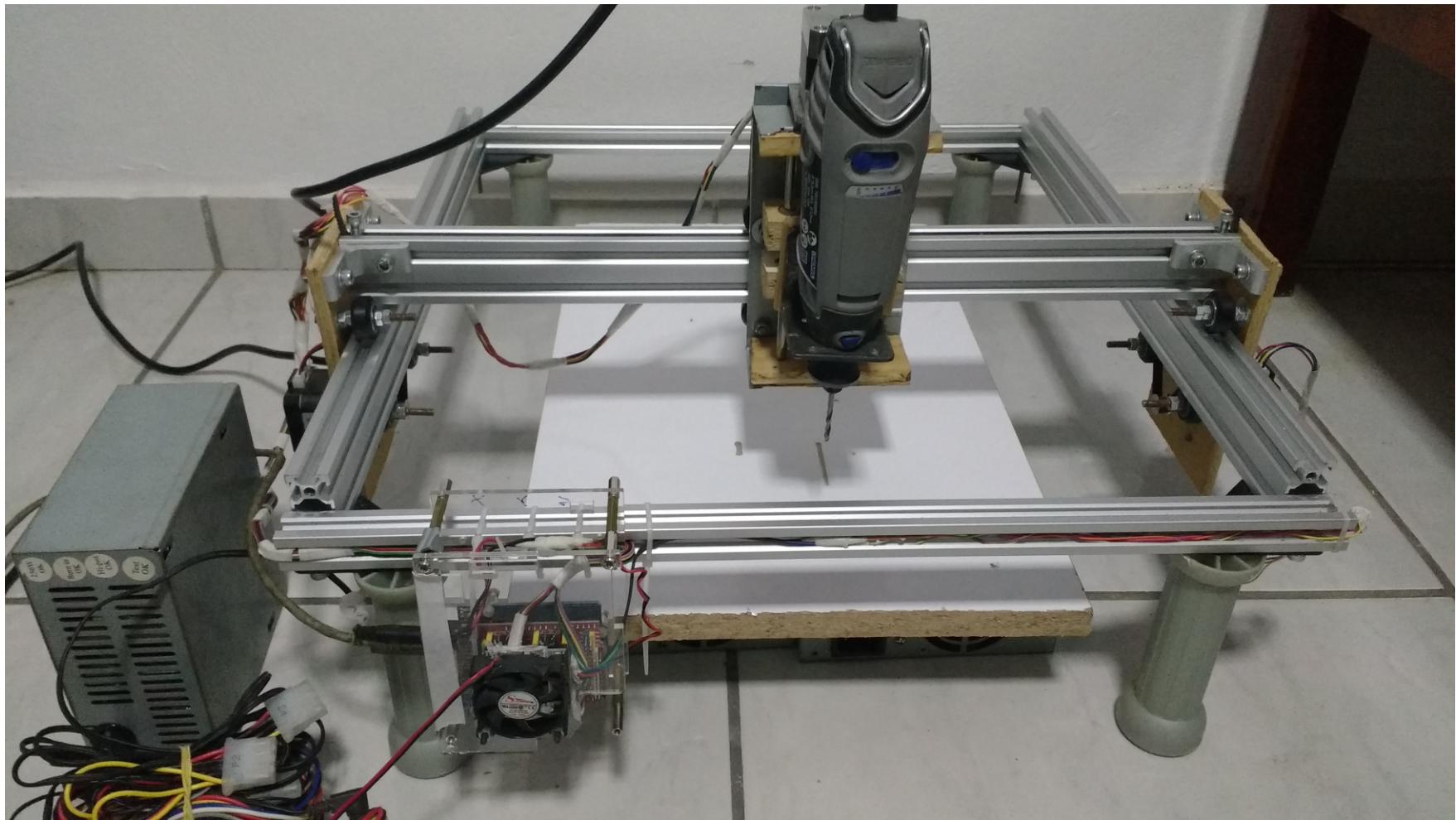
reforço do estrutura

revisão do projeto



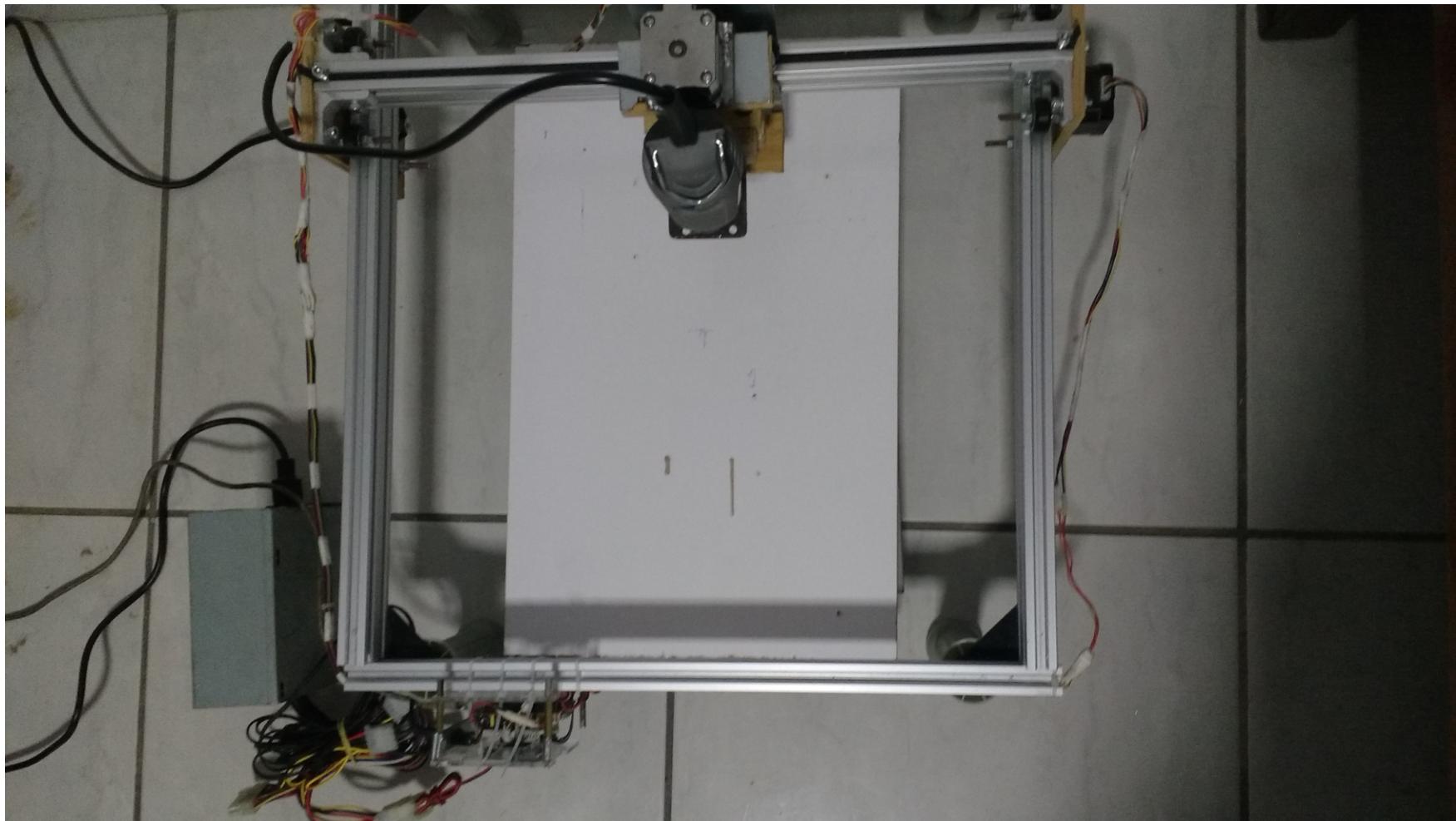
reforço do estrutura

revisão do projeto



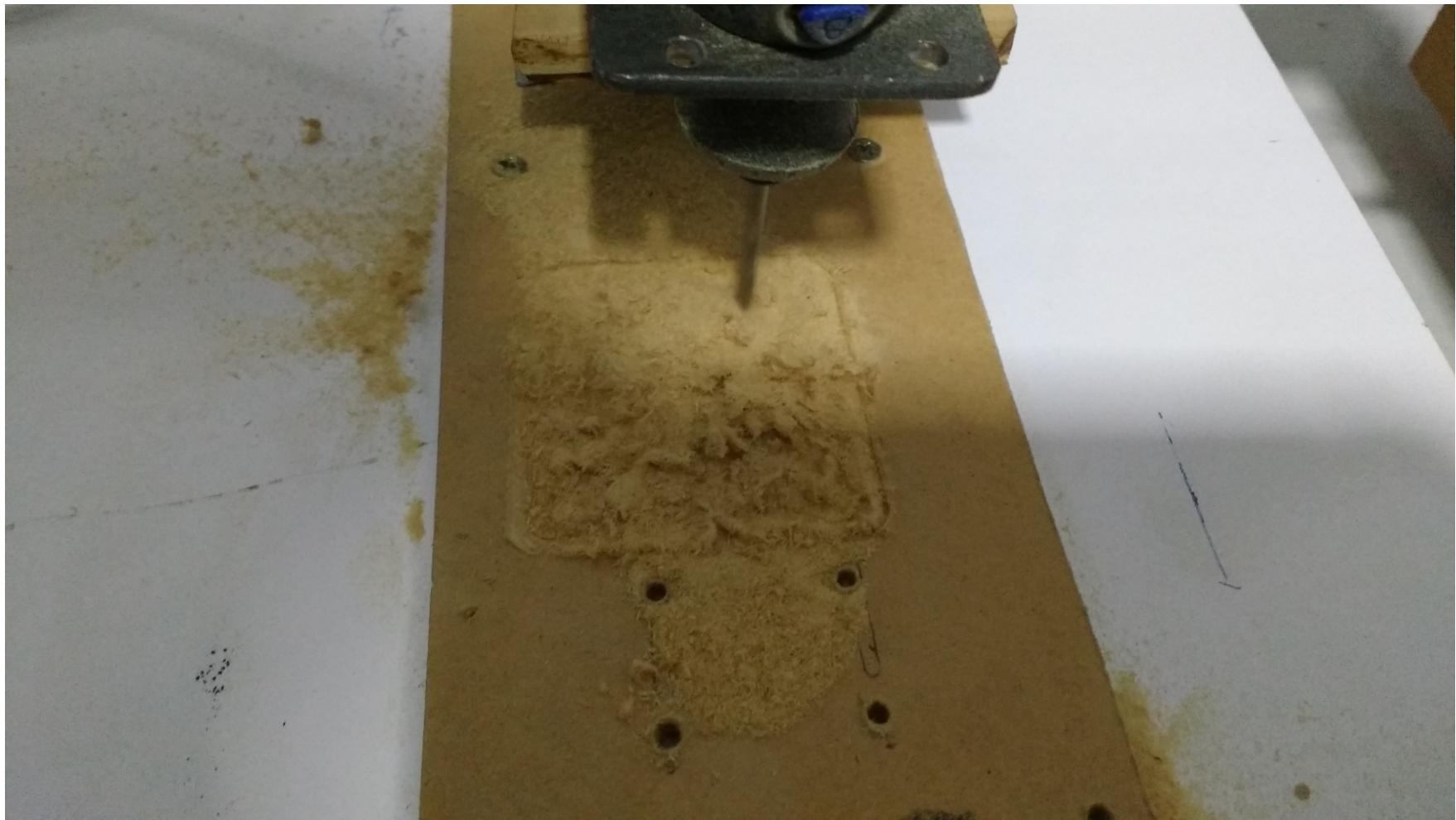
estrutura atual

revisão do projeto



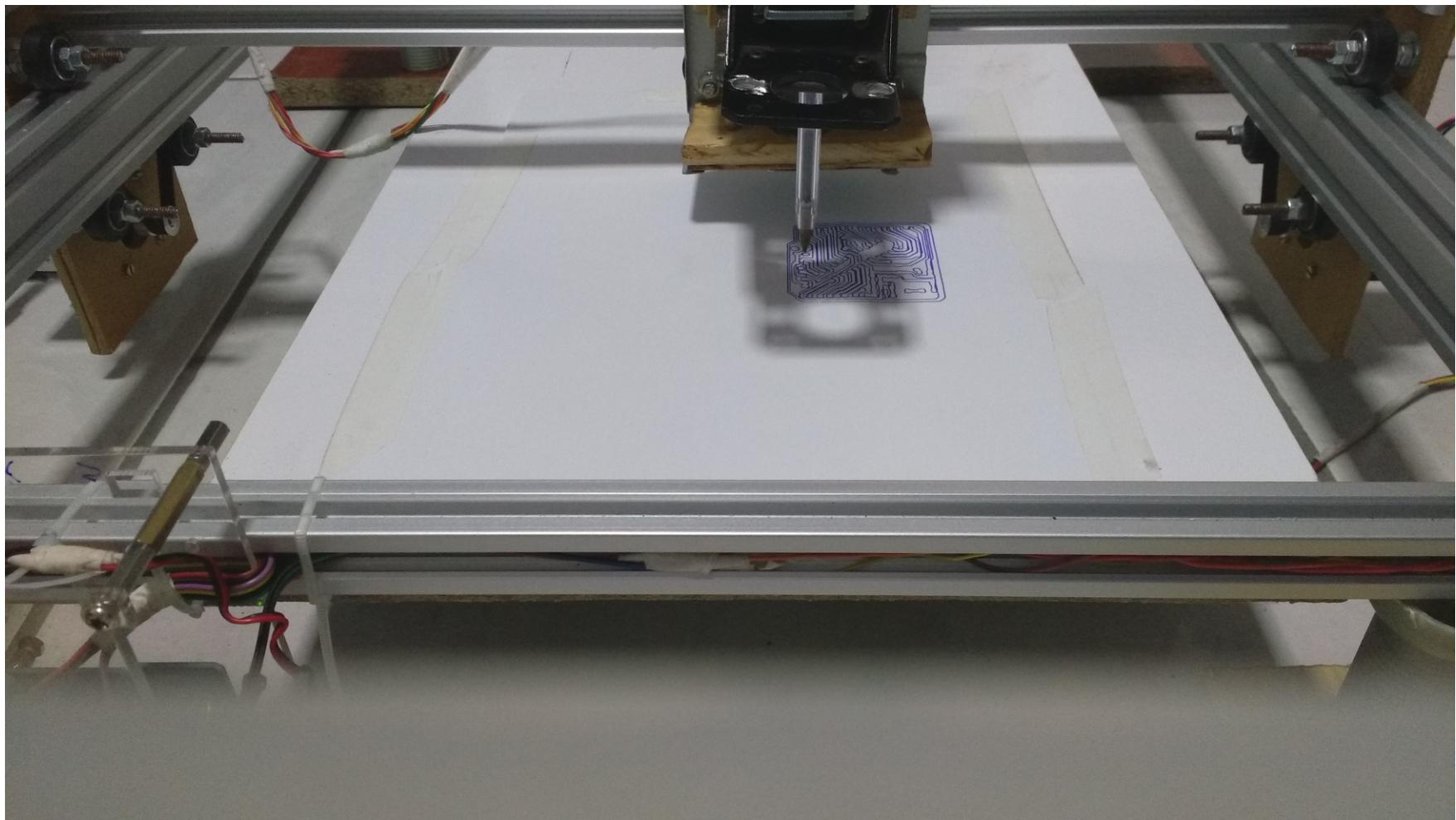
estrutura atual

revisão do projeto



ponta da broca

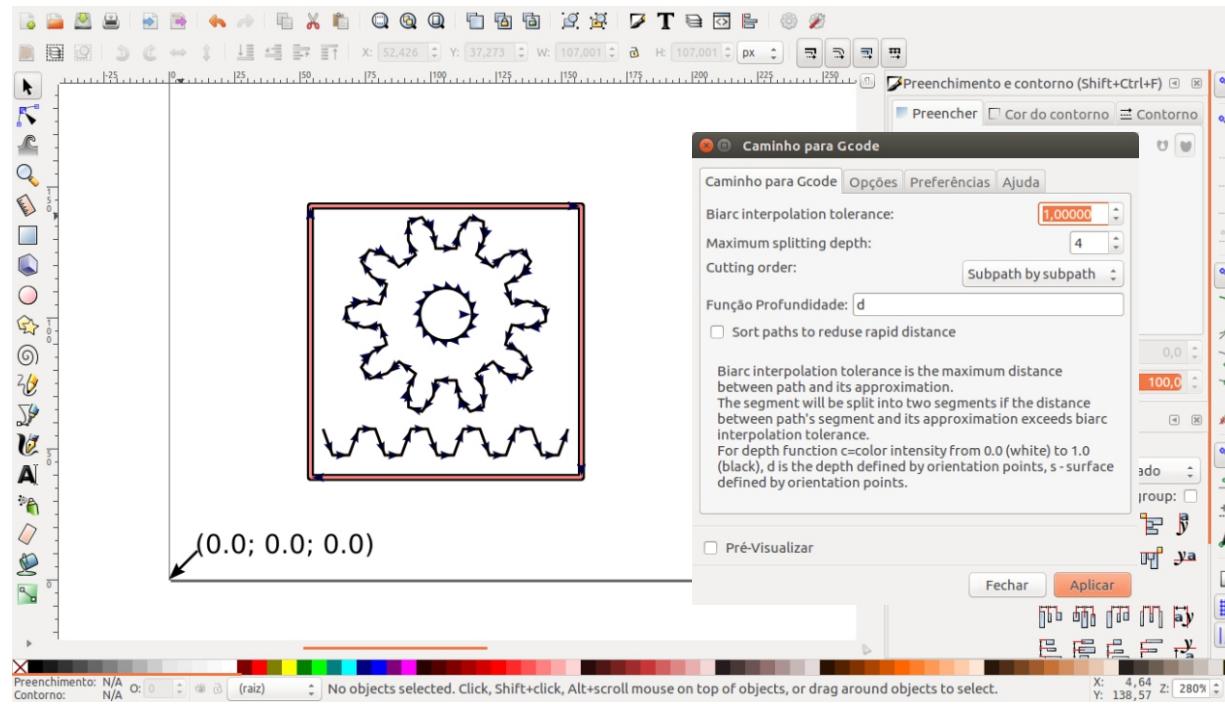
revisão do projeto



ponta da broca

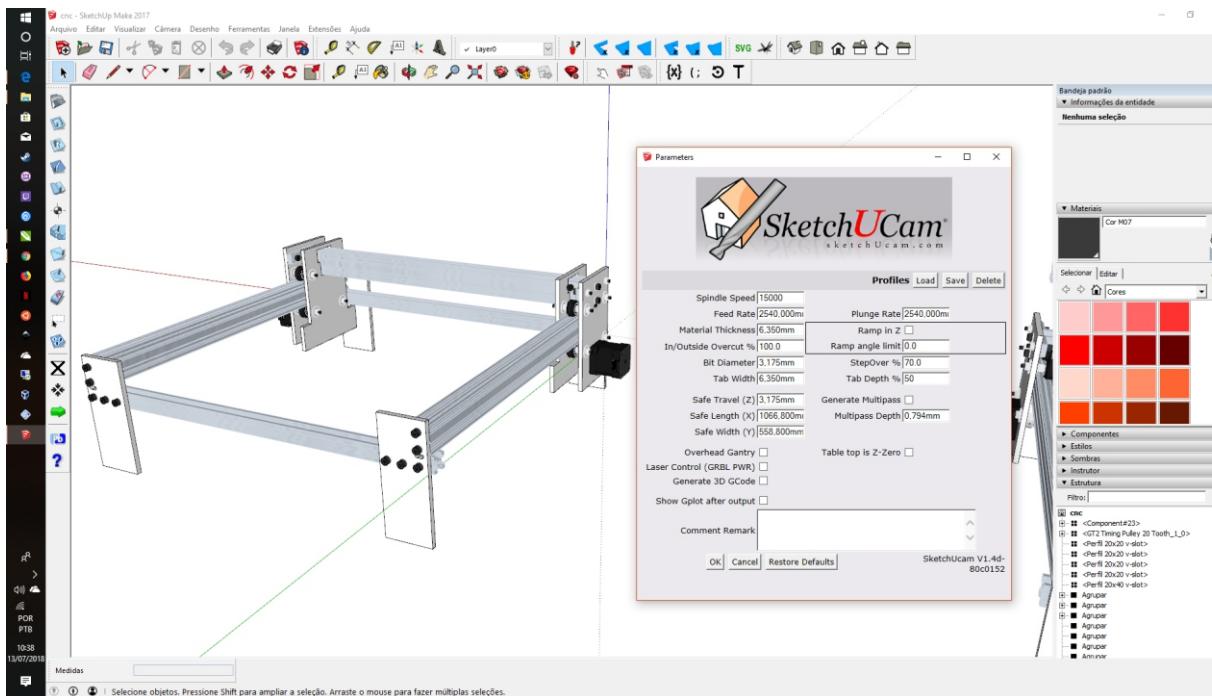
softwares auxiliares

softwares auxiliares



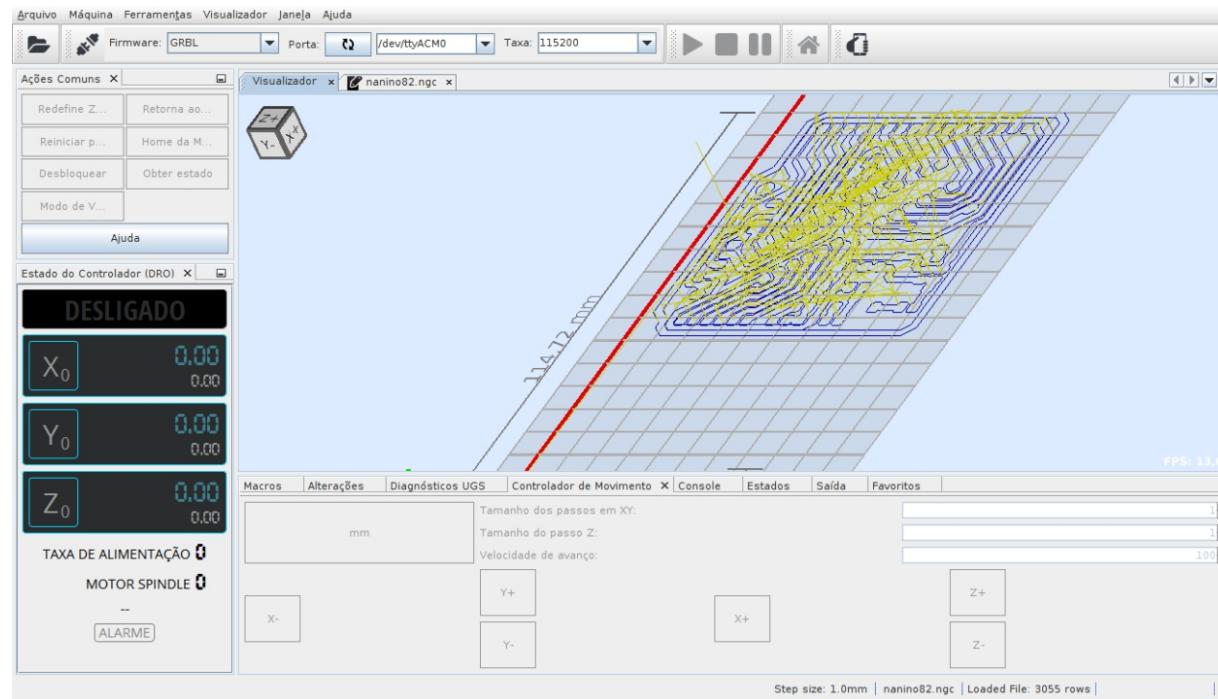
inkscape

softwares auxiliares



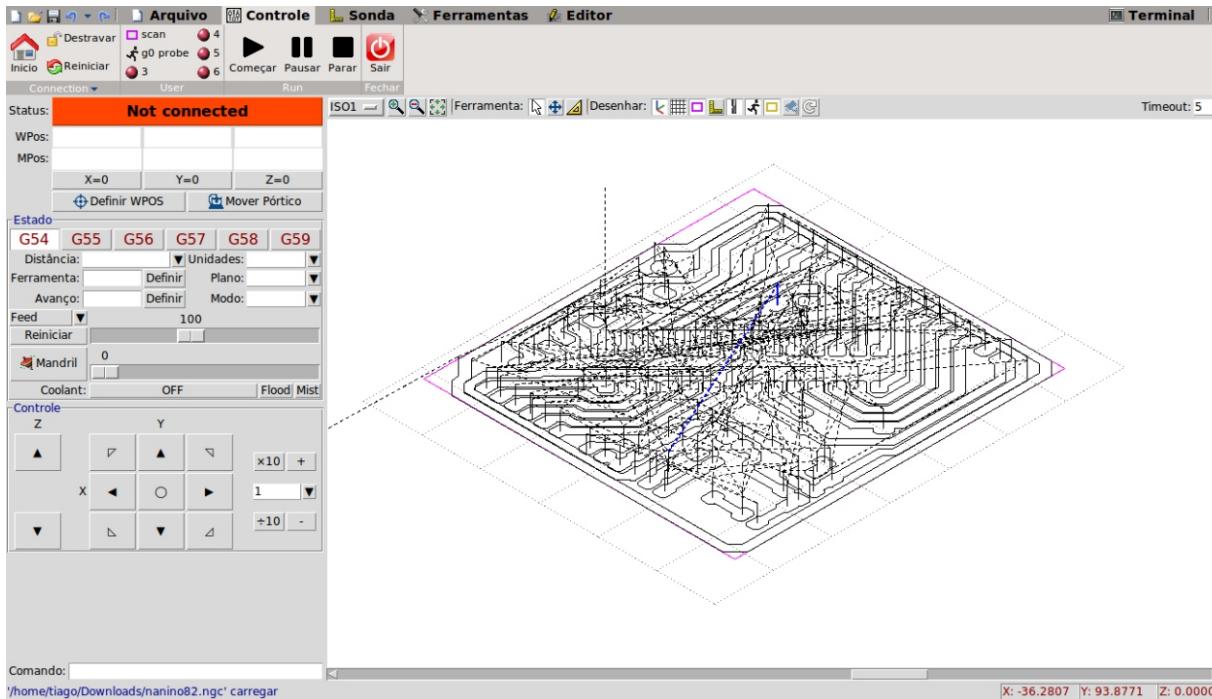
sketchup

softwares auxiliares



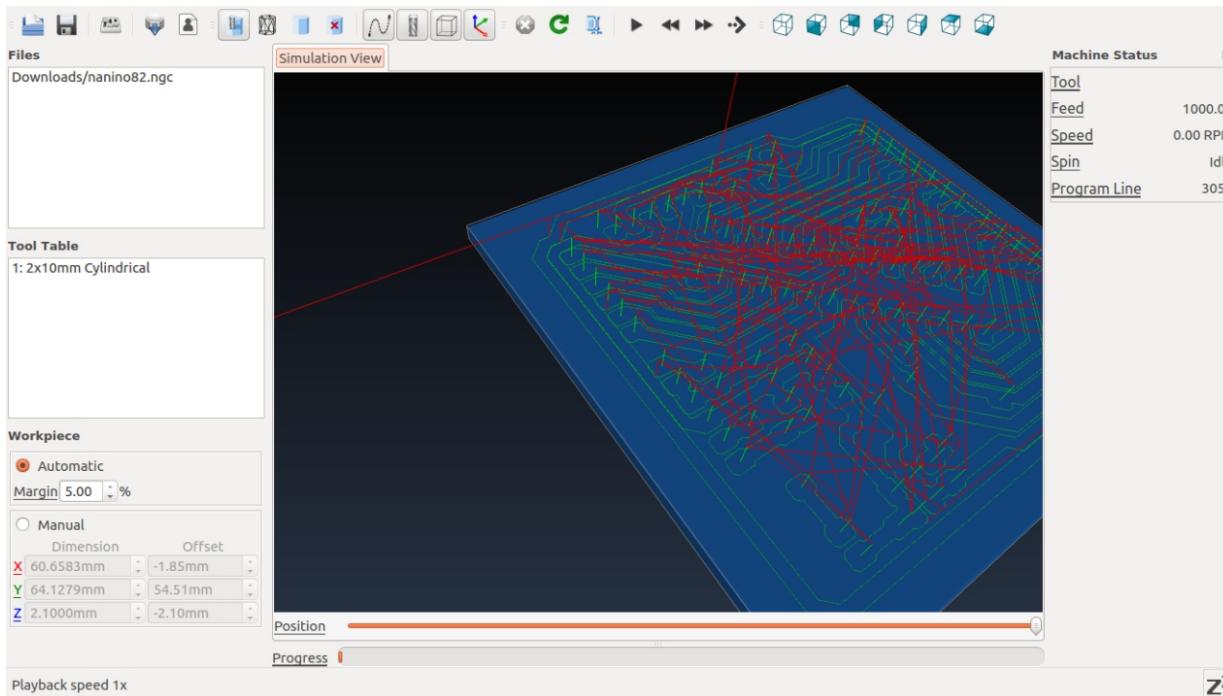
universal g code sender master

softwares auxiliares



bcnc

softwares auxiliares



camotics

propostas futuras

o que esperar do projeto

propostas futuras

- estrutura em alumínio
- aumento da resolução de trabalho
- estrutura trabalhando na vertical
- eixo y em rodas ao invés de trilhos
- utilização na cozinha
- utilização para artesanato
- utilização na construção civil
- utilização na construção de equipamentos eletrônicos
- cortes a laser
- soluções utilizadas em impressão 3d
- diferenciar densidade dos materiais
- hardware próprio
- software próprio

dúvidas