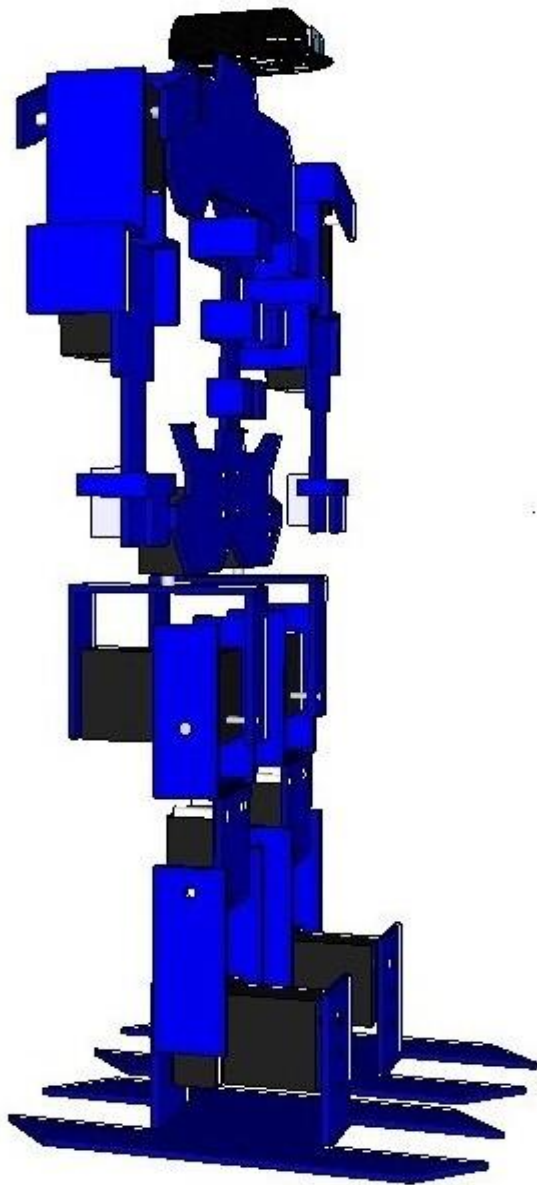


Humanoide de 21 articulacions

Comandat per Bluetooth i Infraroig



2n Batxillerat A
I.E.S. Cendrassos
2012-2013

• Introducció	2
MEMORIA:	
1. Components.....	
1.1 IR.....	3
1.2 Microcontrolador.....	3
1.3 Interruptor.....	4
1.4 Altres elements.....	4
1.4.1 LiPo.....	4
1.4.2 Placa 21 servos.....	5
1.4.3 Placa IR i brunzidor.....	6
1.4.4 Servos	6
1.4.5 Control.....	6
PLANOLS	
1. Disseny.....	
1.1 Dissenys del cos.....	
1.1.1 Primer disseny del cos.....	8
1.1.2 Disseny final del cos.....	10
PLEC DE CONDICIOS	
1. Material.....	12
2. Eines.....	13
PROGRAMACIÓ	
1. Mètodes.....	14
2. Definir les variables.....	16
3. Funcionament del programa.....	18
4. Posició dels servos en “saluda”.....	19
PRESSUPOST	20
• Glossari	21
• Conclusions	22
• Agraïments	25
• Bibliografia	26
• Anexes	27

Introducció

La robòtica és una paraula que tothom ha sentit alguna vegada , i relaciona automàticament amb robot , però , què és un robot?

Doncs bé , un robot és un “ordinador” capaç d’interactuar amb l’entorn, és a dir , donar respostes diferents davant estímuls diferents i que hauria d’obeir les 3 lleis d’Isaac Asimov, (que formen part de la ciència ficció , però que s’haurien d’acomplir) que són:

- I.** Un robot no pot fer mal a un ésser humà o per inacció permetre que un ésser humà prengui mal.
- II.** Un robot ha d’obeir les ordres d’un ésser humà a no ser que entrin en conflicte amb la primera llei.
- III.** Un robot ha de protegir la seva existència sempre que no vagi en contra de la primera i segona llei.

Un autòmat és un robot que no necessita interacció humana per a funcionar bé , però si la pot utilitzar per aprendre .Per exemple , Torres Quevedo va inventar un autòmat que jugava a escacs , no pas d’una manera convencional , sinó que ell posseïa un rei i una torre , ambdues fitxes blanques , i la persona que hi jugava tenia el rei negre.

Curiosament aquest autòmat sempre aconseguia fer escac i mat , ja que “recordava” quina jugada li havia sortit bé davant de cada reacció del seu contrincant.

Aquest projecte però , està encarat als humanoides , que són robots amb forma humana , o que intenten imitar l’estructura del cos humà .

Aquest apartat de la robòtica ha avançat molt en els últims anys , fins al punt de crear-ne algun amb un semblant quasi perfecte . Fins i tot han fet humanoides que ballen i canten a la vegada , que escriuen amb guix o amb bolígraf , que juguen a futbol o a tennis taula , i fins i tot n’hi ha amb una precisió tan gran que agafen un ou amb una mà i amb l’altre li fan donar voltes.

Això fins fa poc podria haver estat catalogat de ciència-ficció , però ara és una cosa tan comú que no se l’hi dona la importància que li pertoca.



Ara puc apreciar la dificultat de crear un humanoide , o un robot senzill que realitzi tasques , per molt senzilles que siguin , ja que darrera d’un moviment tant simple com pot ser moure un braç robòtic de dos articulacions hi ha la construcció d’aquest braç , un estudi previ del que pot fer i finalment la programació del moviment esmentat.

Aquest projecte és una continuació d’un humanoide fet com a projecte final de 4rt , el qual em va agradar i em va ajudar a determinar el que m’agradaria estudiar , i vaig tenir la sort que el meu tutor em va guiar a fer aquest projecte , perquè jo sol no m’atrevia a fer-lo , ja que ho veia molt difícil.

1. Components

1.1 IR

A la llum infraroja (IR) se li pot donar moltes funcions , però a continuació se n'explicarà la comunicació .

La comunicació per infrarojos es basa en un emissor i un receptor . El receptor envia polsos de llum a intervals regulars en funció de la senyal que ha d'enviar , així doncs el receptor rep el nombre de polsos i els envia al microcontrolador que interpreta el nombre de polsos com a una ordre .

Aquest sistema de comunicació és molt econòmic , i al ser llum és ràpid i té poca interferència , el que em permet utilitzar-lo per comandar un robot , ja que si la llum no el toca directament, rebotarà a la paret i el rebrà (sempre que

les parets no siguin negres).



Emisor



Receptor

1.2 Microcontrolador

Un microcontrolador és un element programable que rep senyals , les interpreta , i actua d'una manera d'acord amb el que li hem dit que fes.

Hi ha una gran varietat de fabricants i tipus de microcontroladors , jo uso el **18M2** de PICAXE.

El 18M2 és una versió millorada del 18M , i és compatible amb la placa que comanda els 21 servos, així que al tenir una gran memòria em serveix per guardar ordres i fer que les executi per mitjà de l'IR.



El 18M2 té una memòria total de 2048 bytes , pot fer 8 programes paral·lels (el meu és la versió plus del 18M2) , i suporta fins a 16 entrades i/o sortides i 10 sensors captatius, el programo amb el Programming Editor versió 5.4.3 i aquest és un esquema amb les connexions:

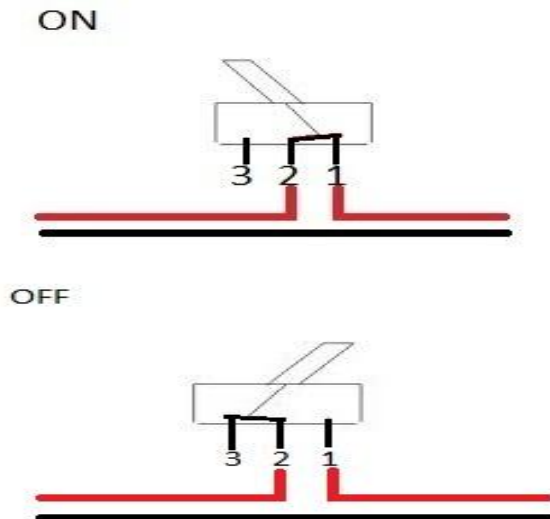
PICAXE-18M2

(DAC / Touch / ADC / Out / In) C.2	1	18	C.1 (In / Out / ADC / Touch)
(SRQ / Out) Serial Out / C.3	2	17	C.0 (In / Out / ADC / Touch)
(In) Serial In / C.4	3	16	C.7 (In / Out) {kb data}
(In) C.5	4	15	C.6 (In / Out) {kb clock}
0V	5	14	+V
(SRI / Out / In) B.0	6	13	B.7 (In / Out / ADC / Touch)
(i2c sda / Touch / ADC / Out / In) B.1	7	12	B.6 (In / Out / ADC / Touch / pwm)
(hserin / Touch / ADC / Out / In) B.2	8	11	B.5 (In / Out / ADC / Touch / hserout)
(pwm / Touch / ADC / Out / In) B.3	9	10	B.4 (In / Out / ADC / Touch / i2c scl)

Aquet microcontrolador té 10 pins, que son els 1,7,8,9,10,11,12,13,17 i 18, que tenen la capacitat d'atorgar a senyals continues un número que les representi . Gràcies a això podem comanar el robot de moltes maneres diferents , fins i tot podríem fer-lo autòmat (amb la petita limitació que la bateria ha de ser canviada i/o carregada per a una persona humana).

1.3 Interruptor

D'interruptor uso un commutador de vaivé (que té tres connexions i dos posicions). Dos de les connexions van connectades al seu cable corresponent , i allà és on s'originarà el tall del corrent. Un dels dos pins ha de ser el del mig , perquè si fossin els dos laterals mai hi hauria contacte.



Aquí es veu com funciona:

La palanca està connectada a una placa que es mou entre els pins 1-2 i el 2-3.

En el moment en que el pin 1 i 2 estan en contacte amb la placa (son els pins connectats) hi passa corrent(ON) , però quan canviem la posició , es a dir que connectem els pins 2 i 3 , deixa de passar el corren (OFF).

1.4. Altres elements

1.4.1 LiPo

Una LiPo (polímer de Liti) és una bateria que pesa poc i és capaç de donar una gran intensitat de corrent , tenen un voltatge de 3,7 V per cel·la , es a dir , dos cel·les en sèrie 7,4 V , tres cel·les 11,1 V , i així successivament (encara que solen anar de 1 cel·la a 4 cel·les).

Les LiPo tenen un petit inconvenient , i és que si baixen de 3 V (per cel·la) poden quedar malmeses.

Tenen un voltatge màxim de 4,2 V .

Les que jo tinc, són de 7,4 V (dos cel·les) i 2000mA·h (és la intensitat que poden arribar a donar per hora) .

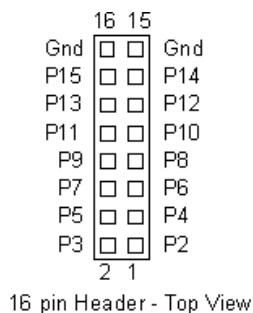
Una pesa 50 g i l'altre 100 g , per la qual cosa són perfectes per l'humanoide , ja que li donen una autonomia d'uns 40 minuts i tenen un pes molt reduït.



1.4.2 Placa 21 Servos

La placa que faig servir és una placa de PICAXE capaç de fer moure 21 servos (anomenada SD21), amb la possibilitat de regular-ne la velocitat i els manté l'últim valor donat (és a dir, conserven la posició), amb això s'evita que el robot pugui caure i trencar-se per falta de senyal als servos. Es pot alimentar de dos maneres diferents:

- Amb dos fons d'alimentació, una de 5V per al microprocessador i una de 7,2 V per als servos.
- Amb una sola font d'alimentació de 7,2V directa al servo i un pont amb regulador de voltatge que el subministra al microprocessador.



Les sortides de la placa que faig servir són les dels servos i la P8 (OUT 0) per al brunzidor.

D'entrada faig servir la P2 (IN0) com a senyal d'IR i P3 (IN1) per al bluetooth. Com que la placa no té l'entrada de Jack Stereo (l'entrada per a programar) s'hi ha d'acoblar igual que el brunzidor i el receptor d'IR.

Per començar s'ha de saber que s'ha de soldar, i on s'ha de soldar.

L'infraroig s'ha de soldar a IN0, i el brunzidor a OUT0, al jack stereo s'ha de fer coincidir el pin Tx amb el Tx de la placa, el GND amb el GND de la placa i l'Rx amb l'Rx de la placa.

Per acabar s'ha de buscar l'esquema de soldadura de l'infraroig, ja que no es pot soldar directament, sinó que s'hi ha de fer un circuit concret per a poder-lo unir amb la placa principal.

Es tria el color dels cables:

Senyal IR: Blanc.

Senyal Brunzidor: Verd clar

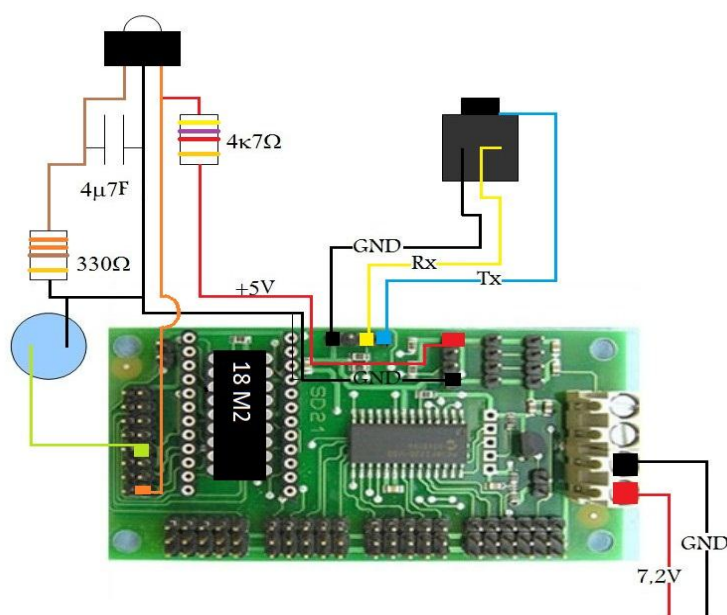
GND jack stereo: Negre

GND IR i Brunzidor: Negre

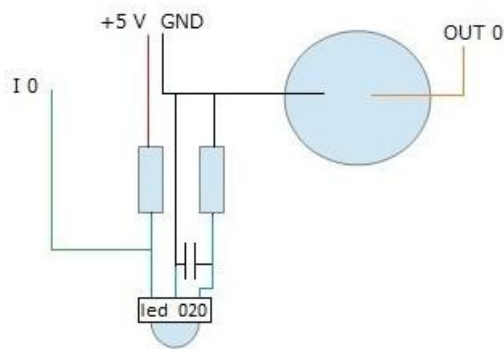
Rx: Blau fosc

Tx: Blau turquesa

Així doncs, l'esquema de la placa acabada és així (els colors de la placa no es corresponen amb els del cable per facilitar-ne la visió):



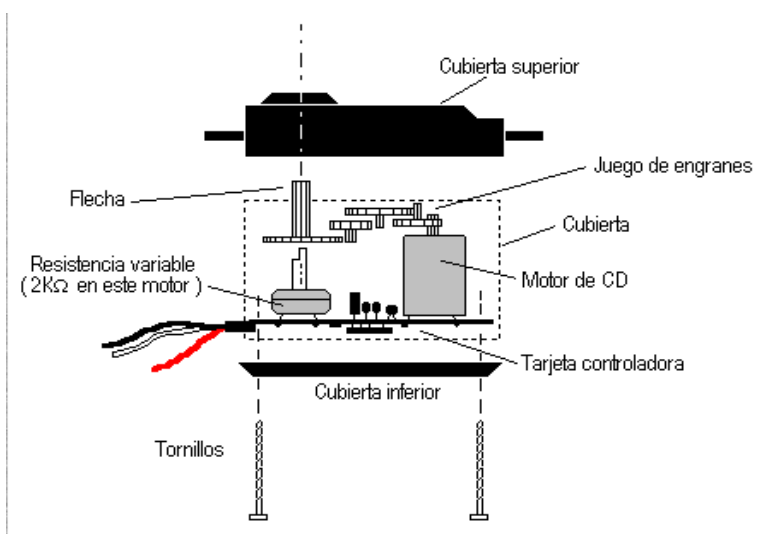
1.4.3 Placa IR i Brunzidor



La placa que conté el receptor d'IR i el brunzidor és una placa expressa per a crear circuits i està feta de manera que es puguin fer les soldadures que es vulguin i les connexions que es necessitin. Així només s'ha de visualitzar l'esquema elèctric sobre la placa i començar a soldar.

La imatge correspon a un esquema de connexions de la placa, on s'especifiquen les entrades i les sortides a les quals s'ha de connectar la placa, si estan amb sèrie o paral·lel i les posicions de les dos resistències i el condensador, i a on.

1.4.4 Servos



Un servo (servomotor) és un motor que sol girar entre 0 i 180°, i es programen per polsos, així un pols de 30 ms (30 mil·lisegons) equival a 0 graus, 150ms a 90° i 260 a 180°. Els servos tenen tres cables, el positiu, el negatiu i el de senyal. L'avantatge del servo és que pots controlar en tot moment la posició en la qual es troba. Consta d'una plaqueta que en controla la posició, d'un motor i d'una reducció per a baixar la velocitat i augmentar-ne la força.

1.4.5 Control

Altres elements com ara les resistències, condensadors i díodes serveixen per a garantir el bon funcionament d'una placa i els seus components.

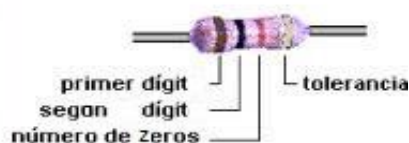
Resistències (resistors)

Una resistència elèctrica té la funció de limitar el voltatge d'un circuit, o d'un dels seus trams, per a poder protegir elements que consumeixen un voltatge menor del que proporciona el generador. Està constituïda per a una bobina (un fil de metall que està enrotllat) per tal d'oposar-se al pas del corrent i dificultar-lo.



Color	Valor
negre	0
marro	1
vermell	2
groc	3
verd	4
blau	5
viola	6
negre	7
negre	8
negre	9

Tolerancia	
10 %	
5 %	



La unitat de resistència és l'Ohm (Ω), i cada una d'elles té un valor assignat, que és la resistència al pas del corrent. Aquest valor ve representat en un codi de colors en 4 franges: la primera i la segona indiquen un número de l'1 al 9, la tercera un exponent de base 10 i la quarta la tolerància.

Condensadors

Un condensador és un element que emmagatzema voltatge i el deixa anar de cop.



El seu funcionament es basa en un principi molt senzill, però això no vol dir que un condensador ho sigui.

Si comparem un circuit tancat d'aigua amb un circuit elèctric podem entendre el funcionament del condensador. En el moment en que la manega es separa del punt d'entrada i sortida hi queda aigua, i on la manega és més ampla hi ha més aigua. Doncs bé, el condensador seria el tram ample de la manega, i es "desconnecta" del generador per a emmagatzemar, i quan és ple es buida de cop.

La unitat de carrega és el fard, però com que els valors són molt més petits es sol usar el microfard.

Díodes

Un díode és un element de protecció que sols deixa passar el corrent en un sentit, ja que està format per dos semiconductors que depenent del costat on els vingui el flux d'electrons el bloquegen o el deixen passar.

Els díodes més coneguts són els LEDs (Light Emitting Díode), que actualment són molt usats en il·luminació i electrònica.

Aquest tipus de díode consumeix 1,4 volts, i té el problema que si inverteixes la polaritat es fon.

Per evitar-ho els LEDs tenen una pota més llarga que l'altre (la que va al pol positiu) i tenen un costat recte que és visible des de la part inferior i està al pol negatiu.

Els LEDs gasten poc i són capaços de produir una bona il·luminació, per això s'usen tant.

Actualment aquests petits emissors són capaços de emetre llum visible (verd, vermell, blau, groc...), infraroja (és l'aspecte que més m'interessa en aquest projecte), es a dir, que aquest component és molt útil tant per a comunicació visual com per a comunicació infraroja, i fins i tot per a certes investigacions.



1. Disseny

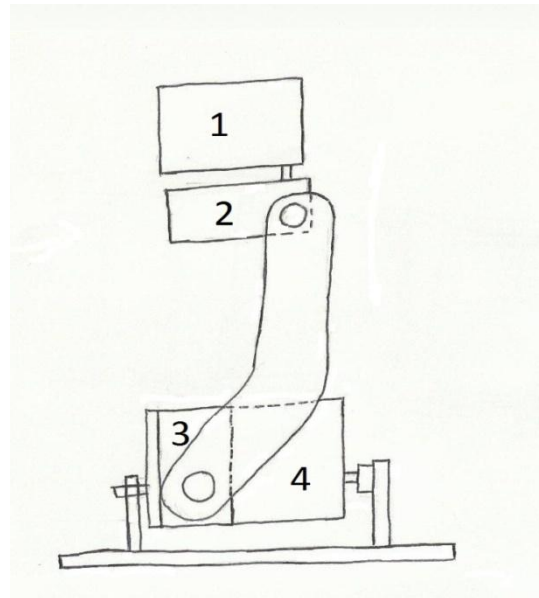
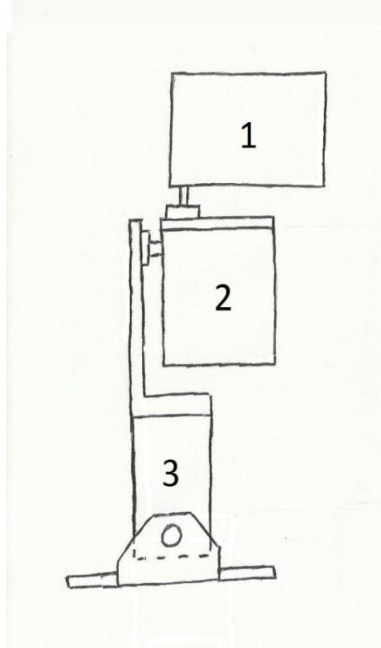
1.1 Dissenys del cos

1.1.1 Primer disseny del cos

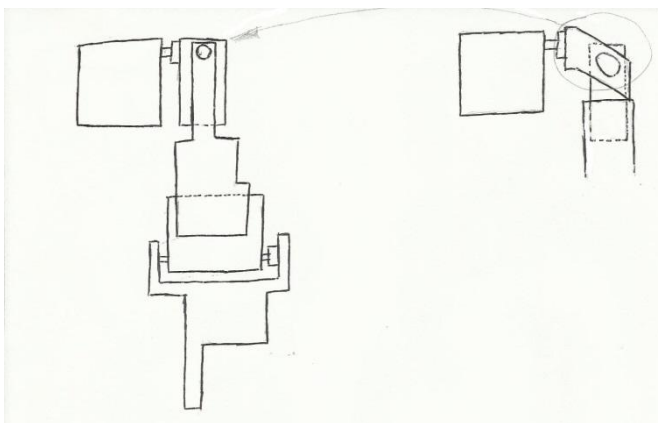
En els primers esbossos no vaig tenir en compte que els motors havien de suportar el pes del robot , així que vaig haver de fer modificacions per tal de poder mantenir la llibertat de moviments però que a la vegada fossin capaços de mantenir el robot dempeus.

El primer esbós del cos va ser la unió de les extremitats prèviament dissenyades i millorades.

La cama , que originàriament comptava amb tres articulacions, va passar a tenir-ne 4 , amb una millor utilització de l'espai.



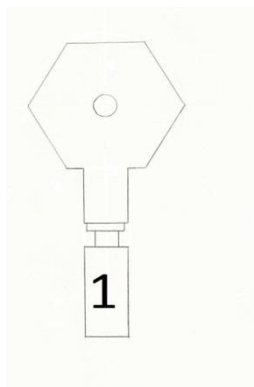
La vista de l'esquerra és frontal i correspon a el primer disseny de tots , mentre que la de la dreta és posterior , s'hi posa una quarta articulació (al turmell) i la seva vista és lateral .



El braç també va ser modificat , però d'una manera més simple ja que només es va modificar la unió de l'espatlla amb l'extremitat mencionada. Més endavant , en dissenys posteriors hi vaig afegir una pinça a cada ma per a completar el braç.

El cap va ser la part més simple, ja que només tenia una articulació i la forma del cap era senzilla.

Estava compost per un hexàgon acabat amb un fragment rectangular per on es collava al servo. Al centre hi havia un orifici on s'hi havia de col·locar un LED, només tenia un grau de moviment.

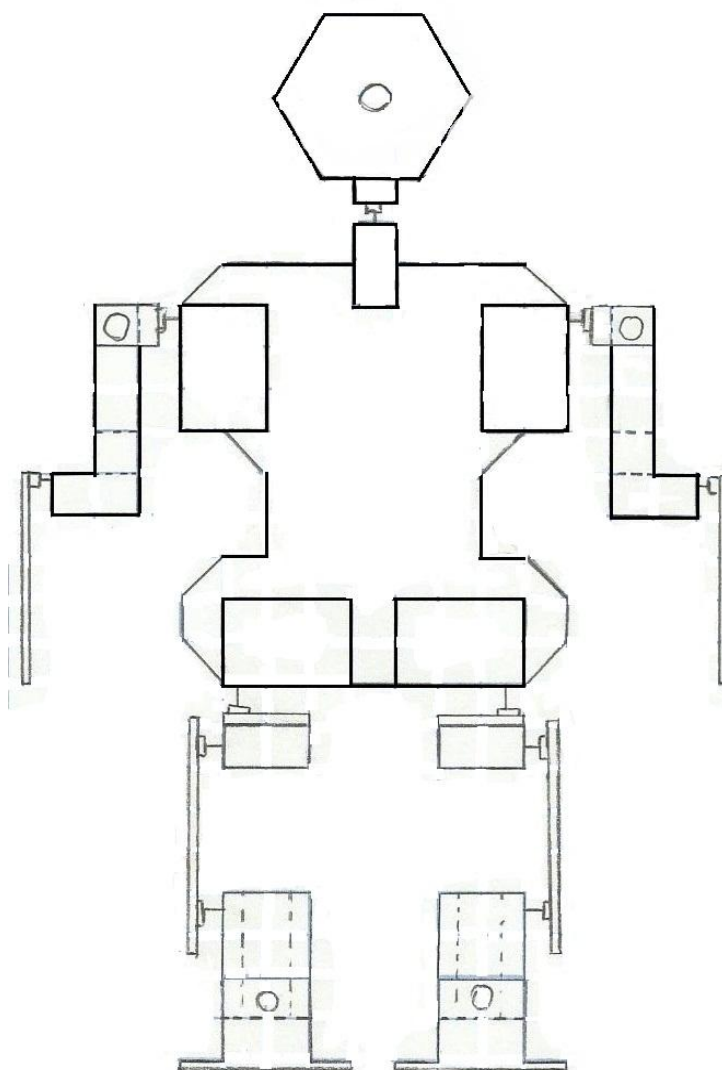


El disseny final del cap no ha canviat tant, ja que encara conserva un sol grau de moviment després d'haver considerat un segon grau, però el pes era més gran del que podia suportar.

El servo 1 és un servo que pot suportar 1,5 Kg·cm, però si es té en compte el parell del segon servo ("l'efecte de la palanca") no té prou força per a sostenir-lo.

Així doncs el servo li permet rotar fins a un màxim de 180° assumint que a 90° sempre mirarà endavant, i 0 i 180 són els costats dret i esquerre respectivament.

Amb aquest primer disseny només vaig haver d'anar modificant articulacions i unions per arribar a tenir un disseny final satisfactori.



1.1.2 Disseny final del cos

En el disseny final del cos buscava que els eixos dels servos no patissin , que el robot pogués estar dret , que tinguessin llibertat de moviment i que tingués una bona estètica (sempre intentant que fos el màxim de lleuger possible).

En total vaig arribar a fer 3 dissenys amb les seves rectificacions corresponents. El primer és el mostrat anteriorment , el segon és un al qual vaig començar a definir la forma dels braços i les cames (adjunt als annexes) i el tercer va sortir de la modificació del cos i de les cames del segon , però conservant-ne els braços.

Aquest disseny compta amb els 21 servos distribuïts per tot el cos donant així la mobilitat que busco en aquest projecte , i em facilita la simulació d'accions plenament humanes.

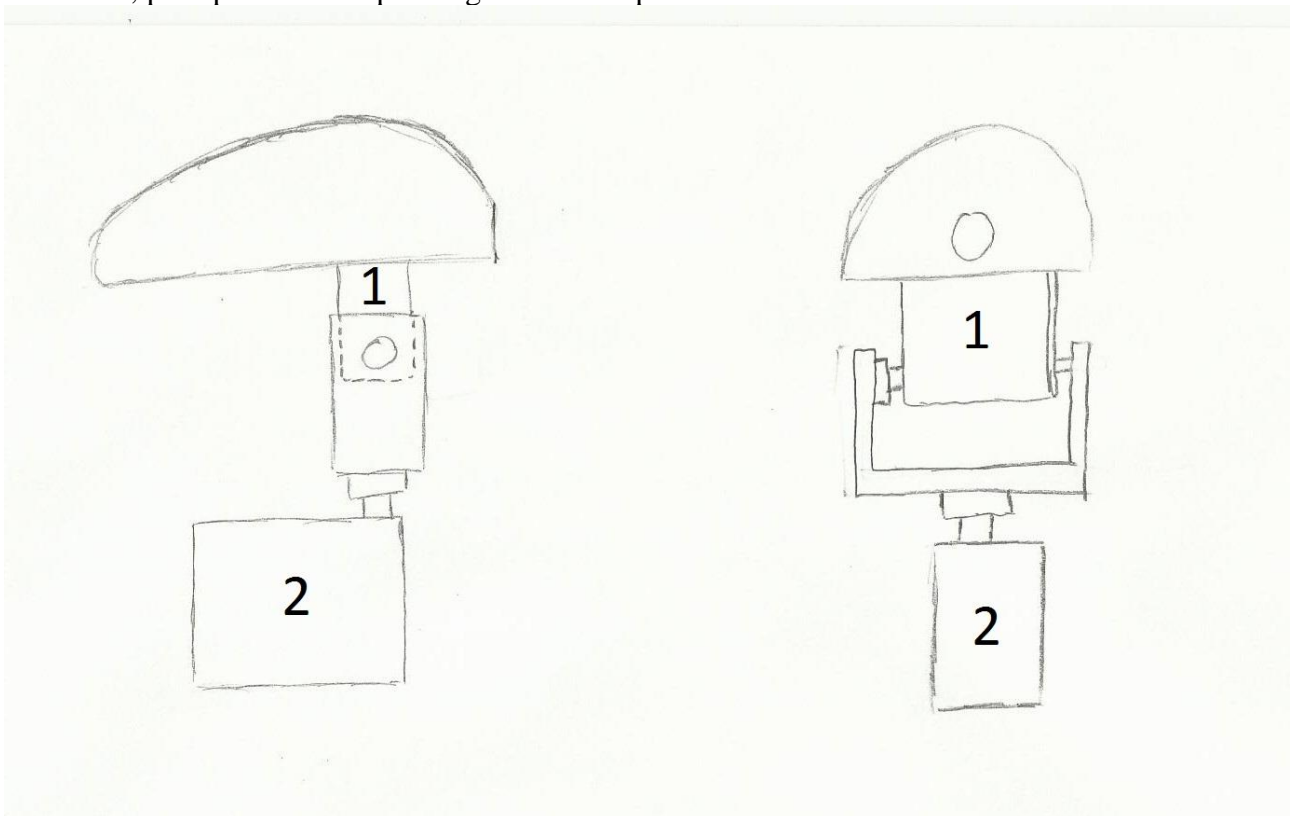
Evidentment , quantes més articulacions hi hagi més difícil serà el seu control i programació.

El cap està format per un ratolí vell , amb una càmera de RF(radiofreqüència) a l'interior , (la idea d'usar un ratolí com a cap ja s'havia fet servir en projectes anteriors).

Els braços compten amb 4 servos cada un , i estan constituïts de 5 peces diferents que constitueixen el suport del servo i donen aparença de braç. El cos està format per una peça per les espatlles , on reposen els braços , una altra per a la columna que uneix les espatlles i la cintura , i suporta la placa, i una darrera per la cintura , que uneix les cames amb la resta del cos .I finalment, les cames , estan formades per 6 servos cada una amb 5 peces diferents que mantenen l'estabilitat del robot i permeten modificar el seu sosteniment sobre el terra.

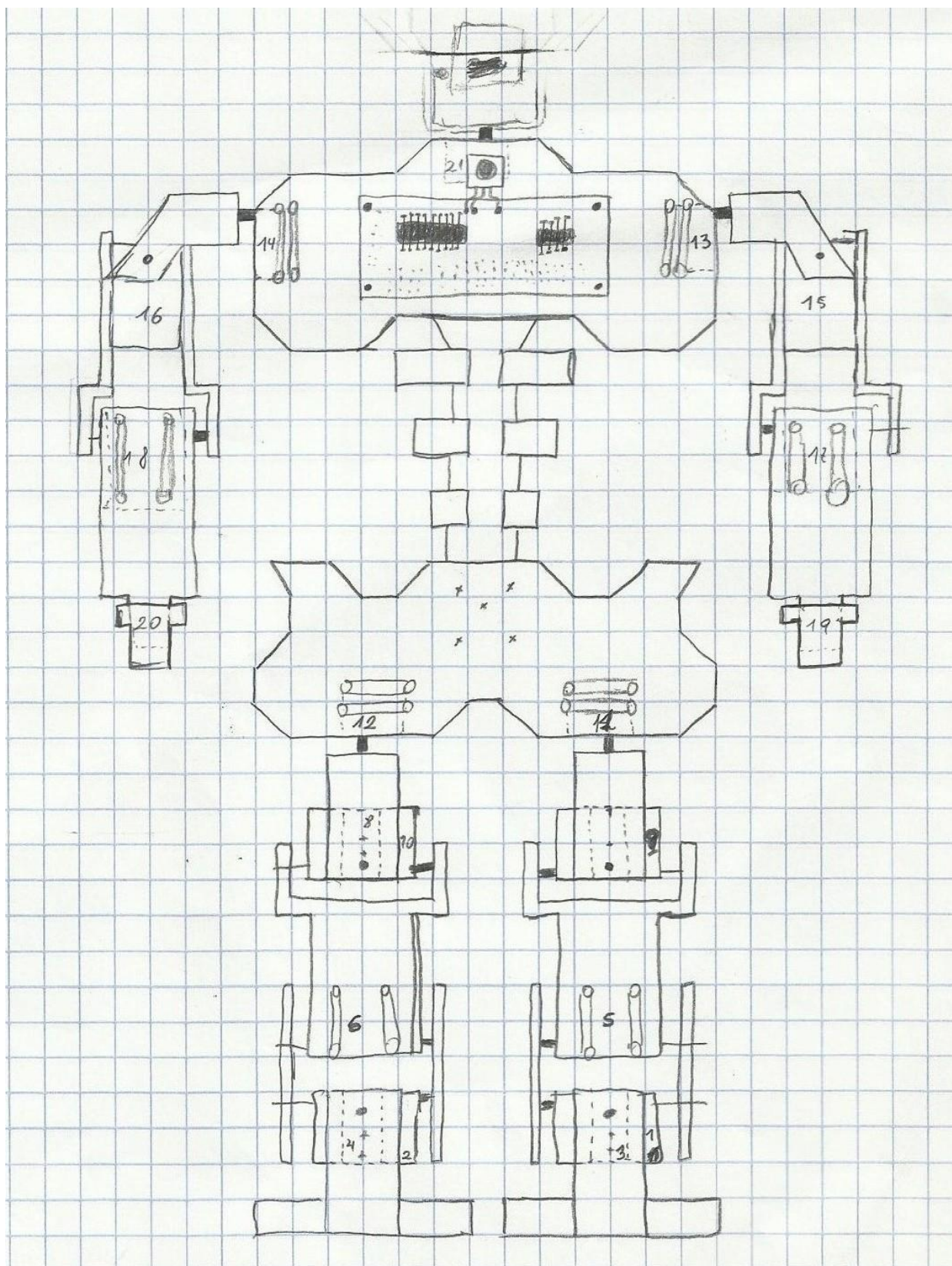
En total el robot està format per 23 peces (més el ratolí del cap i la càmera) i 21 servos de 3 tipus diferents , amb un pes total de 20,58 N (2,1 Kg).

En el segon disseny hi posava una segona articulació al cap , permetent que pogués mirar endalt i avall, però per raons de pes vaig haver de suprimir-lo.



Així doncs , el disseny final es aquest :

El disseny anterior li falta el cap actual , ja que ha sigut una modificació posterior . Fa 50 cm d'altura i 32,5 cm d'ample a les espatlles.



Plec de condicions

1.Material

El material necessari és el següent:

- Metacrilat (1,12 m² gastats) , per a fer el xassís de l'humanoide
- Cargols cabota cònica de 3mm x 9 mm (x18 unitats), per a fer la continuació dels eixos dels servos.
- Visos de 2 mm x 6 mm (x 21 unitats) per fer unions entre dos servos, (quatre vegades) per aconseguir dos graus de moviment.
- “Servo 13 Kg GoTeck GS-5515 MG” (Ref.707311) (x 16 unitats), per a fer les articulacions on es necessita més força per a la mobilitat del robot.
- “Servo 6 Kg piñonera metalica con rodamientos”(Ref.707308) (x 2 unitats), per a fer les articulacions dels colzes.
- “Servo 9 gramos Azul”(Ref.707101) (x 3 unitats), per a fer les articulacions del cap i les dos pinces.
- Cinta adhesiva de doble cara de 2cm d'ample (0,75 M) per a unir la prolongació de l'eix del servo amb el servo, enganxar el servo a el xassís i enganxar les plaques al xassís.
- Brides 360 mm x4,8 mm (x 66 unitats), per enganxar els servos al xassís i mantenir els cables dels servos units.
- Cable de coure amb protecció (32 cm) ,per a fer les unions de la placa principal amb les Li Pos i la placa receptora d'infraroig.
- Femelles autoblocants hexagonals de 3 mm (x 20 unitats) , per a fixar la prolongació de l'eix del servo i la placa al xassís.
- Ratolí vell , per a fer de cap del robot.
- LED 020 , per a soldar-lo a una plaqueta i rebre l'infraroig.
- Brunzidor , per a emetre sons per tenir controlat el programa i la LiPo.
- Condensador de 4,7 µF , per a fer la placa receptora d'infraroig.
- Resistència de 330 Ω , per a fer la placa receptora d'infraroig.
- Resistència de 4,7 K Ω ,per a fer la placa receptora d'infraroig.
- Placa per crear circuits , per crear la placa receptora d'infraroig.
- Interruptor , per a controlar l'encès i apagat de la placa.
- Allargs de 10 cm (x5) , per allargar els cables dels servos perquè arribin a la placa.
- Allargs de 20 cm (x5) , per allargar els cables dels servos perquè arribin a la placa.
- Lipo de 2000 mAh, 15 C, 7,4 V i 50g , per a fer funcionar el robot.
- Lipo de 2000 mAh 25 C , 7,4 V i 100g , per a fer funcionar el robot.
- SD21 , és la placa que permet comandar tot el robot i dona senyal als 21 servos.
- PICAXE 18M2 + , és el microcontrolador que es programa per a la placa SD21.
- Placa receptora/emissora de Bluetooth, és la placa que comunica el bluetooth amb la SD21 per a fer moure el robot.
- Camara Radiofreqüència per a poder veure el que veu el robot i poder dirigir-lo a distancia.
- Estany ,per a soldar.

2.Eines

Les eines que s'han fet servir son:

- Serra de calar , per tallar el metacrilat amb comoditat i precisió sense que es trenqui.
- Trepant de sobretaula , per fer els forats necessaris a les peces amb precisió i comoditat.
- Escalfador elèctric , per poder doblegar el metacrilat i donar-li la forma desitjada.
- Soldador elèctric , per fer les unions entre les connexions necessàries.
- Tornavís estrella, per collar els visos .
- Alicates , per pelar el cable per poder-lo soldar i per cargolar les femelles.
- Cable AXE 027 , per crear comunicació entre la placa i el portàtil.
- Portàtil amb PICAXE PROGRAMING EDITOR , per a poder programar el microprocessador .
- Tisores , per retallar els sobrants de les brides.
- Cúter , per tallar la cinta adhesiva de doble cara amb comoditat.
- Fulls mil·limetrats (6 unitats), per fer les plantilles de les peces del robot.
- Cinta aïllant, per evitar curtcircuits a les connexions.

1.Mètodes de programació

Els xips picaxe són xips relativament econòmics i pràctics , ja que es poden programar les vegades que es desitgi i d'una manera senzilla. Aquesta empresa està dedicada a l'ensenyament ,i intenta facilitar l'accés a l'electrònica. Les plaques que fan anar els seus microcontroladors son barates i

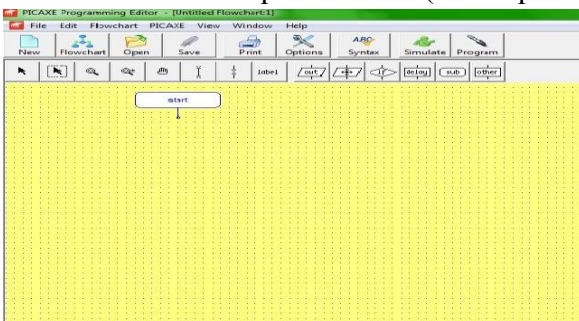
(en aquest cas robòtica)

Hi ha dos maneres de programar un xip picaxe , la primera és mitjançant un diagrama de flux , i la segona és mitjançant un programa escrit en llenguatge BASIC.

Diagrama de flux:

Consta d'una etiqueta inicial (és el principi del programa) de la qual es va creant el programa , afegint-hi ordres unides per una línia que indica en quina direcció i quin ordre s'ha d'executar el que se li ha programat.

Hi ha etiquetes concretes per a cada acció i només s'ha d'indicar la sortida a la qual afecta i/o el valor que ha d'adquirir.

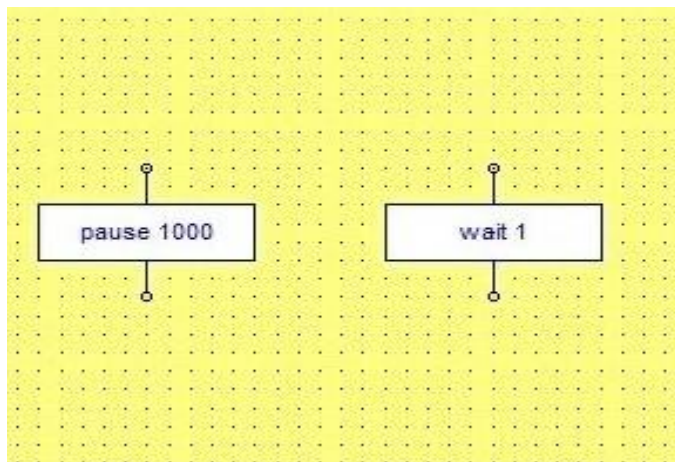


Així doncs hi ha etiquetes per a pausar momentàniament el programa.

Una d'elles és el *pause* , *pause* és medeix en mil·lisegons , per tant si es posa un 10 ,seran 0,01 segons , *pause 100* equivaldrà a 0,1 segons i *pause 1000* és el mateix que pausar durant un segon. La segona etiqueta és la de *wait*, la qual fa servir segons però només amb valors enters, per tant per escriure 2,4 segons(per exemple , no es pot escriure *wait 2,4* sinó que s'ha d'escriure *pause 2400*). Aquestes etiquetes són molt útils quan s'ha de treballar amb programes on el temps hi juga un paper important.

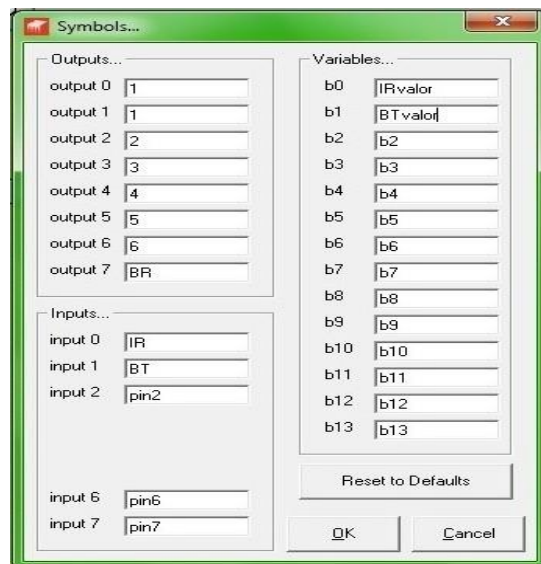
N'hi ha una altre que serveix per a activar una sortida que dona senyal a un element que

funciona amb corrent continu , com ara un LED o un motor elèctric. L'etiqueta s'escriu *high B.X* , on el B.X és el nom de la sortida que s'usa(la X correspon a un numero). L'etiqueta contrària és *Low B.X* , i serveix per apagar o deixar de donar corren a la sortida X.

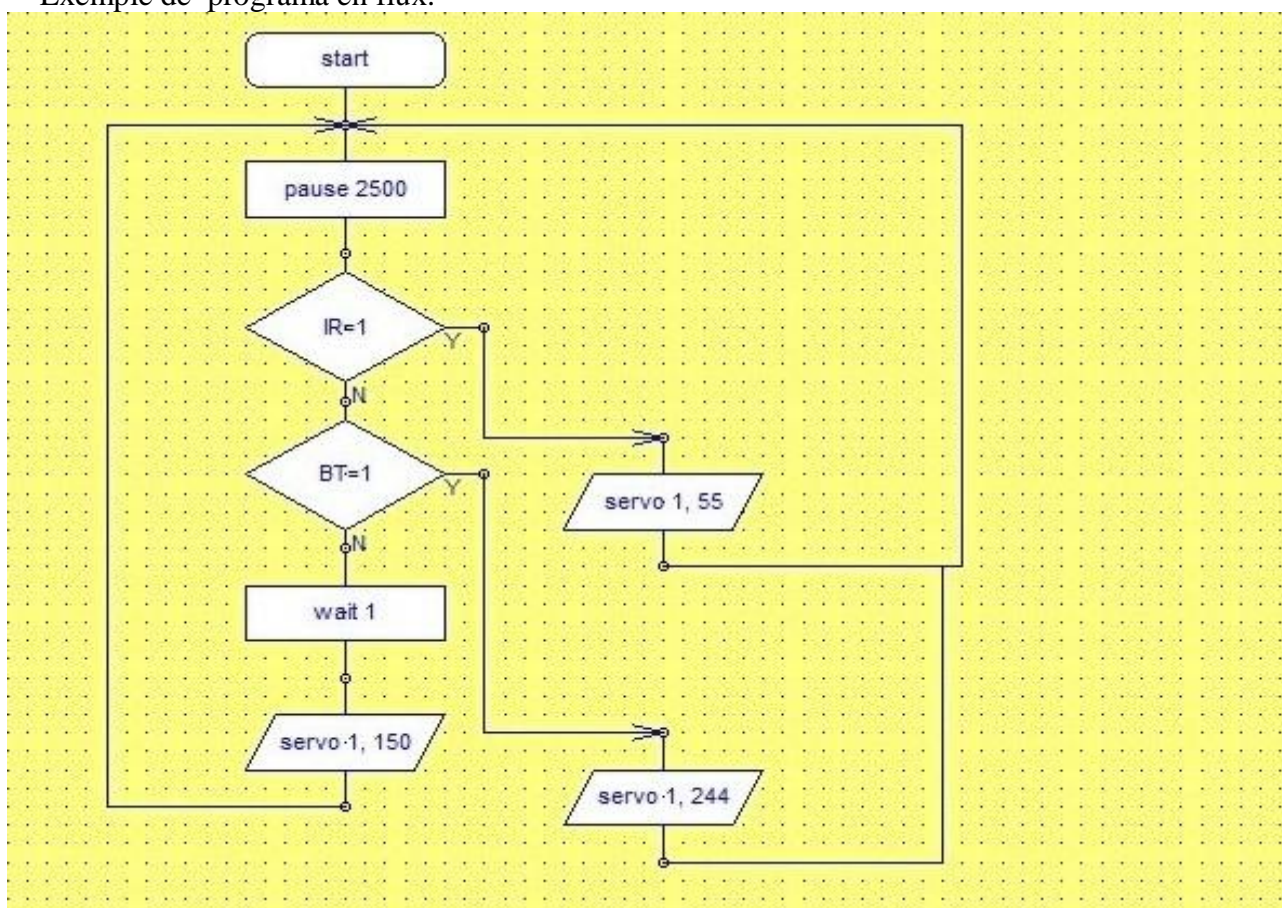


Per definir les variables s'ha d'anar a *flowchat* , i d'allà a *flowchat symbol table*. Aquí és pot anomenar amb el nom que es necessiti les entrades ,sortides i registres que té la placa.

Les output son totes les sortides que té la placa , i les input es com s'anomenen les entrades d'informació , com 'infraroig , la placa de bluetooth , la LDR (sensor sensible a la llum) ... , els que son b i un numero són els registres amb els quals es pot comptar. Els registres son els llocs on es guarden els valors que es recullen de l'infraroig , el bluetooth , les LDR.... o fins i tot on es guarden valors que predetermina l'usuari, com ara els valors inicials dels servos , un grau de rotació , qualsevol valor numèric que s'hagi de guardar , i mes tard , usar.



Exemple de programa en flux:



El programa comença amb l'start , i es para durant 2,5 segons i va a un condicional , si la variable IR és igual a 1 el servo1 es posiciona a 55° i torna a començar. Si IR és diferent de 1 el programa segueix avançant. La següent etiqueta és un altre condicional que mira la variable BT , si és igual a 1 el servo1 es posiciona a 244° i torna a l'inici , i si IR i BT son diferent de 1 , el servo es posiciona a 150°.

Programació en basic:

La programació en BASIC és igual que al diagrama de flux però escrivint el que diu l'etiqueta, així doncs el programa anterior passat a BASIC vindria a ser una cosa així:

```
1
2  symbol 1 = 0                                'es defineixen les variables
3  symbol BR = 7
4  symbol IR = pin0
5  symbol BT = pin1
6  symbol IRvalor = b0
7  symbol BTvalor = b1
8
9      let dirsB = 255
10     comprovavariab:
11
12         pause 2500
13         if IR=1 then gosub posicio+
14         if BT=1 then gosub posicio-
15         wait 1
16         servo 1, 150
17         goto comprovavariab
18
19     posicio+:
20
21         servo 1, 55
22         goto comprovavariab
23
24     posicio-:
25         servo 1, 244
26         goto comprovavariab
27
```

2. Definir les variables

Per definir les variables dels servos es fa servir el següent:

```
symbol servoTE=63      'registre gir servo1 posició base
symbol servoTEp=84     'registre gir servo1 up.
symbol servoTEn=105    'registre gir servo1 down.
symbol baseTE=145      'posició inicial o intermitja servo1.
symbol offsetTE=b1     'valor que girarà el servo
symbol velocitatTE=0   'registre control de la velocitat del servo1.
```

El fragment anterior forma part del programa complert , (concretament de la secció de variables) on hi ha 21 fragments com l'anterior , un per cada servo.

La primera línia guarda el servo com a *servoXY* , on XY corresponen a dos lletres que indiquen la posició del servo, en aquest cas les lletres TE corresponen a Turmell Esquerre.

La segona línia correspon a un registre que s'encarrega de sumar el valor que s'hi guardi a la posició a la qual es trobi el servo en el moment en que s'hi indiqui, per exemple :

```
writei2c velocitatTE , (20)
writei2c servoTEp , (73)
```

Si el servo estigués a 20° abans de l'ordre , estaria a 93 després d'haver passat per aquest fragment . La tercera línia del fragment inicial fa el contrari de la segona fila, resta la seva graduació al servo. La quarta , guarda el valor en el qual el servo té la posició base , es a dir , el grau de gir que ha de mantenir per mantenir el robot d'empeus , és només el valor , per tant s'ha de fer girar el servo fins a la posició *baseXY*(cada servo té la seva corresponent).

OffsetXY s'igual a un valor que tingui importància en la subrutina en la que s'hi iguala , i serveix com a constant de gir.

I finalment el *velocitatXT* serveix per indicar la velocitat a la qual s'ha de moure el servo depenent de l'angle perquè ho faci d'una manera regular. El valor es pot calcular amb l'expressió següent:

$$Tiempo = \left(\frac{Pos.Actual - Pos.Destino}{Reg.Velocidad} \right) * 20ms$$

Aquest es el nombre que s'introdueix en el modulu velocitat , es a dir:

Writei2c velocitatXY , ("Tiempo")

Per últim , hi ha dos símbols més :

Symbol IR = b0 Aquí es guarda el registre b0 com a registre de l'infraroig.

Symbol e_serie = b1 Es guarda el registre b1 com a registre del bluetooth.

3.Funcionament del programa

El programa complet del robot funciona amb dos condicionals , el primer condicional fa referencia a IR (infraroig) i el segon al BT (Bluetooth). El condicional funciona així (ja que els dos condicionals son iguals i nomes canvia el BT per el IR nomes s'hi mostra el IR) :

```
163 *****
164 *****PROGRAMA PRINCIPAL*****
165 *****
166
167 start:
168
169 readi2c 65, (b12)
170     if b12 < 160 then goto bateria_baixa
171
172
173
174     irin[1000,start],c.0,b0  'mira IR i => b0
175     inc b0
176     debug b0
177
178 comprovainfra:
179 if IR = 1 then goto saluda
180 if IR = 2 then goto flexionacames
181 if IR = 3 then goto observa
182 if IR = 4 then goto balla
183 'if IR = 5 then goto camina
184 if IR = 6 then goto cintura
185 if IR = 5 then goto caminados
186
187 'if IR = 7 then goto
188 'if IR = 8 then goto
189 'if IR = 9 then goto
190 'if IR = 10 then goto
191 'if IR = 11 then goto
192 if IR = 10 then gosub posicioservos
193 pause 200
194 goto inici
195
```

A la línia 167 comença amb l'*start* , és una etiqueta que et permet fer començar el programa després de cada subrutina.

A la línia 169 es llegeix el voltatge de la LiPo , es guarda al registre *b12* , i es compara amb el valor *160* (línia 170) .Si es mes baix va a la subrutina *bateria_baixa* que emet una alarma per avisar que s'ha acabat la bateria i s'ha de treure i posar a carregar , ja que si baixa d'aquest valor la bateria podria quedar malmesa , i si *b12* és mes gran o igual al valor *160* el programa segueix avançant. Mira si hi ha senyal d'infraroig , i en cas de que hi sigui , el guarda a *b0* (línia 174) , n'incrementa el valor (linia175) (ja que si envies un nombre es rep l'anterior per qüestió de polsos).I per acabar , si el robot està connectat a l'ordinador, (línia 176) ens mostra els valors que rep de l'IR.

El programa segueix avançant i arriba a *comprovainfra* (línia 178) , és el nom de l'apartat de condicions que hi he dictat , i és un punt d'orientació .

I per acabar hi ha les condicions (ja que totes son iguals explicaré només la primera) , s'agafa el valor *IR* (*b0*) que li ha donat l'entrada *C.0* (el receptor de l'infraroig) i es compara amb *1* . Si és igual , el robot anirà a la subrutina *saluda* ,on mourà el braç com si saludés , ja que se l'hi ha programat així (més endavant se'n mostra el programa).

A cada subrutina s'hi especifiquen uns servos concrets i els moviments que han de fer ,així que s'ha de tenir controlada la posició del servo en cada moment.

4.Posició dels servos en saluda

A continuació hi ha una taula en la que es presenten les posicions dels servos respecte el temps en la ordre de *saluda* (només es mouen els servos HE ,BE i ZE , per tant son els únics als quals hi ha canvi de posició.

Servo	Temps1	Temps2	Temps3	Temps4	Temps5	Temps6	Temps7	Temps8	Temps9
TE(1)	145	145	145	145	145	145	145	145	145
TD(2)	135	135	135	135	135	135	135	135	135
PE(3)	135	135	135	135	135	135	135	135	135
PD(4)	121	121	121	121	121	121	121	121	121
GE(5)	140	140	140	140	140	140	140	140	140
GD(6)	112	112	112	112	112	112	112	112	112
AE(7)	200	200	200	200	200	200	200	200	200
AD(8)	120	120	120	120	120	120	120	120	120
FE(9)	100	100	100	100	100	100	100	100	100
FD(10)	122	122	122	122	122	122	122	122	122
CE(11)	130	130	130	130	130	130	130	130	130
CD(12)	110	110	110	110	110	110	110	110	110
HE(13)	25	225	225	225	225	225	225	225	25
HD(14)	250	250	250	250	250	250	250	250	250
BE(15)	40	170	170	170	170	170	170	170	40
BD(16)	205	205	205	205	205	205	205	205	205
ZE(17)	130	130	90	170	90	170	90	170	130
ZD(18)	120	120	120	120	120	120	120	120	120
ME(19)	240	240	240	240	240	240	240	240	240
MD(20)	150	150	150	150	150	150	150	150	150
CO(21)	141	141	141	141	141	141	141	141	141

A continuació es mostra una taula amb els preus, unitats i material que he usat en aquest projecte :

Concepte	Unitats	Preu	Total
Metacrilat	2 m ²	-	Cortesia rètols Figueres
Placa 21 servos	1	33,46 €	33,46€
Servo 13kg GoTeck	16	13,85 €	279,90€*
Servo 9g Blau	4	3,84 €	
Servo 6kg	2	11,85 €	
Càmera Radio	1	82,95	89,92€
LiPo 7,4V 2A 15C	1	14 €	14 €
LiPo 7,4V 2A 25C	1	14,90€	25,80€*
Convertidor vídeo USB	1	26,95€	26,95€
Allargs Futaba 20 cm	5	0,20€	8,45*
Allargs Futaba	5	0,19€	
Carregador LiPo	1	26,95€	36,65€*
Cinta doble cara	5m	2,4€/m	12 €
Pila 9V	1	2,60€	2,60€
Placa BT i emisor	1	17,50	15,68€
Pack 12 cargols	1	0,40	0,40€
Font alimentació 12V	1	12,85	12,85€
			560,48€

*Han estat comprats per internet i s'ha de tenir en compte el preu d'enviar el paquet.

Glossari

Llum infraroja: és llum amb una freqüència mes baixa de la que nosaltres podem percebre, i es caracteritza perquè tots els cossos que tenen temperatura en desprenen.

Polsos: un pols és una senyal intermitent i irregular en el temps

Byte: és una unitat de memòria, cada byte està format per 8 bits , els quals “recorden” un 1 o un 0 ,i el que guardin ho faran amb combinacions d’aquets 8 bits i l’1 o el 0.

Sensor captatiu: un sensor captatiu és un sensor capaç de extreure informació de l’ambient.

Ampere(A): és la unitat d’intensitat del corrent elèctric , i es pot calcular amb la llei d’Ohm , la qual diu que $V=R \cdot I$ (Voltatge = Resistència x Intensitat).

Jack Stereo: és un sistema de comunicació (molt usat en audio i el més comú en els ariculars).

Tx: Tx és com s’anomena a les sortides d’informació.

Rx: així s’anomenen les entrades d’informació.

GND: Terra , born negatiu d’una font d’alimentació.

Subrutina: és com s’anomena a un programa que forma part d’un altre i que es pot es repetir dins d’un mateix cicle.

Conclusions

Aquest treball de recerca era un experiment per veure si realment m'interessaria estudiar una enginyeria , ja que no tenia molt clar quina carrera volia fer.

El que es plantejava en aquest treball era si podria fer un robot i programar-lo , però no un robot qualsevol , sinó un robot amb 21 articulacions (el problema que té tenir tantes articulacions és que has de tenir més controlada la posició dels motors en tot moment) , i en si seria capaç de dissenyar una estructura que ho suportés , passant per la tria de materials, la selecció de tots els elements que formessin part del projecte , (servos , placa , mètode de control...).

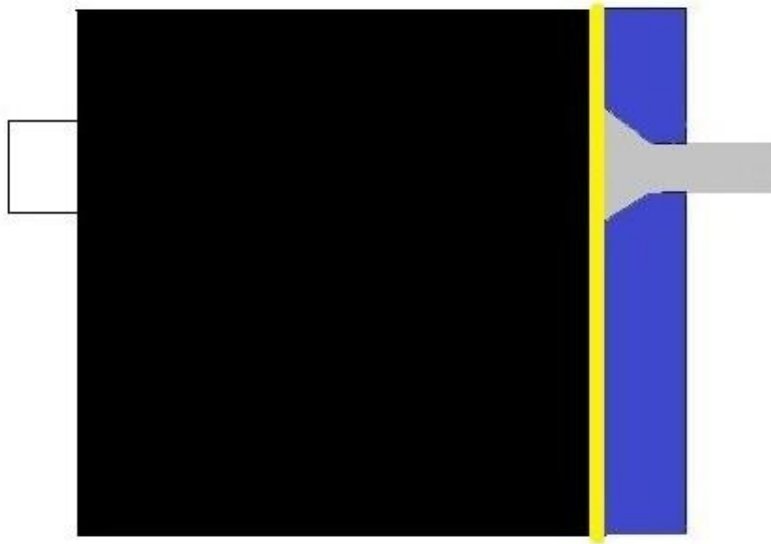
Havent arribat aquí puc afirmar que aquesta experiència amb la robòtica m'ha agradat i m'ha ajudat a dirigir el meu futur acadèmic cap a l'enginyeria.

Per altre banda , tot i que el robot camina, no està acabat del tot. A partir d'ara l'hauré d'anar modificant per aconseguir polir els errors que té. (a la següent pagina hi ha una breu explicació).

El camp de la robòtica ha avançat molt i encara ha d'avançar més , per tant vull formar part d'aquesta empresa per a millorar la qualitat de vida amb petits detalls , (ja sigui un llum que s'encén per sensor de presència , o les persianes quan surt el sol) , i intentar millorar mica a mica.

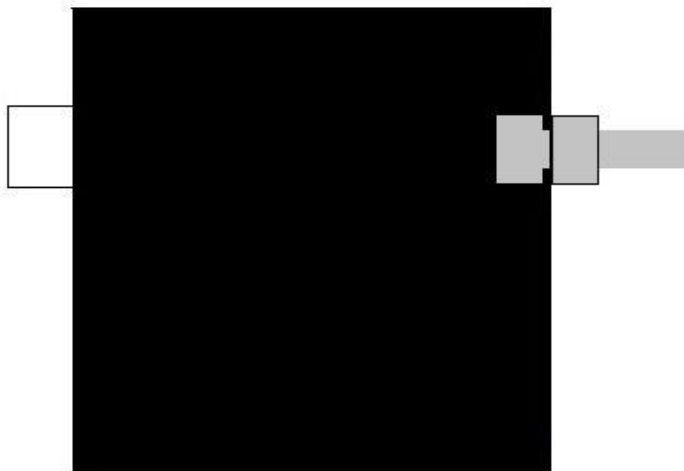
Millores Possibles

Els servos tenen un eix, però aquest eix només surt per un costat. Per aconseguir un millor recolzament s'agafa una lamina de metacrilat s'hi fa un forat de 3 mm a la sortida i una entrada cònica per posar-hi un cargol de cabota cònica, després s'hi enganxa cinta de doble cara evitant que el cargol pugui recular, i s'adhereix l'altre costat de la cinta al servo, de tal manera que l'eix del servo i el cargol quedin alineats i així el servo pateix menys.



El problema que hi ha és que la cinta de doble cara (representat amb groc) patina quan hi ha massa pes i el robot perd estabilitat, s'han de canviar els valors inicials dels servos afectats i les ordres per adaptar-les al canvi.

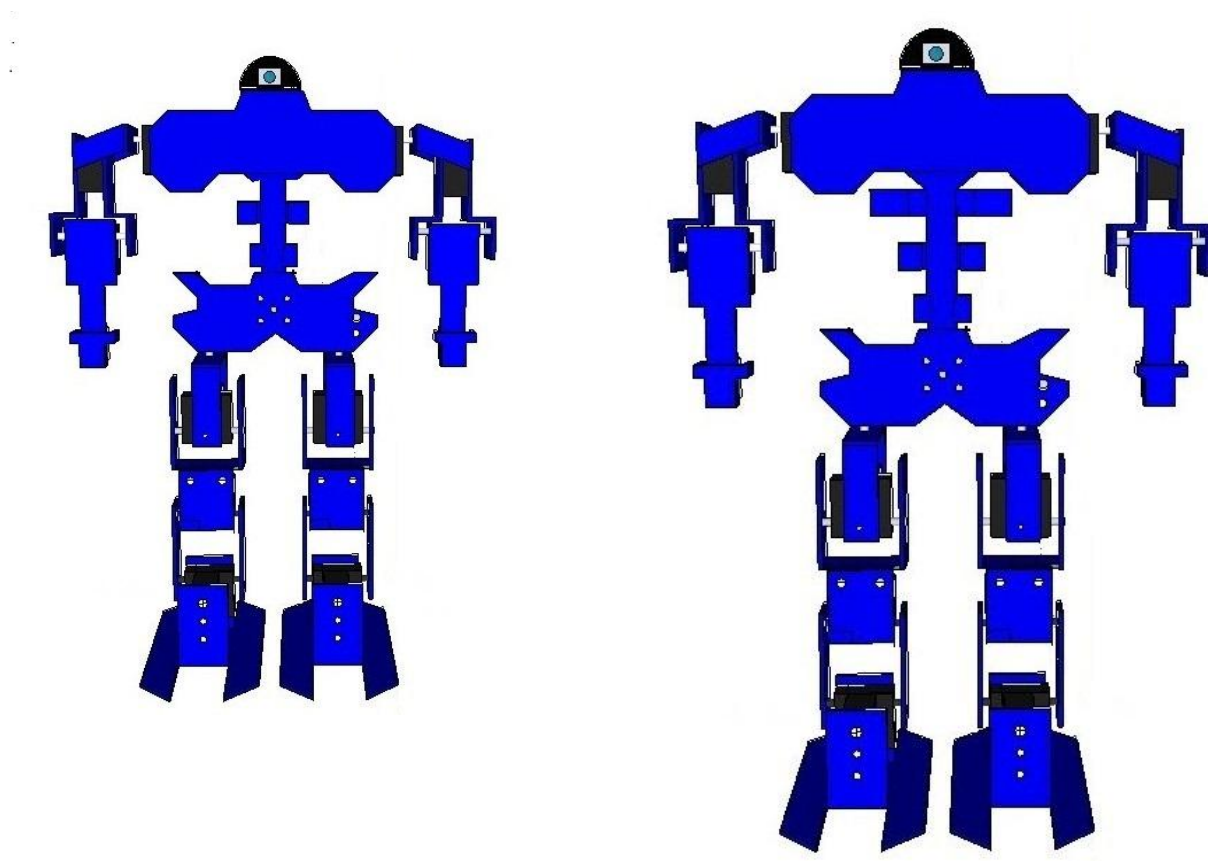
La manera de solucionar-ho podria ser connectar l'eix directament al servo, és a dir, fer un forat a la part posterior de la carcassa, fer-hi passar un cargol i collar-lo per l'altre costat amb una femella, i quedaria així:



El segon problema que té el robot és que al ser de metacrilat és molt trencadís, i quan hi havia una peça que s'havia de collar a algun lloc i anava una mica justa es trencava, a l'hora de foradar també es trencava per la vibració i fins i tot a l'hora de tallar les peces. Això es podria haver evitat amb un material diferent, com ara l'alumini, que també és resistent i lleuger.

També m'agradaria millorar la part electrònica, ja que els cables estan soldats a la placa, per a pròxims projectes els soldaré a pins femella i d'allà els connectaré als pins de la placa.

I per últim, el robot és massa alt, i fa balancejar massa l'estructura. Això es podria arreglar fent una columna més curta i substituint-la per l'actual, i així aconseguir més estabilitat al ésser el robot més baix amb la mateixa base i fent-ne unes peces més reduïdes.



Agraïments

Primer de tot m'agradaria donar les gracies a la meva mare (Marina Comas) per haver-me finançat el projecte i pel seu suport incondicional.

Agrair al meu pare l'haver-me proporcionat totes les eines que hagués de menester .

A les hores del pati ,tres cops per setmana , anàvem al taller a plegar les peces , polir els detalls i parlar amb el tutor per fer els nostres seguiments respectius , així que també dono les gracies a tots els companys amb els que compartia el taller perquè ens ajudàvem els uns als altres i ajudaven a fer mes entretinguda l'estona de treball .

També agrair a rètols Figueres la seva generositat , ja que em van donar uns 2 m² de metacrilat de 3 mm de gruix , translúcid i de color blau fosc , el qual és una mica car .

I per acabar, agrair al meu tutor tota la paciència que ha tingut amb mi a l'hora de revisar els programes i els errors que he comes en aquest projecte.

Bibliografia

- <http://www.alegsa.com.ar/Dic/adc.php> (ADC)
- <http://www.araelectronica.com/componentes-pasivos/condensadores.htm> (condensador)
- <http://www.msebilbao.com/notas/downloads/Hojas%20tecnicas%20del%20SD21.pdf> (placa)
- <http://www.tecnologiafacil.net/page.php?2> (PICAXE)
- <http://www.webelectronica.com.ar/news08/nota08/picaxe.htm> (PICAXE)
- http://www.robotronica.com.mx/catalog/documentacion/picaxe_manual_es.pdf(PICAXE)

ANNEXES

```

1 *****
2 *****DEFINIR VARIABLES*****
3 *****
4 #picaxe 18m2
5 symbol servoTE=63 'registre gir servo1 posició base
6 symbol servoTEp=84 'registre gir servo1 up.
7 symbol servoTEn=105 'registre gir servo1 down.
8 symbol baseTE=145 'posició inicial o intermitja servo1.
9 symbol offsetTE=b1 'valor que girarà el servo
10 symbol velocitatTE=0 'registre control de la velocitat del servo1.
11
12 symbol servoTD=64 'registre gir servo2 posició base
13 symbol servoTDp=85 'registre gir servo2 up.
14 symbol servoTDn=105 'registre gir servo2 down.
15 symbol baseTD=135 'posició inicial o intermitja servo1.
16 symbol offsetTD=b2 'valor que girarà el servo
17 symbol velocitatTD=3 'registre control de la velocitat del servo2.
18
19 symbol servoPE=65 'registre gir servo3 posició base
20 symbol servoPEp=86 'registre gir servo3up.
21 symbol servoPEN=150 'registre gir servo3 down.
22 symbol basePE=135 'posició inicial o intermitja servo3.
23 symbol offsetPE=b3 'valor que girarà el servo
24 symbol velocitatPE=6 'registre control de la velocitat del servo3.
25
26 symbol servoPD=66 'registre gir servo4 posició base
27 symbol servoPDp=87 'registre gir servo4 up.
28 symbol servoPDn=150 'registre gir servo4 down.
29 symbol basePD=121 'posició inicial o intermitja servo4
30 symbol offsetPD=b4 'valor que girarà el servo
31 symbol velocitatPD=9 'registre control de la velocitat del servo4.
32
33 symbol servoGE=67 'registre gir servo5 posició base
34 symbol servoGEp=88 'registre gir servo5 up.
35 symbol servoGEN=150 'registre gir servo5 down.
36 symbol baseGE=140 'posició inicial o intermitja servo5.
37 symbol offsetGE=b5 'valor que girarà el servo
38 symbol velocitatGE=12 'registre control de la velocitat del servo5.
39
40 symbol servoGD=68 'registre gir servo6 posició base
41 symbol servoGDp=89 'registre gir servo6 up.
42 symbol servoGDn=110 'registre gir servo6 down.
43 symbol baseGD=112 'posició inicial o intermitja servo6.
44 symbol offsetGD=b6 'valor que girarà el servo
45 symbol velocitatGD=15 'registre control de la velocitat del servo6.
46
47 symbol servoAE=69 'registre gir servo7 posició base
48 symbol servoAEp=90 'registre gir servo7 up.
49 symbol servoAEn=111 'registre gir servo7 down.
50 symbol baseAE=200 'posició inicial o intermitja servo7.
51 symbol offsetAE=b7 'valor que girarà el servo
52 symbol velocitatAE=18 'registre control de la velocitat del servo7.
53

```

```

54 symbol servoAD=70 'registre gir servo8 posició base
55 symbol servoADp=91 'registre gir servo8 up.
56 symbol servoADn=112 'registre gir servo8 down.
57 symbol baseAD=120 'posició inicial o intermitja servo8.
58 symbol offsetAD=b8 'valor que girarà el servo
59 symbol velocitatAD=21 'registre control de la velocitat del servo8.
60
61 symbol servoFE=71 'registre gir servo9 posició base
62 symbol servoFEp=92 'registre gir servo9 up.
63 symbol servoFEn=113 'registre gir servo9 down.
64 symbol baseFE=100 'posició inicial o intermitja servo9.
65 symbol offsetFE=b9 'valor que girarà el servo
66 symbol velocitatFE=24 'registre control de la velocitat del servo9.
67
68 symbol servoFD=72 'registre gir servo10 posició base
69 symbol servoFDp=93 'registre gir servo10 up.
70 symbol servoFDn=114 'registre gir servo10 down.
71 symbol baseFD=122 'posició inicial o intermitja servo10
72 symbol offsetFD=b10 'valor que girarà el servo
73 symbol velocitatFD=27 'registre control de la velocitat del servo10.
74
75 symbol servoCE=73 'registre gir servo11 posició base
76 symbol servoCEp=94 'registre gir servo11 up.
77 symbol servoCEn=135 'registre gir servo11 down.
78 symbol baseCE=130 'posició inicial o intermitja servo11.
79 symbol offsetCE=b11 'valor que girarà el servo
80 symbol velocitatCE=30 'registre control de la velocitat del servo11.
81
82 symbol servoCD=74 'registre gir servo12 posició base
83 symbol servoCDp=95 'registre gir servo12 up.
84 symbol servoCDn=116 'registre gir servo12 down.
85 symbol baseCD=110 'posició inicial o intermitja servo12.
86 symbol offsetCD=b12 'valor que girarà el servo
87 symbol velocitatCD=33 'registre control de la velocitat del servo12.
88
89 symbol servoHE=75 'registre gir servo13 posició base
90 symbol servoHEp=96 'registre gir servo13 up.
91 symbol servoHEN=117 'registre gir servo13 down.
92 symbol baseHE=25 'posició inicial o intermitja servo13.
93 symbol offsetHE=b13 'valor que girarà el servo
94 symbol velocitatHE=36 'registre control de la velocitat del servo13.
95
96 symbol servoHD=76 'registre gir servo14 posició base
97 symbol servoHDp=97 'registre gir servo14 up.
98 symbol servoHDn=118 'registre gir servo14 down.
99 symbol baseHD=250 'posició inicial o intermitja servo14.
100 symbol offsetHD=b14 'valor que girarà el servo
101 symbol velocitatHD=39 'registre control de la velocitat del servo14.
102
103 symbol servoBE=77 'registre gir servo15 posició base
104 symbol servoBEp=98 'registre gir servo15 up.
105 symbol servoBEn=119 'registre gir servo15 down.
106 symbol baseBE=40 'posició inicial o intermitja servo15.

```

```

107 symbol offsetBE=b15 'valor que girarà el servo
108 symbol velocitatBE=42 'registre control de la velocitat del servo15.
109
110 symbol servoBD=78 'registre gir servo16 posició base
111 symbol servoBDp=99 'registre gir servo16 up.
112 symbol servoBDn=122 'registre gir servo16 down.
113 symbol baseBD=205 'posició inicial o intermitja servo16.
114 symbol offsetBD=b16 'valor que girarà el servo
115 symbol velocitatBD=45 'registre control de la velocitat del servo16.
116
117 symbol servoZE=79 'registre gir servo17 posició base
118 symbol servoZEp=100 'registre gir servo17 up.
119 symbol servoZEN=125 'registre gir servo12 down.
120 symbol baseZE=130 'posició inicial o intermitja servo17.
121 symbol offsetZE=b17 'valor que girarà el servo
122 symbol velocitatZE=48 'registre control de la velocitat del servo17.
123
124 symbol servoZD=80 'registre gir servo18 posició base
125 symbol servoZDp=101 'registre gir servo18 up.
126 symbol servoZDn=128 'registre gir servo18 down.
127 symbol baseZD=120 'posició inicial o intermitja servo18.
128 symbol offsetZD=b18 'valor que girarà el servo
129 symbol velocitatZD=51 'registre control de la velocitat del servo18.
130
131 symbol servoME=81 'registre gir servo19 posició base
132 symbol servoMEp=102 'registre gir servo19 up.
133 symbol servoMEN=131 'registre gir servo19 down.
134 symbol baseME=240 'posició inicial o intermitja servo19.
135 symbol offsetME=b19 'valor que girarà el servo
136 symbol velocitatME=54 'registre control de la velocitat del servo19.
137
138 symbol servoMD=82 'registre gir servo20 posició base
139 symbol servoMDp=103 'registre gir servo20 up.
140 symbol servoMDn=134 'registre gir servo20 down.
141 symbol baseMD=150 'posició inicial o intermitja servo20.
142 symbol offsetMD=b20 'valor que girarà el servo
143 symbol velocitatMD=57 'registre control de la velocitat del servo20
144
145 symbol servoCO=83 'registre gir servo21 posició base
146 symbol servoCOp=104 'registre gir servo21 up.
147 symbol servoCON=137 'registre gir servo21 down.
148 symbol baseCO=141 'posició inicial o intermitja servo21.
149 symbol offsetCO=b21 'valor que girarà el servo
150 symbol velocitatCO=60 'registre control de la velocitat del servo21.
151
152 'Symbol IR = b0
153 *****
154 *****INICI*****
155 *****
156
157 i2cslave $c2,i2cslow,i2cbyte
158 setfreq m8
159 gosub posicioservos

```

```

160 gosub so_inici
161
162
163     w1=500 'temps que tarda en desactivar l'acció de cada tecla (500ms)
164     '      w2=2000 'idem b0 però per les tecles turbo 2 i 0 (1000ms)
165
166     '      serout b.0,T38400_32,("-Hola, sóc el robot-")
167     '      srtxd ("-Hola, sóc el robot-") 'comprovació de prova pel terminal del Prog. Editor
168     (F8).
169
170     *****
171     *****PROGRAMA PRINCIPAL*****
172     *****
173
174     inici:
175     start:
176
177
178
179     readi2c 65, (b12)
180     if b12 < 160 then goto bateria_baixa
181
182     pause 100
183     serin 1,T9600_8,b1
184
185     pause 150
186
187     '      irin[1000,start],c.0,b0 'recepció d'infraroig per l'entrada 0 i ho guarda al registre
188     '      infra (b0)
189     '      inc b0
190     '      'posa els servos en posició correcta
191     Comprovable:
192
193     if b1 = 97 then goto observa 'A
194     if b1 = 98 then gosub posicioservos 'C
195     if b1 = 99 then goto saluda 'B
196     if b1 = 100 then goto flexionacames 'D
197     if b1 = 101 then goto camina 'E
198     if b1 = 102 then goto balla 'F
199     if b1 = 103 then goto cintura 'G
200     if b1 = 104 then goto xuta 'H
201     if IR =  then goto
202     if IR =  then goto
203     if IR =  then goto
204     if IR = 10 then gosub posicioservos
205     pause 200
206     goto inici
207
208     posicioservos:
209
210     writei2c velocitatTE, (7) '(0= no s'utilitza, velocitat= max).((pos

```

```

ini-posfinal)/reg_velocitat)*20ms.
211 writei2c servoTE, (baseTE) 'es posiciona a posició intermitja.
212 pause 100 'espera 2s.
213 writei2c velocitatTD, (7)
214 writei2c servoTD, (baseTD)
215 pause 100
216 writei2c velocitatPE, (7)
217 writei2c servoPE, (basePE)
218 pause 100
219 writei2c velocitatPD, (7)
220 writei2c servoPD, (basePD)
221 pause 100
222 writei2c velocitatGE, (7)
223 writei2c servoGE, (baseGE)
224 pause 100
225 writei2c velocitatGD, (7)
226 writei2c servoGD, (baseGD)
227 pause 100
228 writei2c velocitatAE, (7)
229 writei2c servoAE, (baseAE)
230 pause 100
231 writei2c velocitatAD, (7)
232 writei2c servoAD, (baseAD)
233 pause 100
234 writei2c velocitatFE, (7)
235 writei2c servoFE, (baseFE)
236 pause 100
237 writei2c velocitatFD, (7)
238 writei2c servoFD, (baseFD)
239 pause 100
240 writei2c velocitatCE, (7)
241 writei2c servoCE, (baseCE)
242 pause 100
243 writei2c velocitatCD, (7)
244 writei2c servoCD, (baseCD)
245 pause 100
246 writei2c velocitatHE, (10)
247 writei2c servoHE, (baseHE)
248 pause 100
249 writei2c velocitatHD, (7)
250 writei2c servoHD, (baseHD)
251 let offsetHD=10
252 writei2c velocitatHD, (7)
253 writei2c servoHDp, (offsetHD)
254 pause 100
255 writei2c velocitatBE, (7)
256 writei2c servoBE, (baseBE)
257 pause 100
258 writei2c velocitatBD, (7)
259 writei2c servoBD, (baseBD)
260 pause 100
261 writei2c velocitatZE, (7)
262 writei2c servoZE, (baseZE)

```

```

263 pause 100
264 writei2c velocitatZD, (7)
265 writei2c servoZD , (baseZD)
266 pause 100
267 writei2c velocitatME, (7)
268 writei2c servoME , (baseME)
269 pause 100
270 writei2c velocitatMD, (7)
271 writei2c servoMD , (baseMD)
272 pause 100
273 writei2c velocitatCO, (7)
274 writei2c servoCO , (baseCO)
275 pause 100
276 return
277
278
279 'agafaobjecte:
280 'let offset16 =60
281 'writei2c velocitat16, (10)
282 'writei2c Servo16n , (offset16)
283 'wait 5
284 'writei2c velocitat16, (10)
285 'writei2c servo16p , (offset16)
286 'goto start
287
288 saluda:*****PKDGPS***
289 writei2c velocitatBE , (10)
290 writei2c servoBE, (170)
291 pause 750
292 writei2c velocitatHE , (10)
293 writei2c servoHE, (255)
294 pause 2750
295 writei2c velocitatBE , (10)
296 writei2c servoBE, (baseBE)
297 pause 1750
298 writei2c velocitatZE , (25)
299 writei2c servoZE, (90)
300 pause 950
301 writei2c velocitatZE , (25)
302 writei2c servoZE, (170)
303 pause 950
304 writei2c velocitatZE , (25)
305 writei2c servoZE, (90)
306 pause 950
307 writei2c velocitatZE , (25)
308 writei2c servoZE, (170)
309 pause 950
310 writei2c velocitatZE , (25)
311 writei2c servoZE, (90)
312 pause 950
313 writei2c velocitatZE , (25)
314 writei2c servoZE, (baseZE)
315 pause 550

```



```

316 writei2c velocitatBE , (10)
317 writei2c servoBE, (170)
318 pause 1750
319 gosub posicioservos
320 goto start
321
322 'saluda1:
323 let offsetHD= 250
324 writei2c velocitatHD,(10)
325 writei2c servoHDn ,(offsetHD)
326 pause 2500
327 gosub saluda2
328 gosub saluda3
329 gosub saluda2
330 gosub saluda3
331 gosub saluda2
332 gosub saluda3
333 gosub saluda2
334 gosub posicioservos
335
336 goto start
337
338 bateria_baixa:
339 sound 0,(123,100)
340 pause 500
341 sound 0,(123,100)
342 goto start
343
344 so_inici:
345 sound 0,(123,200)
346 pause 500
347 return
348
349 saluda2:
350 writei2c velocitatBD , (125)
351 writei2c servoBD, (245)
352 pause 700
353 return
354
355 saluda3:
356 writei2c velocitatBD , (105)
357 writei2c servoBD ,(220)
358 pause 700
359 return
360
361 flexionacames:
362 writei2c velocitatBD,(7)
363 writei2c servoBD , (80)
364 pause 100
365 writei2c velocitatBE , (7)
366 writei2c servoBE , (170)
367 pause 2500
368 writei2c velocitatHD,(7)

```



```
369 writei2C servoHD , (110)
370 pause 100
371 writei2C velocitatHE , (7)
372 writei2C servoHE , (197)
373 pause 2500
374 writei2C velocitatBD ,(7)
375 writei2C servoBD , (100)
376 pause 100
377 writei2C velocitatBE ,(7)
378 writei2C servoBE , (150)
379 wait 1
380 writei2C velocitatFE , (6)
381 writei2C servoFE, (177)
382 pause 100
383 writei2C velocitatFD , (6)
384 writei2C servoFD, (62)
385 pause 100
386 writei2C velocitatGD,(10)
387 writei2C servoGD, (235)
388 pause 100
389 writei2C velocitatGE,(10)
390 writei2C servoGE,(28)
391 pause 100
392 writei2C velocitatTE,(6)
393 writei2C servoTE,(100)
394 pause 100
395 writei2C velocitatTD,(6)
396 writei2C servoTD,(193)
397 pause 100
398 writei2C velocitatBD,(7)
399 writei2C servoBD,(baseBD)
400 pause 100
401 writei2C velocitatBE,(7)
402 writei2C servoBE,(baseBE)
403 pause 3500
404 writei2C velocitatGE,(20)
405 writei2C servoGE ,(baseGE)
406 pause 100
407 writei2C velocitatGD,(20)
408 writei2C servoGD ,(baseGD)
409 pause 100
410 writei2C velocitatTD,(50)
411 writei2C servoTD ,(baseTD)
412 pause 100
413 writei2C velocitatTE,(50)
414 writei2C servoTE ,(baseTE)
415 pause 100
416 writei2C velocitatFE,(65)
417 writei2C servoFE ,(baseFE)
418 pause 100
419 writei2C velocitatFD,(65)
420 writei2C servoFD ,(baseFD)
421 pause 100
```

```

422 writei2c velocitatBE,(7)
423 writei2c servoBE ,(150)
424 pause 100
425 writei2c velocitatBD,(7)
426 writei2c servoBD ,(100)
427 pause 100
428 writei2c velocitatFE ,(80)
429 writei2c servoFE, (baseFE)
430 pause 100
431 writei2c velocitatFD ,(80)
432 writei2c servoFD, (baseFD)
433 pause 100
434 writei2c velocitatTE ,(70)
435 writei2c servoTE, (125)
436 pause 100
437 gosub posicionservos
438 goto start
439
440 observa:
441 writei2c velocitatCO,(20)
442 writei2c servoCO,(51)
443 pause 3500
444 writei2c velocitatCO,(20)
445 writei2c servoCO,(191)
446 pause 3500
447 writei2c velocitatCO,(40)
448 writei2c servoCO,(121)
449 pause 100
450 goto start
451
452
453 PAS:
454 writei2c velocitatAD,(10)
455 writei2c servoAD,(125)
456 pause 100
457 writei2c velocitatAE,(10)
458 writei2c servoAE,(220)
459 pause 100
460 writei2c velocitatPD,(10)
461 writei2c servoPD,(143)
462 pause 100
463 writei2c velocitatPE,(10)
464 writei2c servoPE,(185)
465 pause 100
466 writei2c velocitatGE ,(10)
467 writei2c servoGE , (100)
468 pause 100
469 writei2c velocitatHE ,(30)
470 writei2c servoHE , (115)
471 pause 100
472 writei2c velocitatFD ,(10) 'per no caure
473 writei2c servoFD , (112)
474 pause 100

```

```

475 writei2c velocitatFE, (10)
476 writei2c servoFE, (140)
477 pause 550
478 writei2c velocitatCD, (20)
479 writei2c servoCD, (120)
480 pause 300
481 writei2c velocitatCE, (20)
482 writei2c servoCE, (140)
483 pause 100
484 writei2c velocitatPE, (20)
485 writei2c servoPE, (basePE)
486 pause 100
487 writei2c velocitatTE, (10)
488 writei2c servoTE, (185)
489 pause 100
490 writei2c velocitatAE, (10)
491 writei2c servoAE, (baseAE)
492 pause 100
493 writei2c velocitatGE, (20)
494 writei2c servoGE, (baseGE)
495 pause 100
496 writei2c velocitatFD, (20)
497 writei2c servoFD, (142)
498 pause 100
499 writei2c velocitatTD, (20)
500 writei2c servoTD, (145)
501 pause 100
502 writei2c velocitatPD, (20)
503 writei2c servoPD, (68)
504 pause 100
505 writei2c velocitatTE, (20)
506 writei2c servoTE, (baseTE)
507 pause 100
508 writei2c velocitatAD, (20)
509 writei2c servoAD, (140)
510 pause 100
511 writei2c velocitatAE, (20)
512 writei2c servoAE, (210)
513 pause 100
514 writei2c velocitatPE, (20)
515 writei2c servoPE, (120)
516 pause 100
517 writei2c velocitatCD, (10)
518 writei2c servoCD, (95)
519 pause 100
520 writei2c velocitatCE, (10)
521 writei2c servoCE, (115)
522 pause 100
523 writei2c velocitatFE, (10)
524 writei2c servoFE, (baseFE)
525 pause 100
526 writei2c velocitatGD, (10)
527 writei2c servoGD, (152)

```

```

528 pause 100
529 writei2c velocitatFD ,(10)
530 writei2c servoFD ,(82)
531 pause 100
532 writei2c velocitatPD ,(10)
533 writei2c servoPD ,(58)
534 pause 100
535 writei2c velocitatTE, (7)  '(0= no s'utilitza, velocitat= max).((pos
ini-posfinal)/reg_velocitat)*20ms.
536 writei2c servoTE, (baseTE)  'es posiciona a posició intermitja.
537 pause 100  'espera 2s.
538 writei2c velocitatTD, (7)
539 writei2c servoTD, (baseTD)
540 pause 100
541 writei2c velocitatPE, (7)
542 writei2c servoPE, (basePE)
543 pause 100
544 writei2c velocitatPD, (7)
545 writei2c servoPD, (basePD)
546 pause 100
547 writei2c velocitatGE, (7)
548 writei2c servoGE, (baseGE)
549 pause 100
550 writei2c velocitatGD, (7)
551 writei2c servoGD, (baseGD)
552 pause 100
553 writei2c velocitatAE, (7)
554 writei2c servoAE, (baseAE)
555 pause 100
556 writei2c velocitatAD, (7)
557 writei2c servoAD, (baseAD)
558 pause 100
559 writei2c velocitatFE, (7)
560 writei2c servoFE, (baseFE)
561 pause 100
562 writei2c velocitatFD, (7)
563 writei2c servoFD, (baseFD)
564 pause 100
565 writei2c velocitatCE, (7)
566 writei2c servoCE, (baseCE)
567 pause 100
568 writei2c velocitatCD, (7)
569 writei2c servoCD, (baseCD)
570 pause 100
571 return
572
573
574 camina:
575 gosub PAS
576 pause 100
577 serin 1,T9600_8,b1
578 if b1= 102 then goto camina
579 pause 100

```

```

580 gosub posicioservos
581 goto start
582
583
584
585 balla:
586 gosub pas1
587 pause 1000
588 gosub pas2
589 pause 1500
590 gosub pas1
591 pause 1000
592 gosub pas3
593 pause 1500
594 gosub pas1
595 pause 1000
596 writei2c velocitatGE,(50)
597 writei2c servoGE,(135)
598 pause 100
599 writei2c velocitatGD,(50)
600 writei2c servoGD,(125)
601 pause 100
602 writei2c velocitatHE,(50)
603 writei2c servoHE,(baseHE)
604 pause 100
605 writei2c velocitatHD,(50)
606 writei2c servoHD,(baseHD)
607 pause 100
608 writei2c velocitatGE,(50)
609 writei2c servoGE,(baseGE)
610 pause 100
611 writei2c velocitatGD,(50)
612 writei2c servoGD,(baseGD)
613 pause 100
614 goto start
615
616
617 pas1:
618 writei2c velocitatHE,(50)
619 writei2c servoHE,(197)
620 pause 100
621 writei2c velocitatHD,(50)
622 writei2c servoHD,(110)
623 pause 100
624 writei2c velocitatGE,(50)
625 writei2c servoGE,(135)
626 pause 100
627 writei2c velocitatGD,(50)
628 writei2c servoGD,(125)
629 pause 100
630 writei2c velocitatZE , (50)
631 writei2c servoZE , (baseZE)
632 pause 100

```

```

633 writei2c velocitatZD , (50)
634 writei2c servoZD , (baseZD)
635 return
636
637 pas2:
638 writei2c velocitatZE , (50)
639 writei2c servoZE,(30)
640 pause 100
641 writei2c velocitatZD, (50)
642 writei2c servoZD , (30)
643 pause 100
644 writei2c velocitatHD,(50)
645 writei2c servoHD,(baseHD)
646 pause 100
647 writei2c velocitatHE,(50)
648 writei2c servoHE,(255)
649 writei2c velocitatHE,(50)
650 writei2c servoHEp,(60)
651 pause 100
652 writei2c velocitatGD,(50)
653 writei2c servoGD,(baseGD)
654 pause 100
655 writei2c velocitatGE,(50)
656 writei2c servoGE,(baseGE)
657 return
658
659 pas3:
660 writei2c velocitatHD ,(50)
661 writei2c servoHD , (10)
662 pause 100
663 writei2c velocitatHE ,(50)
664 writei2c servoHE , (baseHE)
665 pause 100
666 writei2c velocitatGE,(50)
667 writei2c servoGE,(baseGE)
668 pause 100
669 writei2c velocitatGD,(50)
670 writei2c servoGD,(baseGD)
671 pause 100
672 writei2c velocitatZD ,(50)
673 writei2c servoZD,(250)
674 pause 100
675 writei2c velocitatZE ,(50)
676 writei2c servoZE , (250)
677 pause 100
678 return
679
680 primerpeu:'avança peu dret
681 gosub desequilibriesquerre
682 return
683
684 segonpeu:'avança peu esquerre
685 return

```

```
686
687
688 desequilibriesquerre:
689 writei2c velocitatAD,(20)
690 writei2c servoADp,(30)
691 pause 100
692 writei2c velocitatAE,(20)
693 writei2c servoAEp,(30)
694 pause 100
695 writei2c velocitatPD,(20)
696 writei2c servoPD,(98)
697 pause 100
698 writei2c velocitatPE,(20)
699 writei2c servoPE,(105)
700 pause 100
701 writei2c velocitatGD,(50)
702 writei2c servoGD,(baseGD)
703 return
704
705 desequilibridreta:
706 writei2c velocitatAD,(20)
707 writei2c servoAD,(105)
708 pause 100
709 writei2c velocitatAE,(20)
710 writei2c servoAE,(185)
711 pause 100
712 writei2c velocitatPD,(20)
713 writei2c servoPD,(158)
714 pause 100
715 writei2c velocitatPE,(20)
716 writei2c servoPE,(155)
717 pause 100
718
719 return
720
721 cintura:
722 gosub desequilibriesquerre
723 pause 100
724 writei2c velocitatBD,(50)
725 writei2c servoBD,(80)
726 pause 500
727 gosub posicionservos
728 pause 1500
729 gosub desequilibridreta
730 writei2c velocitatBE,(50)
731 writei2c servoBE,(170)
732 pause 500
733 gosub posicionservos
734 pause 100
735 goto start
736
737 caminados:
738 gosub desequilibriesquerredos
```

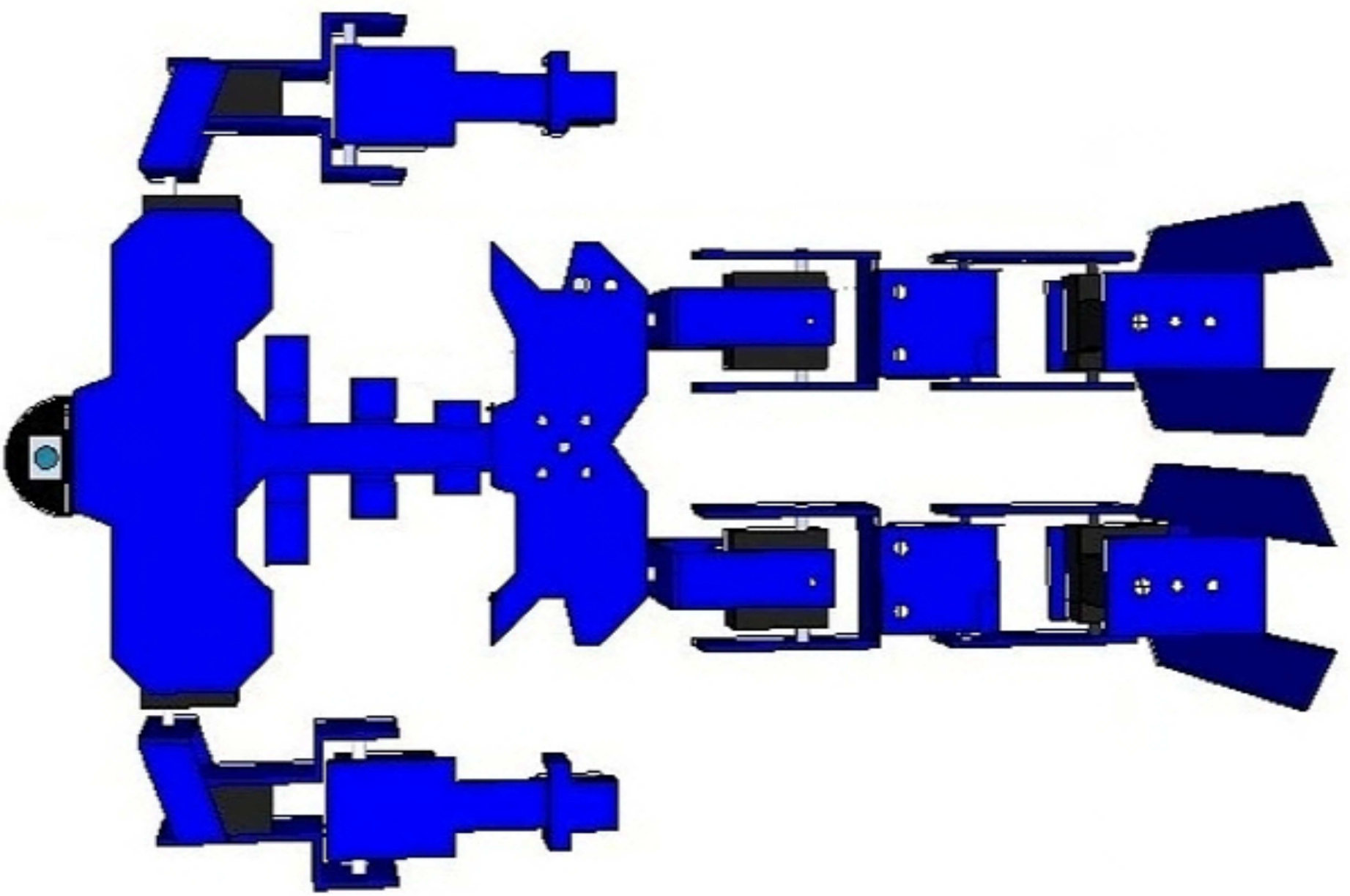


```

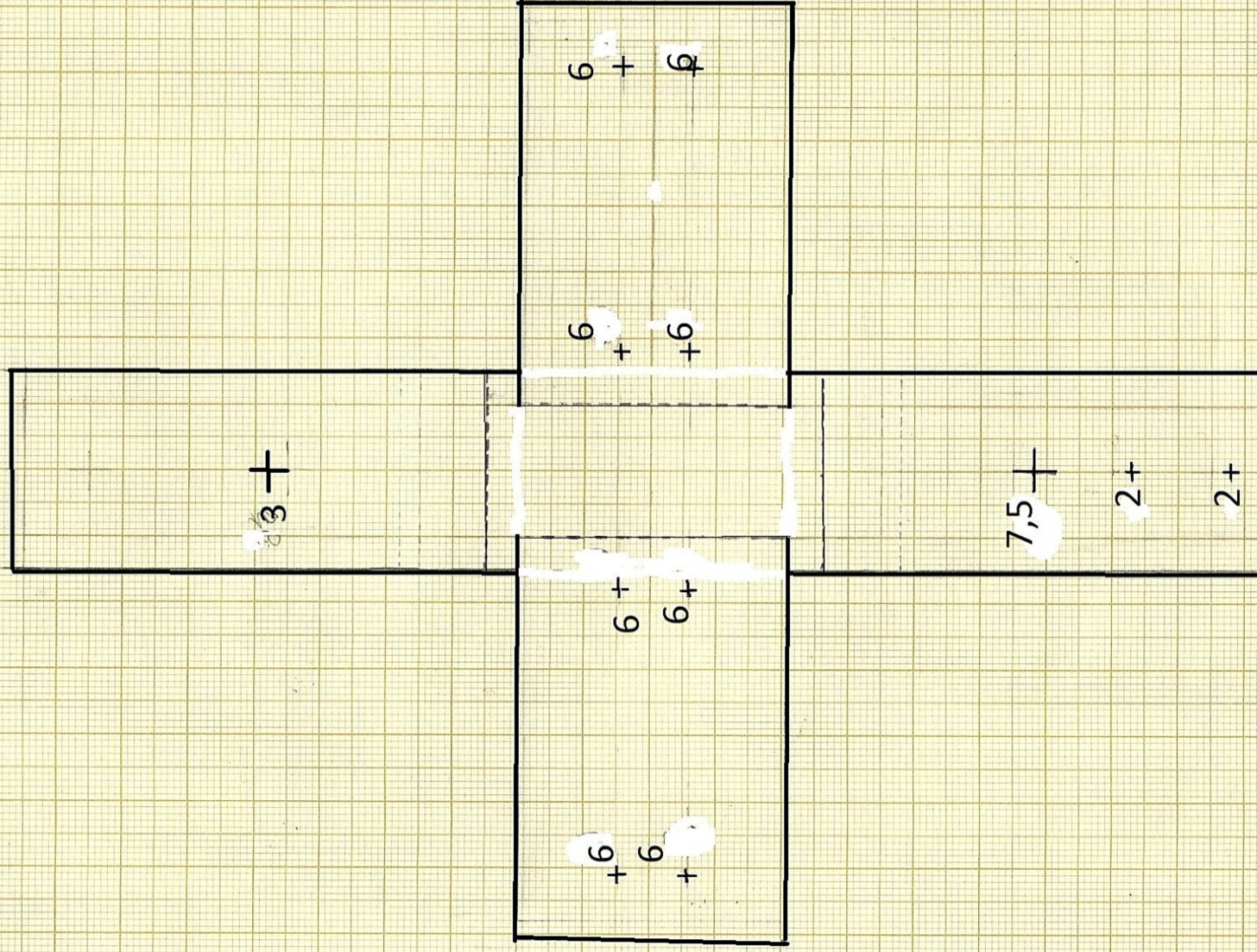
739 gosub flexionacamadreta
740 writei2c velocitatCE,(6)
741 writei2c servoCE,(100)
742 pause 100
743 writei2c velocitatCD,(6)
744 writei2c servoCDn,(30)
745 pause 100
746 'writei2c velocitatPD,(6)
747 'writei2c servoPD,(158)
748 pause 100
749
750 goto start
751
752
753 desequilibriesquerredos:
754 writei2c velocitatAD,(20)
755 writei2c servoADp,(60)
756 pause 100
757 writei2c velocitatAE,(20)
758 writei2c servoAEp,(50)
759 pause 100
760 writei2c velocitatPD,(20)
761 writei2c servoPD,(68)
762 pause 100
763 writei2c velocitatPE,(20)
764 writei2c servoPE,(85)
765 pause 100
766 writei2c velocitatGD,(50)
767 writei2c servoGD,(baseGD)
768 pause 100
769 writei2c velocitatBD,(50)
770 writei2c servoBD,(80)
771 pause 500
772 return
773
774 flexionacamadreta:
775 writei2c velocitatFD , (6)
776 writei2c servoFD, (52)
777 pause 100
778 writei2c velocitatGD,(10)
779 writei2c servoGD, (235)
780 pause 100
781 writei2c velocitatTD,(6)
782 writei2c servoTD,(193)
783 pause 100
784 writei2c velocitatPD,(6)
785 writei2c servoPD,(58)
786 pause 100
787 'writei2c velocitatBE,(6)
788 'writei2c servoBE,(170)
789 pause 100
790 return
791

```

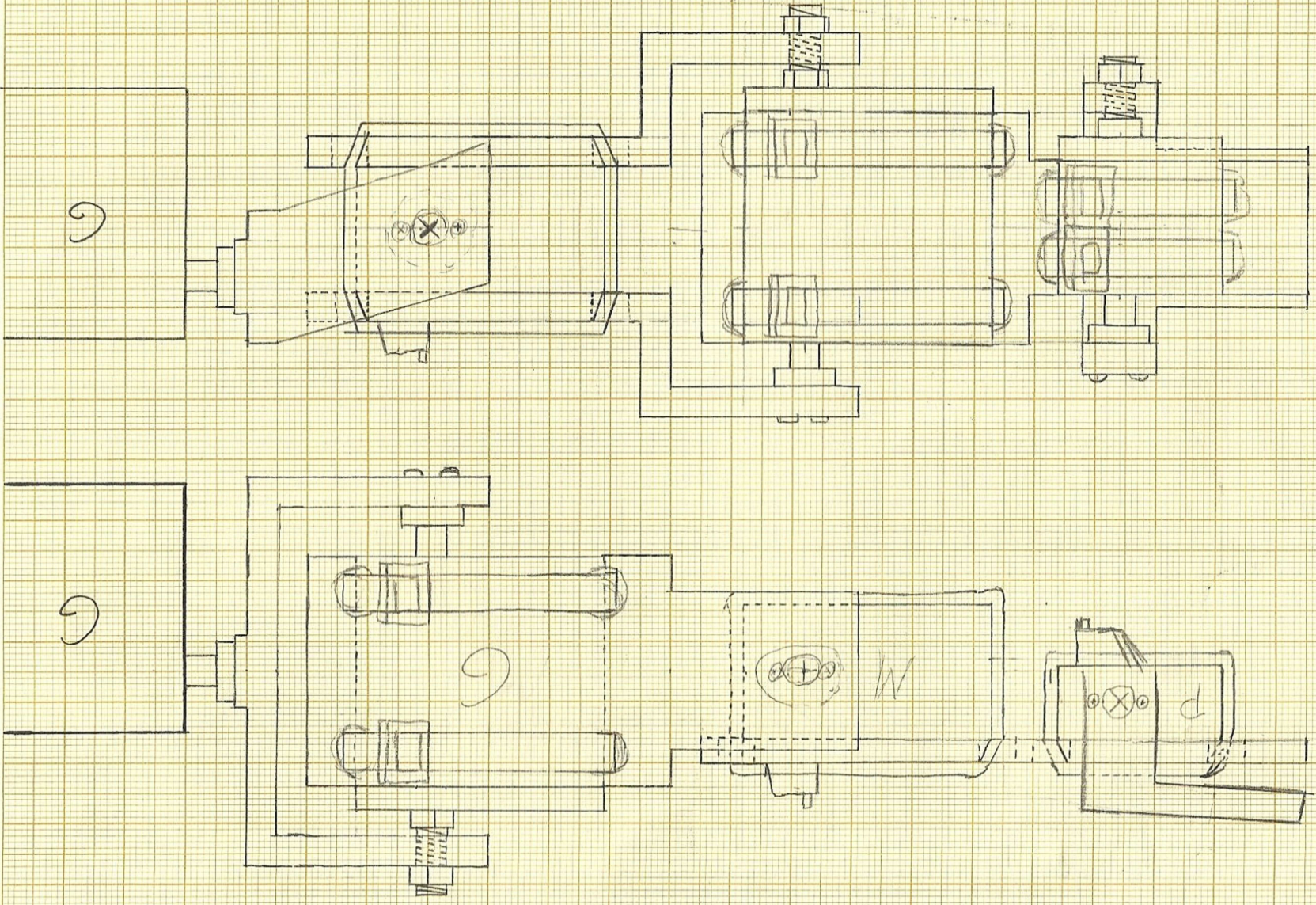
```
792 xuta:
793 gosub desequilibriesquerredos
794 pause 100
795 gosub flexionacamadreta
796 pause 100
797 writei2c velocitatFD , (20)
798 writei2c servoFD, (30)
799 pause 100
800 writei2c velocitatAD,(10)
801 writei2c servoCEp,(10)
802 pause 100
803 gosub posicionservos
804 goto start
```

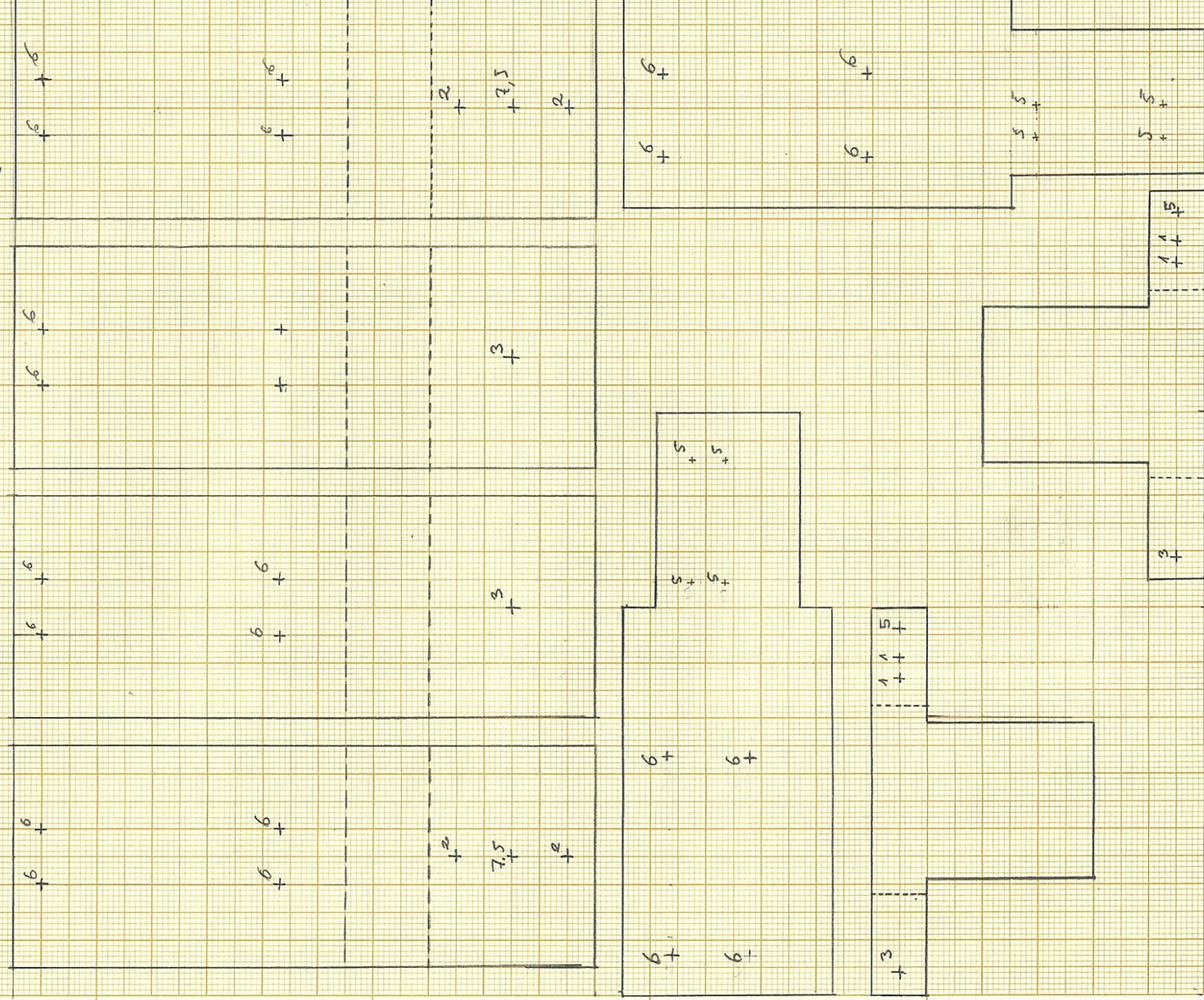
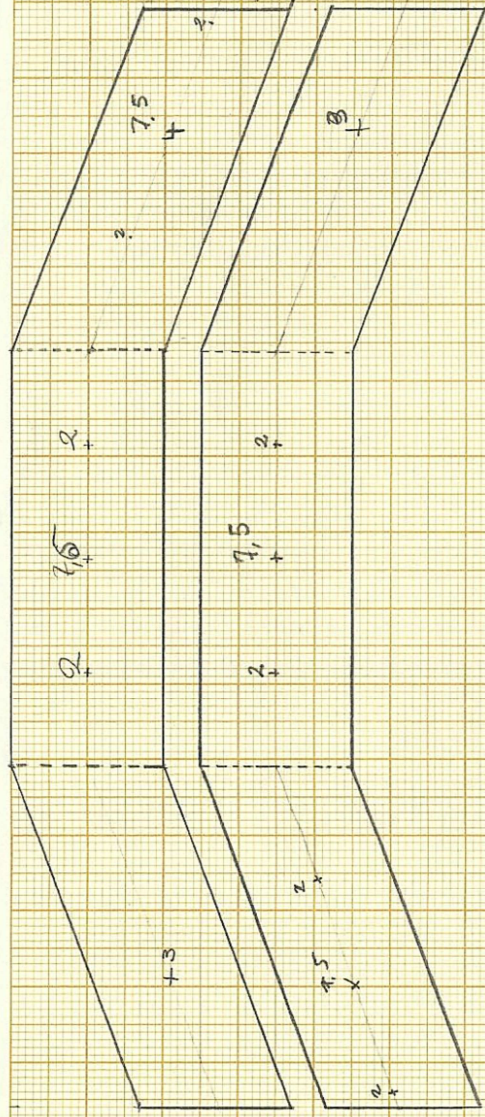


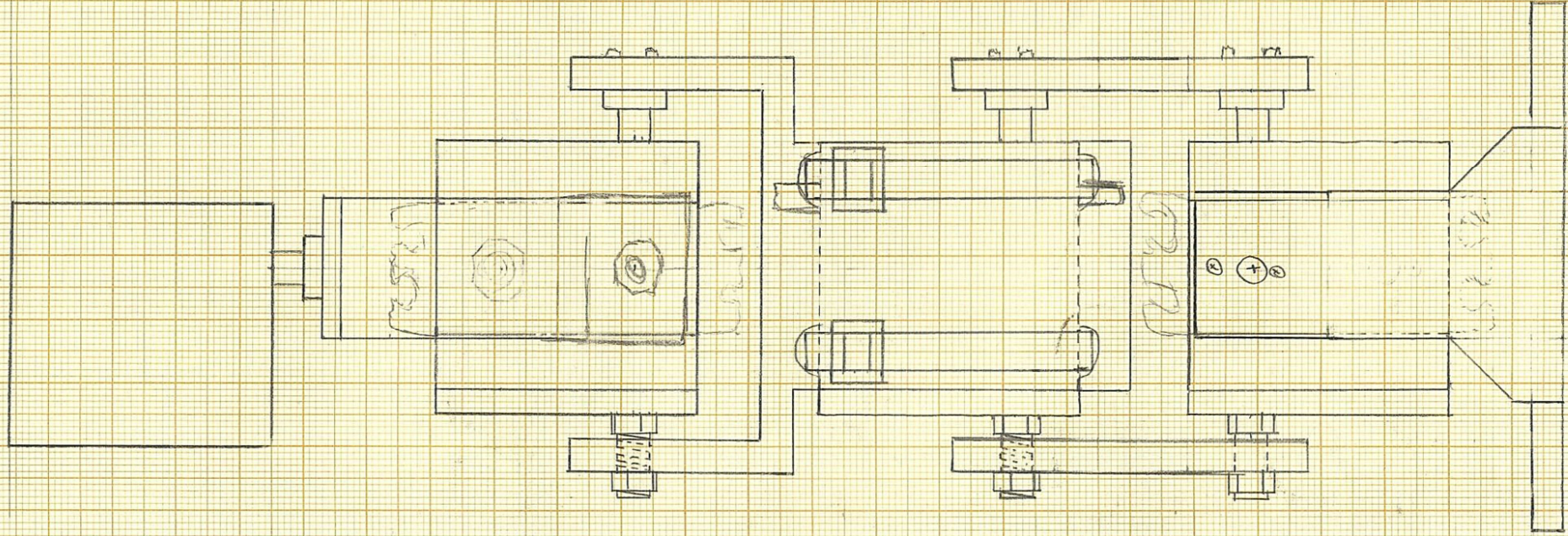




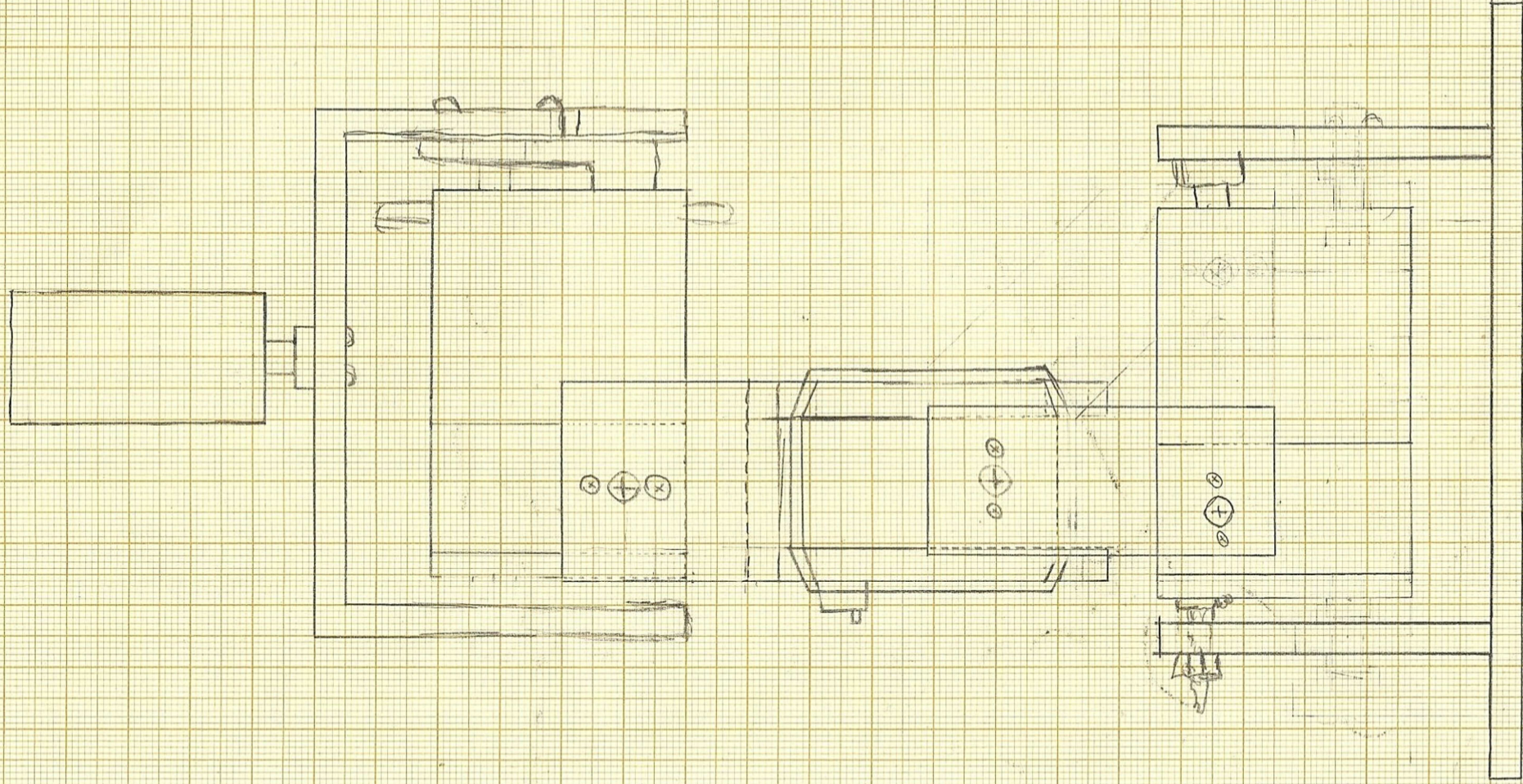
22

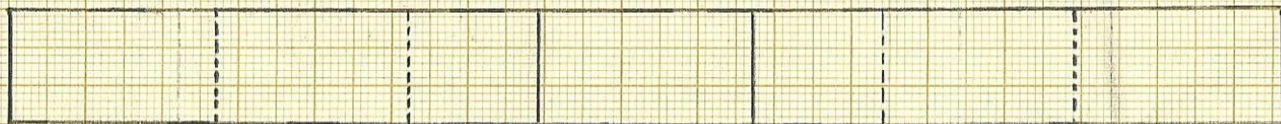
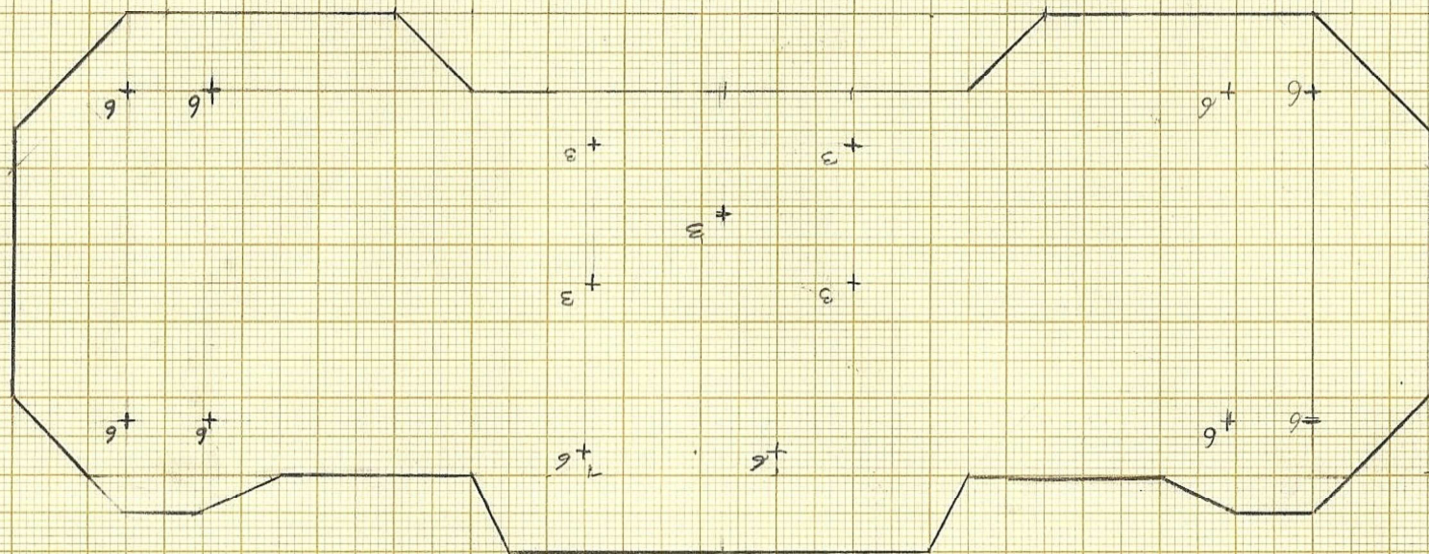
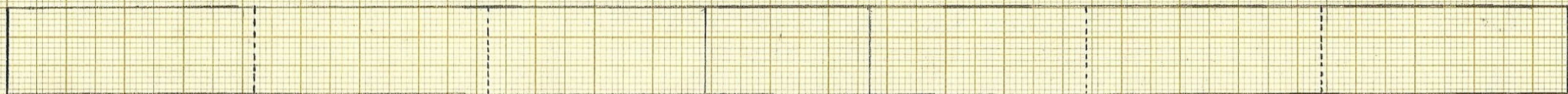




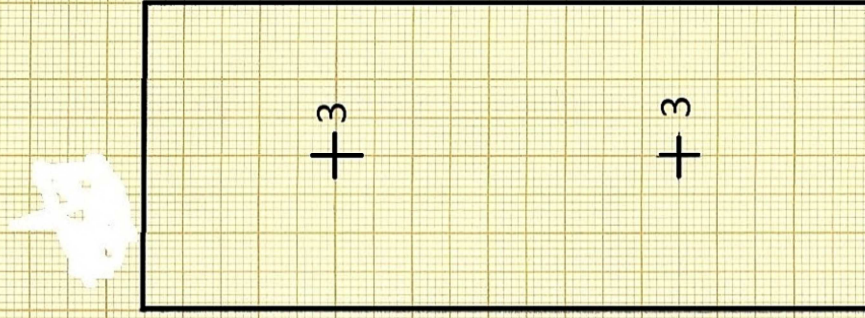
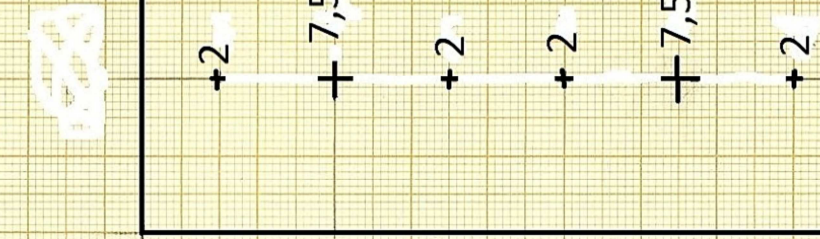
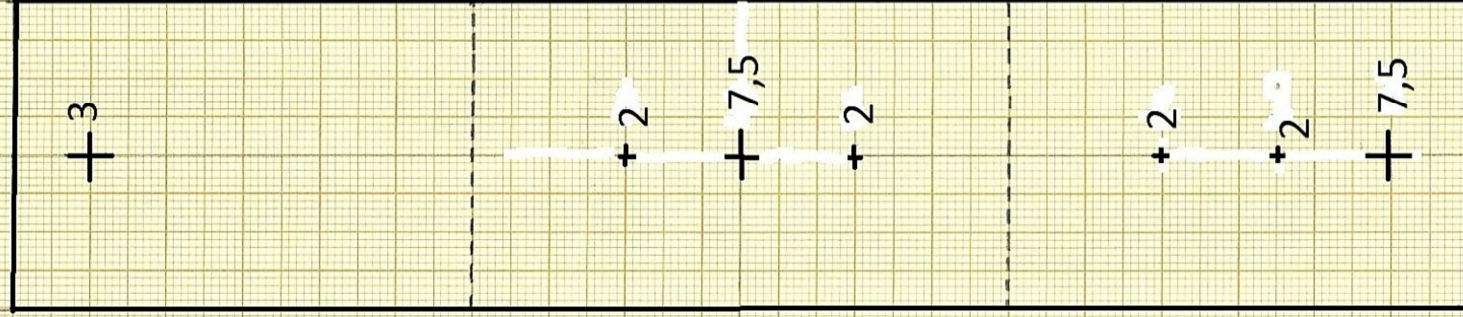


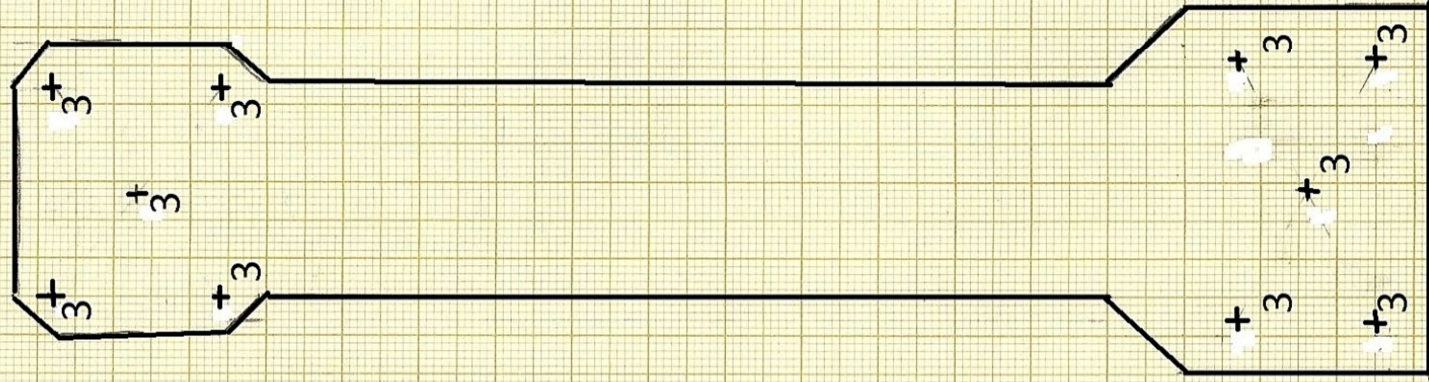
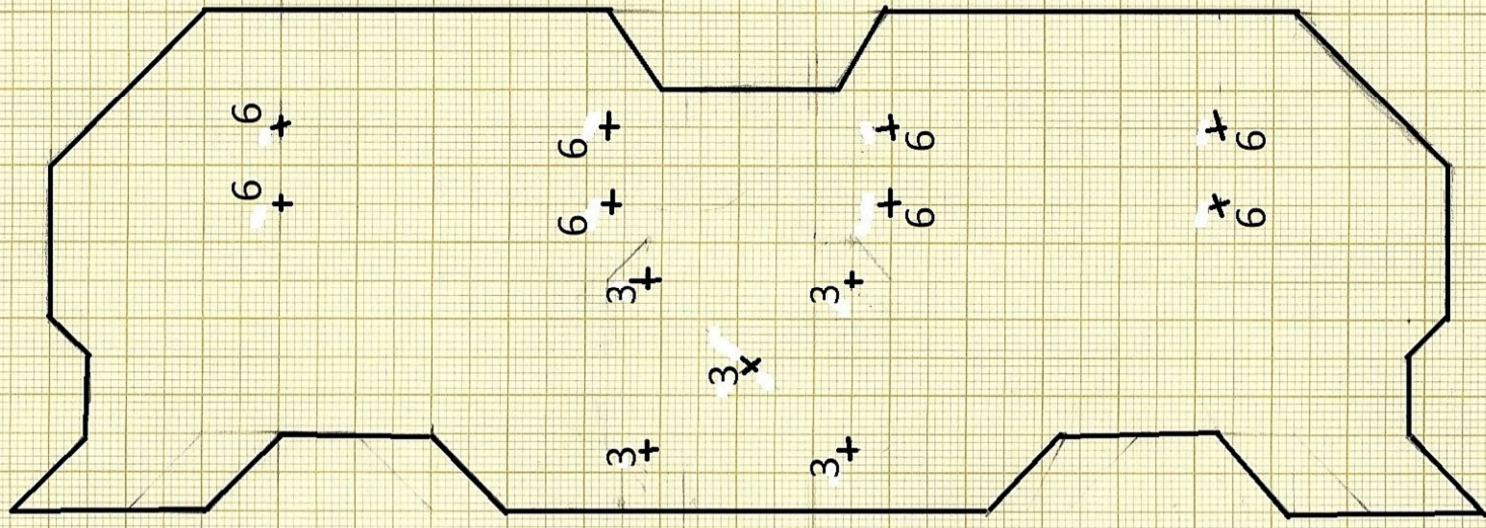
○



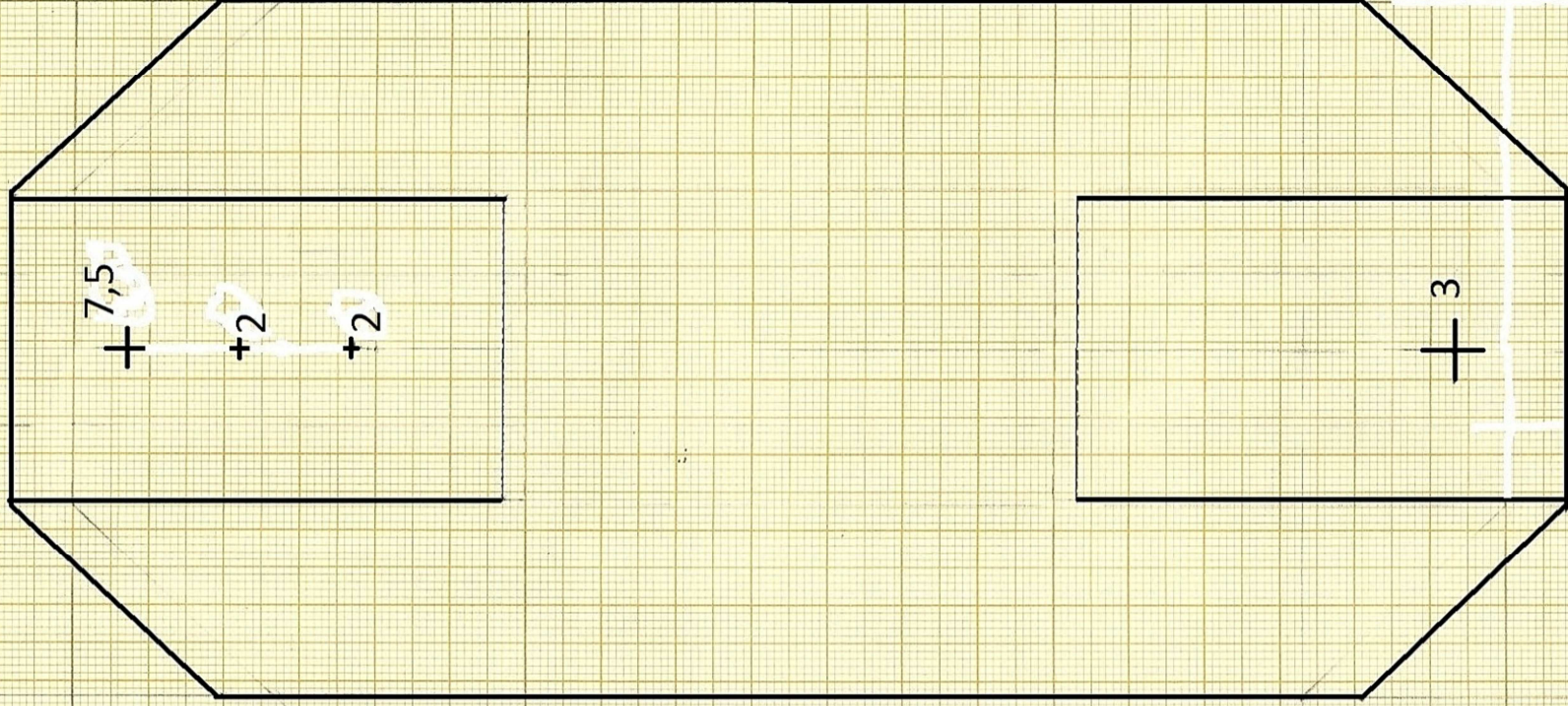


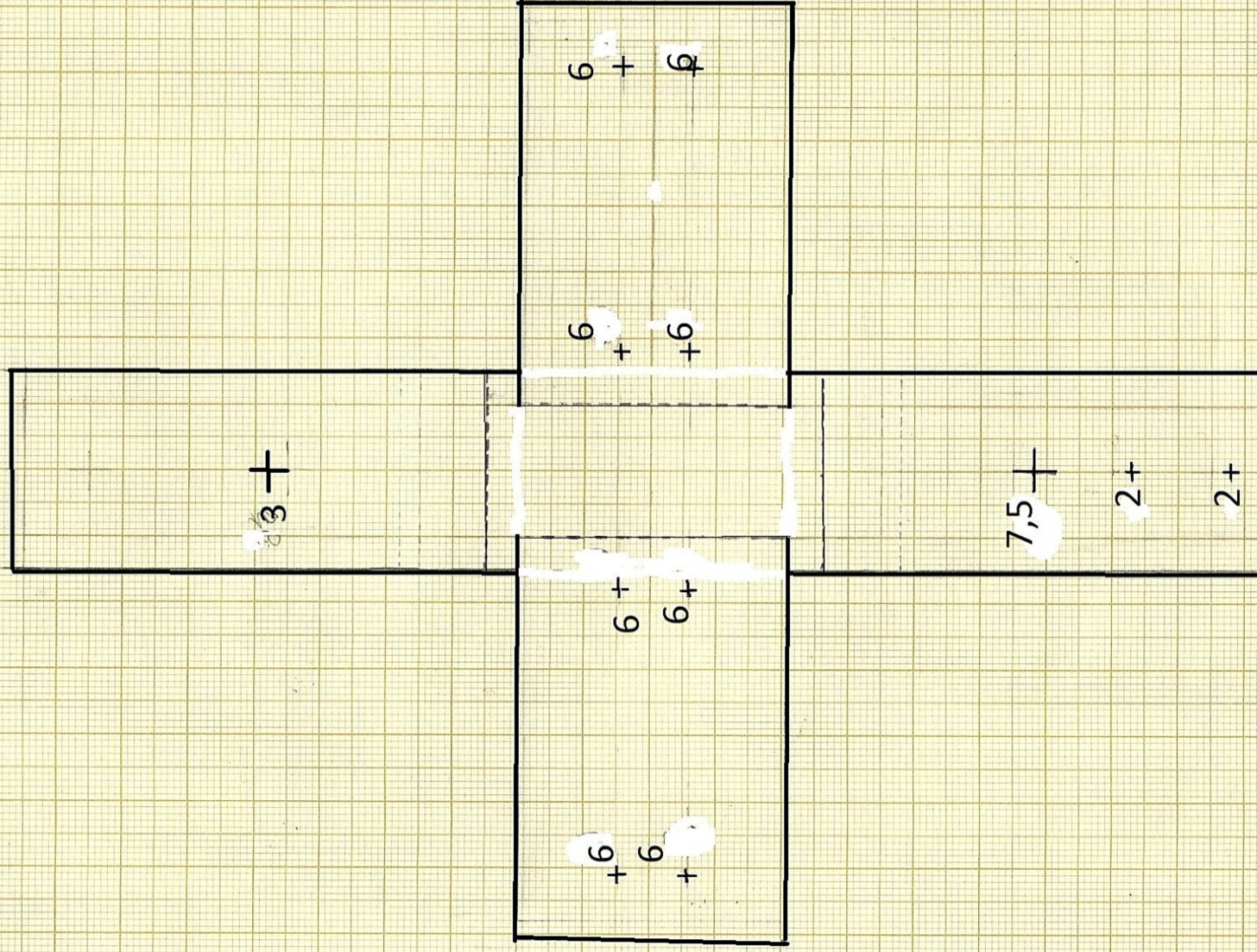
x2





x2





22