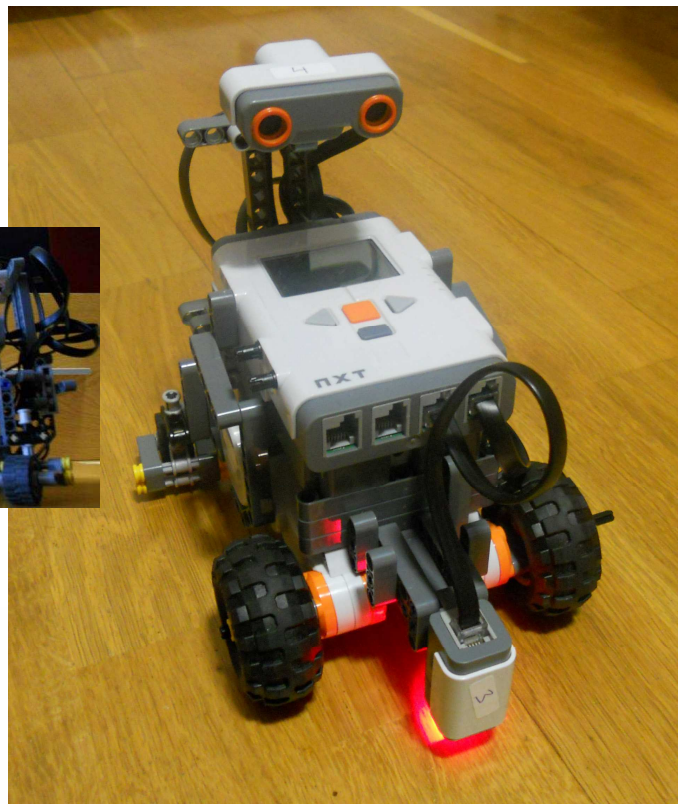
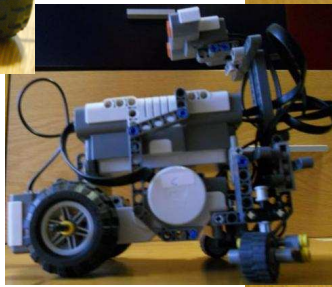


La Robòtica: Construcció d' un seguidor de línies amb LEGO® Mindstorms NXT



IES Alba del Vallès
2n Batxillerat 'B'
17-01-11

Índex

	Pàg.
1. Introducció.....	4
2. Objectius	5
3. Marc teòric.....	6
3.1 La robòtica	6
3.1.1 Definició de robot	6
3.1.2 Inicis	6
3.1.3 Tipus de Robots	8
3.2 Sistemes de control	15
3.2.1 Tipus de sistemes de control.....	17
3.3 Components del robot.....	28
3.3.1 Sensors	28
3.3.2 El controlador	31
3.3.3 Actuadors	32
3.4 Packs de robots educatius.....	33
3.5 Competicions robòtiques	36
4 Treball pràctic.....	38
4.1 Estructura del nostre robot.....	41
4.1.1 Els sensors.....	42
4.1.2 El controlador	43
4.1.3 Els servomotors.....	46
4.2 Pràctica 1: Longituds	47
4.3 Pràctica 2: Velocitats	48
4.4 Software Lego Mindstorms	51
4.4.1. Blocs comuns	55
4.4.2 Blocs d'acció	56
4.4.3. Blocs de sensors	58
4.4.4 Blocs de flux.....	60

4.4.5 Blocs de dades.....	62
4.4.6 Blocs avançats.	63
4.5 Programa final amb control PID.....	64
5.Conclusió.....	71
6. Bibliografia.....	73
7. Annex.....	77

1. Introducció

Per començar amb el meu treball de recerca explicaré perquè he realitzat aquest treball i les raons de la meva tria, així com l'estructura general que té i finalment els principals objectius que m'he plantejat.

El tema escollit per al treball de recerca ha estat la robòtica ja que hi intervenen diverses branques de la tecnologia que he estudiat durant l'ESO i que estudiaré durant el curs actual a Tecnologia Industrial. Una altra de les motivacions de caire més personal, a part de les vinculacions del tema amb la matèria és la meva afició a fer petits projectes tecnològics, i a que sempre m'ha cridat molt l'atenció la robòtica i com poder construir un robot perquè realitzi una funció preestablerta de forma autònoma.

Per tant un cop em van proposar de fer el treball de recerca sobre la robòtica i en concret sobre un robot seguidor de línies de Lego® que havien demanat per l'escola, vaig decidir acceptar ràpidament.

El treball consta principalment de dues parts, una de teòrica i una de pràctica. A la part teòrica explicaré els fonaments teòrics del treball, això inclou una definició de robòtica i de robot, un petit esment a la història de la robòtica i l'explicació de diferents tipus de robots. A més inclouré altres aspectes com l'explicació dels diferents tipus de sistemes de control, els elements que conformen un robot i per acabar dels diferents packs educatius que hi ha al mercat.

En la segona part del treball, la part pràctica, em centraré en parlar del robot que jo he construït, explicaré la feina feta durant tot el procés d'aprenentatge sobre com programar el robot, parlaré, també, dels elements que tindrà el robot i que m'haurà servit, o no, per aconseguir els meus objectius. Seguidament s'inclouran dues pràctiques fetes amb el robot i, per acabar, explicaré com funciona tot el software de Lego Mindstorms que hauré utilitzat per a programar el robot i, també, faré una explicació del programa final amb el control PID.

2. Objectius

Els objectius principals que espero poder complir al finalitzar el meu treball de recerca són els següents:

- Aconseguir aprendre a construir i programar un robot que sigui capaç de seguir una línia mitjançant un sensor de llum.
- Gràcies al pack de Lego® que m'han proporcionat, aprendre com funciona tot el programari del robot, saber fer servir correctament els diferents sensors i saber programar el robot perquè faci el que jo determini.
- Saber diferenciar i aprendre com funcionen les diferents parts de que està compostat un robot.
- Conèixer els diferents tipus de sistemes de control que puc utilitzar per fer la meva tasca i implementar-los, aconseguint fer un robot seguidor mitjançant un control de tipus PID.
- "Desmitificar" o "demostrar" que la robòtica no queda tant lluny de la vida quotidiana i que estem envoltats per molts elements robòtics (sensors, controladors...) que des de fa temps ens fan la vida més fàcil.

3. Marc teòric

3.1 La robòtica

La robòtica és la ciència de l'enginyeria i la tecnologia dels robots, i el seu disseny. La robòtica s'ocupa del disseny, construcció i l'aplicació dels robots i està relacionada amb la seva mecànica, electrònica i software o programació.

El terme robot va ser 'presentat al món' pel dramaturg Txec Karel Čapek al 1920, en la seva obra Rossum's Universal Robots, ja que la paraula txeca *robot* (treballs forçosos) va ser traduïda al anglès com 'robot', però, realment, el terme va ser encunyat per **Isaac Asimov** al 1941.

3.1.1 Definició de robot

Un robot és un agent artificial mecànic o virtual. És una màquina usada per realitzar un treball automàticament i que és controlada per una computadora. Tot i així, no hi ha un consens sobre quines màquines poden ser considerades robots, però sí existeix un acord general entre els experts i el públic sobre que els robots tendeixen presentar algunes d'aquestes característiques: no són naturals, és a dir, s'han creat artificialment, es mouen i poden fer funcionar un braç mecànic, són capaços de sentir i manipular el seu entorn. A més, pot mostrar un comportament intel·ligent o habilitat per prendre decisions basades en l'ambient o una seqüència preprogramada.

De tota manera, no hi ha acords ni una definició precisa de què es considera robot. Fins i tot, Joseph Engelberger, un pioner en la indústria robòtica, va expressar clarament aquesta idea amb aquesta frase: "No puc definir un robot, però en reconec un quan el veig".

3.1.2 Inicis

Pel que fa als robots pròpiament dits se'n poden trobar exemples des de temps molt remots, com per exemple:

- Les primeres màquines autòmats, com un orgue de vent o una màquina que funcionava amb la introducció d'una moneda, daten del S.I a.C. i es creu que

van ser ideades i construïdes per Herón d'Alexandria, ja que era l'autor del llibre on estaven descrites.

A continuació es pot observar un quadre sobre l'evolució dels robots des dels inicis fins l'actualitat.

Any	Descripció	Inventor
S.I a.C	Descripcions de més de 100 màquines autòmats com les esmentades anteriorment.	Herón d' Alexandria, Ctesibius D'Alexandria i altres
1206	Primer robot programable (Vaixell amb 4 músics robotitzats)	Al- Jazari
1495	Cavaller mecànic : Disseny d'un robot hominoide.	Leonardo Da Vinci
1961	Instal·lació del primer robot industrial	George Devol
1975	Braç manipulador programable universal.	Victor Scheinman (Unimation)
2000	Robot hominoide capaç de desplaçar-se de forma bípeda i interactuar amb les persones. Asimo (Honda)	Honda Motor Co. Ltd
2002	Comercialització de Roomba un aspirador robòtic.	iRobot
2007/ 2010	Projecte Aiko, un Robot molt avançat amb aparença humana capaç de relacionar-se, diferenciar menjars, medicaments...	Le Trung



iRomba



Aiko



Asimo

3.1.3 Tipus de Robots

Existeixen una gran varietat de robots que poden ser classificats de diverses formes, per aplicació, per mitjà de locomoció i generacions. Començaré explicant la seva classificació per les seves aplicacions, així doncs trobem:

3.1.3.1 Robots industrials

L'organització internacional per l'estandardització o ISO els defineix com a robots controlats automàticament, reprogramables, manipuladors, multifunció i programables en tres eixos de moviment.

Entre els usos més comuns d'aquests robots destaquen la soldadura (utilitzada, per exemple, en la indústria de l'automoció), la pintura, l'emalatge i emmagatzematge, la inspecció de productes, tot dotat amb una alta resistència, velocitat i precisió.

Aquest tipus de robot fan accions repetitives i 'avorrides' i ajuden a millorar la qualitat dels productes ja que un operari humà, és més fàcil que cometi errors quan, a sobre, s'ha d'afegir que les feines repetitives provoquen un augment fatiga i per tant més errors. Així les empreses dotades de robots, redueixen el nombre d'errors, el preu de cost dels seus productes i els terminis de lliurament ja que els robots són, alhora, més ràpids que els humans, fent que el producte sigui més competitiu al mercat.

La majoria d'empreses utilitzen, per a moltes feines repetitives, de les que s'encarreguen màquines automàtics, disseny assistit per ordinador (*computer aid design*, CAD), fabricació assistida per ordinador (*computes aided manufacturing*, CAM) i control numèric per ordinador (*computer numerical control* CNC), que permet els enginyers controlar per ordinador tot el procés, més ràpidament i efectiva, des del disseny fins la fabricació final.

Un altre avantatge de l'ús de robots reemplaçant humans és que es posa menys en risc la integritat física de les persones, per a depèn quines feines.



Braç robot en una fosa

Com ja he dit la robòtica és un component essencial en qualsevol entorn modern de la fabricació, per tant la majoria de fàbriques estan augmentant l'ús de robots en la producció. Això pot provocar una certa preocupació a la població pel que fa a que els robots “prenguin” llocs de treball als homes ja que milloren el rendiment i els guanys de l'empresa, així com la qualitat dels productes resultants. Per altra banda, aquest fet el que provoca és un augment en la demanda de mà d'obra especialitzada en les noves tecnologies per poder controlar, reparar, programar... els robots; tot i que sovint aquesta demanda de mà d'obra qualificada no compensa les baixes d'altres treballadors poc qualificats.

3.1.3.2 Robots domèstics

Els robots domèstics estan destinats a ajudar-nos a fer les feines diàries de casa per a facilitar-nos la vida i guanyar en benestar i seguretat. Aquests tipus de robots sovint són utilitzats en les feines que estan caracteritzades per les 3 Ds: Avorrit, brut i perillós (dull, dirty and dangerous). Pot ser aquestes feines no són realment perilloses però si que són pesades i brutes. Com per exemple aspirar la casa, tallar la gespa o netejar els fons de les piscines.

Són un producte relativament nou tot i que ja fa temps que



existeixen els neteja piscines, o sense anar més lluny podríem considerar les màquines rentavaixelles.

Actualment s'estan desenvolupant molt els robots aspiradors que són capaços d'aspirar completament una planta de la casa sense caure per escales, xocar, ni deixar-se un pam de brutícia. A més un cop es posen en funcionament si es queden sense bateries tornen a la base per carregar-se i després continuar. A part d'aquests hi ha de molts altres tipus com els robots de

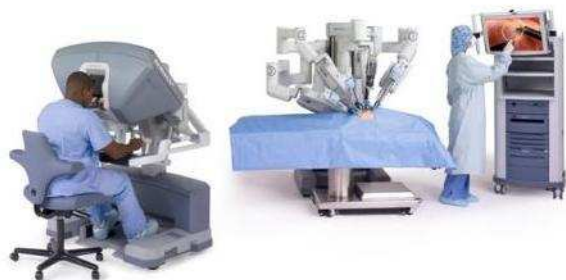


telepresència que vigilen la casa i mostren imatges en temps real; robots de companyia desenvolupats al Japó que donen converses senzilles a la gent gran i no es cansen de sentir sempre les mateixes històries, etc.

← "Spyke" robot de telepresència

3.1.3.3 Robots mèdics

Aquest tipus de robots estan dissenyats per a fer una funció específica relacionada amb el camp de la salut com pot ser transportar



medicines entre les diferents parts d'un hospital, realitzar operacions amb telepresència en les quals un metge controla el robot a distància i aquest és qui opera al pacient i, fins i tot, s'està treballant en la recerca amb nano-robots (robots a escala nanomètrica 10^{-9}) que podrien ser injectats als humans amb infinitud de possibilitats sanitàries.

3.1.3.4 Robots militars i d'exploració

Com molts dels avanços tecnològics, un dels primers camps en utilitzar-los ha estat la guerra.

Durant les últimes guerres, els països amb una tecnologia més desenvolupada han fet servir aquest tipus de robots per explorar cases, teòricament desocupades però que han resultat ser emboscades, sense posar en perill vides humanes; desactivar bombes; o en els anomenats TeleTanks utilitzats pels Alemanys durant la 2ª Guerra Mundial.

A part de l'exploració terrestre en zones perilloses, els robots també s'han sumat en l'ajuda a la cursa espacial i han sigut enviats a altres planetes per recopilar i enviar dades sobre la temperatura, nivell d'humitat, composició de les roques... com és el cas de l'enviat a Mart al 2004 i que ha estat operatiu fins a principis de 2009.



3.1.3.5 Robots educatius

La robòtica educativa és un mitjà d'aprenentatge, en el qual participen les persones que tenen motivació pel disseny i construcció de creacions pròpies. Aquesta part de la robòtica se centra principalment en la creació d'un robot amb la finalitat de desenvolupar de manera molt més pràctica i didàctica les habilitats dels qui el creen. Principalment s'està dirigit als estudiants. A més, es pretén estimular l'interès per les ciències i la tecnologia i, així mateix, fer que els estudiants aconseguixin una organització en grup, discussions que permetin desenvolupar habilitats socials, respectar cadascun el seu torn per exposar i aprendre a treballar en equip.

També podem classificar els robots segons la seva cronologia o per generacions:

3.1.3.6 Primera generació

Manipuladors: Que són sistemes multifuncionals amb un petit sistema de control que permet governar el moviment dels seus elements.

3.1.3.7 Segona Generació

Robots de repetició o aprenentatge : Són manipuladors que es limiten a repetir una seqüència de moviments, prèviament ordenada per un operador humà, fent ús d'un controlador manual o un dispositiu auxiliar. En aquest tipus de robots, l'operari en la fase d'ensenyament, utilitza una paleta de programació amb diversos polsadors o tecles, o de vegades, desplaça directament la mà del robot.

3.1.3.8 Tercera Generació

Robots amb control per ordinador que són manipuladors, controlats per un ordinador, que habitualment sol ser un microordinador. El control per ordinador disposa d'un llenguatge específic, compost per diverses instruccions adaptades al robot, amb les quals es pot confeccionar un programa d'aplicació utilitzant només el terminal del computador, no el braç. A aquesta programació se l'anomena textual i es crea sense la intervenció del manipulador.

Aquest tipus de robot s'està imposant al mercat ràpidament, la qual cosa exigeix la preparació urgent de personal qualificat, capaç de desenvolupar programes informàtics.

3.1.3.9 Quarta generació

Robots intel·ligents: són similars als anteriors, però, a més, són capaços de relacionar-se amb el món que els envolta a través de sensors i prendre decisions en temps real. Aquesta presa de decisions es pot programar segons les variables d'entrada que arriben al robot.

Juntament amb aquestes dues classificacions una altra molt acceptada és la de classificar-los segons el seu mitjà de locomoció.

3.1.3.10 Robots mòbils rodants (Rolling Robots)

Són els robots que es mouen gràcies a la rotació d'algun dels seus elements. Dins aquest grup trobem el més simples que utilitzen 4 rodes (no cal que totes siguin motores), robots amb una o dues rodes que es balancegen i mantenen l'equilibri alhora que es desplacen (segway), robots esfèrics que giren ells mateixos per moure's, i finalment robots amb sis rodes o amb tren de rodolament d'erugues , que és el sistema que utilitzen els tancs per desplaçar-se, per augmentar la tracció.



3.1.3.11 Robots que caminen (Walking Robots)

Aquests robots són capaços de pujar escales, caminar, córrer, però no de moure's en superfícies rocoses o desiguals. La majoria d'aquests són androides, és a dir, tenen forma humana i utilitzen dues 'cames' per caminar.

3.1.3.12 Els robots voladors

Són robots capaços de volar autònomament. Un exemple d'això el trobem al activar el pilot automàtic, que incorporen moltes aeronaus, que és capaç de volar fins la destinació i, fins i tot, fer l'aterratge. Aquests anteriors porten tripulació i comandants de vol però existeixen els anomenats vehicles aeris no tripulats , de l'anglès *unmanned aerial vehicles* (UAVs) que no porten humans a bord.

3.1.3.13 Robots estàtics

Dins aquest grup s'inclourien tots els braços robòtics i altres robots que no varien la seva posició però sí que tenen moviment en algun dels seus eixos.

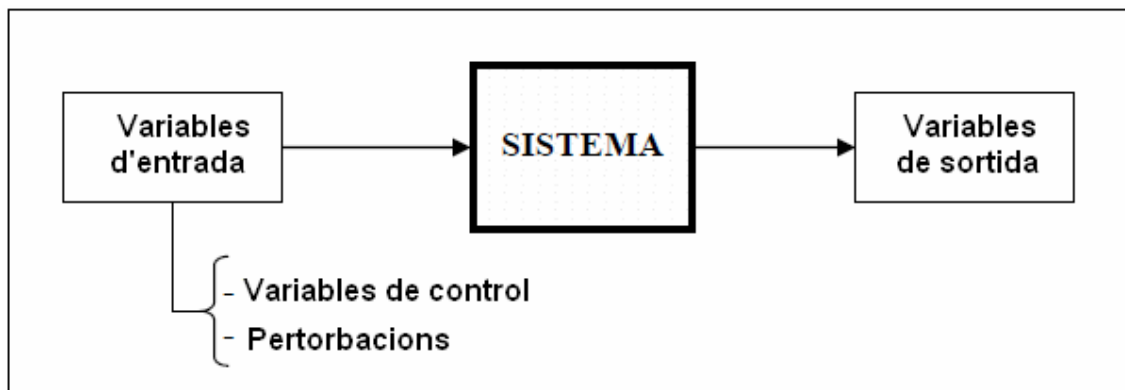
3.1.3.14 Altres mitjans de locomoció

A part d'aquests tipus de locomoció, que són els principals, altres robots es poden moure nedant, com és el cas dels robots submarins; serpentejant com algunes serps robot i altres robots zoomòrfics; finalment en algunes universitats d'EEUU han fet robots capaços d'escalar, anomenats escaladors.

Segons totes aquestes classificacions podem definir el robot amb el qual farem el projecte com un robot educatiu, ja que, com he dit als objectius, m'ajudarà a aprendre quin és el funcionament bàsic d'un robot i les parts que el conformen. Per altra banda també és un robot de quarta generació ja que serà capaç de relacionar-se amb el medi que l'envolti a través de sensors i prendrà diferents decisions segons el programem. Finalment, segons el seu mitjà de locomoció, serà un robot rodant ja que utilitzarà 4 rodes, dues de motores i dues de seguidores que en el nostre cas seran "rodes boges".

3.2 Sistemes de control

Els sistemes de control automàtic són objectes o sistemes que , en rebre un senyal d'entrada, realitzen una funció de forma automàtica sense la intervenció de les persones.



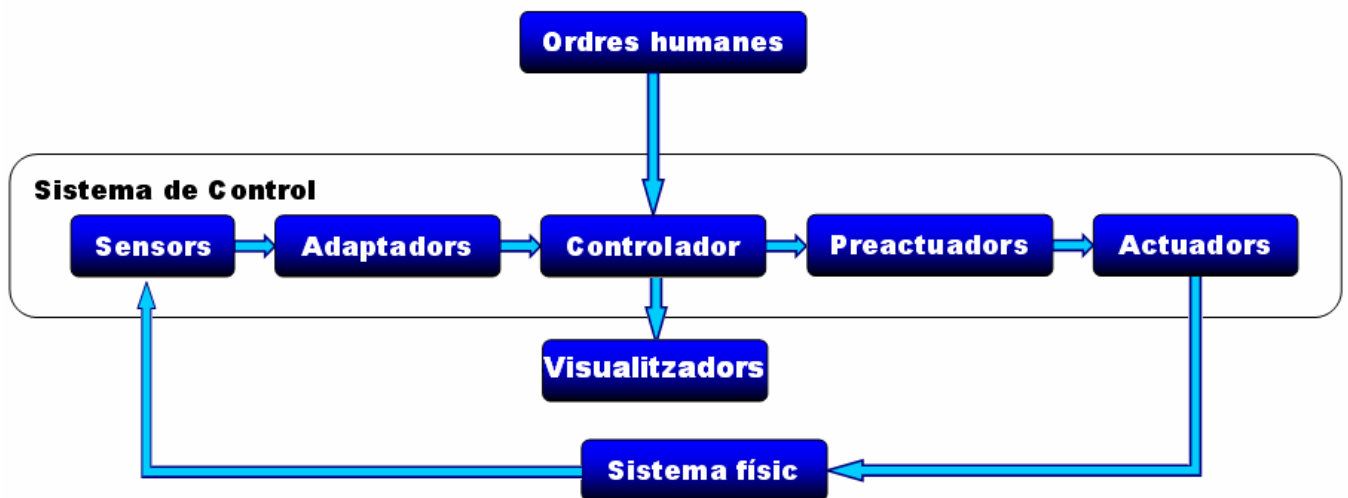
A l'esquema observem que el sistema, segons els senyals d'entrada, que són les variables que hi entren, emet unes variables de sortida. La finalitat d'un sistema de control és aconseguir, mitjançant les variables d'entrada, unes determinades variables de sortida amb uns valors prefixats.

A les variables d'entrada s'hi poden infiltrar unes pertorbacions que són irregularitats desconegudes que alteren el desenvolupament normal del sistema. Normalment, aquestes pertorbacions no es poden mesurar, però els seus efectes sobre les variables permeten detectar la seva presència. La utilització d'un sistema de llaç tancat permet corregir els problemes ocasionats per la presència de pertorbacions al sistema.

Un sistema de control ideal ha de permetre aconseguir el seu objectiu seguint aquests requisits:

1. Garantir l'estabilitat i ser robust davant de pertorbacions.
2. Ser tan eficient com sigui possible, segons un criteri preestablert, evitant comportaments bruscs i irrealment.
3. Que sigui fàcil de posar en pràctica i còmode d'usar en temps real amb l'ajuda d'un ordinador.

Un sistema de control esta format dels següents elements:



*(Aquest esquema correspon a un sistema de control de llaç tancat)

· Sistema físic: és la part d'un sistema real que pot controlar el sistema de control. El sistema de control pot controlar diferents magnituds com lluminositat, temperatura, posició, etc. del sistema físic.

- Sensors: són la part del sistema que s'encarreguen de recollir informació d'una o mes magnituds del sistema físic, segons allò que es vulgui controlar .
Ex : sensor de presència, de lluminositat, infrarojos, etc.
- Adaptadors: el senyal que surt del sensors a vegades no és adequat per arribar al controlador, en aquests casos utilitzem els adaptadors.
- Controlador: és la part del sistema de control que s'encarrega de donar ordres al actuadors a partir de les senyals que rep dels sensors i de les ordres i programacions de l'usuari, això ho fa comparant una variable física amb el valor desitjat, calculant-ne l'error i actuar per intentar corregir-lo. És per al robot com el cervell per als humans.
- Preactuadors: s'encarreguen d'adaptar els senyals elèctrics que surten dels controladors per tal que siguin prou potents per activar els actuadors.
- Actuadors: Un actuator és un dispositiu inherentment mecànic la funció del qual és efectuar una força o fer moure un altre dispositiu mecànic. Els actuadors poden ser pneumàtics, hidràulics, elèctrics. Ex: motors elèctrics per fer moure un cotxe teledirigit, cilindres pneumàtics per moure objectes...
- Visualitzadors: és la part que, dins o fora del sistema estant, s'encarrega d'informar a l'usuari del funcionament del sistema de control (Pantalles LCD, Leds ...).

3.2.1 Tipus de sistemes de control

Depenent del tipus de robot que es vulgui controlar hi ha diferents tipus de control. En el cas que ens ocupa tenim 2 sistemes de control bàsics que són, el llaç obert i el llaç tancat.

3.2.1.1 Llaç Obert

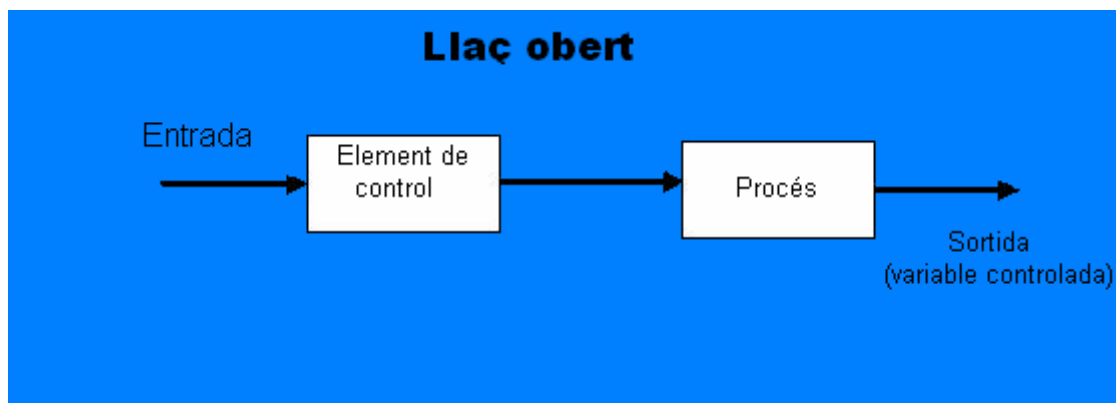
Són els sistemes on la sortida no afecta l'acció de control. En un sistema en llaç obert no es mesura la sortida ni es realimenta per comparar-la amb l'entrada. Per tant a cada entrada li correspon una condició operativa fixa; com a resultat, la precisió del sistema depèn del calibratge. Enfront pertorbacions el sistema de llaç obert no fa la tasca desitjada com seria convenient. A la pràctica, el control en llaç obert només s'utilitza si es coneix la relació entre l'entrada i la sortida i si no hi ha pertorbacions internes ni externes.

Exemple: En un llum d'escala nosaltres polsem el botó quan volem que s'encengui la llum que romandrà encesa durant un cert període de temps preestablert. Tot i que nosaltres arribem al nostre destí (dins de casa) la llum seguirà encesa. (No es controla si es útil en tot moment, o no)

Característiques:

- Són senzills.
- Res assegura la seva estabilitat davant una pertorbació.
- La sortida no es compara amb l'entrada.
- Les pertorbacions els afecten notòriament.
- La precisió depèn del previ calibratge del sistema.

Elements bàsics:



- Element de control: Aquest element determina l'acció que es realitzarà donada una entrada al sistema de control.

- Element de procés: És el sistema on es controlarà la variable (el procés que es vol controlar).

3.2.1.2 Llaç tancat

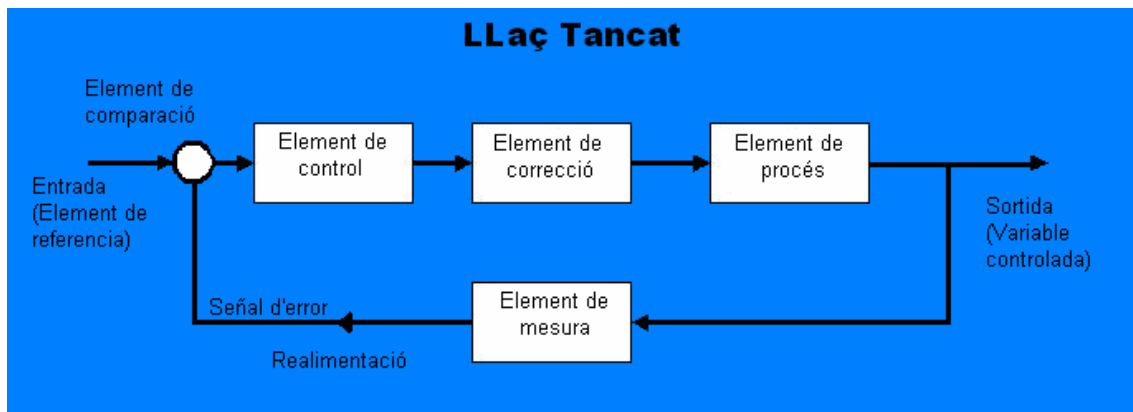
Definició: Els sistemes de control de llaç tancat també anomenats sistemes de control realimentats utilitzen un sistema de control on s'alimenta el controlador amb el senyal d'error d'actuació, que és la diferència entre el senyal d'entrada i la sortida de realimentació per així reduir l'error i portar la sortida del sistema de control a un valor convenient i consegüentment que el robot dugui a terme una acció desitjada de la millor forma possible. El control en llaç tancat sempre implica l'ús d'una realimentació per reduir l'error del sistema.

Exemple: En un llum amb sensor de presència a l'entrada d'un pàrking, la llum s'encendrà, al detectar moviment, i romandrà encesa durant un curt període de temps preestablert (que es pot canviar). Un cop passat el temps tornarà a utilitzar el sensor per detectar si hi ha algú i encendre's o no. Això permet estalviar energia.

Característiques:

- Són complexos i amplis en quantitat de paràmetres.
- La sortida es compara amb l'entrada i això afecta al control del sistema.
- La seva propietat de retroalimentació.
- És més estable a pertorbacions i variacions internes.

Elements bàsics:



- Element de comparació: Aquest element compara el valor de referència de la variable a controlar amb el valor mesurat que s'obté a la sortida, i produeix un senyal d'error que indica la diferència del valor de sortida i el valor de referència.
- Element de control: Aquest element decideix quina acció prendre quan es rep un senyal.
- Element de correcció: Aquest element s'utilitza per produir un canvi en el procés per eliminar l'error.
- Element de procés: És el sistema on es controlarà la variable (El procés que es vol controlar).
- Element de mesurament: Aquest element produeix un senyal relacionat amb la condició de la variable controlada, i proporciona el senyal realimentat a l'element de comparació per determinar si hi ha o no error. (Sensors)

Tipus de Llaç tancat:

Dins dels sistemes de control retroalimentats, o de llaç tancat, podem distingir fonamentalment entre diferents subtipus que ens permetran, amb la seva combinació, que el control efectuat sigui més precís, és a dir, que la resposta s'acosti més al valor que desitgem obtenir, i que s'arribi a aquesta referència més ràpidament. Seguidament parlaré dels controls de tipus proporcional (P), Proporcional Integral (PI), Proporcional Derivatiu (PD), Proporcional Integral Derivatiu (PID) i per acabar del control tot o res.

Abans de començar a parlar dels diferents tipus de control cal definir alguns paràmetres:

- Distingim principalment dues fases per les quals passa un procés automàtic: el règim transitori i el règim permanent.

1) El règim transitori és la fase que comprèn des del moment en que es dona l'ordre fins que el sistema respon i aconsegueix la resposta desitjada.

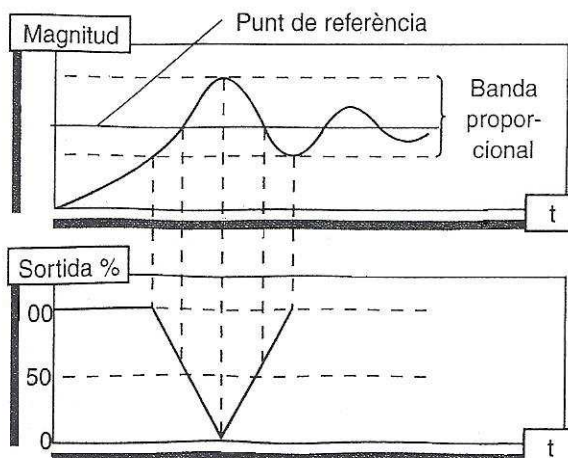
2) El règim permanent és la part del procés en la qual el sistema ha de mantenir la resposta aconseguida solucionant pertorbacions o altres problemes que es presentin.

- La consigna o punt de referència és el valor al qual es vol que el sistema arribi. Ex: En un aire acondicionat, la consigna seria la temperatura a la que volem que l'habitació estigui, tot i que és difícil que s'hi arribi exactament sense variacions.

- L'error o offset que és la diferència entre el punt de referència i la resposta del sistema o estat real d'aquest. Seguint l'exemple anterior, si la consigna és arribar a 24°C però la temperatura real en un instant determinat es 24.2°C, l'error seria 0.2°C. Aquest error va variant amb el temps i es tendeix a fer-lo ínfim.

3.2.1.2.1 Proporcional

El regulador proporcional provoca que la sortida canviï proporcionalment al valor del error calculat. Consisteix en el producte entre una constant proporcional o guany proporcional (K_p) i l'error ($e(t)$). Si donem un valor alt a K_p la resposta serà ràpida però es produirà un error elevat i això pot fer que el sistema sigui inestable. Per això s'ha d'arribar a un compromís en que la millora del règim transitori no comporti un empitjorament del regim permanent.



El regulador proporcional permet un control de manera que la sortida és proporcional a la desviació respecte la consigna, dins de la banda proporcional que ve determinada per dos valors, un

per sobre de la consigna i un altre per sota , entre els quals la variable controlada es mou. Un cop que la magnitud està dins la banda proporcional, la sortida disminueix progressivament i proporcional la seva potència.

Triar la banda proporcional adequada és molt important ja que si triem una banda massa estreta apareixerien oscil·lacions que farien que el sistema no arribes a estabilitzar-se completament i si la banda proporcional fos molt ampla no apareixerien tantes oscil·lacions però el valor del offset pot ser bastant elevat.

L'expressió matemàtica que defineix la sortida o resposta del controlador en funció de l'error i la constant de proporcionalitat és:

$$P_{out} = K_p e(t)$$

On: P_{out} → És la sortida de l'acció proporcional.

K_p → Constant de proporcionalitat.

$e(t)$ → Error o desviació.

3.2.1.2.2 El regulador proporcional derivatiu (PD)

Aquest regulador consisteix en la suma de l'acció proporcional com en el cas anterior i l'acció derivativa.

L'acció derivativa, a la pràctica, el que fa és dur a terme una acció predictiva ja que si es coneix quina serà la resposta del sistema i l'error que provocarà ens hi podem avançar per corregir-la, és a dir, anticipar l'efecte de l'acció proporcional per estabilitzar més ràpidament la variable controlada.

La funció de l'acció derivativa és mantenir l'error mínim corregint-lo proporcionalment a la velocitat a la que es produeix per evitar que s'incrementi. L'acció derivada actua en el règim permanent i només quan es produeix una pertorbació (modificant el valor absolut de l'error) , si l'error és constant aquesta acció no actua.

Sumant l'acció derivativa a la proporcional aconseguim que el sistema sigui menys oscil·latori, millorant així el règim permanent; tenir un sistema millor

preparat enfront les pertorbacions ja que l'acció derivada fa que el sistema hi respongui amb més rapidesa i els errors transitoris seran menors.

L'expressió matemàtica que mostra la resposta del controlador de l'acció derivativa és la següent:

$$D_{out} = K_d \frac{d}{dt} e(t)$$

On: D_{out} → És la sortida de l'acció derivativa.

K_d → Constant derivativa.

$e(t)$ → Error o desviació

La sortida d'aquesta acció dependrà de la constant, i de l'acció derivativa, $d/dt \cdot e(t)$ que és el resultat de restar a l'error anterior (d'un interval molt petit de temps anterior) l'error actual, i que suposa que serà igual a l'error següent.

Si K_d és gran aleshores allarguem el temps d'estabilització del sistema, però redueix les oscil·lacions, per contra, si K_d és petita llavors tenim un temps d'estabilització més ràpid, però unes oscil·lacions majors.

La constant derivativa és el producte de la constant de proporcionalitat (K_p) pel temps d'acció derivada (T_d)

$$K_d = K_p T_d$$

La suma de les accions Proporcional i Derivativa (PD) quedaria de la següent forma:

$$u(t) = \underbrace{K_p e(t)}_P + \underbrace{K_p T_d \frac{de(t)}{dt}}_D$$

On: $u(t)$ és la sortida de l'acció combinada PD

3.2.1.2.3 L'acció Proporcional Integral

És la suma de l'acció proporcional i l'acció Integral. El propòsit de l'acció integral és disminuir i eliminar l'error en l'estat permanent o estat estacionari. El terme integral és proporcional a la magnitud i a la duració del error. Quan hi

ha un error, que no s'ha corregit, el terme integral els va sumant, per després multiplicar-los per una constant (K_i) i així corregir aquest error.

Així doncs amb la combinació d'aquesta acció amb la proporcional el que es vol aconseguir és reduir l'offset generat al utilitzar un regulador proporcional amb una K_p alta per millorar l'estat transitori que, com ja he dit, provocaria inestabilitat a l'estat permanent.

L'acció integral també es durà a terme quan es produeixi una pertorbació permanent i ajudarà a reduir els efectes del soroll.

L'expressió matemàtica de l'acció integral queda definida per:

$$I_{out} = K_i \int_0^t e(\tau) d\tau$$

On: I_{out} → És la sortida de l'acció integral.

K_i → Constant integral.

$e(t)$ → Error o desviació.

Si K_i és gran, els errors en el estat estacionari s'eliminen més ràpidament, però provoca un sobrepic major, en canvi si K_i és petit, alentim el temps d'estabilització de l'error reduint també l'excés o sobrepic.

La K_i pot ser trobada dividint la constant de proporcionalitat entre la constant de temps integral T_i

$$K_i = \frac{K_p}{T_i}$$

La suma de les expressions de les accions Proporcional i Integral (PI) seria:

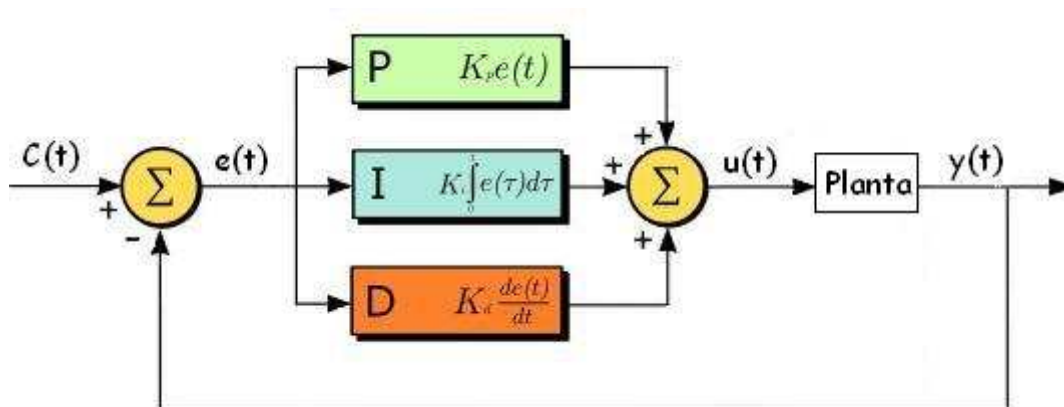
$$u(t) = \boxed{K_p e(t)} + \boxed{K_i \int_0^t e(t) dt}$$

P I

On: $u(t)$ és la sortida de l'acció combinada PI

3.2.1.2.4 Proporcional Integral Derivatiu

El control de tipus PID sorgeix de la combinació simultània de les tres accions comentades anteriorment per efectuar un control òptim de qualsevol procés. En el següent esquema es pot observar com funciona un sistema controlat per un regulador PID:



Podem observar que arriba una consigna que és el senyal d'entrada $C(t)$, corresponent al que seria el valor ideal, a la que se li resta el valor real $y(t)$ de la magnitud que volem controlar i se n'obté l'error $e(t)$. Aquest error es tracta per les accions proporcional, integral i derivativa que se sumen per donar lloc a una resposta $u(t)$ obtinguda mitjançant un control PID. El sistema o planta efectua la resposta i torna a mesurar el nou valor real en que es troba la magnitud controlada.

- L'acció proporcional permet reduir les oscil·lacions del règim transitori, tot i que sempre genera un error.
- L'acció derivativa actua ràpidament per tal de corregir els canvis ocasionats per pertorbacions externes que es produeixen durant l'estat permanent. Aquesta acció ens indica la velocitat a la que es produeix l'error per tal d'estimar a un temps determinat quin serà l'offset i així modificar per endavant la sortida.

· L'acció integral corregeix l'offset i ajusta el valor de la variable al punt de referència o consigna. Això ho fa 'mirant enrere' per determinar quina ha sigut la dinàmica de la variació de l'error.

El control PID s'utilitza per a obtenir un control precís en processos continus tals com control de pressió, força, cabal , control de velocitat en automòbils (Control de creuer) , etc.

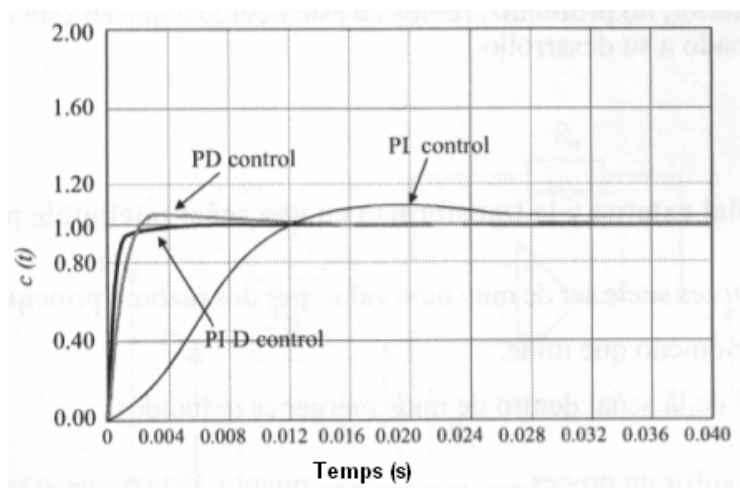
El comportament d'un regulador PID correspon a la suma de les tres accions a dalt esmentades i el seu comportament s'expressa amb la següent equació:

$$u(t) = \underbrace{K_p e(t)}_P + \underbrace{\frac{K_p}{T_i} \int_0^t e(t) dt}_I + \underbrace{K_p T_d \frac{de(t)}{dt}}_D$$

Observem:

- Les equacions matemàtiques de les tres accions: proporcional (P), integral (I) i derivativa (D).
- La sortida del controlador, u(t), que s'obté de la suma de P+I+D.

En la següent imatge es pot observar una gràfica comparativa entre els controls PI, PD i PID.



A la gràfica es mostra la variació de l'estat real de la variable controlada respecte del temps amb els diferents tipus de

controlador. En aquest cas la consigna és 1.00

3.2.1.2.4 Control tipus tot o res

Un sistema de control tot o res és aquell en que la sortida només adopta dos estats: connectat i desconnectat o màxima i mínima sortida.

Un exemple típic d'aquest control són els termòstats en que un interruptor automàtic connecta o desconnecta tota la potència disponible.

L'expressió matemàtica de l'acció d'aquest sistema de control seria

$C(t) = 1 \rightarrow$ si $E(t) > R(t)$ On: $C(t)$ és la sortida del controlador.

$C(t) = 0 \rightarrow$ si $E(t) < R(t)$ $E(t)$ és la consigna.

$R(t)$ és la realimentació o valor real mesurat.

3.2.1.3 Comparació Llaç Obert vs Llaç Tancat

Avantatges del llaç tancat:

- Més exacte a l'hora dels valors real i requerit.
- Menys afectat per les pertorbacions i canvis en les característiques dels components , per tant ens permet controlar sistemes inestables i fins i tot els que tenen errors de modelatge.
- La velocitat de resposta s'incrementa i, per tant, l'ample de banda és major, és a dir, l'interval de freqüències en els quals el sistema respondrà augmenta.

Inconvenients llaç tancat:

- Existeix una possibilitat d'inestabilitat.
- El sistema és més complex i, per tant, més car.
- Es pot introduir el problema del soroll al fer la mesura del valor de realimentació .

Inconvenients llaç obert:

- Mai es coneix el sistema, com a molt es pot conèixer un model aproximat, per la qual cosa no es pot aconseguir l'invers perfecte.

-No compensa pertorbacions en el sistema.

3.3 Components del robot

El robot que construiré està format per una sèrie de components que serveixen per crear un estructura física sobre la qual muntaré sensors, el controlador i actuadors que em serviran per dur a terme el control del robot.

3.3.1 Sensors

Els sensors són aparells electrònics que s'encarreguen de captar magnituds físiques i transformar-les en senyals electrònics comprensibles pel controlador. Simulen els sentits humans o animals en els robots. La gran característica que tenen els sensors, és que són sensibles a determinats estímuls que, en captar-los, es transmeten com a senyal a un altre dispositiu o sistema,



Sensor d'ultrasons



Sensor de temperatura

que els utilitza com a informació, bé per a efectuar algun control, algun accionament, bé per elaborar alguna taula, etc. Els sensors són utilitzats en objectes quotidians com els botons d'ascensors sensibles al tacte, les cèl·lules fotoelèctriques als peatge i en alguns llums que podem augmentar o atenuar una llum. Hi ha infinitud de variables de mesura que pot captar un sensor, algunes d'elles són, per exemple: temperatura, intensitat lumínica, distància, etc; i per això s'utilitzen sensors de llum, de temperatura, d'ultrasons...



Sensor de llum

De sensors se'n poden trobar de molts tipus i qualitats fet que afecta al seu preu. Tots tenen unes característiques tècniques comunes que ens permetran saber la seva qualitat:

- Precisió: és l'error de mesurament màxim esperat.
- Abast: Distància màxima en que el sensor capta la magnitud.
- Sensibilitat: Quantitat de valors d'una magnitud que el sensor és capaç de captar. Es pot aplicar tant als valors intermedis (matisos o tonalitats) com als valors màxim i mínim.

És la relació entre la variació de la magnitud de sortida i la variació de la magnitud d'entrada.

- Resolució: és la mínima variació de la magnitud d'entrada que pot captar.
- La mida: Poden haver sensors de varies mides, però es tendeix a empetir-los perquè ocupin menys espai i arribin a llocs minúsculs.
- Altres factors com són el consum energètic, la fiabilitat i la resistència a factors externs, al pas del temps, etc.

Per a fer el meu projecte dispo de una sèrie de sensors, alguns dels quals no utilitzaré, que estan inclosos dins el pack de Lego® Mindsotrms Education NXT. El disseny dels



sensors del Lego® esta especialment enfocat a augmentar al seva resistència. Estan fets de plàstic dur, per així augmentar la seva durabilitat, desgast i fer que siguin més robustos ja que estan destinats " al treball a les aules". Tenen forma de prisma rectangular (capsa) i en una base tenen el sensor pròpiament dit i a l'altra oposada el connector on s'endolla el cable que transmet les dades al bloc controlador.

A continuació explicaré els diferents sensors dels quals dispo:



- 1- Sensor de rotació intern del motor.
- 2- Sensor ultrasònic
- 3- Sensor de llum
- 4- Sensor de so
- 5- Sensor de tacte.

1- Els sensors de rotació estan al interior dels 3 servomotors dels que dispo, la seva funció és mesurar l'angle de rotació del eix o l'espai recorregut per unitat de temps. Aquest tipus de sensors són utilitzat, per exemple, en el control de velocitat de creuer dels automòbils.

2- També dispo d'un sensor d'ultrasons que serveix per detectar i determinar la distància del robot a una paret o qualsevol altre obstacle. El sensor emet uns ultrasons que reboten a la superfície de l'objecte concret i depenent del temps que triga en tornar és capaç de calcular la distància a la que el robot s'hi troba. Podem trobar un exemple d'utilització d'aquest sensor en les ajudes d'aparcament que porten molts vehicles 'nous' que fan sonar un "bip" que augmenta de freqüència amb la proximitat del cotxe al vehicle del darrera o del davant.

3- Un tercer sensor és el sensor de llum que, en el cas que ens ocupa, és el que farem servir principalment. El sensor de llum o sensor fotoelèctric és el sensor que respon al canvi en la intensitat lluminica. En el meu cas servirà al robot per fer-li saber si es troba , o no, sobre la línia negra que haurà de seguir. Disposa d'un component que emet la llum i un que rep la llum generada per aquest emissor.

4- El sensor de so el que fa es captar els sons de l'ambient i utilitzar-los per alguna funció com seguir la direcció del so. El sensor detecta els decibels mitjançant una escala logarítmica i per defecte ens dona el valor de 0 al 100 %.

Segons els nivells de so pot determinar el "lloc on es troba":

4-5% Una casa silenciosa.

5-10% Algú parlant de lluny

10-30% Una conversa prop del sensor
o música a volum normal.

30-100% Gent cridant o música a
volum alt.

5- El pack de Lego® també em proporciona dos sensors de tacte (touch sensor). El sensor de tacte és un sensor de posició que capta si el robot ha arribat , o no, a un lloc determinat de l'espai mitjançant el contacte directe del robot (per part del sensor) amb un cos físic. El sensor detecta quan l'estem pressionant i quan no. Un exemple d'això el podem trobar en les tecles de l'ordinador.

3.3.2 El controlador

El controlador és la part física del sistema s'encarrega d'executar una acció segons les ordres que nosaltres li hem donat (programat) i les variables d'entrada que li arriben dels sensors, a través dels ports d'entrada, les quals processa.

El controlador és un circuit integrat o xip que inclou en el seu interior les tres unitats funcionals d'una computadora o ordinador : unitat central de processament, memòria i unitats d'entrada i sortida.

-La unitat central de processament , CPU o simplement el processador, és el component del controlador que interpreta les instruccions contingudes als programes i processa les dades.

- La memòria és un dispositiu que reté dades informàtiques durant un interval de temps. Solen ser d'accés ràpid i poden ser volàtils, quan la informació emmagatzemada s'esborra al aturar el dispositiu , o no volàtils.
- Les unitats d'entrada i sortida permeten al controlador "comunicar-se amb l'entorn". Les unitats d'entrada són les que fan arribar informació als controladors procedent de sensors, un ordinador, etc; mentre que les de sortida són les que utilitza el controlador per enviar respostes processades als diferents actuadors. L'arribada/enviament d'informació del controlador no sempre es fa mitjançant cables ja que pot tenir instal·lat un dispositiu bluetooth, com és el cas del bloc Lego®, o altres dispositiu de xarxes wireless.

Els controladors o microcontroladors (en aquest cas a escala micromètrica 10^{-6}) són molt utilitzats en el nostre dia a dia: gairebé a cada habitatge hi ha un, sinó més, ordinadors els quals en porten varis; les modernes rentadores digitals en porten un; i la majoria de vehicles el duen instal·lat per controlar quan canviar l'oli, la pressió de les rodes, etc.

3.3.3 Actuadors

Els actuadors són dispositius capaços de transformar energia hidràulica, pneumàtica o elèctrica en un altre tipus d'energia com cinètica o tèrmica amb la finalitat de generar un efecte sobre un procés automatitzat. L'actuador rep l'ordre d'un regulador o controlador i en funció d'aquesta genera l'ordre per activar un element final de control. Podem trobar diferents tipus d'actuadors, aquí hi ha alguns exemples:

-Cilindres pneumàtics i hidràulics: Realitzen moviments lineals gracies als fluids que utilitzen com l'aire o l'oli , respectivament.

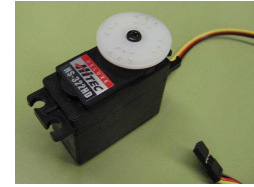
-Vàlvules: N'hi ha de comandament directe, motoritzades, electropneumàtiques, etc. S'utilitzen per a regular el cabal de gasos i líquids.

-Resistències calefactores : utilitzades per escalfar.

-Motors elèctrics: com és el cas dels servomotors que utilitzarem.

Em centraré en explicar el funcionament d'un servomotor (motor elèctric) que serà el que utilitzaré per fer el projecte.

Definició: Un servomotor o servo és un dispositiu semblant a un motor de corrent continu però el que el fa diferent de qualsevol altre motor elèctric és que és capaç de situar-se en una posició determinada (en graus, rotacions, radiants...)

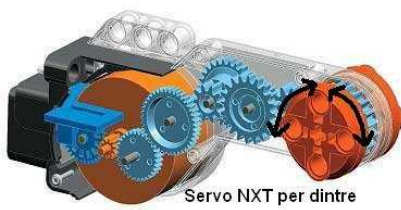


i fer-ho a una velocitat també controlada. Els elements que s'encarreguen de detectar la seva posició s'anomenen "encoders". El servo està conformat per un motor, una caixa reductora, un circuit de control i tres cables, dos dels quals li proporcionen energia i un altre per on rep dades. En el cas que estudiem, energia i dades del controlador arriben pel mateix cable.

Un dels principals avantatges dels servos és que són capaços de proporcionar un "alt" parell motor en relació a la seva mida.

Els servomotors són molt utilitzats en tasques diàries com per exemple en un lector de CD/DVD per expulsar o introduir aquest, els cotxes radio control i en la robòtica, com és el nostre cas.

Nosaltres disposem de 3 servos de l'NXT que, en el cas que ens ocupa, ens serviran per fer girar les rodes i, així mateix, moure el robot. Els motors disposen d'un tacòmetre que els serveix per controlar i alhora graduar la velocitat i també incorporen un sensor de rotació.



3.4 Packs de robots educatius

En l'àmbit dels robots educatius hi ha diverses marques que ofereixen els seus productes, no solament Lego®, que és la que utilitzaré. Entre aquestes destaquen Fischer® amb la gamma de productes fischertechnik; robotgroup® amb diverses versions de robots educatius; robotis® amb l'hominoide Bioloid i la marca de la que dispo Lego® amb els diferents packs de Lego® Mindstorms NXT i altres propostes per a nens més petits (Lego® WeDo):

Començaré parlant de la casa Lego® que és la que ens servirà per a fer el projecte. Lego® és una empresa de joguines danesa reconeguda principalment pels seus blocs de plàstic interconnectables.



A més d'això gracies a la gran flexibilitat de construccions factibles amb les seves peces han desenvolupat una sèrie de robots educatius molt flexibles amb els quals es poden crear molts aparells a petita escala.

Per una banda existeix la sèrie Lego® WeDo dedicada a nens a partir de 7 anys on poden construir 12 models predissenyats i molts altres més, amb sensors simples i un motor que es connecta als seus controladors. Es poden programar comportaments amb una eina extremadament simple, fàcil i divertida per iniciar-se en la robòtica.

Per altra banda existeix també la sèrie Lego® mindstorms NXT, que es posà en venda al 1998, destinada a nois i noies una mica més grans, que depenent del set, permet fer projectes més avançats amb nombrosos sensors, servomotors i altres accessoris que es poden comprar a part. El programari del NXT és un software que permet programar el robot gràficament i està basat en LabView.



A part del pack Education , del qual en som posseïdors, n'hi ha d'altres amb més peces i amb caixes d'expansió fent que el preu del producte pugui anar de 250€ a 600€. Més informació sobre Lego® i Lego® mindstorms a <http://www.legoeducation.us/> i <http://mindstorms.lego.com/en-us/Default.aspx>



A continuació parlaré de la gamma de productes de Fischer® que és una marca d'Alemanya i va començar a oferir aquest tipus de productes al 1945. Com a empresa de robots educatius la seva proposta és ensenyar als estudiants una capacitat de pensament lògica, entendre la tecnologia i desenvolupar la seva

fantasia i creativitat.

Disposen de diversos packs destinats a diferents edats des dels 5 anys fins a més de 9. Dintre d'aquests rangs d'edats hi ha diferents "joguines" cada una amb les peces necessàries per construir uns tipus de robots predissenyats encara que hi ha caixes d'ampliació per comprar més peces, motors o sensors, això



podria suposar un desavantatge respecte els packs de Lego® que son més flexibles i amb un set bàsic pots construir molts robots diferents. Un avantatge respecte la tecnologia d'aquesta casa es que combina parts de l'àmbit de la robòtica amb tecnologia pneumàtica, amb l' energia solar i d'altres. La casa Fischer® posa a disposició dels seus usuaris un software propi, per a programar els seus productes, mitjançant una aplicació de programació gràfica. Alguns dels packs que més s'assemblarien al que nosaltres estem utilitzant serien ROBO TX Explorer que tindria un preu d' aproximadament 200€.

Més informació sobre fischertechnik a:

<http://www.fischertechnik.de/en/desktopdefault.aspx>

A més d'aquestes dues conegudes marques del joguet, també s'han endinsat en el món de la robòtica moltes altres empreses com és el cas de Robotis®. Aquesta és una empresa coreana amb l'objectiu , també educatiu, fer entendre el que es un robot des de nivells bàsics, com el seu nom en angles indica " un robot és..." (Robot is). Com les altres dues anteriors, també te dues games de productes :



- Ollio que estan destinats als més menuts per desenvolupar la creativitat, ciència, lògica juntament amb l'entreteniment. Aquesta sèrie incorpora nombroses peces que es poden combinar entre elles (semblant al Lego®) i un motor, però no incorpora ni controlador ni programari. Això fa que siguin relativament barats amb uns preus de 25 a120€ aproximadament i depenent del tipus de set.

- Bioloid, que és el robot més avançat d'aquesta casa i està destinat a tant alumnes de batxillerat com universitaris. El robot pot tenir formes zoomòrfiques o d'hominoide. Incorpora un gran nombre de servos , un controlador i disposa d'un programari iconogràfic propi basat en llenguatge C.



Pot tenir un preu molt variat que oscil·la de 300€, el model més bàsic, fins 1000€, el més complex. Es pot trobar més informació a:

http://www.robotis.com/xs/bioloid_en i

http://www.robotis.com/xs/ollo_en

3.5 Competicions robòtiques

A l'estat espanyol s'hi realitzen periòdicament diverses competicions de robots. Tot seguit parlaré d'una d'àmbit internacional que se celebra a Catalunya i

d'una de les més conegudes a l'estat espanyol , destinada als robots Lego®, la First Lego League, que també se celebra en altres països.

Des de fa 3 anys a la UPC de Terrassa se celebra el Concurs Internacional de Robòtica JET. La gent que s'inscriu ha de presentar el robot que han construït en una de les 6 categories que hi ha: Sumo i minisumo, consisteix en treure el robot contrincant del quadrat de joc; Velocistes i rastrejadors, dues categories semblants que consisteix en curses de robots rastrejant una línia negra; JET bot, categoria reservada als robots construïts en el taller JET, un taller dut a terme per estudiants de l'Escola Tècnica Superior d'Enginyers Industrials i Aeronàutics de Terrassa; Categoria lliure, en que hom pot presentar el seu autòmat que no encaixi en les categories abans esmentades.

Cada categoria disposa d'un reglament propi i els robots han de ser 'homologats'(complir el reglament) abans de dur-se en pràctica la competició. Els guanyadors reben premis econòmics i, els primers de cada categoria, material electrònic per a seguir fent recerca amb autòmats.

La FIRST LEGO league (FLL) és una competició que té com a objectiu donar a conèixer les noves tecnologies als nois i noies de nou a setze anys. Per això cada any tracta un tema tecnològic diferent. La competició se celebra en diverses ciutats d'Espanya així com altres de la resta del món.

La competició està estructurada en tres parts:

- La primera és un campionat utilitzant el robot. Només hi ha dos minuts i mig per ronda. El robot, a sobre d'un tauler de joc, ha d'intentar aconseguir el màxim nombre de punts complint diverses missions. Per complir-les, el robot ha de polsar palanques o agafar objectes i moure'ls. Pot utilitzar els accessoris que vulgui si han estat construïts utilitzant peces LEGO.
- El segona part és un projecte científic. El tema és el mateix que el de tota la competició, el passat any 2010 havia de tractar de l'enginyeria biomèdica. El projecte tractava de descobrir formes innovadores per curar lesions, superar les predisposicions genètiques, i maximitzar el potencial del cos, juntament amb el propòsit de portar una vida més saludable.

- L'últim bloc es el projecte tècnic. Consisteix en explicar el procés que ha seguit l'equip des de que va començar fins el dia de la competició.

Per el moment, no ens hem plantejat assistir amb el nostre robot a cap competició d'aquestes, però amb els coneixements adquirits, i ja que ens agrada la superació personal a l'hora de programar i millorar el robot, no ho descartem per a un futur no gaire llunyà.

4 Treball pràctic

A continuació, en aquesta part del treball, em dispo a explicar el meu treball pràctic sobre el cos teòric anterior.

El que m'he plantejat de fer, seguint els meus objectius plantejats, és construir, gràcies al pack Lego® Mindstorms Nxt Education proporcionat per l'IES Alba del Vallès, un robot seguidor de línies utilitzant un control tipus PID.

Començaré parlant dels meus primers passos amb el pack de Lego®, seguidament explicaré el material que utilitzaré (tipus de sensors) i després parlaré del software gràfic amb el que s'ha de programar el robot.

- Tant bon punt vaig rebre el pack del Mindstorms, vaig començar a ordenar les més de 600 peces que hi havia a la caixa.

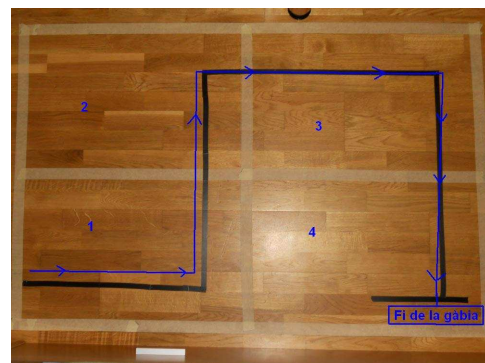




- A continuació vaig muntar el robot de tres rodes que proposen les instruccions de muntatge de Lego. Aquest robot està compost per dues rodes motores que són al davant i una roda 'boja' al darrera que 'balla' tot seguint el camí que imposen les dues del davant.

El tutor del treball i una altra professora, em van plantejar que un cop finalitzat el muntatge del robot comencés a fer unes unitats didàctiques (5) d'un curs de Lego® Mindstorms, pensat per a professors. Aquestes unitats estan formades per teoria sobre el que s'està fent en cada moment i les ordres que es donaran al robot, i uns exercicis per fer amb el software de programació per anar introduint nous conceptes del programari; es comença amb simples moviments de les rodes endavant i enrere, i es va complicant fins aconseguir fer un seguidor de línies negres amb un control de velocitat per so, introduint entre mitges bucles, bifurcacions, blocs lògics, etc.

A cada unitat hi ha un, o més "exercicis incrementals" que parteixen de la base d'un rectangle d'1x1.5 m. El rectangle està delimitat, en el nostre cas, per línies blanques per on el robot ha d'avançar depenent de les unitats des del primer quadrant fins al últim seguint la trajectòria



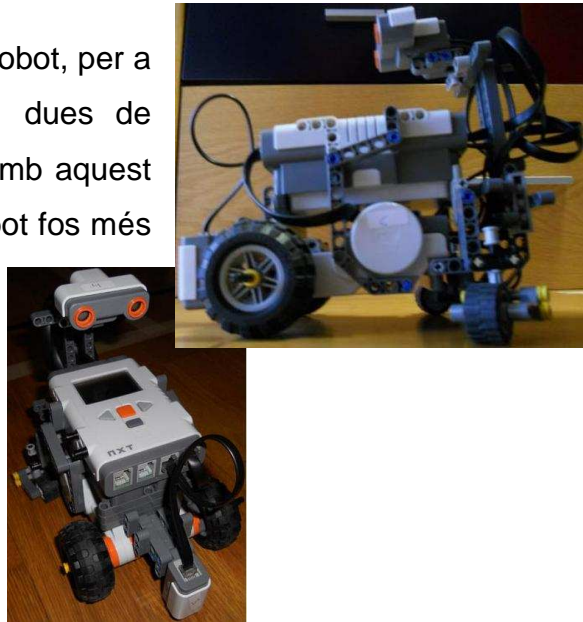
mostrada a la imatge fins arribar al quart quadrant, també anomenat final de la gàbia. (La imatge correspon a un dels últims exercicis en que ja seguia la línia negra per fer el recorregut)

http://www.edu365.cat/imagina/robots/lego_nxt/index.htm

Un cop vaig començar a programar el robot, tot seguint els passos de les unitats didàctiques, perquè es comencés a moure i alhora familiaritzar-me amb el tipus de llenguatge gràfic, que utilitzaria finalment per a fer el robot seguidor de línies, ens vam adonar que, quan el robot havia de fer segons quins girs i depèn quins moviments, la roda del darrera, que teòricament només havia de seguir les del davant, es quedava encallada en algun punt i impedia el correcte

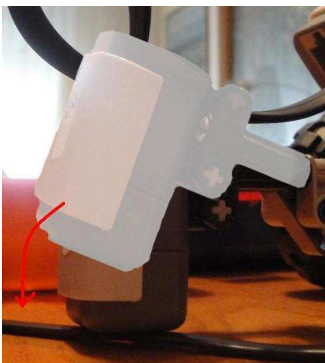
desenvolupament de la trajectòria del robot fent que aquest no anés del tot recte o impedit-lo avançar.

Per això vaig pensar en redissenyar el robot, per a fer-ne un que disposés de 4 rodes, dues de motors i dues de boges, al darrera. Amb aquest disseny el que preteníem era que el robot fos més compacte, robust i aconseguir solucionar el problema de les rodes boges esmentat.



Un cop solucionat aquest problema vaig haver de tornar a començar de nou les unitats sobre la programació ja que programant els mateixos exercicis anteriors molts moviments quedaven alterats.

Aleshores, sorgí un altre inconvenient: La roda C (connectada al motor de la banda esquerra) patinava al fer els gir i això afectava significativament al nombre de rotacions que s'havien de programar per fer, ja que uns cops patinava més i d'altres menys. Aquest problema es va solucionar simplement canviant l'eix que uneix la roda amb el motor ja que per alguna maniobra fortuïta aquest havia quedat una mica doblegat. Amb això es va aconseguir reduir l'efecte considerablement, tot i que, algun cop les rodes s'encallen perquè, com nosaltres, tampoc és perfecte.



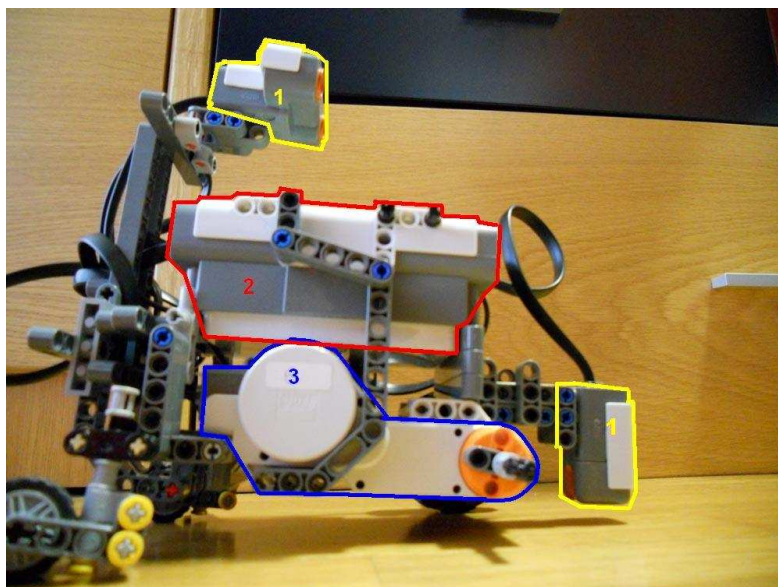
Una altra de les millores que s'han introduït al robot és la modificació de la part davantera, just on s'acobla el sensor de llum. A les instruccions de muntatge proposaven un mètode amb el qual el sensor no quedava perpendicular amb el terra i això ens podia significar problemes amb les lectures del sensor en el futur, ja que com explicarem, el sensor emet una llum i

en capta la quantitat relaxada i si no esta en posició vertical el rebot de la llum no és el més òptim.

Un cop solucionats tots els problemes , tocava seguir treballant fins acabar totes les unitats didàctiques, al final de les quals ja vaig ser capaç de fer que el robot seguís les línies negres. Cal dir que en aquesta primera fase el robot seguia les línies oscil·lant molt i no gaire ràpid. Per això tal i com hem vaig marcar com a objectiu vaig seguir practicant per aconseguir que les seguís mitjançant un control de tipus PID. Donat que a les unitats didàctiques havia treballat amb altres sensors, vaig pensar en incloure, si era capaç, el sensor d'ultrasons perquè , tot seguint la línia, si el robot detectava un obstacle s'aturés per no xocar.

4.1 Estructura del meu robot

Al robot que he construït s'hi poden diferenciar perfectament les diferents parts de les que consta:



1- Sensors per captar informació de l'entorn.

2- Controlador, per controlar el robot segons les variables d'entrada i la programació.

3- Actuador, en aquest cas un servomotor, per executar les ordres donades pel controlador, en el nostre robot, per moure's.

4.1.1 Els sensors

Els sensors que utilitzaré són:

- Per una banda el sensor de llum, (light sensor) que és un sensor que capta el grau de lluminositat de l'ambient. És a dir, detecta el nivell d'il·luminació i la transforma en un tant per cent (%). Aquest sensor pot, a més de captar llum, emetre'n ell mateix. Això hem resulta molt útil ja que el que vull és que enfoqui cap al terra, per després poder captar la llum reflectida pel material sobre el que es troba. D'aquesta manera el robot és capaç de distingir les diverses tonalitats del terra.

El sensor no diferencia colors, només veu el blanc, el negre i una infinitat de tonalitats de gris.



El sensor emet llum per l'orifici 1 i capta la llum reflectida, o la llum de l'ambient, per l'orifici 2. La llum que emet el sensor és de color vermell.

Hauré de connectar aquest sensor a un dels ports d'entrada (1,2,3,4) del bloc controlador, amb el cable especial de Lego® ja esmentat. El sensor de llum anirà connectat al port 3 que és el predefinit pel programari.

Aquest sensor es pot calibrar: Calibrar un sensor és adaptar-lo a les condicions de llum de l'entorn en particular on desenvoluparà la seva tasca, en el cas al que ens referim per aquest sensor, és dir-li al robot el que és blanc i el que és

negre en aquest entorn. El calibratge convé fer-lo a l'inici de cada activitat o si se sospita que ha pogut haver-hi algun canvi en el nivell de lluminositat.

Els sensors de Lego ja venen calibrats de sèrie i per a fer les unitats didàctiques no ens ha fet falta calibrar-los de nou, ja que les condicions, en aquest cas, llumíniques no varien. Però com que no se la llum ambiental que hi haurà al lloc on es farà la presentació del projecte els haurem de calibrar aleshores.

Per a aconseguir que el robot s'aturi al trobar un obstacle haurem d'incloure també un sensor ultrasònic (*ultrasonic sensor*).

El sensor ultrasònic és un sensor que mesura la distància a la què el robot es troba d'un obstacle emetent i captant ultrasons. El seu principi de funcionament és el mateix que el que utilitzen els ratpenats per a orientar-se. El sensor emet ultrasons que al xocar contra un objecte reboten.



Llavors el sensor els capta i mesurant el temps que triguen en anar i tornar calcula la distància entre l'objecte i el robot. El sensor pot expressar la distància en centímetres o polzades. El sensor d'ultrasons pot mesurar fins a 250cm (100 in) amb un marge d'error d'uns 3 cm. Malgrat que aquest sensor es força precís, quan els ultrasons reboten sobre superfícies molt petites, primes o esfèriques el sensor perd precisió.

El sensor s'ha de connectar al bloc controlador, a un dels ports d'entrada. En el cas que ens ocupa, el connectarem al port 4, també predefinit.

4.1.2 El controlador

Com ja he explicat, el controlador és la part del sistema de control que rep les nostres ordres i n'ordena la seva execució. També rep la informació dels sensors i la processa. És la part més important del robot, el seu cervell.

El bloc de l'NXT del que dispo per a fer el treball està equipat amb un microcontrolador ARM7 de 32 bits, que inclou 256 Kb de memòria Flash i 64 Kb de memòria RAM. Disposa també de diverses unitats d'entrada i sortida.

El meu controlador és híbrid ja que les senyals que hi entren i surten poden ser tant analògiques com digitals.

Unitats d'entrada:

- Port USB: Que ens permet entrar la programació feta amb el programari de Lego® Mindstroms NXT
- Els ports de la part inferior del bloc per on entren les dades provinents dels sensors. Hi ha 4 ports i cada sensor en té assignat un predeterminat.
- Els propis botons del controlador que utilitzem per encendre, apagar i moure'ns pel menú.
- Bluetooth que permet controlar-lo sense fils.

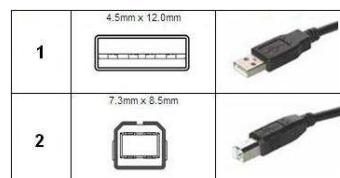
Unitats de sortida:

- Les unitats de sortida o els ports A, B i C, respectivament, són les que ens serviran per connectar el bloc NXT amb els servomotors.
- Pantalla LCD que ens mostra el menú i altres informacions sobre el robot, com la bateria, el programa que està executant...



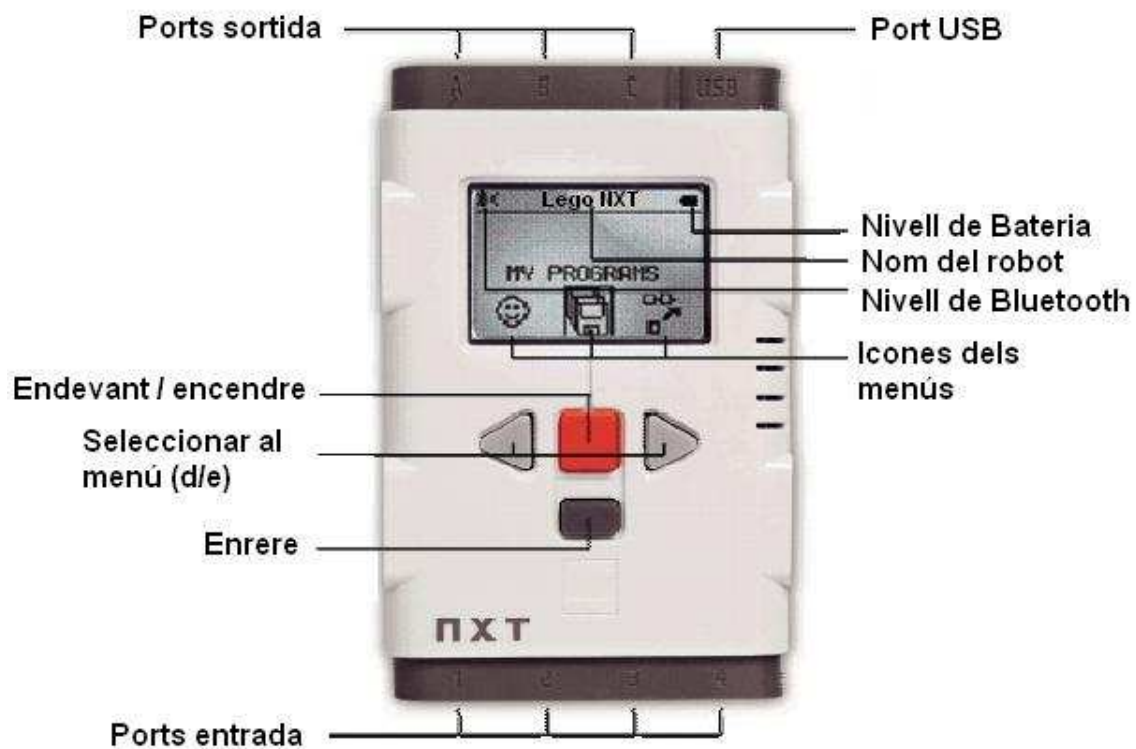
- També disposem d'un altaveu al controlador. Correspon a les 4 obertures horitzontals a la part central, per on surt el so.

El cable USB que s'ha de connectar té un connector de cada tipus a cada extrem.





El cable que s'utilitza per connectar el controlador amb els sensors i actuadors és especial de Lego® , tot i que s'assembla molt al telefònic.



Dins el controlador, hi ha instal·lat un programa intern de sèrie que disposa de 6 menús, que són:



My files (els meus arxius): On estan guardats els sons i tots els programes que descarreguem des del software de l'ordinador.



Bluetooth: Aquesta opció ens permetrà utilitzar el bluetooth i connectar, sense fils, el bloc amb altres aparells amb bluetooth.



Nxt program: Ens permet programar el robot sense necessitat de descarregar-li informació del programari informàtic. Les opcions són molt limitades.



Settings (preferències): Dins d'aquest menú podem esborrar arxius, canviar el volum, veure la versió del robot i posar-lo en mode dormir (*sleep mode*).



Try me : Aquest menú serveix per determinar si els sensors i motors funcionen.



View: Permet visualitzar diferents variables a través dels sensors, com la llum ambiental, la distància al sensor ultrasònic, el nivell de soroll.

4.1.3 Els servomotors

El meu robot està equipat amb dos servomotors que mouran les dues rodes davanteres segons les ordres donades. Aquests servomotors dels que disposem ens permetran poder graduar la seva posició dins el seu rang d'acció. Això ens anirà bé a l'hora d'efectuar segons quins exercicis on controlem el moviment mitjançant el nombre de rotacions de les rodes o els graus o radiants girats.



Els servo motors de Lego® pesen 80g, funcionen amb una tensió de 9V i ens poden arribar a proporcionar un parell motor de 3.8 Kg/cm

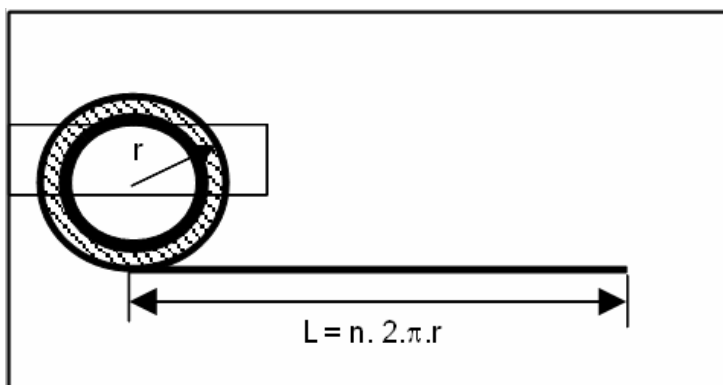
4.2 Pràctica 1: Longituds

Materials

Robot, metre, peu de rei i marcador.

Objectiu

1. Es tracta de determinar el nombre de graus que s'han d'introduir als motors per arribar a una distància d'un metre. L'experiment s'ha de repetir per determinar la precisió.



$$\alpha = n \cdot 360$$

$$\alpha = (L / 2 \cdot \pi \cdot r) \cdot 360$$

Per tant:

$$\alpha = (100 / 2 \cdot \pi \cdot 2,78) \cdot 360$$

$$\alpha = 2060,99 \approx 2061$$

L = longitud a recórrer (cm)

r = radi de les rodes (cm) → En el nostre cas 2.78 cm

α = angle a girar ($^{\circ}$)

n = voltes que les rodes giraran

Distància: 1m → Graus: 2061°	Fre		Flotació	
	Distància 1 cm	Distància 2 cm	Distància 1 cm	Distància 2 cm
Potència				
100	100	100	102	102
75	100	100.5	101	101
50	100	100	100	99.8
25	100	101	101	100

2. Quan la duració es posa a il·limitat els motors no funcionen indefinidament

Distància Il·limitat	Flotació	
Potència	Distància 1 (cm)	Distància 2 (cm)
100	23	33
75	23	22
50	13.5	14
25	5	4.5

4.3 Pràctica 2: Velocitats

Materials

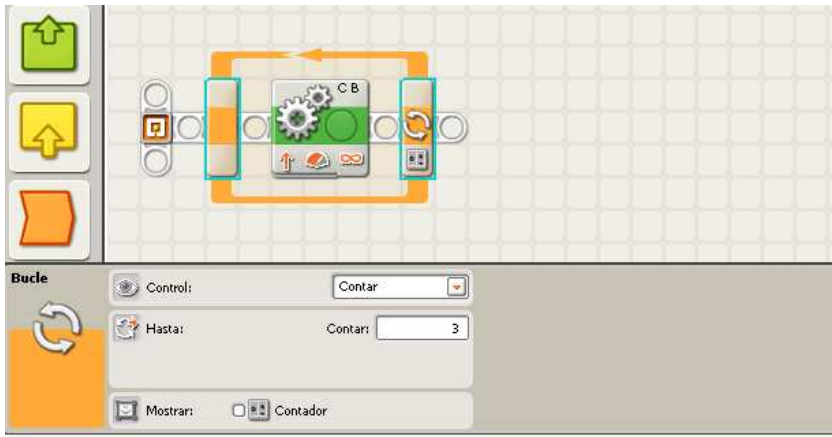
Robot, metre, peu de rei, marcador, cronòmetre

Objectiu

1. Es tracta de determinar la velocitat del robot per recorre la distància de 2 metres amb diferents potències. L'experiment s'ha de repetir per determinar la precisió.

Distància: 1m → Graus: 4122	Flotació		Fre	
	v 1 (m/s)	v 2 (m/s)	v 1 (m/s)	v 2 (m/s)
100	0.393	0.384	0.366	0.379
75	0.258	0.263	0.276	0.267
50	0.174	0.175	0.182	0.177
25	0.077	0.077	0.077	0.076

2. Quan la duració es posa a il·limitat s'ha d'incloure un bucle. El farem 3 cops.



Distància Il·limitat	Flotació contar=3	
Potència	v 1 (m/s)	v 2 (m/s)
100	0.228	0.282
75	0.129	0.195
50	0.1	0.1
25	0.038	0.030

Conclusions de les pràctiques :

Pràctica 1 → En aquesta pràctica s'hi pot observar que, tot i variar la potència dels motors, quan s'ordena al robot que freni al finalitzar, aquest s'atura gairebé al valor exacte que havíem establert ($2061^{\circ} \rightarrow 1 \text{ m}$). En flotació, en canvi, la distància real que recorre varia una mica més, degut a que en acabar les rotacions el robot segueix movent-se gràcies a la inèrcia que porta.

També podem veure que si el que fem és donar-li durada il·limitada als robots, tot i que no funcionen indefinidament, ara la distància que recorren sí que varia segons la potència subministrada.

2- En la segona pràctica s'observa que les velocitats, ja sigui aturant el robot amb una frenada o amb flotació, no varien gaire.

La velocitat màxima que ha assolit el robot és de 0.393 m/s .

Si convertim el resultat, amb la màxima potència (100) pot anar a :

$$W = V/R = 0.393/0.028 = 13.93 \text{ rad/s}$$

$$13.93 \text{ rad/s} \cdot 1 \text{ volta}/2\pi \text{ rad} \cdot 60 \text{ s}/1 \text{ min} = 133.01 \text{ rpm}$$

Per altra banda els resultants que he obtingut fent l'exercici amb la durada il·limitada als motors i repetint l'acció 3 vegades mitjançant un bucle, varien molt les velocitats abans trobades, fet que crec que és degut a un error humà ja com que el temps que estava en moviment era curt l'error pot haver sigut causat a l'hora de prémer el cronòmetre per iniciar i aturar el temps.

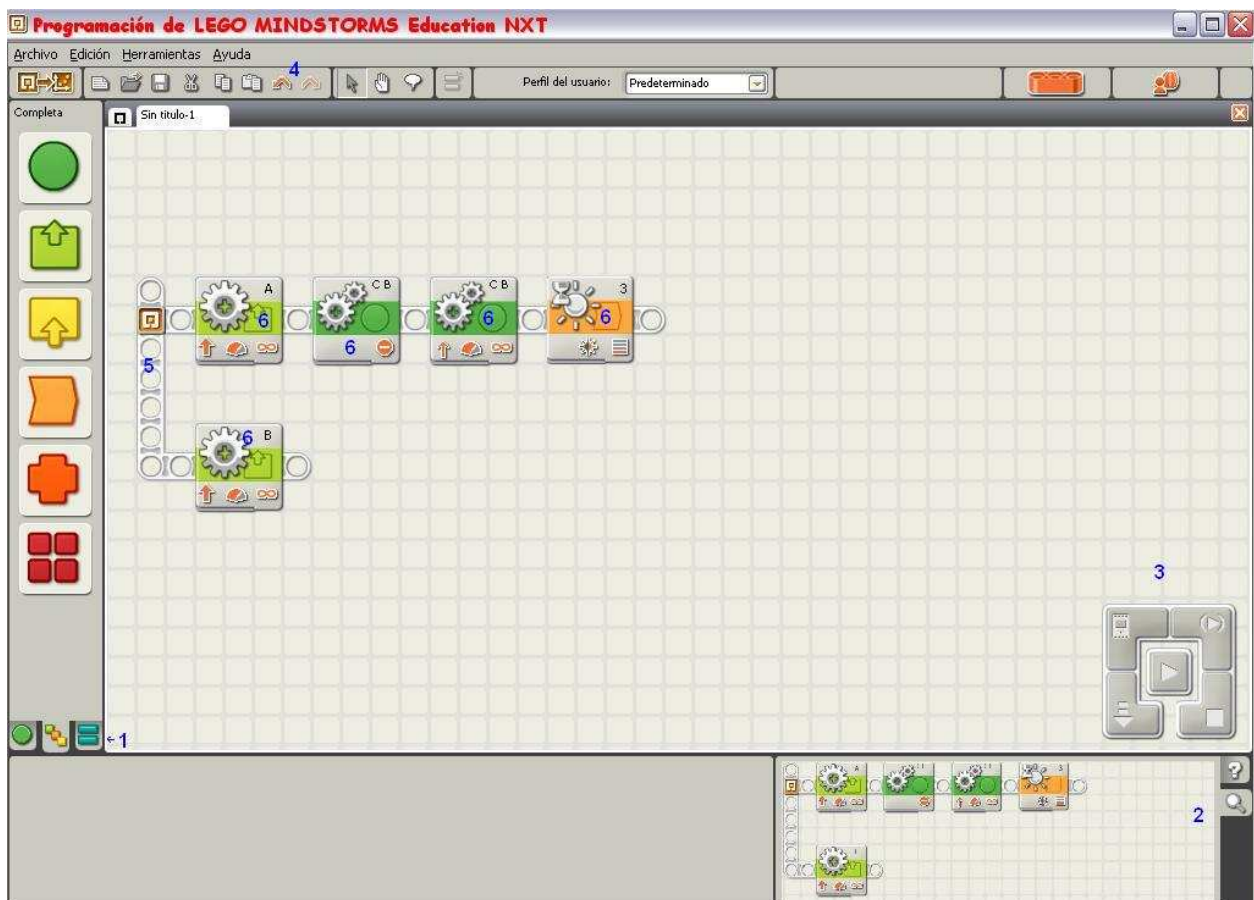
També s'hi pot comprovar que al duplicar, la potència del motor, la velocitat no es duplica i l'increment d'aquesta tampoc és proporcional.

4.4 Software Lego Mindstorms

Lego mindstorms és el programari , proporcionat per LEGO® que em permetrà programar el robot de forma "senzilla". La programació es fa mitjançant un software basat en LabView que utilitza llenguatge G (Gràfic), la seva característica principal és que el seu us és senzill tant com per professionals com per persones aficionades al tema.

El sistema de programació funciona per blocs. Cada bloc correspon a una acció. Els blocs es col·loquen arrossegant-los des de les paletes (on hi ha els diferents blocs) fins la biga de seqüència. Un cop el programa és descarregat al robot aquest executarà les accions dels blocs una darrera l'altre, segons l'ordre establert. També hi ha l'opció que executi diverses accions simultàniament.

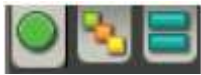
Seguidament explicaré com funcionen els elements més importants per al nostre projecte i alguns coneixements bàsics per saber programar amb el software de Lego® Mindstorms.



Finestra del programari

En la imatge anterior de la finestra que es veu al obrir el programa i iniciar un nou projecte observem:

1- Permet seleccionar entre les diferents paletes de les que disposem. A les paletes estan disposats els diferents blocs.



El primer símbol de l'esquerra pertany a la paleta comú. En aquesta paleta hi ha els blocs ja preparats per a ser arrossegats cap a la biga de seqüència.

Si seleccionem el símbol del mig, se'ns obrirà la paleta completa on hi ha totes les opcions que podem incloure als nostres projectes. Dins aquesta paleta hi ha 6 grups d'icones, blocs



comuns, blocs d'acció, blocs de sensor, blocs de flux, blocs de dades i blocs avançats.

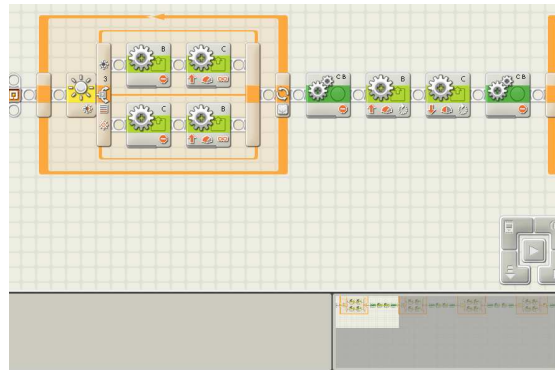
Situant-nos amb el cursor sobre cada icona s'obren els blocs que hi ha dins de cada grup i que són els que arrossegarem a la biga de seqüència.

També podem observar el nom del projecte, tot i que amb qualsevol altra paleta també és possible.




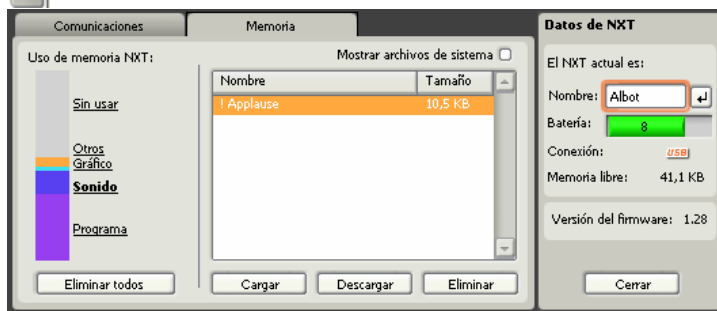
Selecciónant el tercer botó, arribem a la paleta personalitzada on hi ha blocs que pot generar l'usuari (Mis bloques) i que poden ser arrossegats, generant al programa un conjunt d'accions predefinides. Ex: Si creem un bloc personalitzat on dins hi hagi els blocs necessaris perquè el robot avanci 1m i giri 180° sobre el seu eix, cada cop que inserim aquest bloc a la biga, el robot sabrà que ha d'executar totes les accions que hem definit, sense tenir que posar diferents blocs. S'utilitza per compactar el programa.

2- Si seleccionem en aquest requadre la lupa, podem desplaçar-nos per tot el programa que hem creat; l'àrea seleccionada per la lupa és la que es mostrarà augmentada a la pantalla.



3- En aquest requadre disposem de varies icones per descarregar el programa al robot i per veure informació del robot:

 → Aquesta icona ens serveix per veure informació del robot connectat.



Podem observar:

El nom del robot


La bateria restant del robot


La versió del firmware


La memòria lliure


L'ús de la memòria

El tipus de connexió (USB en aquest cas)

 → Ens permet descarregar i executar (sense haver de confirmar-ne l'execució al robot) l'àrea seleccionada del programa.

 → Fent clic aquí es descarregarà tot el programa al robot, posteriorment per executar-lo haurem de dir-li 'Run' dins el menú del Bloc.

 → L'icona d'aturada ens permet aturar qualsevol acció, ja sigui descarrega de dades al robot, o si aquest esta connectat aturar el programa que s'estigui executant.

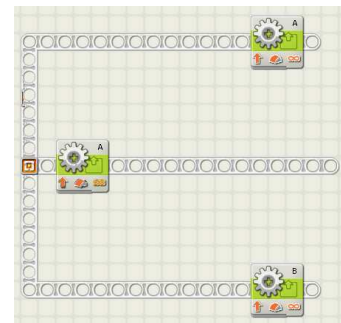
 → El botó central del requadre serveix per descarregar i executar al robot el projecte sencer que tenim en l'aplicació.



Amb aquesta barra , d'esquerra a dreta, podem: crear un projecte nou, obrir-ne un de guardat, desar el projecte actual, retallar una part seleccionada del programa que estem creant, copiar una part seleccionada del programa que estem creant, enganxar la part copiada o retallada, desfer i refer; també podem canviar el cursor, per editar els blocs, moure la pantalla que observem per poder veure altres parts del projecte i escriure anotacions al projecte.

5- Biga de seqüència.

La biga de seqüència serveix per connectar tots els blocs i tots ells han d'estar connectats amb una de les tres (Nº màxim de desdoblaments possibles) per poder funcionar. Si un bloc no hi esta connectat aquesta acció no es dura a terme per part del robot. Es poden fer tres accions simultànies que corresponen al màxim de desdoblaments de la biga.



6- Diversos blocs que es poden arrossegar des de les paletes fins a la biga de seqüència. Cada bloc te assignada una funció específica, fer moure un motor, els dos, visualitzar quelcom a la pantalla, fer lectures amb els sensors, comparar variables...



A continuació explicaré els blocs que he utilitzat per a fer les unitats didàctiques i que utilitzaré per a fer l'últim projecte. Com que he fet molts exercicis per practicar, he fet servir la paleta completa, ja que disposa de més blocs.

4.4.1. Blocs comuns

Dins de la paleta completa, quan ens situem sobre la primera fila (common blocs) se'ns desplega una pestanya on hi ha varis blocs.



El més important és el 'move' que serveix per fer moure, una roda o les dues, així com aturar-les.



Si seleccionem el bloc 'move' ens apareix a la part inferior de la finestra el següent quadre que ens permet modificar diverses opcions del bloc.

Move ↗



- La opció port ens serà útil per determinar en quin port tenim connectats els motors i quins volem que es moguin.
- La direcció ens permet que les rodes girin endavant, enrere o s'aturin
- El volant permet fer que alhora que les rodes giren el robot vagi girant, reduint la velocitat d'una o altra roda.
- La potència és on definim la velocitat amb que es mouran les rodes, podem augmentar-la o disminuir-la.
- La duració la farem servir per ordenar al robot quin nombre de rotacions, graus o temps volem que girin els motors, o si volem que ho facin il·limitadament (tot i que no restaran girant per sempre).

- La acció següent permetrà determinar si, un cop acabat d'executar el bloc volem que el robot freni en sec o no (s'aturarà com a conseqüència del fregament després d'uns cm).

En els common blocs també disposem d'altres blocs com són:

Gravar / reproduir :



Aquest bloc serveix per guardar una sèrie de moviments. Es a dir, primer el carregues al robot amb *Record*, l'engegues i mous els motors durant el temps que hi has posat. En acabar el tornes a carregar al robot amb el mateix nom però amb *Play* i al engegar el programa el robot repetirà, mes o menys, els moviments que has fet. S'ha de tenir en compte al fer servir aquest bloc que no es completament precís.

També hi ha els blocs, so, visualitzar, espera, bucle i bifurcació, que apareixen classificats també en altres grups de blocs i s'explicaran aleshores.

4.4.2 Blocs d'acció



Bloc motor:







Amb aquest bloc es fa girar una única roda, però es pot fer de forma més complexa que amb el bloc move.




D'aquest bloc podem modificar:

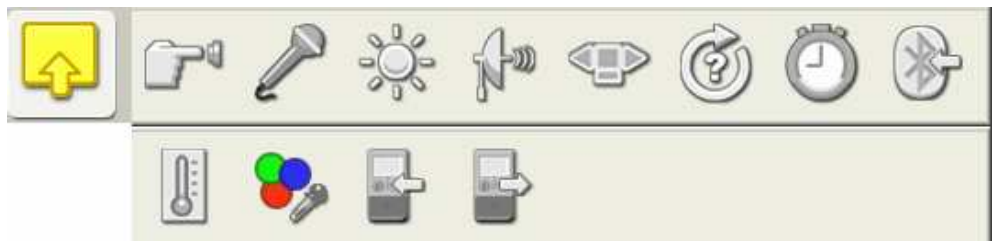
- El port on anirà connectat el servomotor
- La direcció en la que girarà aquest, endavant, endarrere o parar de girar.
- L'acció ens permet controlar la velocitat del motor que pot ser constant, sempre igual; Rampa de pujada, augmenta progressivament fins arribar a la màxima; Rampa avall, disminueix progressivament fins arribar a aturar-se.
- La potencia del motor (0 a 100).
- Control: Al activar la opció el robot variarà la seva velocitat en funció del terreny per mantenir-la estable i evitar pèrdues.
- La duració és on s'escull quant de temps giraran els motors. Aquesta quantitat de temps pot ser indefinidament (il·limitat) o es pot donar en segons, en graus, o en rotacions.
- En l'apartat esperar, si activem la casella, el robot esperarà a que finalitzi d'efectuar-se aquesta acció, en aquest cas girar les rodes, abans d'iniciar la següent. Es pot desactivar quan volem posem 2 blocs motor per a dues rodes diferents ja que si el segon no ha d'esperar a que acabi el primer els dos aniran alhora.
- L'opció acció següent permetrà determinar si, un cop acabat d'executar el bloc volem que el robot freni en sec o no (s'atura com a conseqüència del fregament després d'uns cm)

Dins dels blocs d'acció també trobem:





 <p>El bloc so, fa que el robot emeti un conjunt de sons o produeixi una nota musical durant un cert temps.</p>	 <p>El bloc visualitzador fa que aparegui un text o imatge a la pantalla del NXT</p>
 <p>Aquest bloc permet enviar un missatge (un text, un numero o un verdader o un fals) d'un NXT a un altre via bluetooth.</p>	 <p>Aquest bloc permet encendre una làmpada, connectada a un dels 4 ports amb un cable especial, i que emeti llum vermella, blava o verda.</p>






	<p>El bloc làmpada permet encendre una llum, amb intensitat variable, connectada a un dels ports per un cable especial.</p>
---	---

4.4.3. Blocs de sensors




Introduint aquest blocs al programari podem obtenir lectures dels sensors, fer accions segons si un sensor detecta una determinada variable, etc.

	<p>El sensor de contacte: Al introduir-lo al projecte podem definir el port on està connectat i com ha d'estar el sensor per considerar-lo activat; al xocar, al estar pressionat o al no estar-ho.</p>		<p>El bloc del sensor de so ens permet comparar el valor de so ambiental amb un altre definit (de 0 a 100) perquè, si el primer és més gran >, o més petit<, que el segon, considerem el sensor activat.</p>
	<p>El bloc botons del NXT ens deixa escollir quin botó s'ha de prémer perquè s'activi el bloc i com ha d'estar aquest; premut, no premut o quan es fan les dues accions seguides (bumped).</p>		<p>El sensor de rotació activa el bloc quan les rodes hagin girat un cert nombre de rotacions o graus preestablert. Es pot seleccionar el port del motor a controlar i també reiniciar aquest sensor per esborrar rotacions acumulades anteriorment.</p>

 <p>El bloc temporitzador fa una lectura del temps que porta funcionant el programa i el compara amb un valor establert per activar-se o no. També es pot reiniciar per no acumular temps anteriors, si el que volem es controlar la durada d'una acció determinada.</p>	 <p>En aquest bloc escull quin tipus de missatge t'enviaran. Pots escollir de text, un numero o un verdader o fals (Logic). Un cop fet això, al costat de comparar amb : s'escriu un text un nombre o un verdader o un fals. Si allò que tu hem escrit és igual al del missatge que reps, llavors s'activa el bloc.</p>
 <p>El bloc Sensor de color permet detectar diferents colors i mesura de la intensitat de la llum (ambient o reflectida). També és capaç d'enviar un senyal lògic (veritable o fals), el color actual o la lectura de llum.</p>	 <p>Els blocs entrada i sortida del mesurador de consum permeten saber i llegir els valors proporcionats pel mesurador de consum, tant d'entrada com de sortida, que poden ser: Voltatge(V), Potencia(W), Energia(J).</p>
 <p>El bloc del sensor de temperatura (tot i que no disposem d'aquest sensor) ens permetria saber la temperatura de l'ambient mesurat en Celsius o Fahrenheit.</p>	

Dins aquesta pestanya també trobem dos dels blocs dels sensors que utilitzaré per a la versió final del programa.

 El bloc del sensor de llum és el que principalment utilitzarem en el cas que ens ocupa. Aquest sensor detecta la llum ambiental i si activem la casella de generar llum el que capta és la seva pròpia llum reflexada.



El bloc permet seleccionar en quin port esta connectat el sensor i amb quin valor s'ha de comparar la lectura perquè es consideri

activat o no. Pot donar senyal d'intensitat lluminosa o un valor lògic.



El bloc del sensor ultrasònic ens serà útil per determinar a quina distància es troba un objecte. Té un abast màxim d'uns 250 cm (1 polzada).



El bloc permet seleccionar el port del sensor, la distància amb que compararem la lectura per activar, o no el sensor, i les unitats de mesura.

4.4.4 Blocs de flux



Els blocs que ens apareixen al posar el cursor sobre la icona a la paleta són:

Bloc d'espera: La seva funció és fer que el robot resti sense fer cap acció fins que es compleixi una condició de temps o sensorial. El bloc permet controlar el port mitjançant el qual rebrà la informació, que esperi un temps determinat o a l'activació d'un sensor i comparar el valor de lectura del sensor amb un altre que definim.



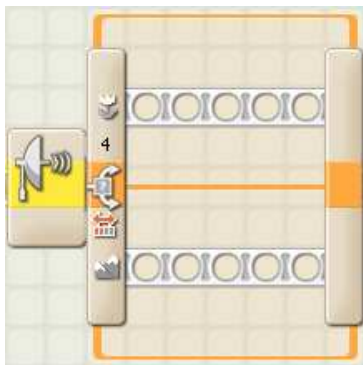
Bloc d'espera (pot variar la icona segons el sensor o el temps)



Aquest bloc el que fa és finalitzar el programa així com els motors, làmpades, sons... No te opcions que es puguin modificar.



Un cop inserit el bucle a la biga de seqüència podem introduir dins seu altres blocs. El que fa és repetir tot el procés contingut en el seu interior un cop ha acabat de fer l'últim d'aquests blocs. Es pot establir una durada il·limitada, o que el programa deixi d'executar el bucle i passi a l'acció següent després de que s'activi un sensor, hagi passat un temps que determinarem o s'hagi repetit el bucle un cert nombre de vegades, també definit per nosaltres.



El bloc de la bifurcació ens permet realitzar dues accions diferents si es compleix, o no, una condició que nosaltres haurem de predefinir. Pot funcionar amb un valor lògic (vertader o fals) o mitjançant la lectura d'un sensor que compara amb la condició que nosaltres hem definit.

Si la condició es compleix executa els blocs d'una de les dues bigues de seqüència que ha creat i si no els blocs de l'altre biga. Si la condició es compara amb la lectura d'un sensor ens és útil per definir el port en que aquest esta connectat i altres característiques dels sensors de manera similar com si fos un bloc sensor.

4.4.5 Blocs de dades

Si situem el cursor sobre la icona de blocs de dades se'ns obrirà una pestanya com la següent, on podem distingir diversos blocs.



	<p>El bloc lògic ens serveix per realitzar una operació lògica que podem definir (AND, OR, NOT, XOR) amb les variables d'entrada que es connectaran mitjançant un cable de dades pel costat esquerra d'A i B. D'aquest símbol sortirà el resultat de la operació.</p>		<p>El bloc matemàtic permet fer operacions com sumar, restar, multiplicar, dividir, fer arrels o valors absoluts. Com en l'anterior, els valors d'entrada arriben pel costat esquerra d'A i B, i la solució surt pel costat .</p>
	<p>El bloc de comparació ens determinarà si un numero de l'entrada A és més gran, més petit o igual que el de l'entrada B. Ens donarà un resultat lògic (vertader o fals). D'aquest símbol sortirà el resultat de la operació.</p>		<p>Aquest bloc és el bloc d'interval que ens permet saber si un número (introduït per un cable de dades a través d'A, B o #). La senyal lògica de sortida s'envia a través de la sortida de .</p>
	<p>El bloc aleatori genera un numero aleatori entre un màxim i un mínim (pot anar de 0 a 100) definits per l'usuari, i l'envia per al sortida de #</p>		<p>Bloc Variable, amb el que podem escriure o llegir una variable lògica, numèrica o textual.</p>
	<p>El bloc de constant ens permet definir una constant lògica, numèrica o textual.</p>		

4.4.6 Blocs avançats.

Per últim, a la paleta comú, torbem els blocs avançats, que com el seu nom indica son per usuaris amb més coneixement del programari i que jo pràcticament no he fet servir, per tant explicaré el que jo he utilitzat en alguna de les unitats didàctiques i anomenaré la resta.



Bloc 'número a text' que permet convertir un número (com el valor d'un sensor) a un text que podrà , per exemple, ser visualitzat a la pantalla del NXT. El valor numèric s'introdueix a través del # i el text resultant surt amb un cable de dades des de **T**.

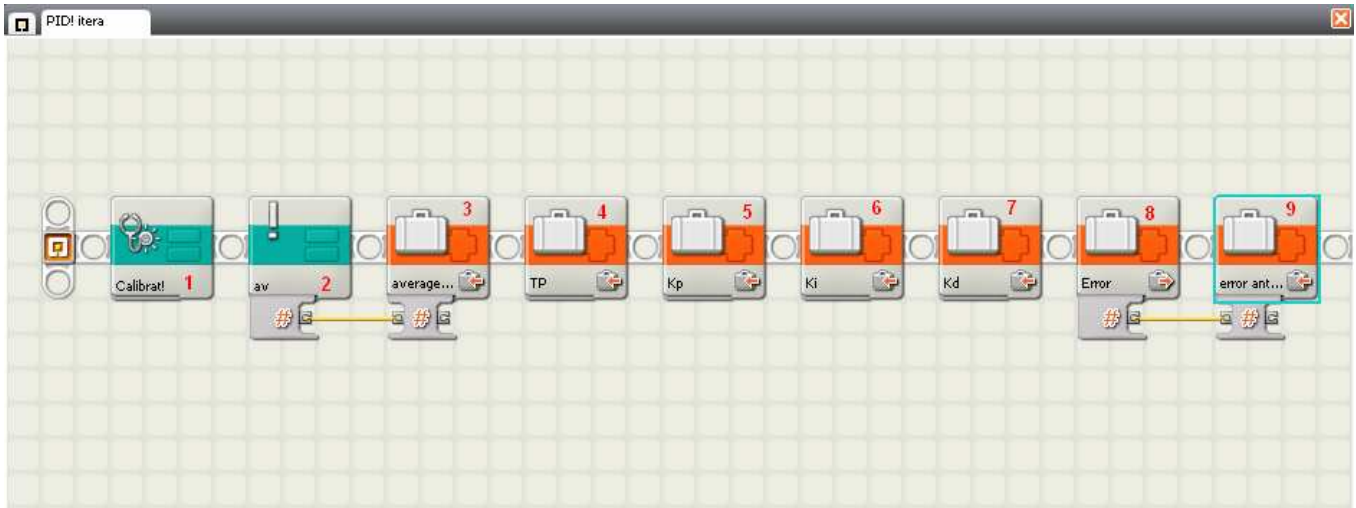


El bloc de calibrar ens permetrà establir un valor màxim (100%) i un de mínim (0%) per als sensors de llum i so.

La resta de blocs, d'esquerra a dreta i ometent els ja esmentats, són: text, mantenir, accés a arxius, reiniciar motors, iniciar datalog, aturar datalog i, a la part inferior, connexió Bluetooth, respectivament.

4.5 Programa final amb control PID

El meu objectiu era programar el nostre robot perquè fos capaç de seguir la línia, sense gaires oscil·lacions i amb una velocitat raonablement ràpida. Per això, mitjançant el programari de Lego Mindstorms he creat aquest programa que explicaré a continuació. Imatge de la primera part del programa principal



1- El primer bloc correspon a un bloc personalitzat anomenat ' Calibrat!' i, que com el seu propi nom indica, ens servirà per a calibrar el sensors.

2- El segon bloc, també és personalitzat i és el que ens proporciona el valor mitjà (average) entre el valor màxim i mínim (Blanc i negre) de lectura dels sensor.

3- Al tercer bloc, un bloc de variable, s'introdueix el valor de l'average o target power, a partir de la sortida del segon bloc, on és calculat, per utilitzar-lo posteriorment.

4- En aquest bloc podem definir el target power, o valor amb el qual el error hauria de ser 0, i els dos motors funcionarien a la mateixa velocitat.

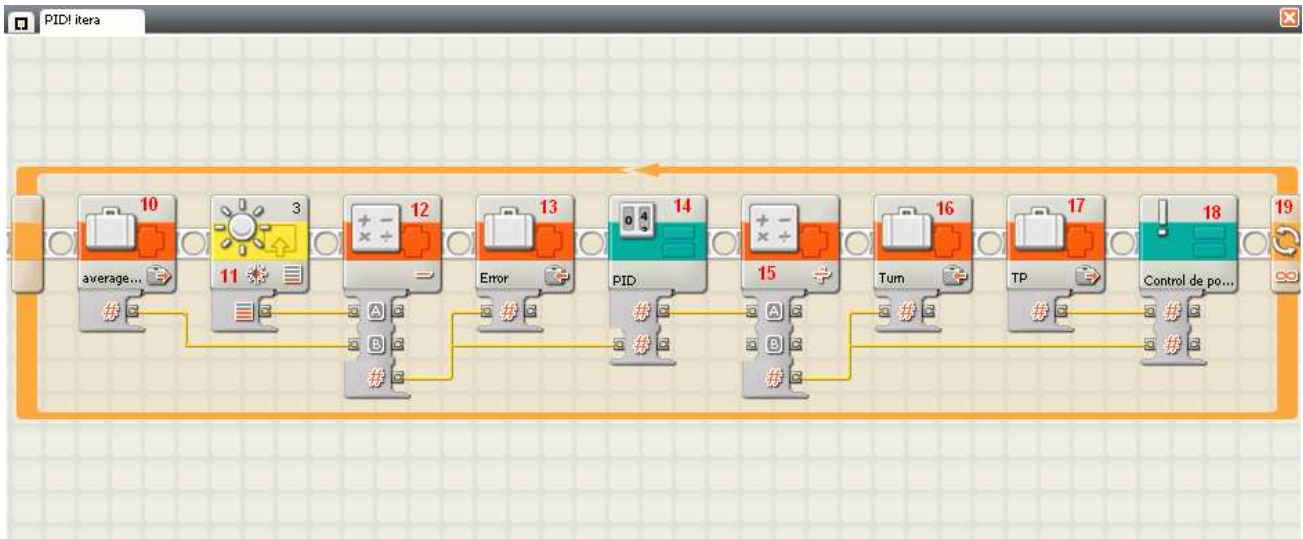
5- En el següent bloc podem definir la Kp (constant de proporcionalitat) per a fer-la servir més tard.

6- En aquest bloc podem definir la Ki (constant integral) per a fer-la servir en un altre bloc posteriorment.

7- En el següent bloc podem definir la Kd (constant derivativa) per a utilitzar-la més tard.

8- El bloc d'aquesta variable el que fa és llegir l'error que calcularem a continuació. El fet que encara no l'haguem calculat no influeix, un cop se sàpiga s'escriurà en aquesta variable automàticament.

9- En aquest pas el que fem és guardar l'error que tenim com a error anterior que farem servir a l'hora de calcular l'acció derivativa.



Segona part del programa principal

10-11-12→ Al bloc average(10) que havíem escrit abans (3) i que recordem que és la consigna, li restem(12) la lectura del sensor (11) per obtenir l'error. Del bloc matemàtic surt un cable de dades que porta aquest error.

13- Introduïm a una nova variable l'error calculat amb els tres blocs anteriors mitjançant el cable de dades.

14- L'error també és introduït en el bloc personalitzar PID, on es calcularà el valor de l'acció proporcional, integral i derivativa.

15- El valor de K_p , K_d i K_i està augmentat 100 vegades al definir-lo, perquè el programari no deixa posar una K de 1'25. Així doncs, posem una K , per exemple, de 125 i el resultat de l'acció PID el dividim entre 100.

16- Al valor de l'acció PID l'anomenarem turn (gir) i el que aconseguim aplicant aquest valor al control de potència de les rodes, és convertir el valor d'aquesta acció a un valor de potència d'una o altra roda per girar.

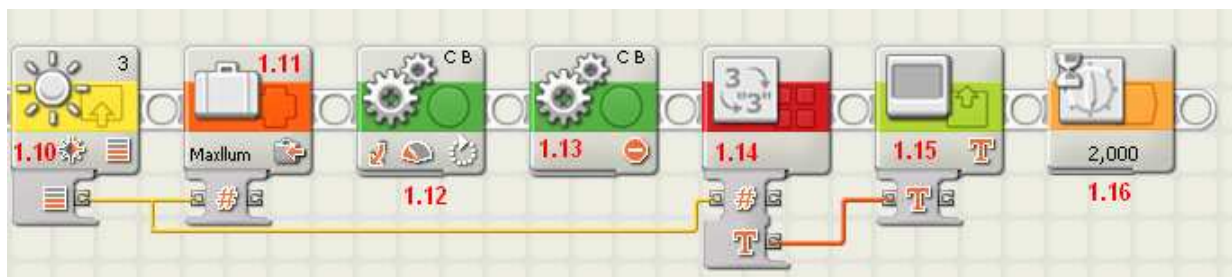
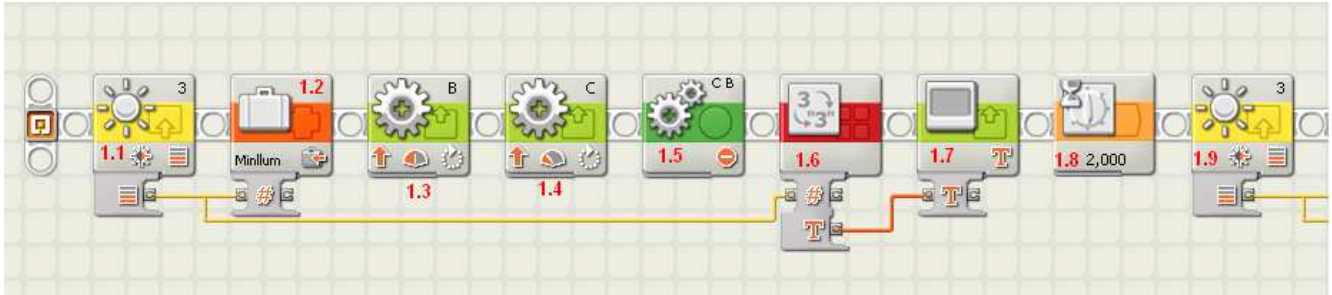
17- Bloc de la variable T_p que hem definit amb anterioritat.

18- Bloc personalitzat de control de potència on entren els valors T_p i $Turn$.

19- Tots aquest blocs de la segona part estan inserits dins d'un bucle perquè es repeteixin aquestes accions indefinidament.

A continuació explicaré els 4 blocs personalitzats dels que he parlat:

Bloc ' Calibrat! ' (1), que té dues parts:



1.1 – El robot, amb el sensor de llum col·locat a sobre la línia negra, pren nota del valor de llum reflexada.

1.2 – El valor refelxat a la línia negra és el valor mínim, i s'emmagatzema al bloc Minllum.

1.3 – 1.4 → Aquests blocs mouen el robot endavant (per arribar a una superfície que considerarem blanca) i alhora girant, això s'aconsegueix donant més potència a una roda que a l'altre.

1.5 - Aturem el robot

1.6 - Convertim el valor numèric del valor Minllum a text perquè el visualitzador pugui mostrar-lo en pantalla.

1.7 - El bloc visualitzador mostra el valor mínim en pantalla.

1.8 - Perquè el que visualitzem s'estigui una estona en pantalla per poder veure el valor desitjat definim que esperi (el mostri) duran 2 segons amb aquest bloc.

1.9 – 1.10 (El mateix bloc) Aquí el sensor de llum capta el valor de la superfície 'blanca' , ja que el robot s'hi havia desplaçat.

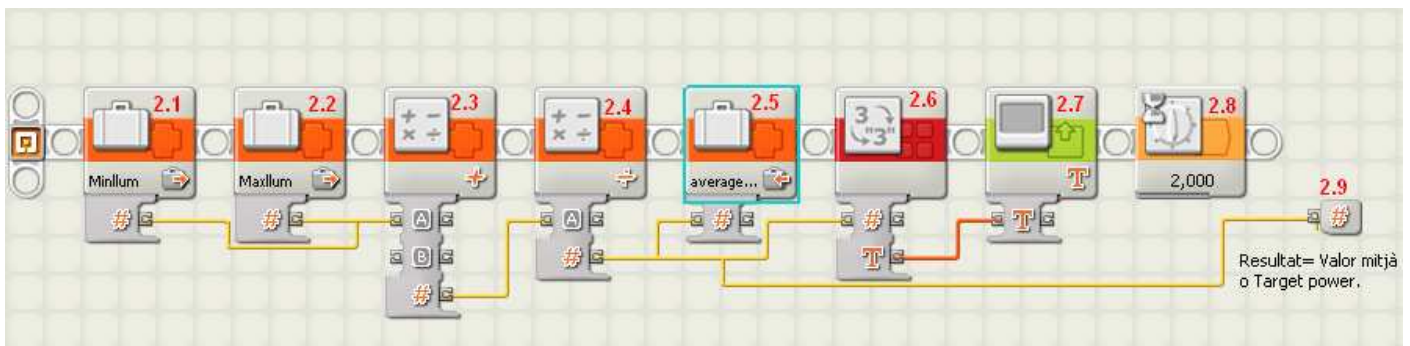
1.11 – El valor de la superfície blanca serà el valor màxim de llum i estarà al bloc Maxllum

1.12 – El robot tornarà a la posició des de la qual començarà a seguir la línia, gràcies a l'acció del bloc move.

1.13 – Aturem el robot

1.14 - 1.15 - 1.16 → Amb aquest blocs el que farem és mostrar en pantalla durant 2 segons el valor de reflex màxim de llum, de la mateixa manera que ho feiem per visualitzar el valor mínim.

Bloc av (2):

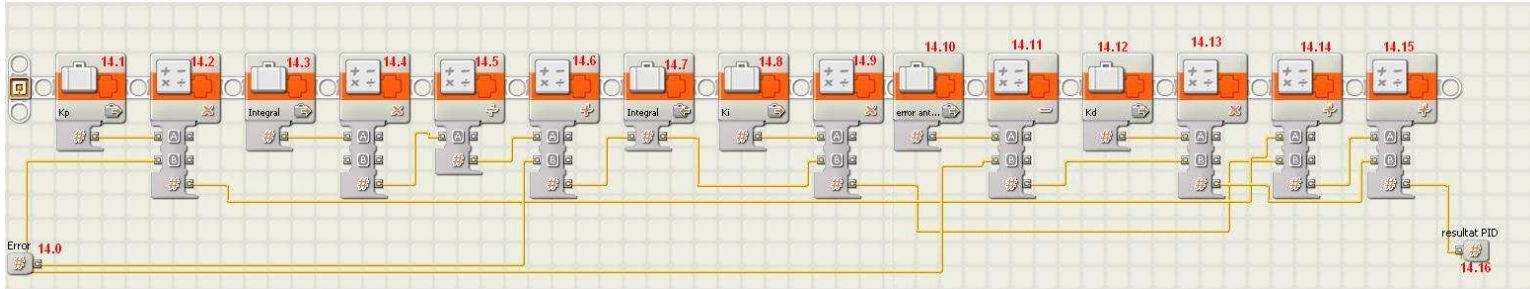


2.1 a 2.5 → El que fem amb aquests blocs és fer la mitjana entre el valor màxim de llum i el mínim. Primer se sumen el valor Maxllum(2.2) i Minllum (2.1) i es divideixen entre 2 (2.4). Aquest valor l'anomenarem average i l'introduïm al bloc average(2.5)

2.6 – 2.7 – 2.8 → Amb aquest blocs visualitzarem a la pantalla el valor average o consigna durant 2 segons, com ja hem fet en passos previs.

2.9 El valor de la consigna s'introdueix en aquest 'bloc', que ens servirà per, després en el programa principal, utilitzar-lo traient-lo del bloc personalitzat mitjançant un cable de dades.

Bloc PID (14) on es realitzarà l'acció proporcional, integral i derivada.



14.0 – El valor de l'error que ha estat introduït mitjançant un cable de dades al bloc personalitzat ubicat al programa principal.

14.1- 14.2 → Acció proporcional: Multipliquem, amb el bloc matemàtic (14.2) el valor de la K_p (14.1), definit al programa principal, per l'error (14.0).

14.3 – 14.7 → Definim el valor de l'integral que multiplicarem per K_i : Sumem (14.6) el valor d'integral (14.3) més el valor de l'error (14.0). El valor de l'integral, com hem dit, és la suma dels errors. Com que amb el nostre programari al sumar aquest error contínuament obtindríem un valor de l'integral molt gran, que no aniria bé al nostre seguidor, el que fem és multiplicar el valor integral per 1 (14.4) i dividir-lo per 3 (14.5), així cada cop reduïm el valor integral a 1/3 de l'anterior. El valor integral es va guardant en la variable integral (14.7).

14.8- Variable amb el valor de K_i

14.9- Acció Integral: Es multiplica el valor integral per la K_i per obtenir la sortida de la funció integral.

14.10- Variable amb l'error anterior, calculat prèviament al programa principal.

14.11- A l'error que acabem de tenir un instant abans li restem l'actual, per saber com varia aquest error. El que fem és estimar que l'error següent de manera que l'error futur serà igual a la variació de l'error més l'error actual. Així ens hi podem avançar per corregir-lo.

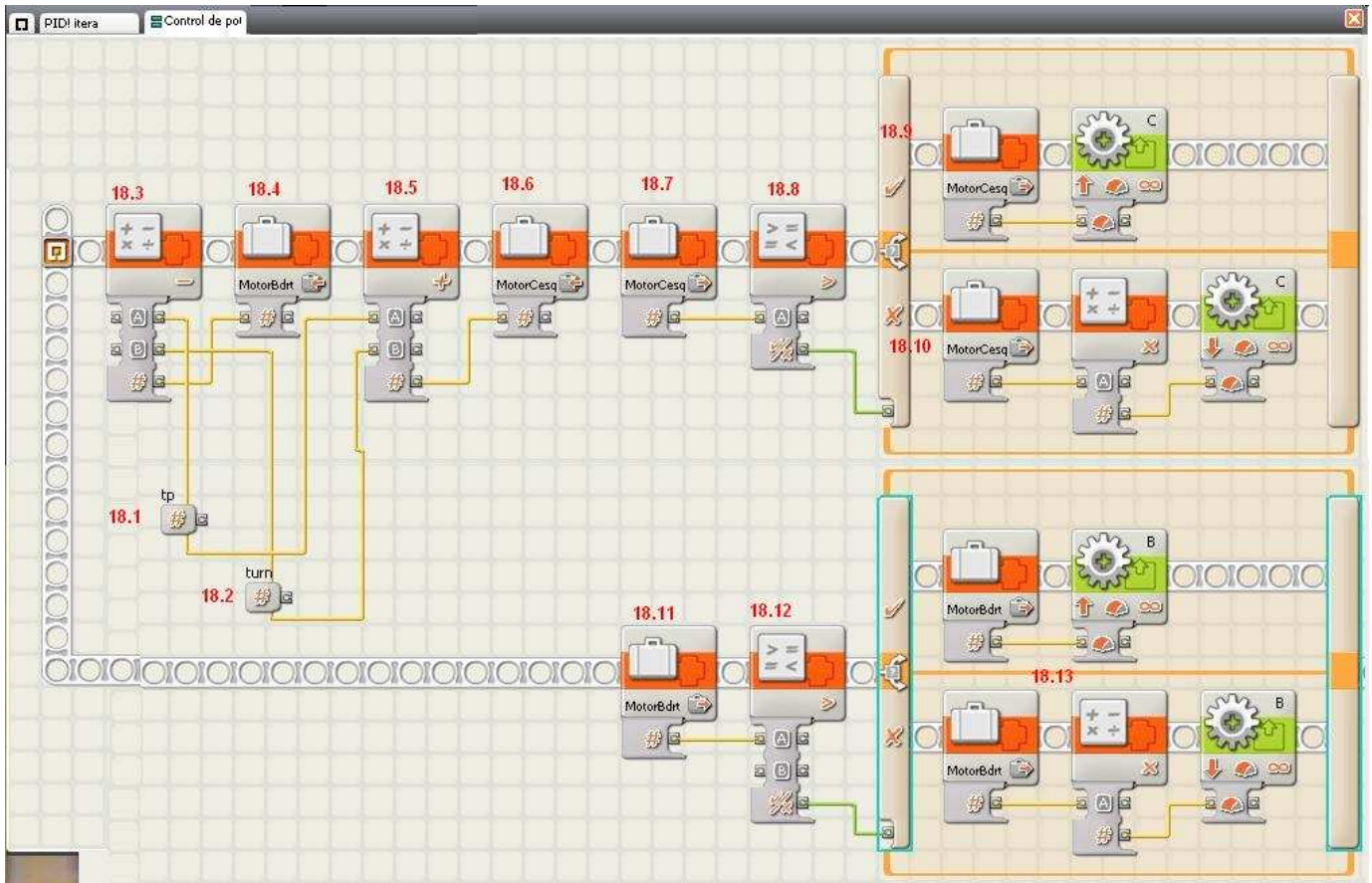
14.12 Constant derivada, definida amb anterioritat.

14.13 Acció derivada: Multipliquem el valor de l'error futur estimat per la K_d per intentar solucionar-lo.

14.14 – 14.15 → Sumem l'acció $P + I + D$, S'ha de fer en dos blocs ja que cada bloc només de dues entrades, la sortida del primer ($P+I$) la sumem, en el segon, a l'acció D .

14.16 – Obtenim de la suma de PID el resultat d'aquesta acció.

Bloc Control de potència (18.)



18.1 -Tp: Target Power, potència amb la qual els dos motors van a la mateixa velocitat si l'error és 0. Prové de les entrades que el bloc personalitzat té al programa principal.

18.2 - Turn(gir): Valor obtingut de la conversió feta al programa principal de l'acció del PID al valor de gir o variació de potència que tindran els motors.

18.3 – 18.4 → Restem $T_p - \text{Turn}(18.3)$ i guardem el valor a (18.4) que és el bloc de la variable de potència del motor B, que en el cas que ens ocupa és el de l'esquerra.

18.5 – 18.6 → Sumem $T_p + \text{Turn}(18.5)$ i guardem el valor a (18.6) que és el bloc de la variable de potència del motor B, que en el cas que ens ocupa és el de l'esquerra.

Aquest dos passos anteriors els tenen una explicació que és senzilla d'entendre amb un cas pràctic: Si definim el $T_p=50$, i el valor de Turn, en

aquest cas i depenent de l'error, és 10, això voldrà dir que el robot es troba sobre la superfície blanca i ha de girar més el motor C que el B per tant la roda C es mourà a la velocitat de 60 i la B a la de 40, corregint la posició. Si per contra el valor calculat pel gir és -10 (el robot és sobre la línia) la potència de C seria 40 i la de B, $60 [50 - (-10) = 60]$, girar el robot cap 'a fora de la línia'.

Tot i semblar una contradicció, no ho és. El que fem, al disposar només d'un sensor de llum és seguir la vora de la línia, perquè si seguíssim la línia en si no sabríem per quin costat (dret o esquerra) el robot se surt, i per tant, cap a quin costat girar.

18.8 a 18.13 → En aquest blocs el que fem és corregir un petit error que es podria donar al fer les operacions $T_p + Turn$ i $T_p - Turn$. Podria donar-se el cas que el valor de Turn fos més gran que T_p i el valor sortís negatiu. Centrem-nos en la bifurcació de la part superior (18.8 a 18.10), ja que la inferior és el mateix però amb l'altre motor. Comparem si el valor final de potència (18.8) és més gran que 0 (positiu) amb el bloc comparador (18.9). Si ho és, (\checkmark) el valor de la potència és traspasat directament al bloc motor (18.10). Si no ho és (X), multipliquem per -1, és a dir, canviem de signe, i donem la potència al bloc motor, però seleccionem la direcció cap enrere, ja que el valor inicial era negatiu.

Amb la bifurcació inferior (18.11 a 18.13) repetim el mateix procés però ara amb el motor B.

Per acabar, la versió final de programa ja està dissenyada, ara només falta definir els valors de les T_p , K_p , K_d i K_i . Això ho farem mitjançant la iteració, és a dir, anar provant valors fins que aconseguim el resultat esperat. Els valors no seran sempre aleatoris, començarem amb un valor de T_p baix, i unes constants també baixes i segons el comportament del robot (oscil·lacions, error estacionari, pèrdua de la línia...) les variarem d'acord amb els coneixements obtinguts a la part teòrica sobre els sistemes de control PID.

El robot no seguirà la línia sobre un fons blanc, com primerament estava previst, ja que la banda de mesura era molt ampla i feia oscil·lar molt l'autòmat. La línia, feta amb cinta negra, està enganxada sobre un tauler de fusta de color marró clar que ens serà suficient perquè el robot sabés en tot moment si estava o no sobre la línia a seguir. El material pensat inicialment era plàstic blanc no i anava tant bé com el material que finalment usarem.

5.Conclusió

En aquesta , la part final del treball, analitzaré el resultat que he obtingut amb el treball. Primer parlaré sobre el compliment, o incompliment, dels objectius plantejats inicialment, a continuació faré una reflexió sobre el que he après fent un treball tan elaborat com aquest ho és i per últim conclouré el treball amb els agraïments a les persones que han fet possible el treball.

En iniciar el treball de recerca, amb el tutor i la 'cotutora', de la que ara parlaré, ens vam plantejar una sèrie d'objectius a complir:

El primer era aconseguir fer un robot que seguís una línia, i s'ha de dir que l'he aconpleert amb escreixos ja que a més de fer que la seguís he aconseguit també el quart objectiu; en aquest em proposava conèixer diferents tipus de control explicats a la part teòrica i implementats a la pràctica, i aconseguir que el seguidor fos regulat amb l'acció del PID. Cal dir, però, que tot i reduir molt les oscil·lacions del seguidor inicial proposat a les unitats didàctiques inicials i augmentar la velocitat, m'hagués agradat que el robot anés a major velocitat reduint d'igual manera les oscil·lacions. Quan em vaig plantejar el treball també volia dominar perfectament el programari, però no sabia de la complexitat d'aquest. He arribat a un bon nivell de coneixement, però sempre es pot augmentar. El tercer i cinquè objectiu els he anat desenvolupant al llarg de tot el treball explicant les parts de les que consta el robot i posant exemples clars i entenedors respectivament. Un dels objectius, que era que el robot s'aturés a l'arribar a un obstacle, tot i no estar marcat com a tal, que em vaig plantejar a mesura que experimentava amb el sensors, també l'he pogut aconpleir tot i la dificultat de combinar això amb el control de l'acció PID.

Respecte els coneixements obtinguts amb el treball, he de dir, que al començar a fer un treball tan extens com aquest, un no es pot imaginar la complexitat que tindrà finalment i la gran quantitat de coses que es poden arribar a aprendre.

En el cas que ens ocupa el treball m'ha servit per a aprendre molt sobre els sistemes de control, sobre els diferents elements dels robots, els tipus que hi ha i a fer una resolució analítica dels problemes que anaven sorgint a l'hora de programar i els errors que tenia el robot en el seu camí seguint la línia. Per acabar, el treball de recerca també m'ha ajudat a comprendre la dificultat de programar un robot perquè prengui decisions i es mogui autònomament, la gran quantitat d'hores que requereix la programació quan es vol que el resultat sigui el més òptim possible i que, tot i que, la teoria digui una cosa, pot ser que a la pràctica el que s'ha de fer sigui diferent, com en el cas del nostre fons del tauler que finalment no ha estat blanc.

Finalment volia agrair l'ajut als familiars , amics que en algun moment m'han proporcionat ajuda per a fer el tauler, millorar parts del treball, etc.; i a l'institut, per l'esforç econòmic a l'hora de proporcionar-me el pack de Lego® .

També volia agrair immensament la col·laboració de quatre professors. Primerament al Sr. ***** , un professor de l'IES Mollet, pels llibres sobre robòtica que em va cedir per a fer el treball. A la ***** que m'ha proporcionat molta informació, i durant els inicis del treball quan no hi havia un tutor decidit em va començar a guiar en aquesta recerca. L'***** , que és el meu tutor del treball de recerca i que m'ha guiat en tot el procés i corregit, per millorar, fins a l'últim moment el treball i que , gràcies a ell, aquest treball és el que és. I per concloure, volia fer una menció especial a la ***** , qui podríem considerar 'co-tutora', que em va donar la gran oportunitat de fer el treball sobre el Lego®, m'ha aconsellat en tot moment sobre la programació del robot, ha corregit algunes parts del treball i ha estat ajudant-nos fins que el seu estat d'embaràs li ha permès. Així doncs els meus més sincers agraiments a aquestes persones ja que sense la seva ajuda aquest treball no hauria estat possible.

6. Bibliografia

Pàgines web consultades:

http://www.inpharmix.com/jps/PID_Controller_For_Lego_Mindstorms_Robots.html

<http://www.superrobotica.com/VisualSC.htm>

<http://www.razorrobotics.com/>

http://freakshare.net/files/ih4gdzl2/Fndamentos_de_Rbotica.rar_char.rar.html

<http://translate.google.es/translate?hl=es&langpair=en%7Ces&u=http://www.gizmag.com/go/7872/>

<http://www.robotic-lab.com/blog/tutoriales/robot-sigue-lineas/>

<http://librosdigitalesfree.blogspot.com/search?q=robotica>

http://store.pleoworld.com/merchant.mvc?Screen=PROD&Store_Code=PleoWorld&Product_Code=INL-INNVOLABS-

<http://www.robotic-lab.com/foros/>

<http://www.pleoworld.com/Home.aspx>

<http://es.wikipedia.org/wiki/Rob%C3%B3tica>

<http://www.allonrobots.com/index.html>

http://www.edu365.cat/imagina/robots/lego_nxt/index.htm

<http://www.bogatech.org/>

<http://roboteros.com/>

<http://mapardo.wordpress.com/2009/06/10/introduccion-a-la-robotica/>

<http://www.edu365.cat/batxillerat/recerca/comfer/index2.htm#questions>

<http://chikiotaku.com/proyecto-aiko-una-verdadera-persecom>

<http://www.youtube.com/watch?v=W5q1IIAOZqs>

[http://onlinelibrary.wiley.com/journal/10.1002/\(ISSN\)1556-4967](http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1556-4967)

<http://www.iso.org/iso/home.html>

http://es.wikipedia.org/wiki/Organizaci%C3%B3n_Internacional_para_la_Estandarizaci%C3%B3n

<http://www.robotica-personal.es/2010/08/nueva-caja-de-recursos-817-piezas-lego.html>

http://en.wikipedia.org/wiki/Industrial_robots

<http://www.uvic.cat/estudi/61>

<http://www.concursroboticajet.etseiat.upc.edu/>
<http://www.buildaplane.org/start-a-project.php>
http://www.xanascap.cat/butlleti_xanascap/octubre_2009/index.htm
<http://www.robolot.org/>
<http://www.enginycat.cat/>
<http://www.discoverengineering.org/>
<http://www.pathwaystotechnology.org/>
http://usuarios.multimania.es/automatica/temas/tema2/pags/la_lc/lalc.htm
http://www.xtec.cat/~jsolson7/eso/cv_auto/automatismes.htm
<http://www.xtec.es/~rcasella/projecte/pagina5.htm>
<http://www.msebilbao.com/tienda/index.php>
<http://www.mitecnologico.com/Main/Se%F1alesAnalogicasYDigitales>
http://es.wikipedia.org/wiki/Se%C3%B1al_anal%C3%B3gica
http://es.wikipedia.org/wiki/Se%C3%B1al_digital
http://www.kalipedia.com/tecnologia/tema/senales-analogicas-senales-digitales.html?x=20070822klpingtcn_137.Kes
<http://www.scribd.com/doc/17702998/Tipos-de-Controladores>
<http://www.biblioteca.upibi.ipn.mx/Archivos/Material%20Didactico/Apuntes%20para%20la%20asignatura%20de%20instrumentaci%C3%B3n%20y%20control/cap4.pdf>
<http://es.wikipedia.org/wiki/Microcontrolador>
<http://www.iberobotics.com/?q=node/11>
<http://www.amazings.com/ciencia/noticias/291010b.html>
<http://www.scitech-news.com/2010/10/robots-could-improve-everyday-life-at.html>
<http://www.news.cornell.edu/stories/Sept10/PersonalRobotics.html>
<http://econfuture.wordpress.com/>
<http://www.amazon.com/gp/product/1448659817>
<http://www.lacatedralonline.es/innova/caleidoscopio/6711-gobernanza-etica-de-robots>
http://cgi-pe.com/descarga/Ing.%20Gustavo%20Kato_Robotica%20CGI.pdf
<http://www.offnews.info/verArticulo.php?contenidoID=22425>
<http://www.econectados.com/2010/08/la-etica-de-los-robots-en-el-ejercito/>
<http://www.mundo-geo.es/ciencia>

http://www.kalipedia.com/tecnologia/tema/cronologia-inteligencia-artificial-ia.html?x=20070821klpinginf_81.Kes&ap=1

<http://www.diccionari.cat/lexicx.jsp?GECART=0103102>

http://www.slideshare.net/ptah_enki/perturbaciones-470726

<http://www.superrobotica.com/Sensores.htm>

http://books.google.es/books?id=iaeP4wq-G14C&pg=PA59&lpg=PA59&dq=ELEMENTS+TEORIA+DE+CONTROL&source=bl&ots=1pojIPMppB&sig=8TKcG7CaIE_r_RfgcYZI4OW9k8&hl=ca&ei=Yk3mTO6dKNKG5AavgNn4Ag&sa=X&oi=book_result&ct=result&resnum=1&ved=0CCMQ6AEwAA#v=onepage&q&f=false

http://es.wikipedia.org/wiki/Ingenier%C3%ADa_autom%C3%A1tica

<http://upcommons.upc.edu/pfc/bitstream/2099.1/3393/5/41056-5.pdf>

http://books.google.es/books?id=Vbd11zVvk_QC&printsec=frontcover&dq=sistemas+de+control&hl=ca&ei=oaL_TP6eDIzrOca7rc0M&sa=X&oi=book_result&ct=result&resnum=1&ved=0CCsQ6AEwAA#v=onepage&q=sistemas%20de%20control&f=false

<http://www.youtube.com/watch?v=4CZTWvQxyE8>

http://www.youtube.com/watch?v=DMF_O9-BEsY

<http://jrvanwhy.homelinux.net/linefollow.htm>

http://es.wikipedia.org/wiki/Proporcional_integral

<http://es.oocities.com/jeeesusmeeerino/procesos/teoriapid/teoriapid.html>

<http://mundobot.com/tecnica/pid/sppid.htm>

http://www.robotica-urjc.es/index.php/QUAV:_Control_de_estabilizacion

<http://www.udb.edu.sv/Academia/Laboratorios/electronica/Sistemas%20de%20Control%20Automatico/guia5SCA.pdf>

<http://www.ing.uchile.cl/~leherrer/IQ57A/apunte2.htm>

<http://cazaresrobotica.blogspot.com/2010/10/caracteristicas-de-un-sistema-de.html>

http://www.automatas.org/hardware/teoria_pid.htm

<http://www.eng.newcastle.edu.au/~jhb519/teaching/caut1/Apuntes/PID.pdf>

http://es.wikipedia.org/wiki/Lego_Mindstorms

<http://mindstorms.lego.com/en-us/history/default.aspx>

http://robots-argentina.com.ar/Sensores_general.htm#microf

http://www.kalipedia.com/tecnologia/tema/componentes-robot.html?x=20070821klpinginf_92.Kes&ap=1
<http://www.philohome.com/sensors/legorot.htm>
<http://proton.ucting.udg.mx/dpto/tesis/quetzal/TES35-01.html>
<http://rbtntxt.blogspot.com/2009/01/bluetooth.html>
<http://blog.electricbricks.com/2010/05/sigue-lineas-labview-maquinas-estado/>
<http://en.wikipedia.org/wiki/Micro>
<http://www.offnews.info/verArticulo.php?contenidoID=22425>
http://cgi-pe.com/descarga/Ing.%20Gustavo%20Kato_Robotica%20CGI.pdf
<http://www.cannic.uab.es/aplicacions/programaArgo/Argo.htm>
<http://www.aprenderobotica.com/page/que-es-un-microcontrolador>
http://es.wikipedia.org/wiki/Sistema_de_control
<http://www.aprenderobotica.com/group/eslaprimeravez>
<http://www.cannic.uab.es/docencia/cannicframedocencia.htm>
<http://www.cannic.uab.es/docencia/RobotsInstituts/RobotsInstituts.htm>
<http://cucabot.netfirms.com/cero.html>
<http://www.cannic.uab.es/docencia/RobotMobil/RoboticaMobil.htm>
http://www.sapiensman.com/control_automatico/
<http://upcommons.upc.edu/pfc/bitstream/2099.1/3330/5/34059-5.pdf>
http://ca.wikipedia.org/wiki/Sistema_de_control
<http://www.monografias.com/trabajos6/sicox/sicox.shtml#tec>
<http://ceres.ugr.es/~angel/docencia/robin/Apuntes/Tema%205.pdf>

Llibres consultats:

Iovine, Jhon (2004), [Pic Robotics- Mc Graw Hill](#) – Tab robotics ISBN 0-07-137324-1

Predko, Myke (2003), [Robot DNA series- Mc Graw Hill- Tab robotics](#) ISBN 0-07-140851-7 (P/N 141098-8)

Williams, Karl (2004), [Build Your Own Humanoid Robots](#), - McGraw Hill- Tab Robotics- ISBN 0-07-142274-9

lovine , Jhon (1998), [Robots Androids and Animatrons](#) , -Mc Graw Hill- Tab electronics
ISBN 0-07-032804-8

Grau, Jordi; Manrique, Emili; Rodriguez, Pedro; (2008), [Tecnologia 4 ESO](#) – La galera- Text- ISBN 978-84-412-1511-5

José A. Martínez Gea, Manuel Ángel Villar Martínez, José A. artínez Asís, Francisco Manjón-Cabeza Muñoz. [Tecnología. Volumen IV](#). Editorial MAD. 2003. ISBN: 84-8311-945-5

Joan Joseph, Jaume Garravé, Francesc Garófano, Francesc Vila. [Tecnologia industrial, Segon de batxillerat](#). Serie Fluvià, Mc Graw Hill. ISBN: 978-84-481-6157-6

7. Annex

La part del treball annex està en un CD entregat conjuntament amb el treball de recerca. Aquesta part inclou:

- Document explicatiu de l'annex, anomenat Llegeix-me.
- Les fotos del robot (procés de muntatge, organització del pack Lego, etc.)
- Vídeos :
 - Vídeo de la demostració del funcionament del robot que vaig fer en la exposició del meu treball de recerca.
 - Vídeos comparatius del robot seguint la línia amb control Tot o Res i amb el control PID.