

Tecnologia electrònica i robòtica

III. Annex.

Guia ràpida per programar un robot LEGO Mindstorms

0. Índex

1. Introducció
2. L'entorn de programació
3. La programació
 - 3.1. Motors
 - 3.2. Sensors i temporitzadors
 - 3.2.1. Temporitzador (*timer*)
 - 3.2.2. Sensor de contacte (*touch sensor*)
 - 3.2.3. Sensor de color o llum (*color sensor*)
 - 3.2.4. Sensor de soroll (*sound sensor*)
 - 3.2.5. Sensor d'ultrasons (*ultrasonic sensor*)
 - 3.3. Bucles (loop), condicions (if, interruptor o switch) i Espera
 - 3.3.1. Bucles (*loop*)
 - 3.3.2. Condicions (if, interruptor o Switch)
 - 3.3.3. Espera

1. Introducció

Els LEGO Mindstorms són robots programables. La programació és realitzada mitjançant blocs. Això vol dir que no necessitem conèixer cap llenguatge de programació per poder utilitzar-lo. Només necessitem saber com funcionen.

El robot obeeix les ordres que li donem. Aquestes poden ser aleatòries o respondre a algun estímul (inputs). Les accions que poden realitzar els LEGO Mindstorms són:

- a) Moure el motor B i/o C que farà girar la roda dreta i/o esquerra respectivament.
- b) Moure el motor A que obrirà o tancarà les pinces.
- c) Emetre un so a determinar entre uns preestablerts.
- d) Visualitzar una figura o lletres al display.

Pel que fa als inputs del robot, tenim:

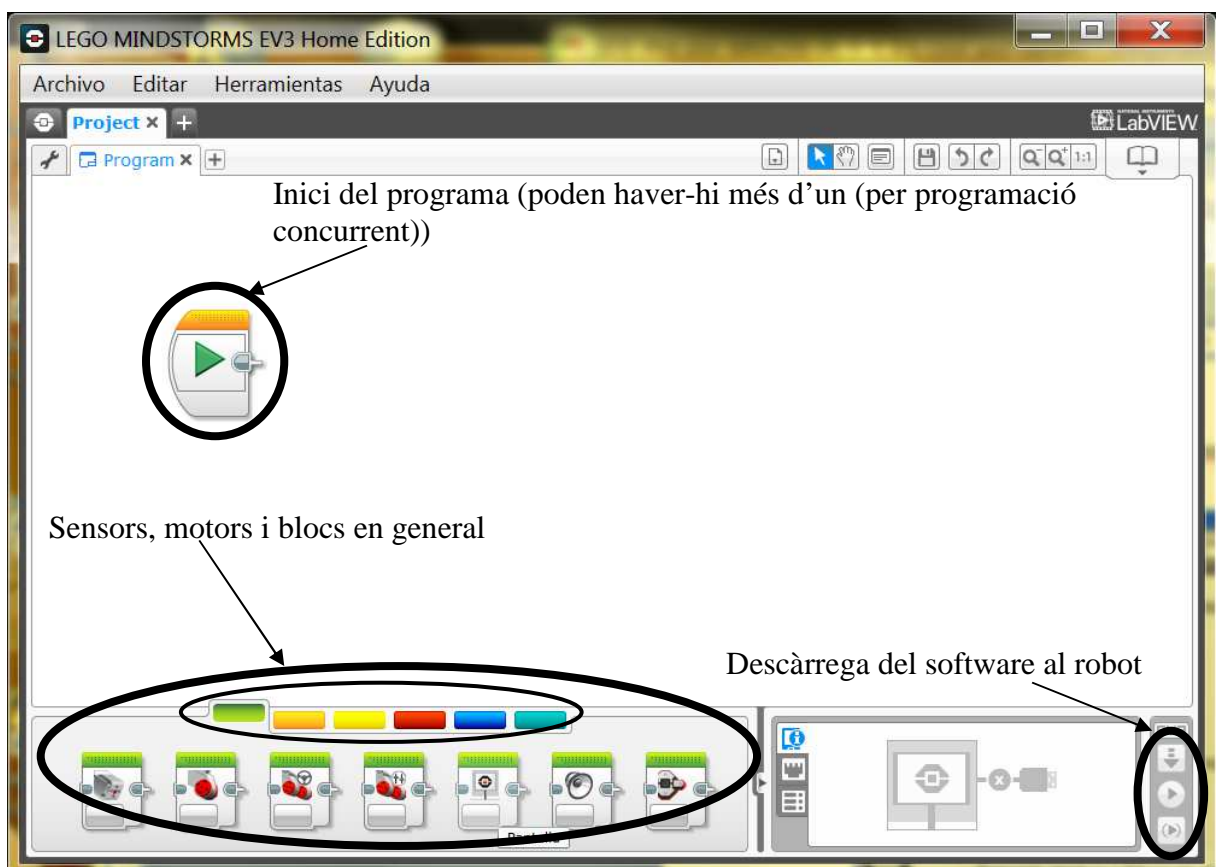


- a) Sensor d'ultrasons: Ens diu si l'objecte que està situat davant del sensor està més a prop o més lluny d'una distància determinada per nosaltres.
- b) Sensor de llum (sensor de color): Està enfocant al terra. La informació que ens proporciona té a veure amb el color sobre el que es troba (bàsicament discernirem entre negre o blanc).
- c) Sensor de soroll: Aquest sensor ens dóna la informació sobre el nivell d'intensitat sonora en decibels (dB).
- d) Sensor de contacte: Ens diu si algun objecte està prement l'interruptor del sensor.

El robot es pot programar de manera seqüencial (una única operació en cada moment) o concurrent (poden executar-se diferents programes a l'hora (en paral·lel), podent realitzar així diverses accions simultànies).

2. L'entorn de programació

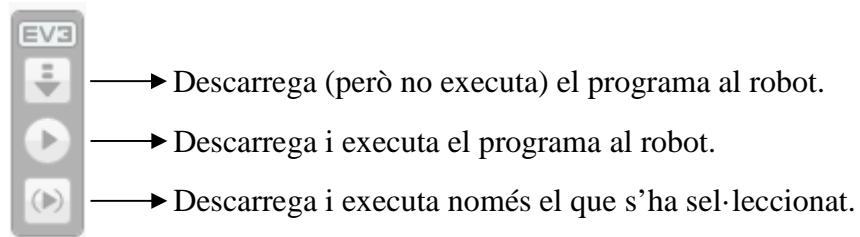
L'entorn de treball del programa Mindstorms és el següent:



Prement els botons d'abaix a l'esquerra, anem arrossegant els diferents blocs al lloc del programa que desitgem.



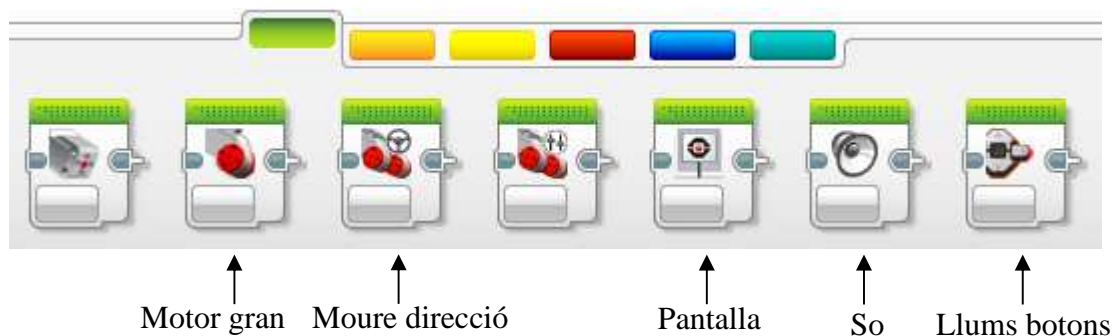
Per a poder provar el software en el robot, tenim els botons d'abaix a la dreta:



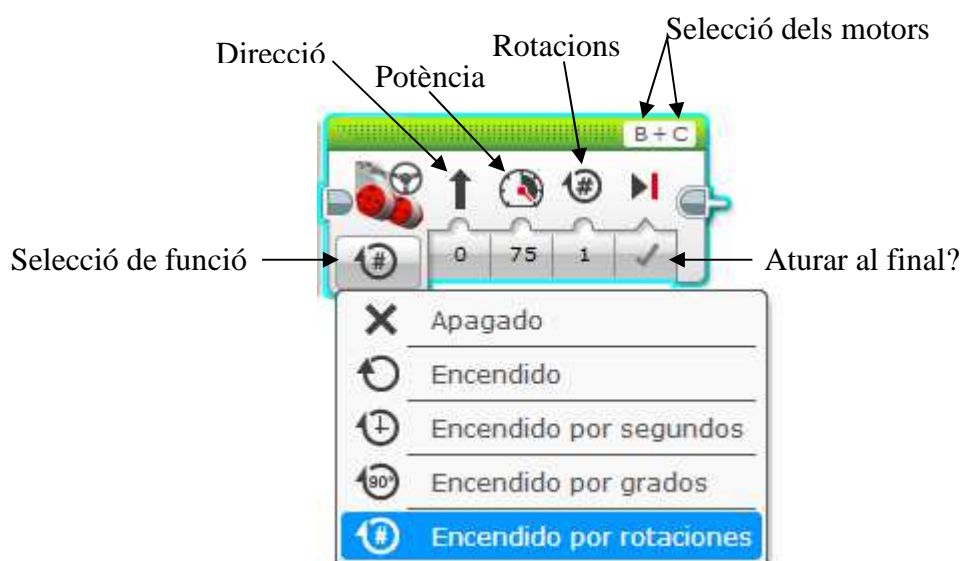
3. La programació

3.1. Motors

Les funcions corresponents als motors els podem trobar al primer menú (“acció”, de color verd):



Pel que fa als motors, bàsicament utilitzarem la funció “Moure direcció” per controlar les dues rodes del robot i “Motor gran” per controlar les pinces del robot (un únic motor). Els diferents paràmetres de la funció “Moure direcció” són:

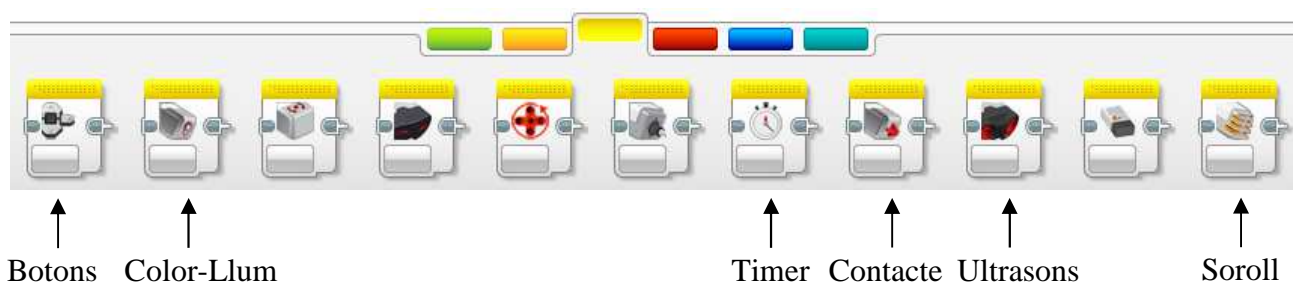


Es poden seleccionar diferents funcions i paràmetres:

- Primer seleccionarem els motors sobre els quals actuarà aquesta instrucció (les dues rodes del robot són les B i C).
- Seguidament triarem una funció. Encès i apagat són per mantenir en moviment continu o per parar-ho respectivament. Les altres opcions permeten controlar quant giren les rodes, ja sigui per temps, per graus o per nombre de rotacions.
- Respecte la Direcció (valors de -100 fins a 100): 0 per anar en línia recta, >0 per anar cap a la dreta i <0 per anar cap a l'esquerra.
- La potència (valors de -100 fins a 100) està relacionada aproximadament amb la velocitat de gir de les rodes. El signe també controla la direcció de gir de les rodes.
- Rotacions serveix per indicar els graus, el nombre de rotacions o els segons (depenent de la funció seleccionada) de la rotació.
- La funció d'aturar serveix per indicar si, al final de l'acció programada, els motors es pararan bruscament o si "es deixaran anar" per inèrcia del moviment.

3.2. Sensors i temporitzadors

Els sensors els podem seleccionar a la pestanya groga:



També hem de dir que podem fer servir el bloc Esperar (a la pestanya taronja) (apartat 3.3.3) per tal d'esperar (el programa no farà res) fins que un dels sensors detecti algun condició concreta (per exemple fins que la intensitat de llum no superi un cert valor).



3.2.1. Temporitzador (*Timer*)

Una de les opcions és la del Temporitzador. Té diverses funcions. Amb Reiniciar, es pot posar a 0 el valor del rellotge intern del mòdul Ev3. Amb Mesura, podem obtenir el valor del timer, i amb Comparar es pot comparar el valor del timer amb un valor especificat (per veure, per exemple, si ha superat un cert temps).



3.2.2. Sensor de contacte (*touch sensor*)

Amb la funció Mesura obtenim l'estat actual del sensor de contacte (Presionat, no presionat o en contacte). Amb la funció Comparar, a més de l'estat actual, podem obtenir un senyal lògic segons si es troba en un estat específic.



També podem seleccionar el port al que està connectat el sensor en qüestió a la part de dalt a la dreta. Això no cal tocar-ho, doncs els valors que apareixen per defecte concorden amb la configuració real en el robot.



3.2.3. Sensor de color o llum (*color sensor*)

Aquest sensor es troba situat a sota el robot. El nostre robot avalua la lluminositat del terra que està trepitjant. Podem programar el sensor perquè actuï quan la lluminositat estigui per sota o per sobre d'un cert nivell.



Es pot programar per sobre d'un valor de lluminositat o per sota. En el cas que veiem a l'anterior imatge, està programat per actuar quan el nivell de lluminositat estigui per sota de 50 sobre 100 (0 seria negre, 100 seria blanc).

A l'igual que l'anterior bloc, aquest també permet calibrar els nivells 0 (mínim) i 100 (màxim).

3.2.4. Sensor de soroll (*sound sensor*)

Amb el sensor de so o soroll s'obté els valors d'intensitat acústica en decibels, o es compara amb algun nivell que podem fixar nosaltres, de forma semblant a altres funcions de sensors.





El nivell de soroll va de 0 en cas d'absència d'aquest i fins a 100.

També es pot calibrar.

3.2.5. Sensor d'ultrasons (*ultrasonic sensor*)

Avalua la distància dels objectes que té davant. Podem mesurar fins a dos metres i mig, i discernir entre si l'objecte està per sobre o per sota d'un cert valor.

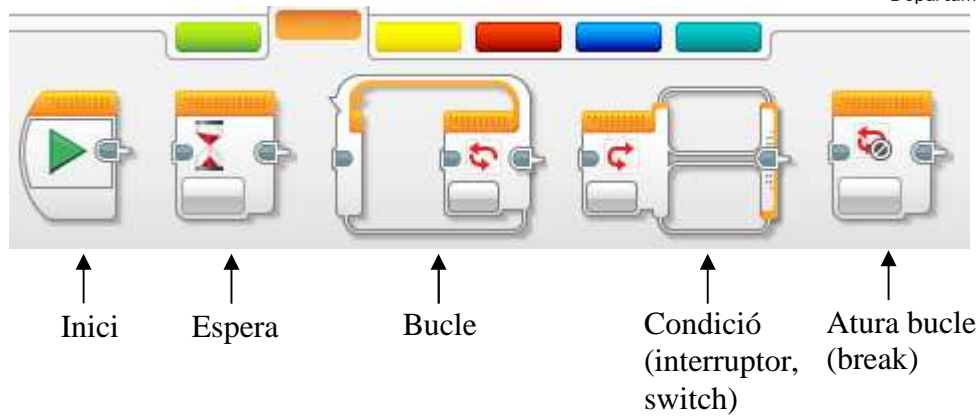


La lectura de la distància es pot donar en polzades o en centímetres.

3.3. Bucles (*loop*), condicions (*if*, *interruptor o switch*) i Espera

Els podem trobar a la pestanya taronja (control de fluxe):



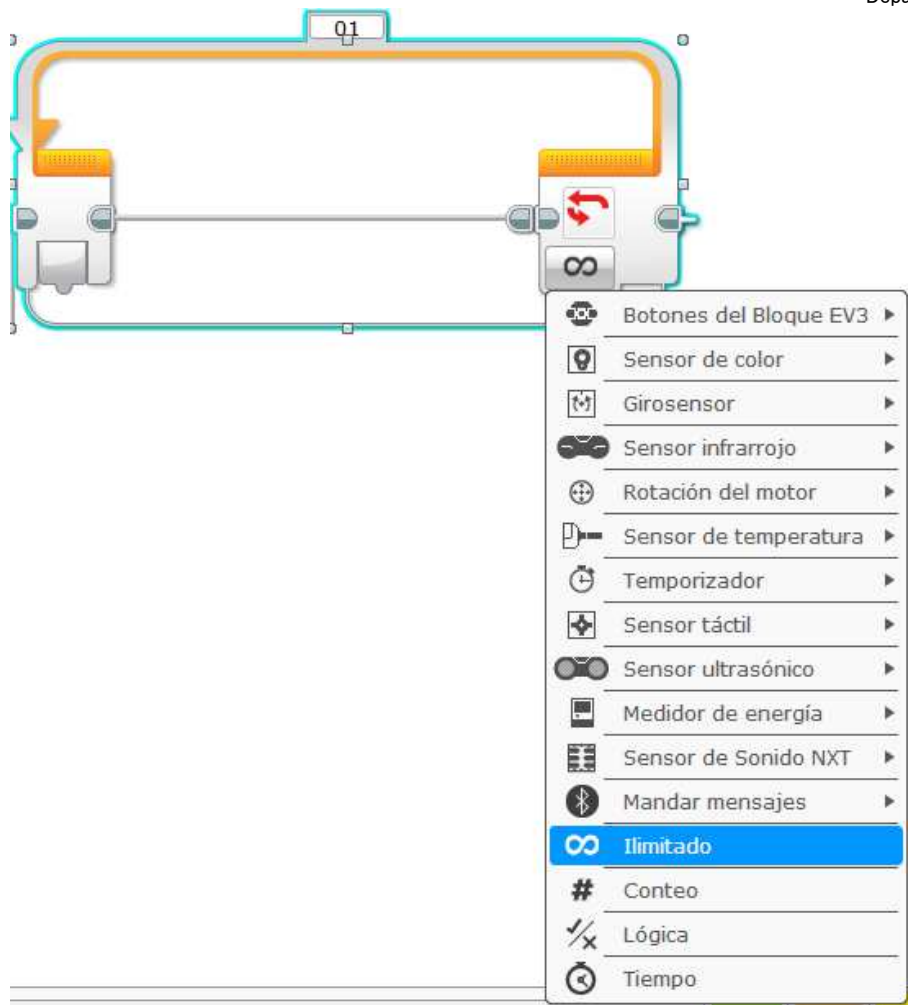


En programació és habitual l'ús de bucles i condicions. El primer ens permet realitzar una part del programa tantes vegades com vulguem o fins que canviï alguna condició. El segon ens permet realitzar unes operacions o unes altres en funció d'alguns paràmetres. Òbviament ambdós es poden combinar. Poden haver-hi bucles dintre de condicions i a l'inrevés.

3.3.1. Bucle (*loop*)

Podem programar el bucle perquè es realitzi una part del programa un nombre concret de vegades o durant un temps concret. També es pot realitzar indefinidament o fins que un sensor ens ho indiqui (sobrepasant algun nivell, per exemple).



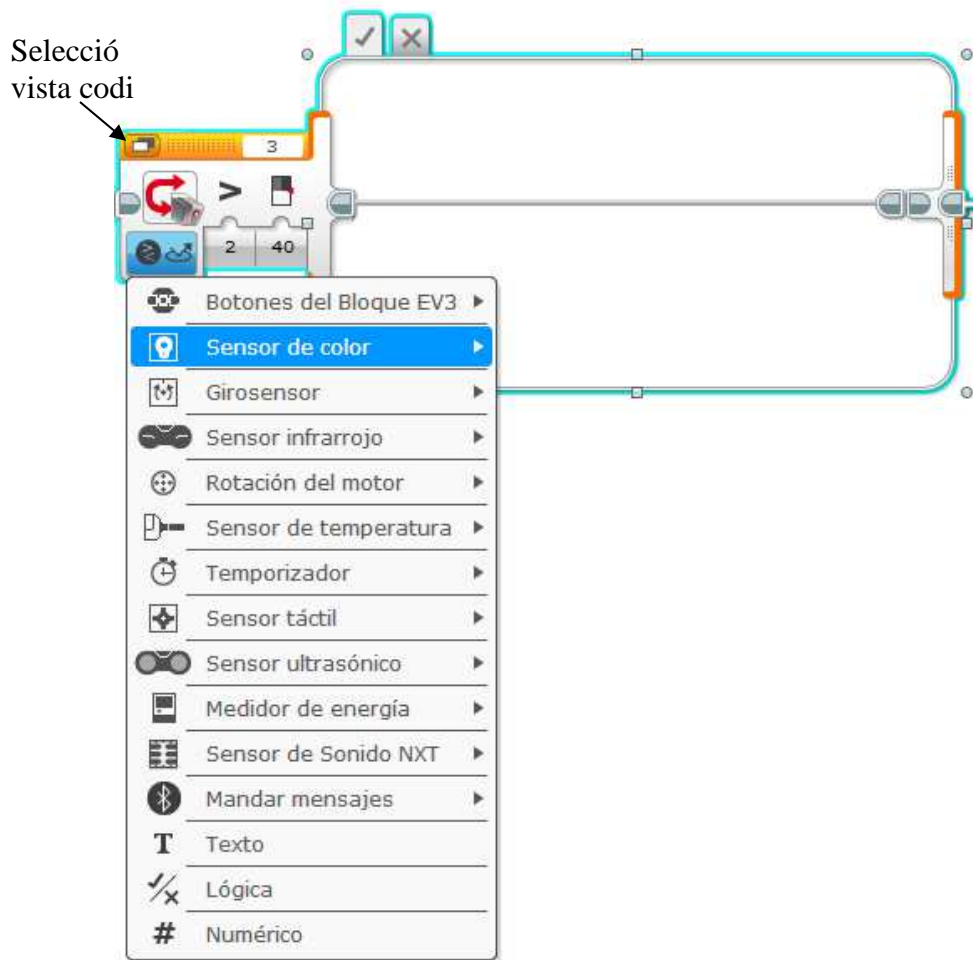


El menú desplegat a la dreta ens permet programar quan s'ha de finalitzar el bucle. Per exemple, si serà infinit (*il·limitat*), dependrà d'un sensor (com per exemple, fins que no es detecti un soroll acústic per sobre del nivell 50), dependrà d'un temps transcorregut concret o d'un comptador (es realitzarà el bucle n vegades) o mitjançant una funció lògica (això seria una funció provinent d'un altre bloc).

3.3.2. Condicions (if, interruptor o Switch)

Els sensors que podem utilitzar en el cas dels Mindstorms, a més de donar-nos el nivell del sensor en un moment determinat, també ens permeten comparar el nivell actual amb un valor concret. Les condicions, però, que veurem ara ens permeten triar entre diferents accions segons algunes indicacions dels sensors, per exemple.



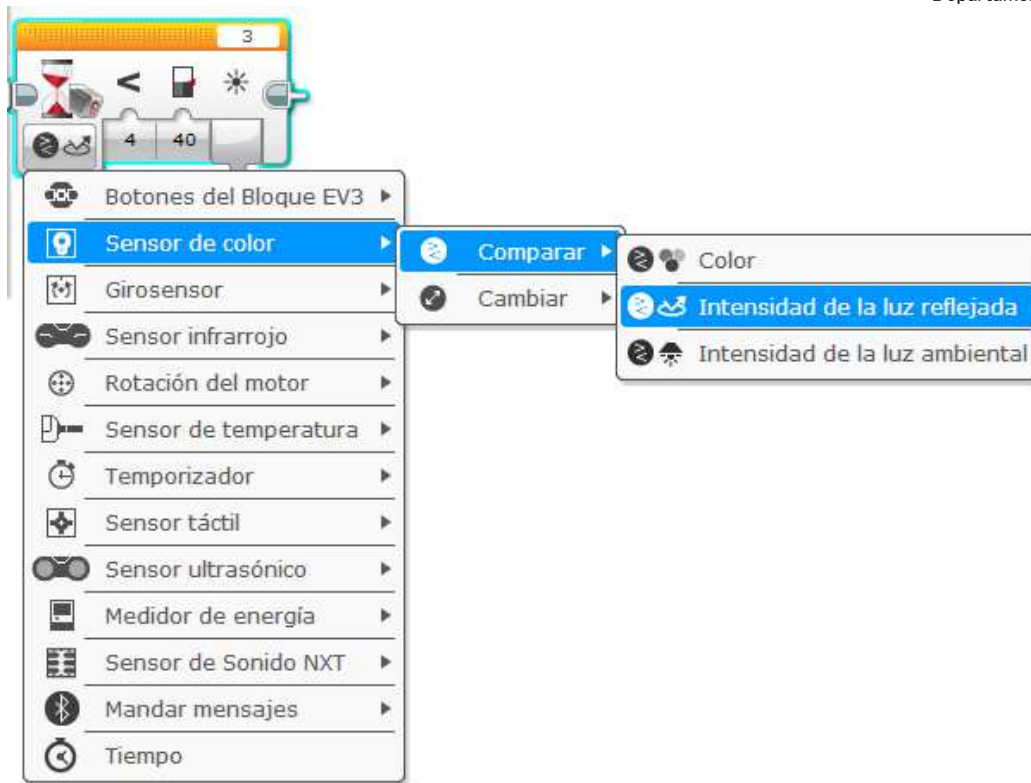


La condició que determina el codi que s'executarà el determina la part de l'esquerra. En aquest cas concret, la condició ve determinada pel sensor de llum. Si en arribar a aquesta part del programa la intensitat de llum que arriba al sensor està per sobre del nivell 40, llavors executarà el codi que hi ha a la pestanya √. En cas contrari, executa el codi de la pestanya X.

3.3.3. Espera

Si volem que el nostre programa esperi (el programa s'atura en aquest bloc) a que es compleixi una certa condició d'algun sensor (per continuar l'execució del programa), per exemple, podem fer servir el bloc Espera:





En el cas de la imatge anterior, el programa s'aturaria en aquest punt del programa fins que el sensor de llum ens doni una intensitat per sota del nivell 40.

