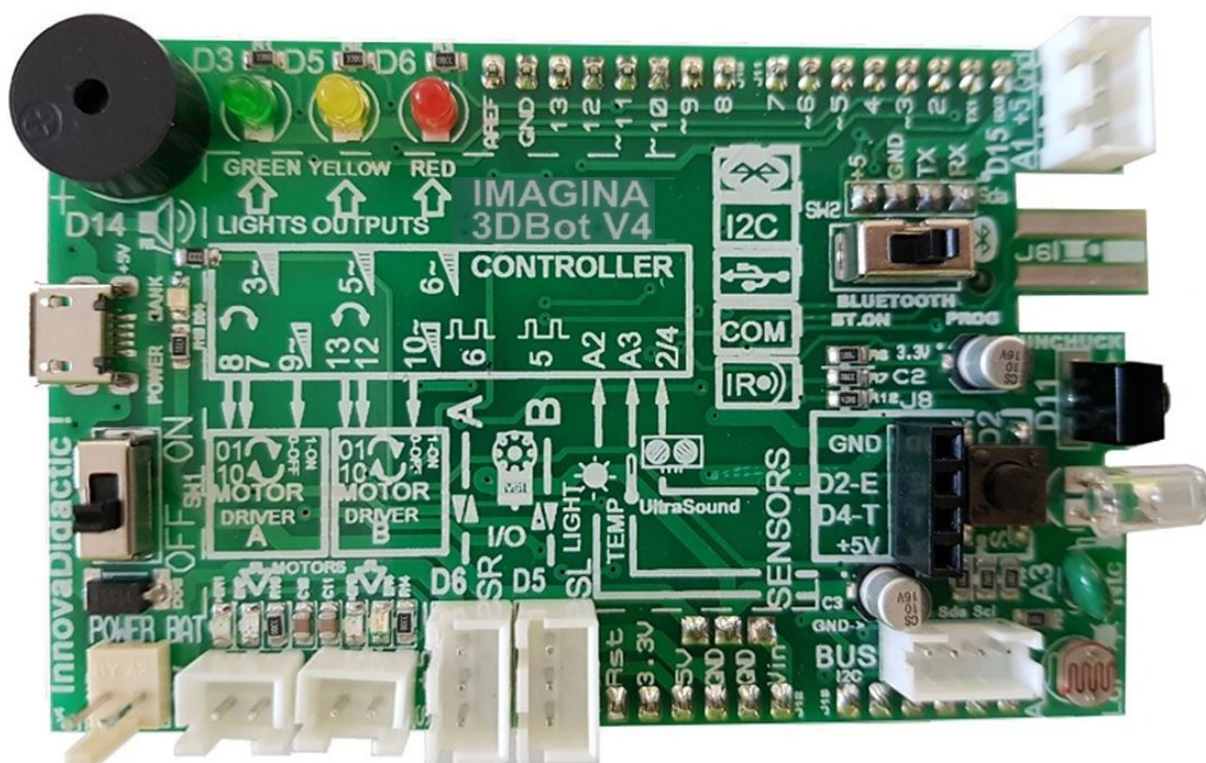


GUIA D'USUARI PLACA IMAGINA 3dBOT ARDUINO



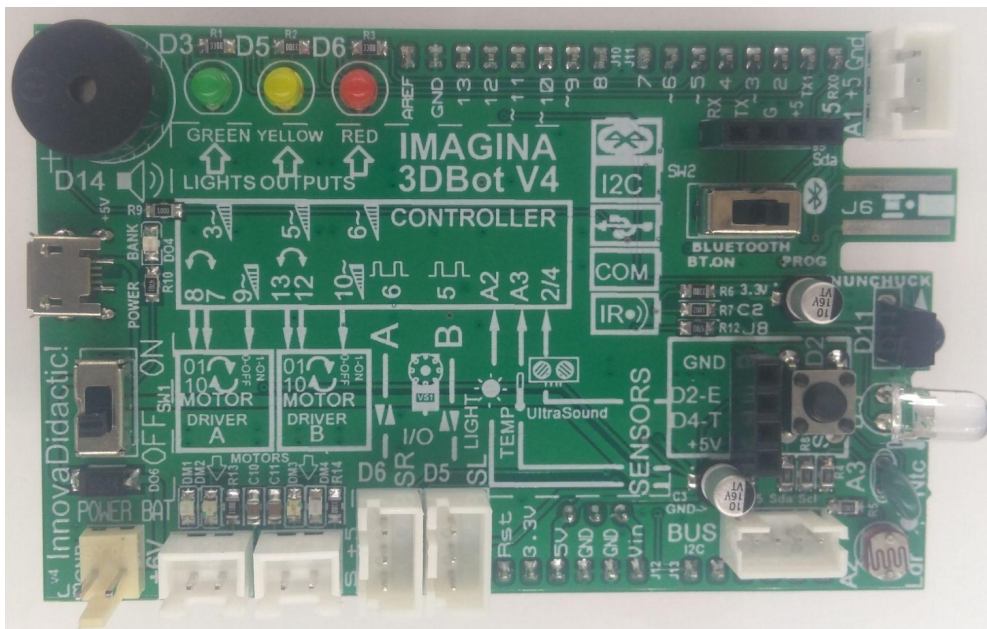
Índex

1.- Introducció	3
2.- Funcionament d'un sistema de control programat	4
3.- Components de la placa Imagina 3dBOT	6
4.- Programació amb ArduinoBlocks	9
5.- Instal·lació de ArduinoBlocks	11
5.1.- Instal·lació ABConnector:	12
6.- Activitats amb la placa Imagina Arduino	13
6.1.- LEDs i Brunzidors	14
6.2.- Polsador i LDR	19
6.3.- Temperatura NTC	31
6.4.- Receptor IR	34

Aquesta placa és programable amb Snap4A, ArduinoBlocks i Microblocks.
El present manual utilitza com a plataforma de programació ArduinoBlocks.

1.- Introducció

Aquesta guia explica la placa Imagina 3dBOT i alguna possible pràctica d'exemple que es podria programar amb ArduinoBlocks i Imagina 3dBOT. La placa Imagina 3dBOT és una Shield, això significa, que és una placa que ha d'anar col·locada a sobre d'una altra que contingui el sistema de control, com pot ser una placa UNO o una ESP32. La placa té nombrosos sensors, actuadors i entrades i sortides que permeten fer infinitat de projectes.

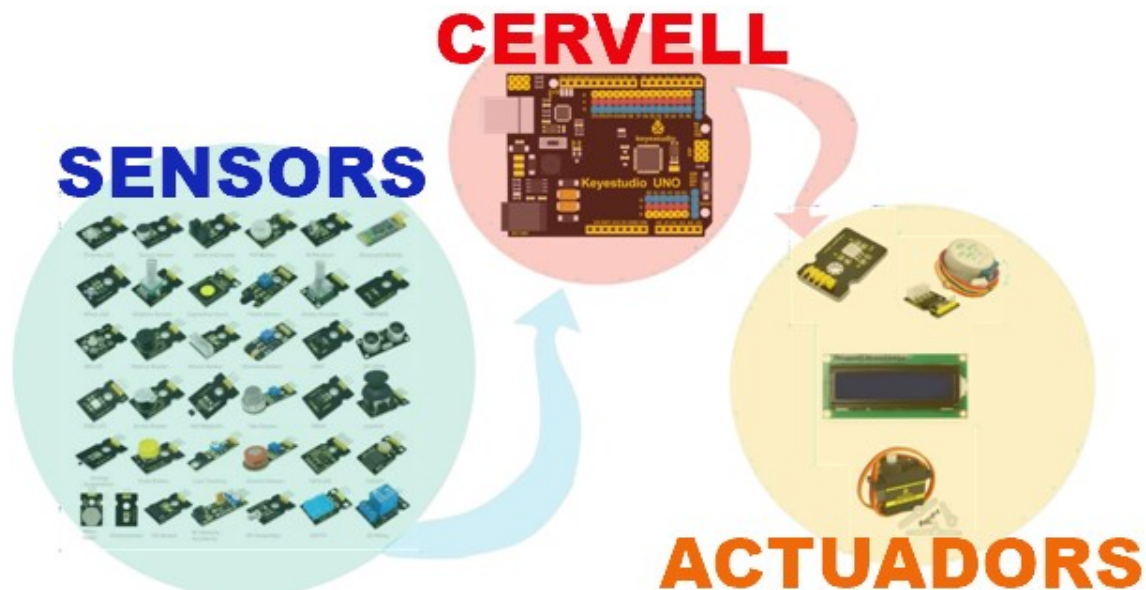


La guia presenta una explicació dels components de la placa i algun exemple de com usar-los en els projectes que elaborarem.

Per realitzar la programació amb la placa Imagina TDR STEAM utilitzarem un llenguatge de programació visual basat en blocs, l'ArduinoBlocks. Aquest llenguatge/programa té un gran potencial i versatilitat.

2.- Funcionament d'un sistema de control programat

Un sistema de control programat funciona de manera similar a la d'un ésser humà. Quan el nostre cervell rep informació dels sentits (oïda, olfacte, gust, vista i tacte) analitza la informació, la processa i dona ordres als nostres músculs per realitzar moviments, donar estímuls a les cordes vocals per emetre sons, etc. Els 5 sentits equivalen a entrades d'informació, i la veu, els músculs, etc. serien les sortides.

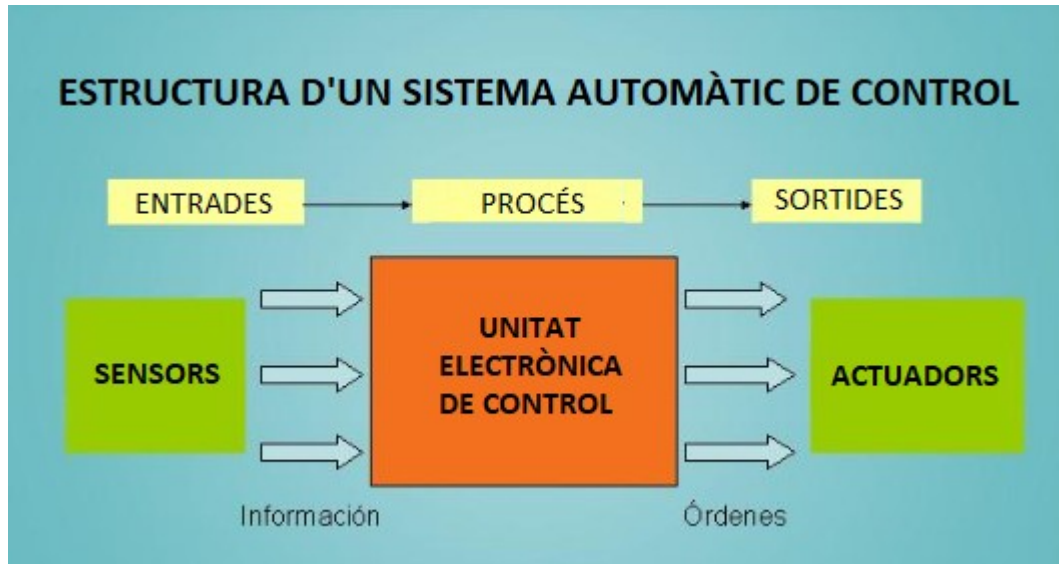


En el cas d'un sistema de control programat, un microcontrolador fa la funció de cervell. Aquest component electrònic conté unes entrades d'informació on es connecten els sensors de llum (LDR), de temperatura (NTC), de so, etc., i també té sortides, on es connecten els motors, LEDs, bronzidors, etc.



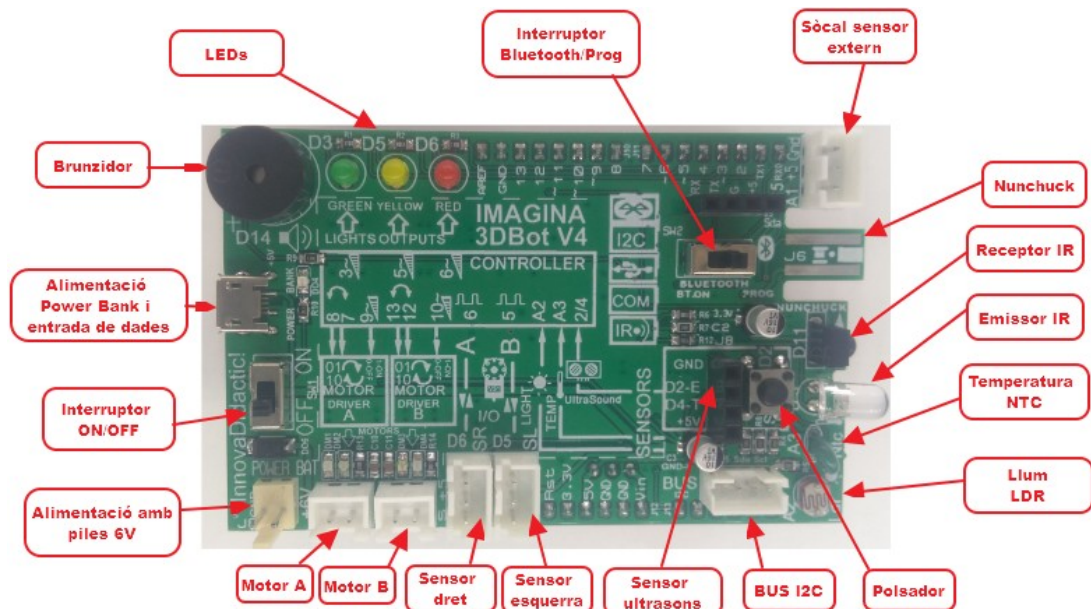
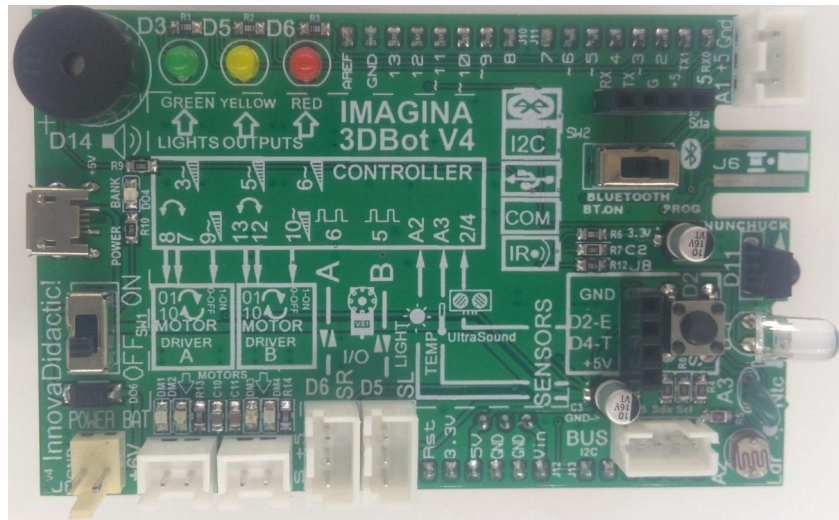
La diferència principal és que, mentre que el nostre cervell ha anat aprenent el que ha de fer al llarg de la nostra vida a base d'estímuls, la memòria del sistema programat és buida, és a dir, no sap el que ha de fer i, per tant, cal que nosaltres li diguem com ha d'actuar en funció dels senyals que rep dels sensors. Aquestes ordres es generen mitjançant el codi del programa que introduïm al sistema de control.

En tot sistema de control automàtic, els sensors (entrades) converteixen les magnituds físiques, químiques, etc. en elèctriques, creant així la informació. Aquesta informació s'envia a la unitat de control (procés) que processa la informació i genera les ordres mitjançant senyals elèctriques que seran convertides en els actuadors (sortides) en altres tipus de magnituds.



3.- Components de la placa Imagina 3dBOT

La placa Imagina 3dBot Arduino és un shield que serveix d'acoblament per a plaques i així utilitzar els components que hi porta incorporats a través de la placa que hi ha connectada a sobre.



Taula d'entrades i sortides

Entrades/Sortides analògiques	Entrades/Sortides digitals
A0 Brunzidor	D0 Bluetooth RX
A1 Sensor de línia fotoelèctric, sensor PIR, servo o qualsevol altre sensor	D1 Bluetooth TX
A2 Sensor de llum (resistència LDR)	D2 Sensor Ultrasons Polsador
A3 Sensor de temperatura (resistència NTC)	D3 Díode LED verd Emissor infraroig
A4	D4 sensor d'ultrasons
A5	D5 Díode LED groc Sensor de línia Esquerra
	D6 Díode LED vermell Sensor de línia dret
	D7 Motor A
	D8 Motor A
	D9 Motor A
	D10 Motor B
BUS I2C/Nunchuk	D11 Receptor Infraroig
	D12 Motor B
	D13 Motor B
	D14
	D15

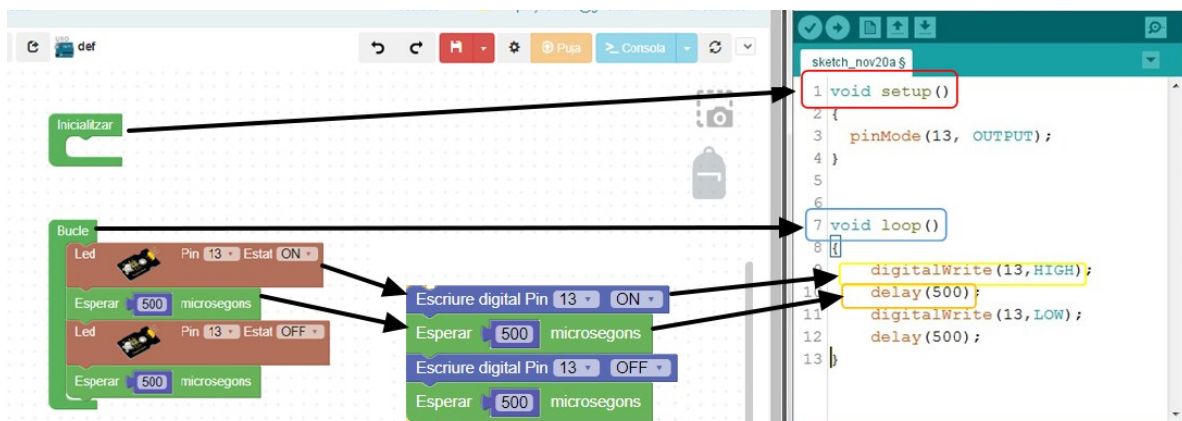
Finançat per

La finalitat d'aquesta Shield és la creació d'un robot motoritzat gràcies al driver o controlador de motors. La placa Imagina 3D BOT la podem posar acoblada a una ESP32 o a una placa UNO. A més a més, com es mostra a la taula d'entrades i sortides es poden connectar a la shield (Imagina 3dBOT) dos motors, sensors i actuadors als pins lliures que conté la placa.

Per programar la placa podem fer-ho des d'ArduinoBlocks creant un projecte de la placa Imagina (+ la placa de sota).

A l'esquerra de la pantalla del nou projecte ens apareix una agrupació de blocs anomenat 3D BOT. A partir d'aquests blocs podrem programar la placa per tal de programar el robot de forma fàcil i senzilla preparada expressament per a la placa.

Per realitzar la programació la podem fer mitjançant la IDE d'Arduino o mitjançant ArduinoBlocks creant un projecte de la placa Imagina (+ la placa que tingui a sota). Com podem veure són dos sistemes diferents. La programació d'Arduino es fa a terme mitjançant instruccions (a la dreta de la imatge) i en ArduinoBlocks es realitza mitjançant blocs (a l'esquerra de la imatge). A la següent imatge es fa una comparació de codi entre ArduinoBlocks i Arduino IDE.

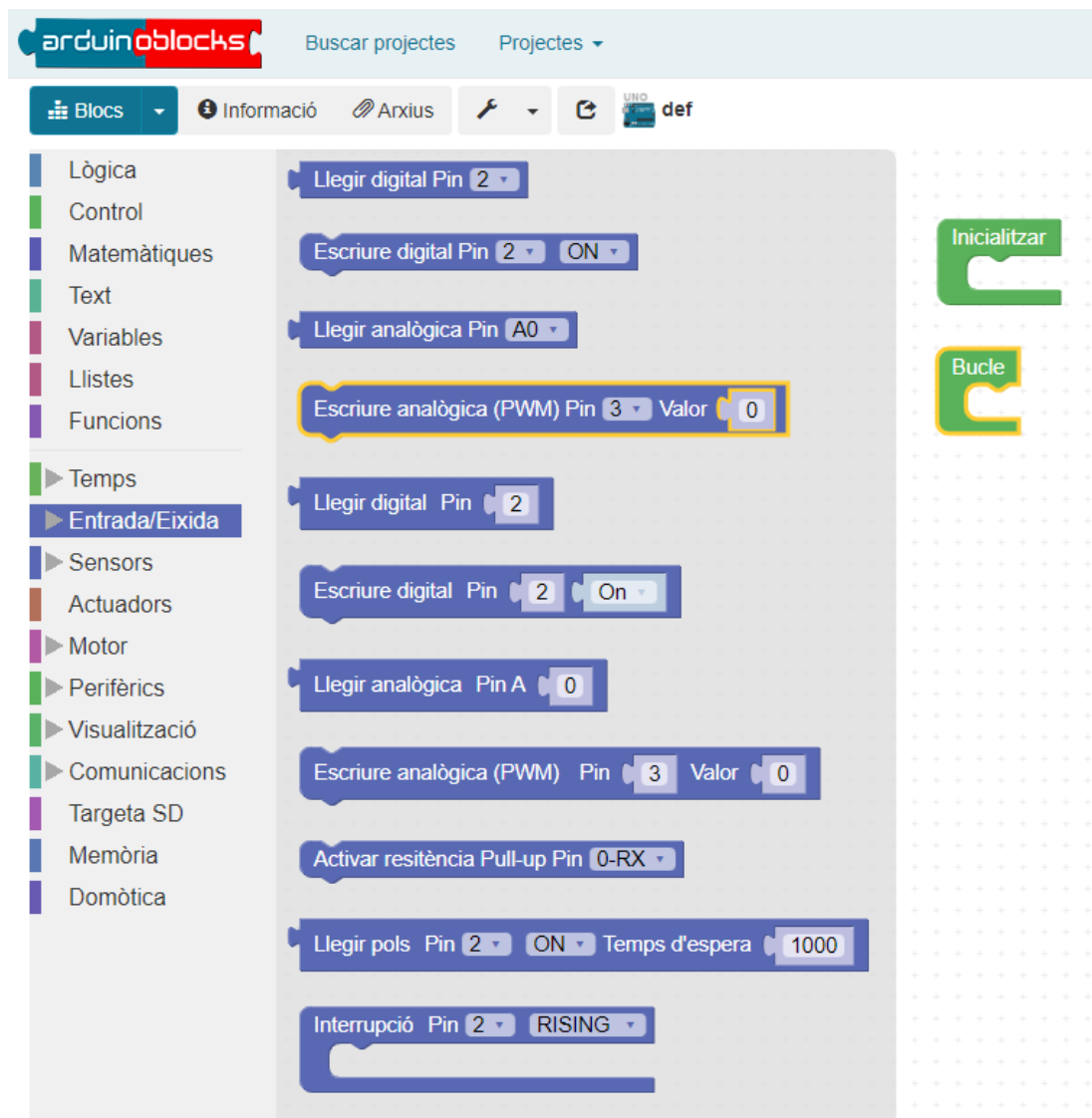


Utilitzar ArduinoBlocks simplifica i fa més intel·ligible el codi, cosa que permet iniciar-se en el món de la programació d'un mode més amigable. ArduinoBlocks també permet programar de diverses formes, ja que té blocs específics que realitzen les mateixes funcions, però que es poden entendre de forma senzilla.

4.- Programació amb ArduinoBlocks

Arduino és un llenguatge de programació gràfic pensat perquè els infants aprenguin a programar amb plaques Arduino a partir d'uns 8-9 anys.

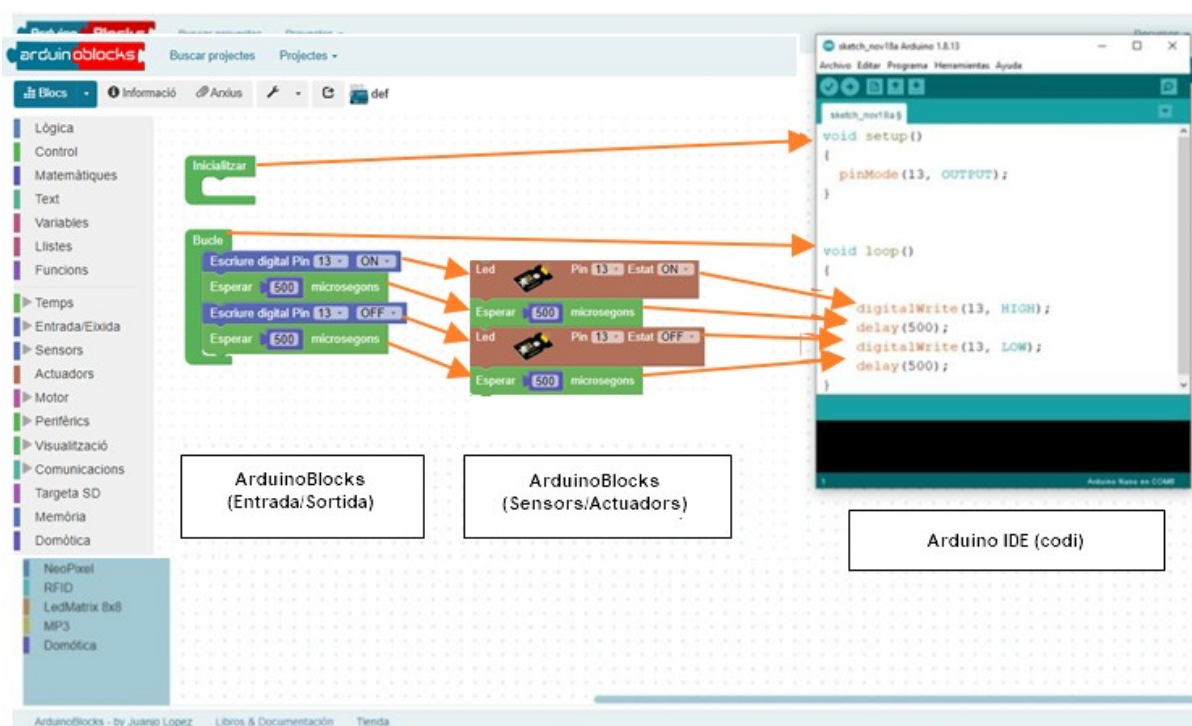
Els diferents blocs serveixen per llegir i escriure les diferents entrades i sortides de la placa, així com programar funcions lògiques, de control, etc.



En aquest manual s'usarà ArduinoBlocks dedicat a l'ús de la placa Imagina 3dBOT i amb aquests blocs podrem programar les entrades i sortides de la placa perquè dugui a terme les tasques que volem.

Podem programar ArduinoBlocks de diferents maneres, ja que té múltiples blocs. També permet exportar el codi per a l'IDE d'Arduino.

En aquest manual utilitzarem, principalment, la programació amb blocs específics per Imagina 3dBOT, perquè és més fàcil d'entendre.



Nota important:

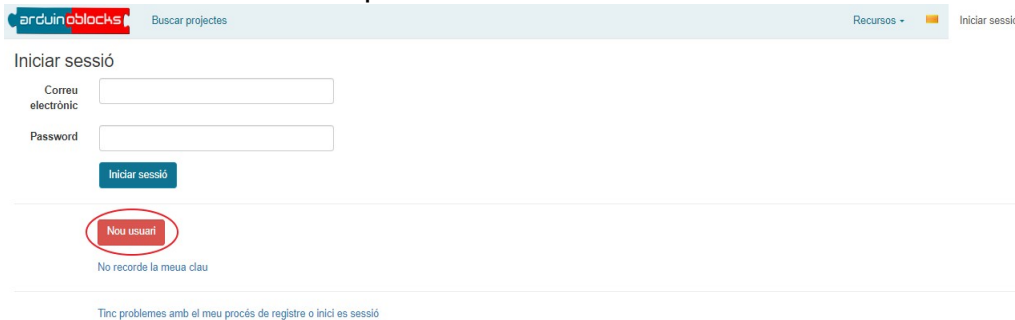
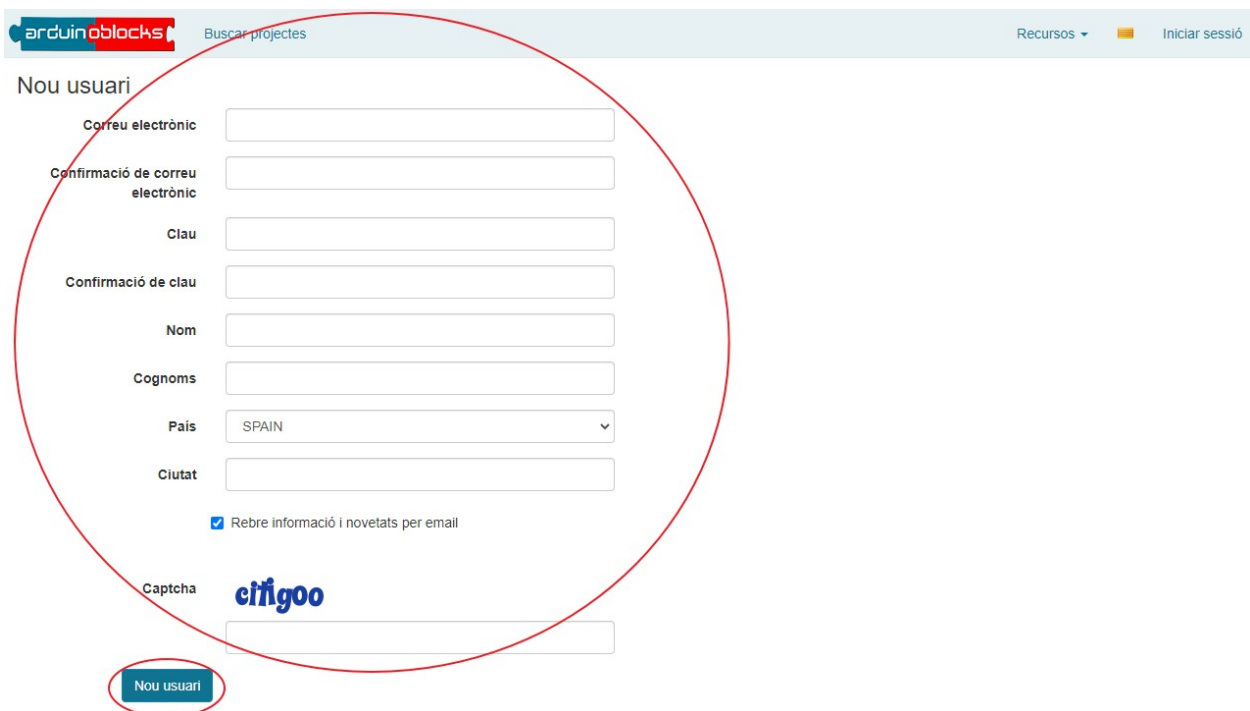
Cal que quan vulguem pujar els programes a la placa l'interruptor bluetooth/prog estigui a la posició de prog

5.- Instal·lació de ArduinoBlocks

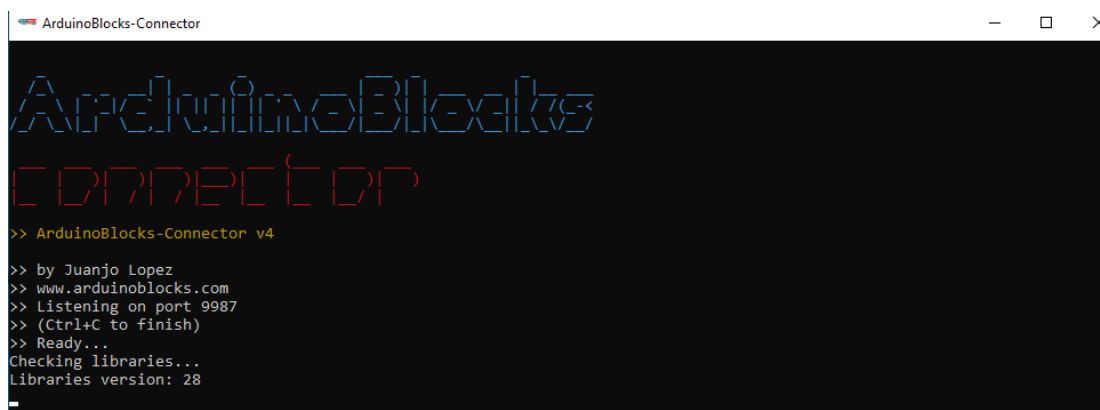
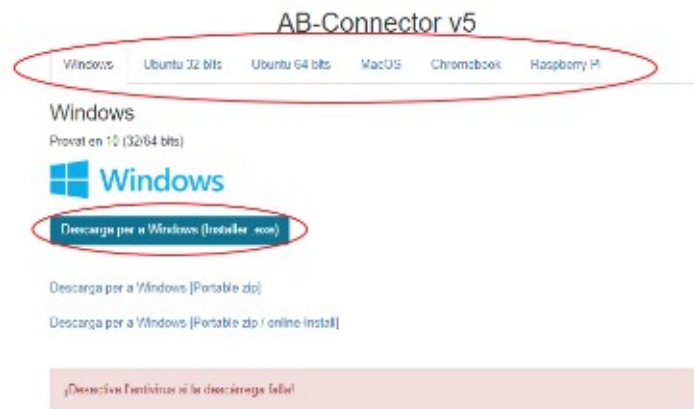
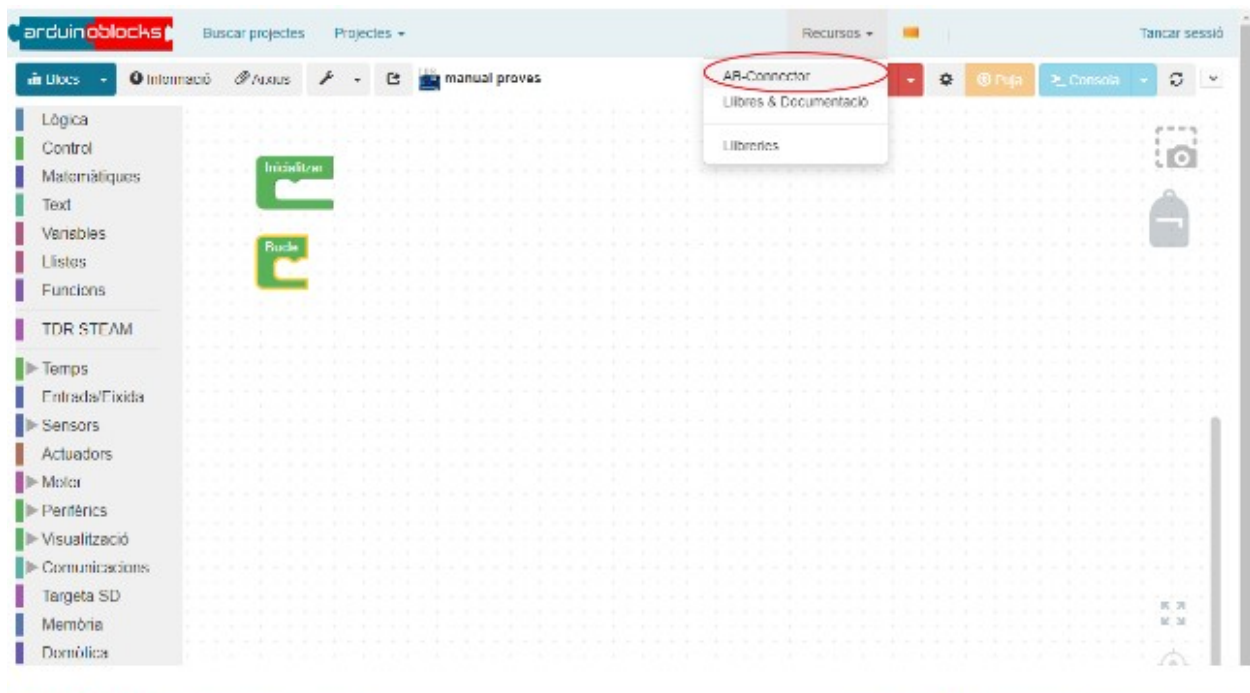
ArduinoBlocks treballa en línia, però hem d'instal·lar un petit programa que serà l'encarregat de connectar el nostre programa amb la placa Keyestudio UNO. Aquest programa basat en Python s'anomena Connector.

Primer hem de crear un compte en ArduinoBlocks (www.arduinoblocks.com) i llavors instal·lar el software **Connector**.

- Per crear un compte a ArduinoBlocks:

5.1.- Instal·lació ABConnector:



6.- Activitats amb la placa Imagina Arduino

A continuació proposem una sèrie d'activitats per aprendre a programar la Imagina 3d BOT amb ArduinoBlocks pas a pas.

Per fer-ho, obrirem l'**ArduinoBlocks** i obrirem també el **Connector** (per poder connectar-nos a la placa) i crearem un projecte de 3D Bot (+ ESP32 si s'escau).

Un cop creat el projecte, a la columna esquerra de la pantalla veurem les agrupacions de blocs classificats en funció de la seva topologia que podrem usar per programar. Els sensors i actuadors que farem servir amb la Imagina 3dBOT podrem trobar-los a l'agrupació anomenada 3DBOT on hi ha els blocs que emprarem a les activitats següents, a banda d'elements de control, lògica, temps, etc. Així i tot, es poden usar per a moltes més activitats que les suggerides.

A la zona de programació trobarem que sempre hi ha dos blocs verds, un d'inicialitzar i un de bucle. Aquests blocs sempre apareixen en iniciar un nou programa. Tot el que posem dins del bloc **Inicialitzar** només s'executarà la primera vegada que es posi en marxa el programa, mentre que si es col·loquen dins del **Bucle**, s'executaran indefinidament fins que apaguem la placa.

Per desplaçar els blocs a on volem caldrà prémer el botó esquerre del ratolí i arrossegat el bloc fins on vulguem deixar-lo i deixar de prémer el botó.

Per eliminar un bloc podem prémer el bloc, desplaçar-lo fins a l'esquerra de la pantalla i deixar de prémer el bloc.

Per fer servir els elements de la placa Imagina 3dBOT seleccionarem l'agrupació de blocs.

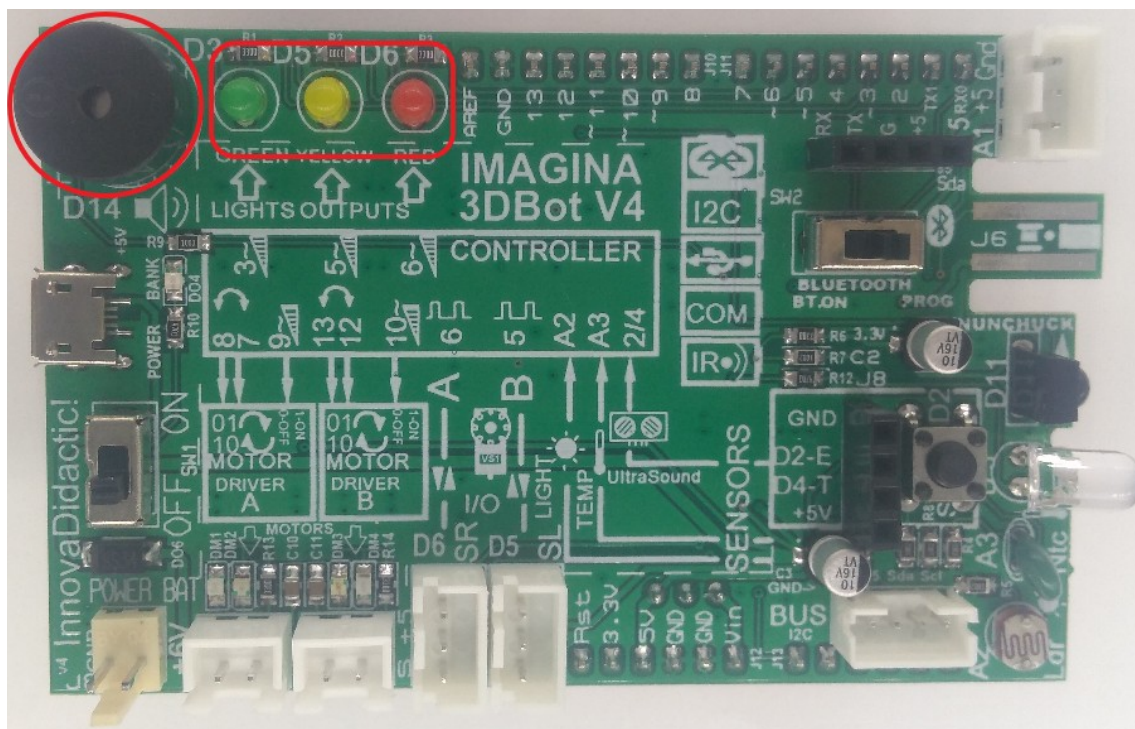
6.1.- LEDs i Brunzidors

Un LED (Díode Emissor de Llum) és un díode semiconductor que emet llum. S'utilitzen com indicadors en molts dispositius i cada vegada amb més freqüència d'il·luminació. Els LEDs presenten molt avantatge sobre les fonts de llum incandescent com un consum d'energia molt menor, major temps, menys mida, gran durabilitat i fiabilitat.

El LED té una polaritat, un ordre de connexió i en connectar-lo el revés no funciona. Per evitar que es cremi sempre ha d'anar connectada amb una resistència en sèrie. Per limitar el corrent elèctric que el travessi.

El brunzidor (buzzer en anglès) és un transductor electroacústic que produeix un so o brunzit continu o intermitent. Serveix com a mecanisme de senyalització o avís i s'utilitza en múltiples sistemes com en automòbils o en electrodomèstics.

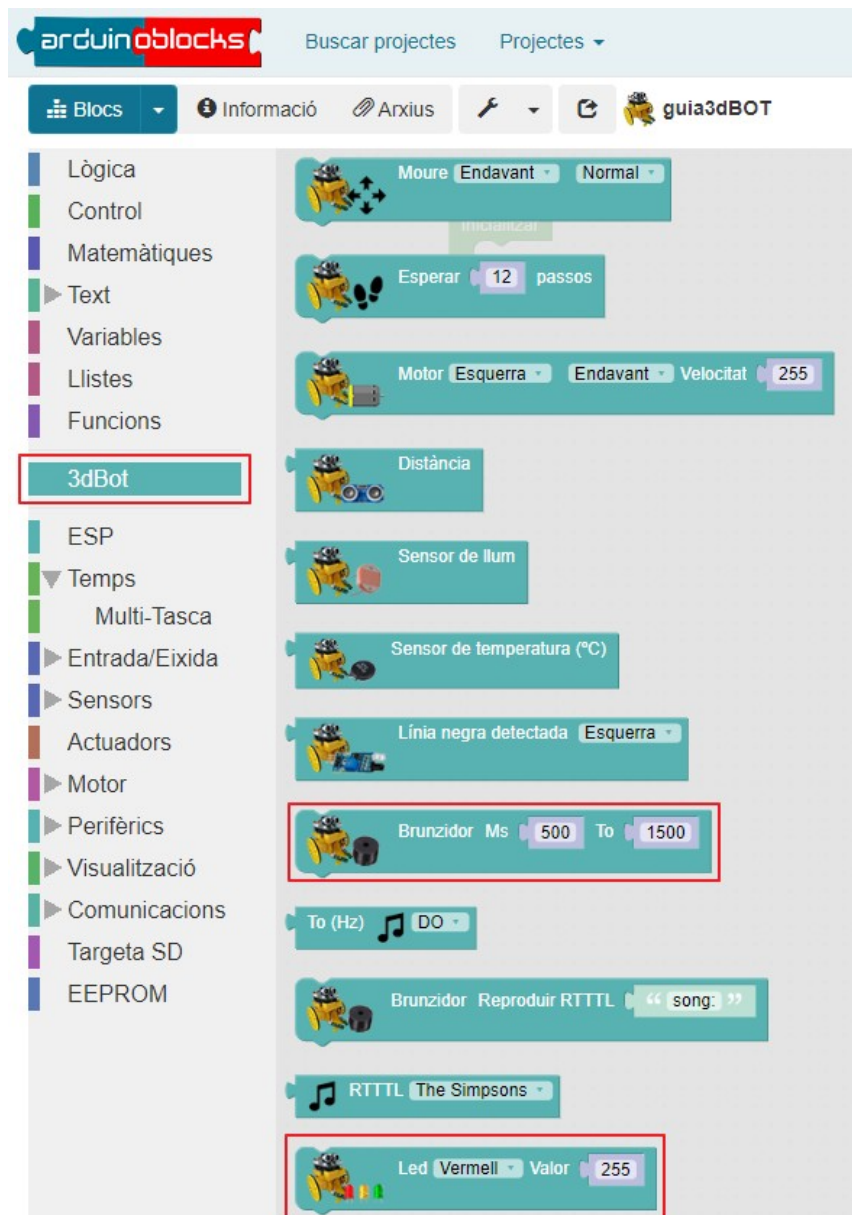
Podem reproduir sons constants a diferents Hz o amb diferents notes musicals i també podem reproduir diferents varietats de cançons obrint el desplegable del quart element de brunzidors.



Brunzidor a dalt a l'esquerra i LEDs al seu costat dret

Presentem una activitat que consisteix a simular un semàfor, fent una seqüència amb els LEDs de la placa 3DBOT verd, groc i vermell que cada 5 segons s'engegui un i s'apaguin els altres. Alhora, quan el LED groc estigui engegat (ha de fer pampallugues) el brunzidor haurà de fer sons al mateix ritme i quan el LED vermell estigui engegat el brunzidor faci un so constant.

Per aquesta activitat utilitzarem els següents blocs de l'agrupació de blocs de 3dBOT.



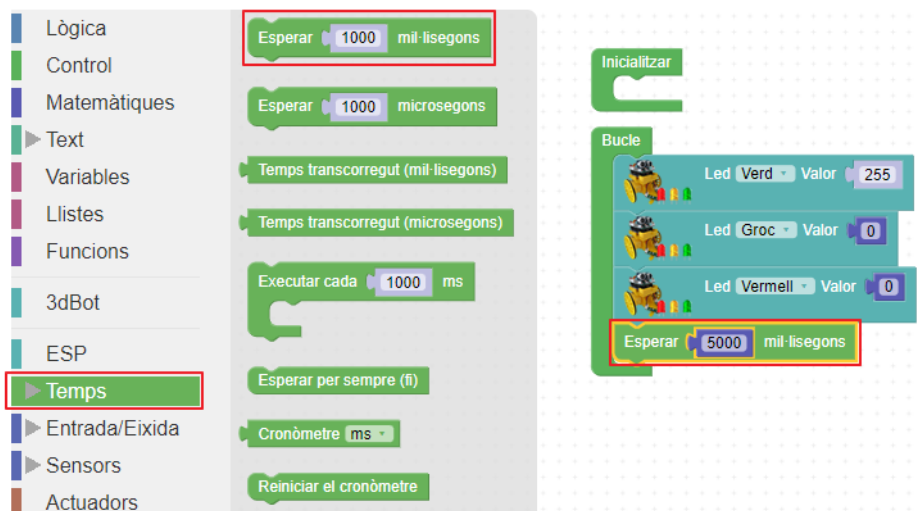
En el bloc de LED, el desplegable que conté serveix per poder triar el color del LED, poden seleccionar entre el LED verd, el groc i el vermell. A part, també té un valor, que indica la intensitat amb la qual es vol que el LED faci llum, sent 0 quan el LED està apagat i 255 quan el LED està engegat amb la màxima lluminositat.

Així doncs, per començar introduïrem 3 blocs de LEDs, un per cada color, dins del bucle. Al LED verd li donarem el valor de 255, mentre que als altres dos LEDs els hi posarem un valor de 0.

És important recordar que si posem un LED a 255 (l'engeguem) continuarà engegat fins que no li posem el valor de 0 (l'apaguem).



Un cop engegat el LED verd i apagats la resta de LEDs cal que posem un temps d'espera al final d'aquesta seqüència per fer que funcionin durant 5 segons (1000 mil·lisegons = 1 segon) aquestes instruccions i que posteriorment continuï fent la resta d'instruccions del bucle.



Un cop acabada la seqüència del LED verd, cal afegir la seqüència del LED groc. Aquesta és una seqüència de 5 segons que fa una intermitència conjuntament amb un brunzidor. Per fer-ho, haurem de seleccionar una estructura de control que repeteixi les vegades que volem que es repeteixi; necessitem fer-ho per fer una seqüència d'intermitència repetidament.



Posteriorment, ens fa falta posar a dins el bloc de repeticions que a cada iteració posi el LED groc engegat i amb un brunzidor durant 0,5 segons i que posteriorment s'apagui el LED groc i s'esperi durant 0,5 segons més. Aquesta seqüència l'haurèm de repetir 5 vegades en lloc de 10.

El bloc del brunzidor té dos termes a modificar: els Ms que són els mil·lisegons que el brunzidor estarà produint so i el To que és el volum al qual es vol produir el so.

Ens hauria de quedar alguna cosa similar al següent:



Darrerament, cal afegir-hi la seqüència de 5 segons que faci que el brunzidor produeixi un so constant i de mentre el LED estigui engegat.

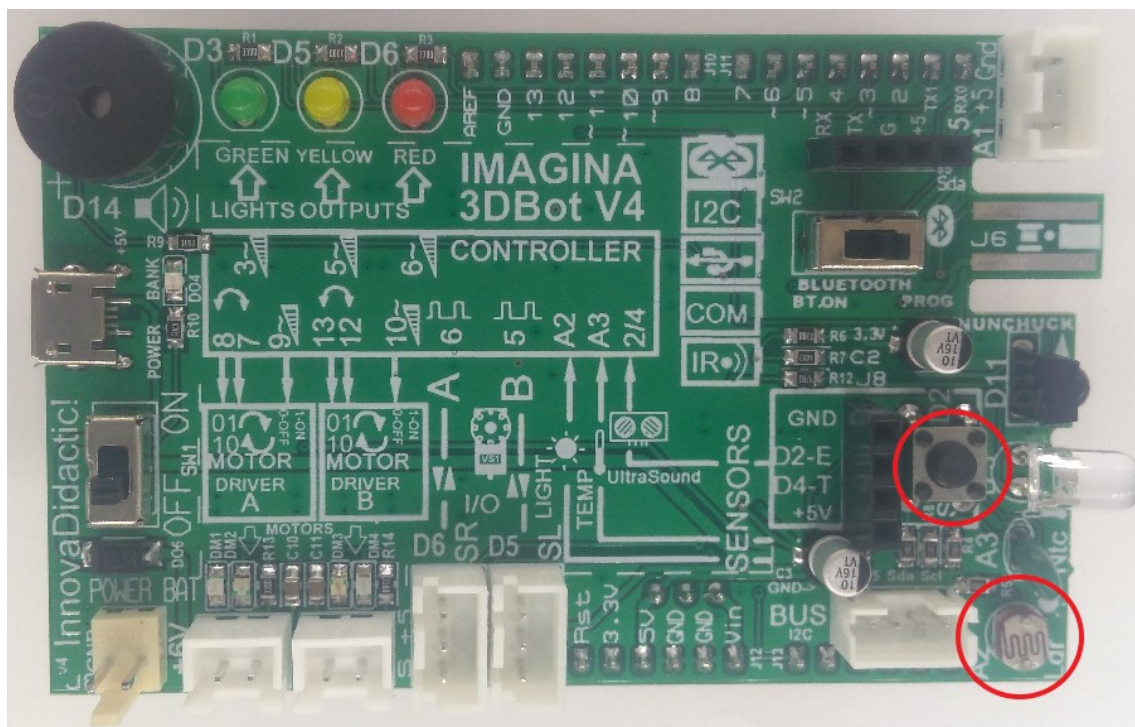


Cal recordar que per pujar el programa a la placa, l'interruptor de Bluetooth/Prog ha d'estar al costat de Prog.

6.2.- Polsador i LDR

Un polsador és un dispositiu que solament s'activa només quan està sent polsat, a diferència d'un interruptor que quan s'activa continua activat fins que no es desactiva (Normalment Obert) o que només es desactiva quan està sent polsat (Normalment Tancat).

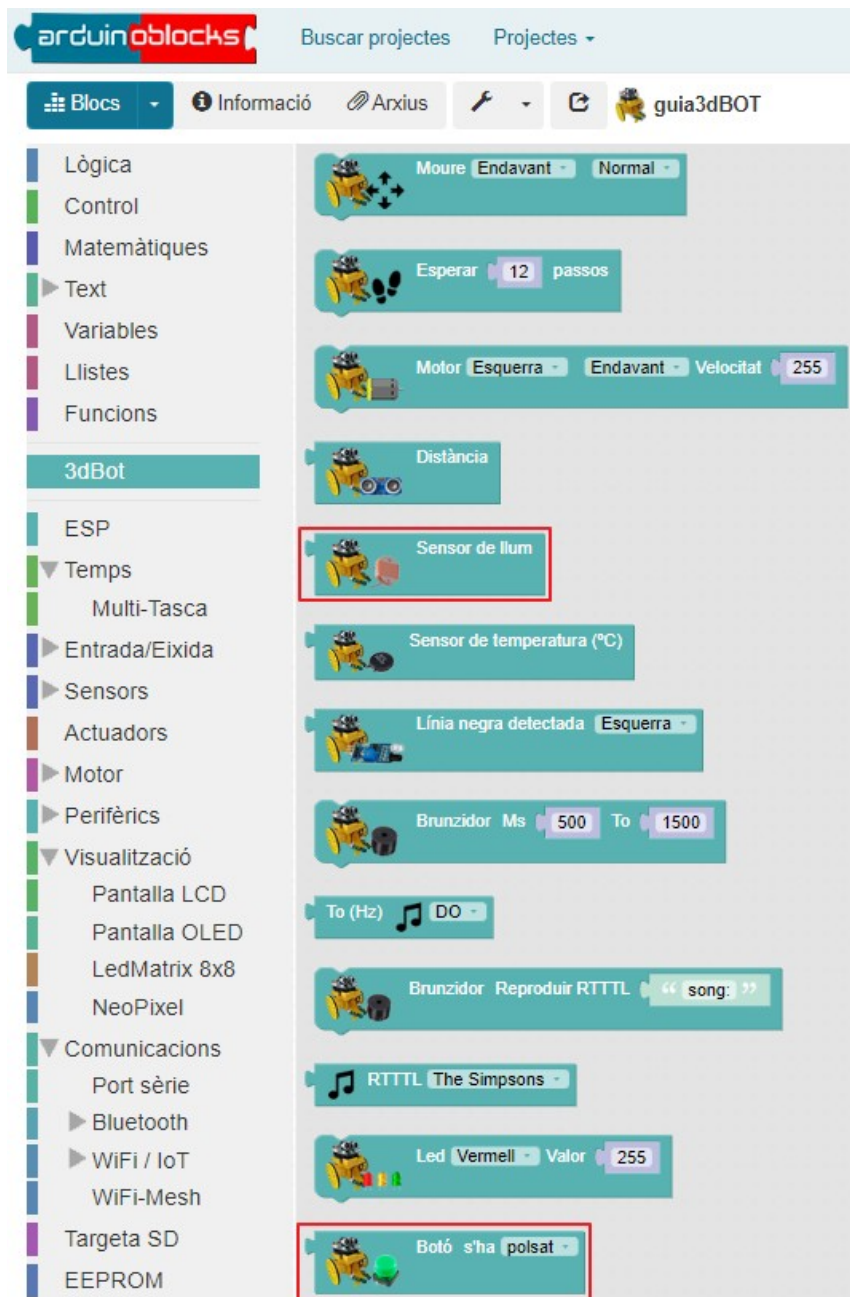
Un LDR (Light Dependent Resistor) és un resistor que varia el valor de la resistència elèctrica depenent de la quantitat de llum que s'incideixi sobre seu. En la seva programació cal saber que quan la quantitat de llum que rep és màxima equival a 4095 i quan és mínima equival a 0.



LDR cantonada dreta i Polsador una mica més amunt

Es proposa la següent activitat: Quan premem el polsador llegirem el valor del sensor de llum (LDR) i depenent del valor engegarem un valor o un altre. Si el valor del sensor de llum és menor de 500, engegarem el LED vermell. Si el valor del sensor de llum és més gran o igual que 500 i més petit que 3000 engegarem el LED verd. I si el valor del sensor de llum és més gran o igual que 3000, llavors engegarem el LED groc.

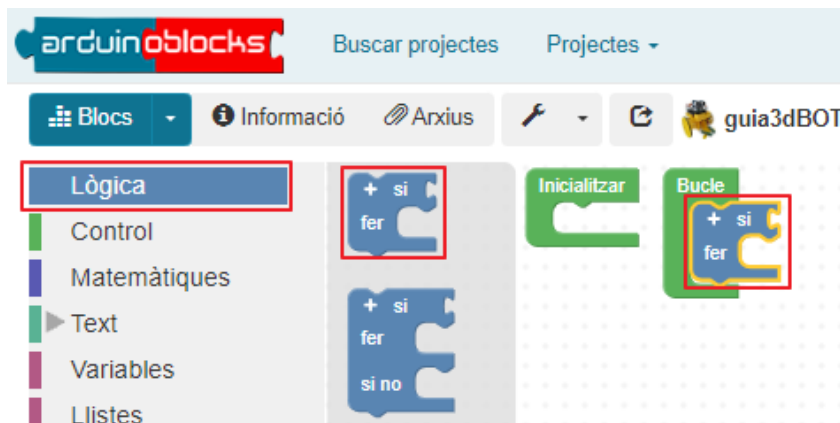
Per començar, haurem de localitzar on tenim el bloc del polsador i del sensor de llum.



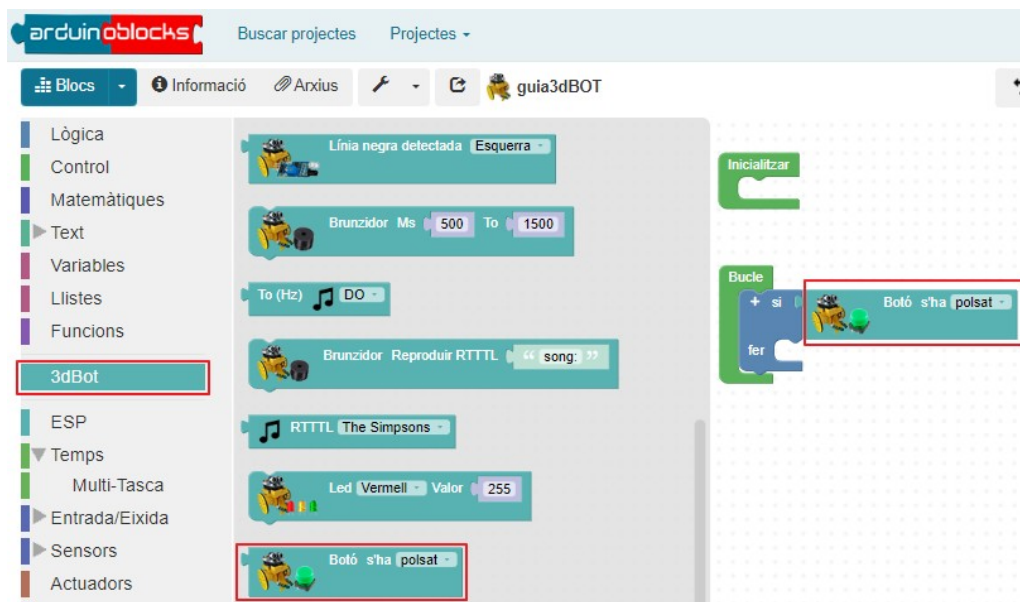
El sensor de llum, com s'ha esmentat anteriorment, el valor que retorna el bloc del sensor de llum va de 0 a 4095.

El botó té dues posicions: si s'ha polsat (que donarà un senyal quan es premi el polsador) i s'ha soltat (que donarà un senyal sempre que no es premi el polsador).

Primer de tot, haurem de posar un bloc que compari si han premut o no el botó. Per fer-ho, necessitem posar la comparació dins del bucle



I on a dins la condició (si) hi posarem el bloc de si han polsat el botó.



Posteriorment, haurem de crear una variable on guardarem el valor del sensor de llum.

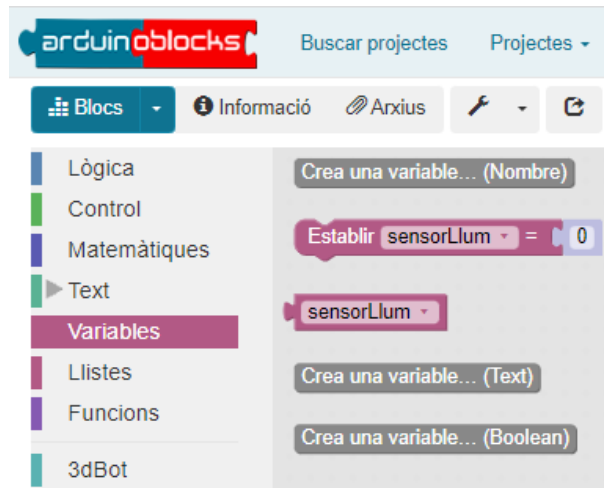
A l'agrupació de variables hi trobarem el següent:



Podem crear tres tipus de variables: Nombre que és per valors numèrics, Text que és per guardar, per exemple, una paraula i la Booleana que és per guardar valors que siguin o Cert o Fals.

Ara nosaltres farem servir una variable de Nombre. Quan fem clic per crear una variable ens apareixerà una finestra on haurem de posar un nom, però sense espais.

Un cop hàgim creat una o més variables ens apareixerà el següent:



Si tenim més d'una variable, un cop hàgim arrossegat qualsevol dels blocs de color magenta a la zona de programació a través del desplegable del bloc podrem seleccionar altres variables que hàgim creat.



A aquesta variable li assignem el sensor de llum de l'agrupació de blocs 3dBOT de la columna de l'esquerra de la pantalla.



Un cop hem assignat el valor del sensor a la variable ja podem utilitzar directament el valor de la variable quan l'hàgim d'utilitzar per posar a les condicions.

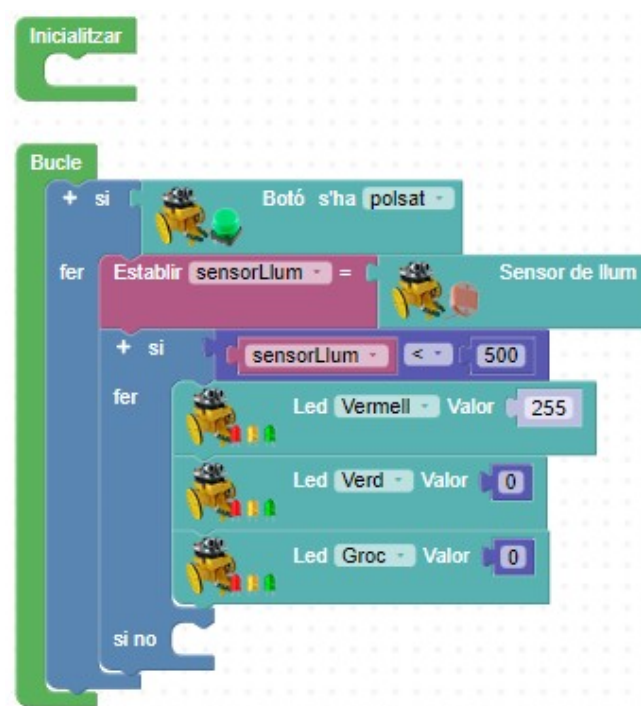
Seguidament d'establir el valor de la variable haurem de posar les condicions que se'ns proposen. Quan el valor del sensor de llum sigui superior a 500 haurem d'engegar el LED vermell i apagar els altres dos LEDs. Haurem d'utilitzar el següent bloc de condició, ja que volem que si és menor de 500, faci una acció, però que si en fa una altra relacionada amb el sensor, en faci una altra o una altra.



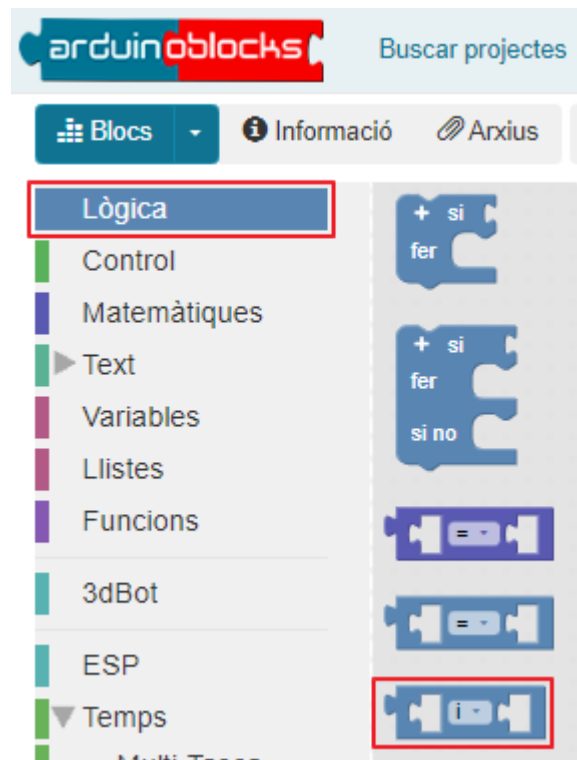
Per comparar dos valors numèrics (fent clic sobre el símbol de comparació entre els dos espais podem canviar el tipus de comparació) posant els blocs corresponents a cada costat de la comparació



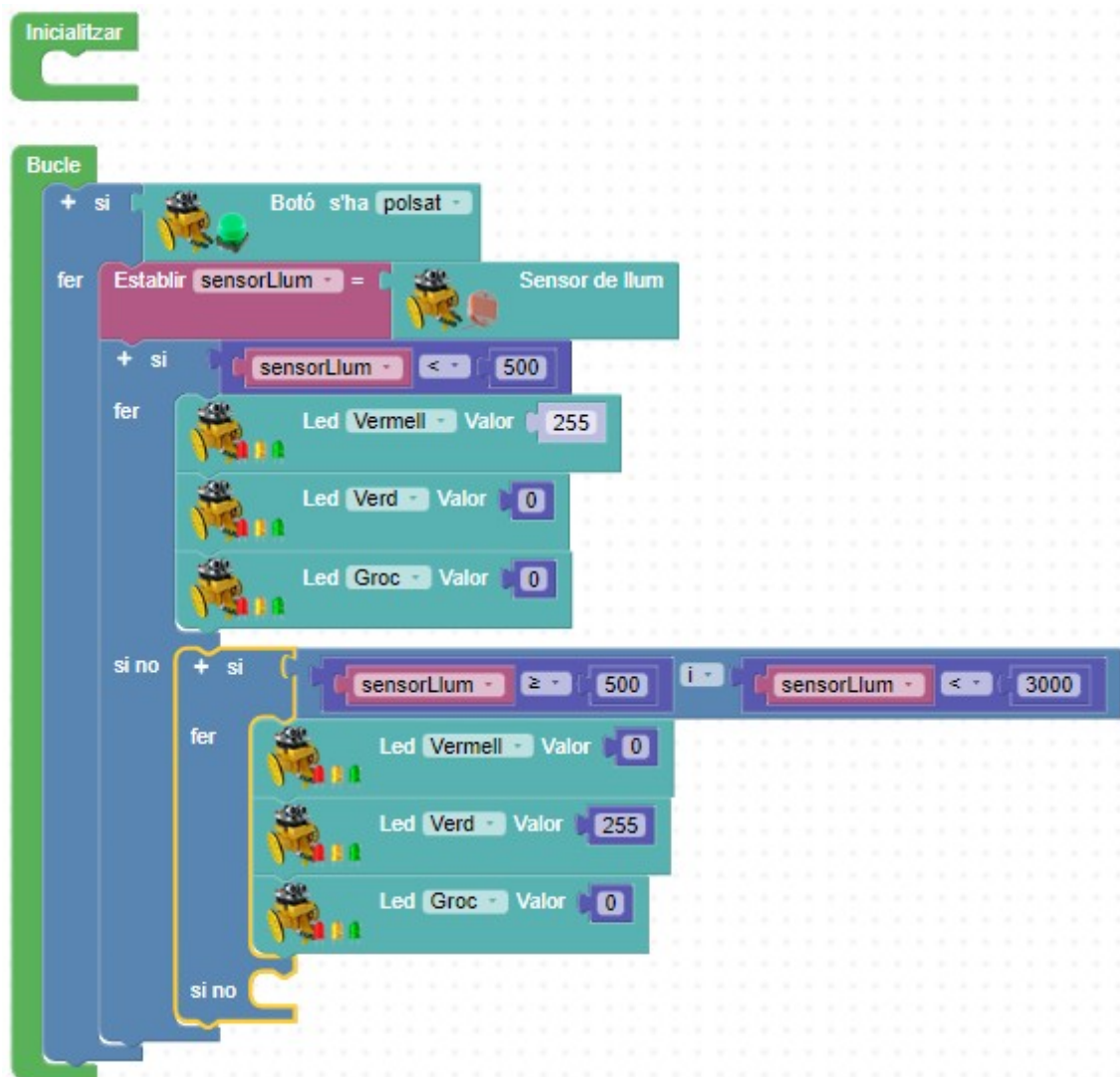
Un cop hàgim afegit la comparació i les accions que s'ha de dur a terme quan es compleixi la condició haurem de tenir alguna cosa similar al següent:



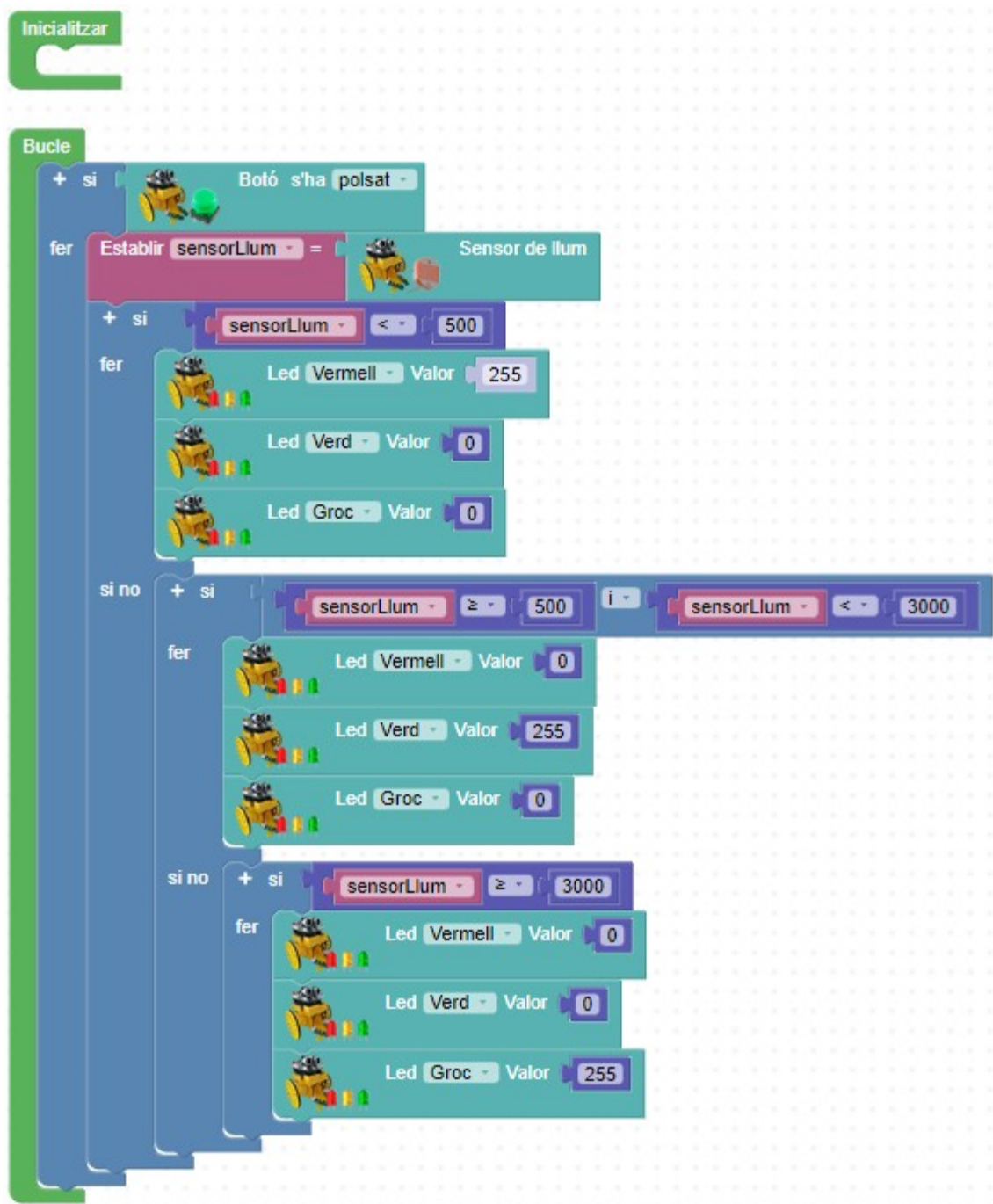
Ara afegirem la següent condició si .. fer .. sino ... dins del si no del bloc anterior.
Ara, però, necessitem més d'una condició que s'ha de complir per poder engegar el LED verd. Per fer-ho, podem utilitzar un bloc que ens permet comprovar si dues condicions són certes o no alhora.



Utilitzant la **i** podem posar dues condicions a cada costat del **i**. Perquè la condició del **i** sigui certa, les dues condicions que té hi ha a dreta i a esquerra hauran de ser certes. Si una de les dues condicions o totes dues són falses, la condició del **i** serà falsa.



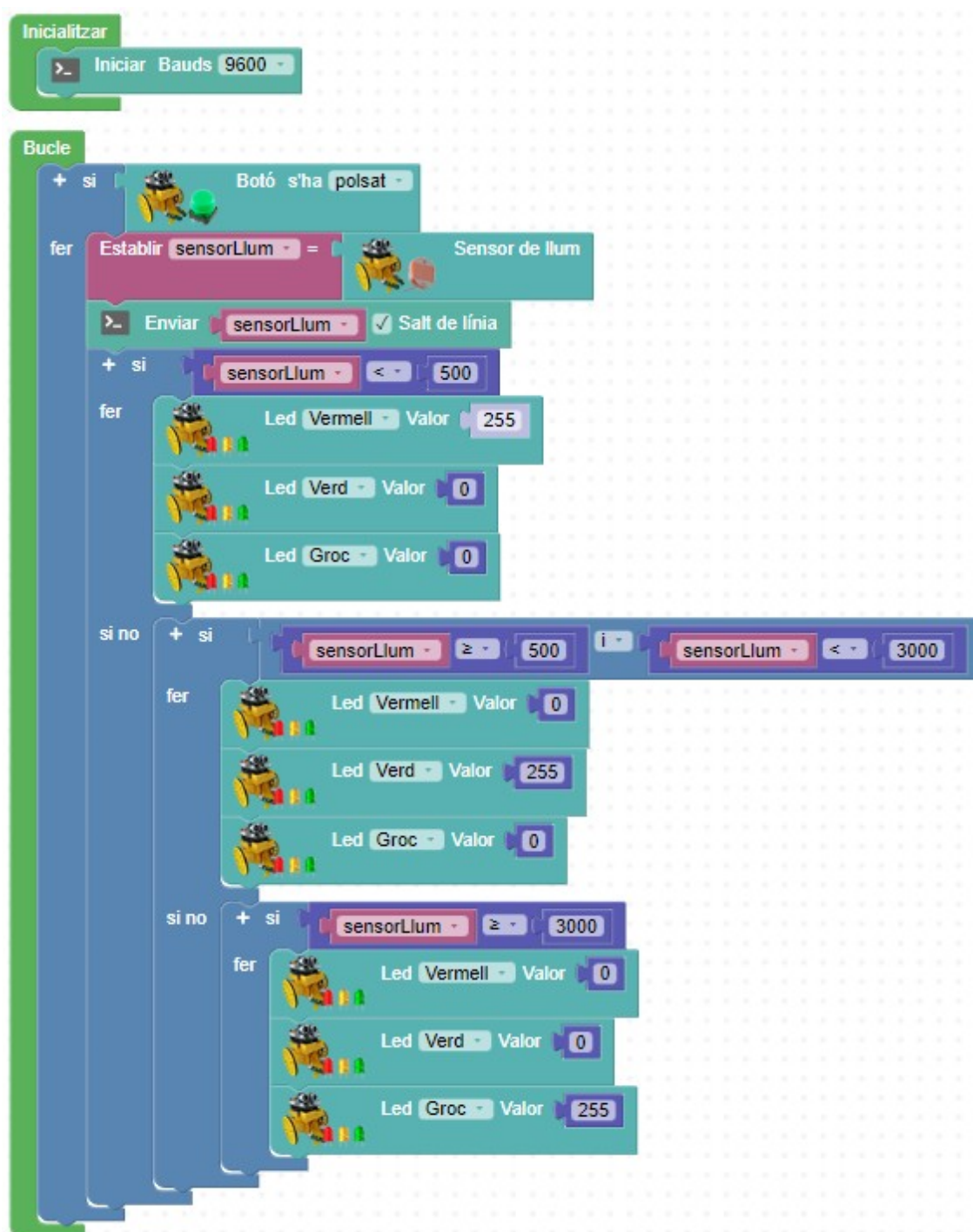
Per acabar l'activitat, ens faltaria posar la condició de si és més gran o igual que 3000 el que rep el sensor de llum.



Aquesta pot ser una possible solució al problema de l'activitat. Finalment, el que podríem fer seria utilitzar la consola de l'ArduinoBlocks per poder veure el valor de la variable del sensor de Llum quan premem el botó. Per fer-ho, haurem de seleccionar l'agrupació de Comunicacions i posteriorment Port Sèrie.



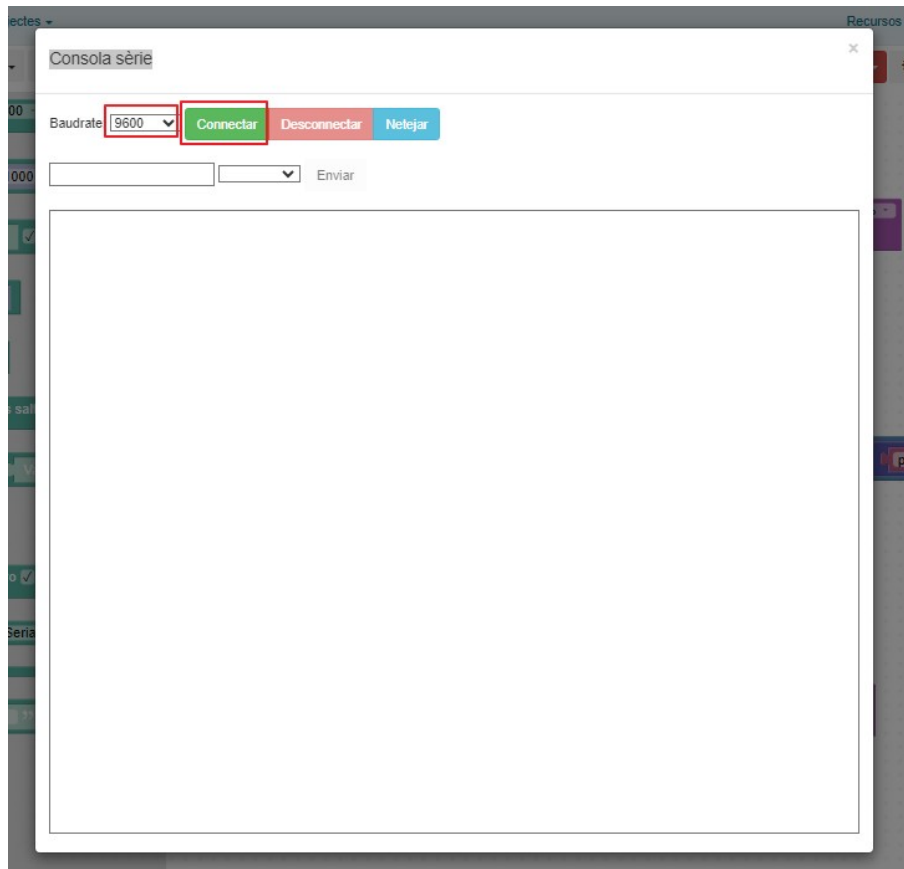
Haurem d'arrossegar el bloc d'Iniciar Bauds al bloc d'Inicialitzar seleccionant el 9600 en el desplegable. També haurem d'arrossegar un bloc per Enviar la variable del sensor de llum un cop hàgim establert un valor a la variable. Per fer-ho, haurem d'arrossegar el bloc d'enviar i posteriorment posar-hi el bloc petit de variable a dins l'espai buit del bloc d'enviar.



Fent això hem inicialitzat el port en el qual volem comunicar-nos i enviar el valor del sensor llum. Per poder veure el valor a la consola de l'ArduinoBlocks, haurem de prémer consola



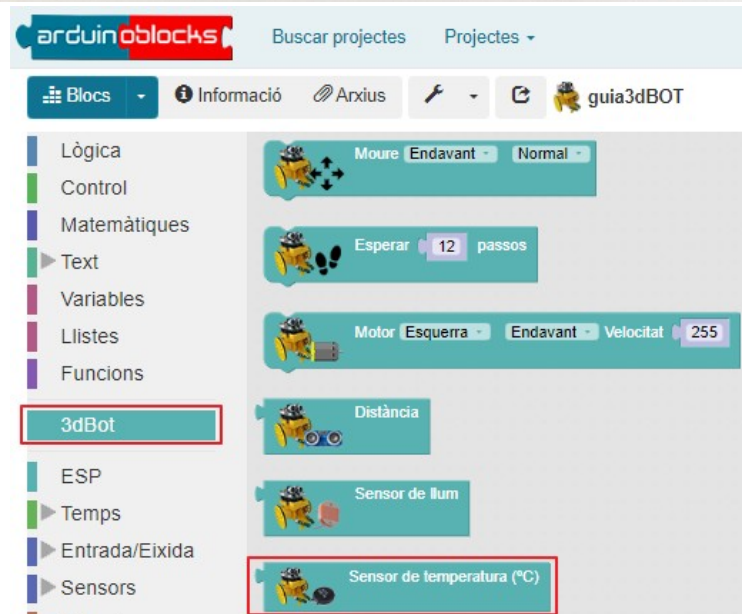
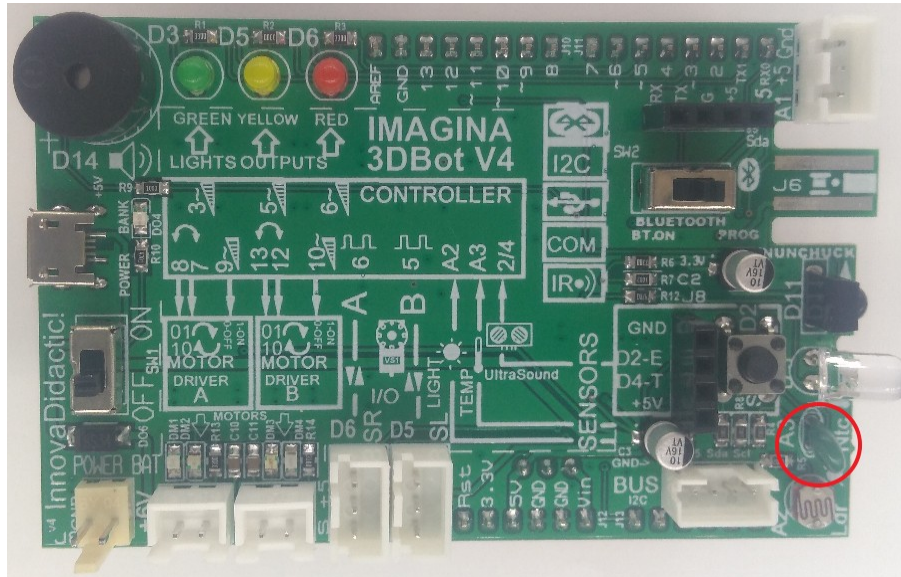
I posteriorment prémer el botó connectar, assegurant-nos que també tinguem el port 9600



Cada vegada que premem el botó de la placa ens hauria d'aparèixer a la consola un valor entre 0 i 4095.

6.3.- Temperatura NTC

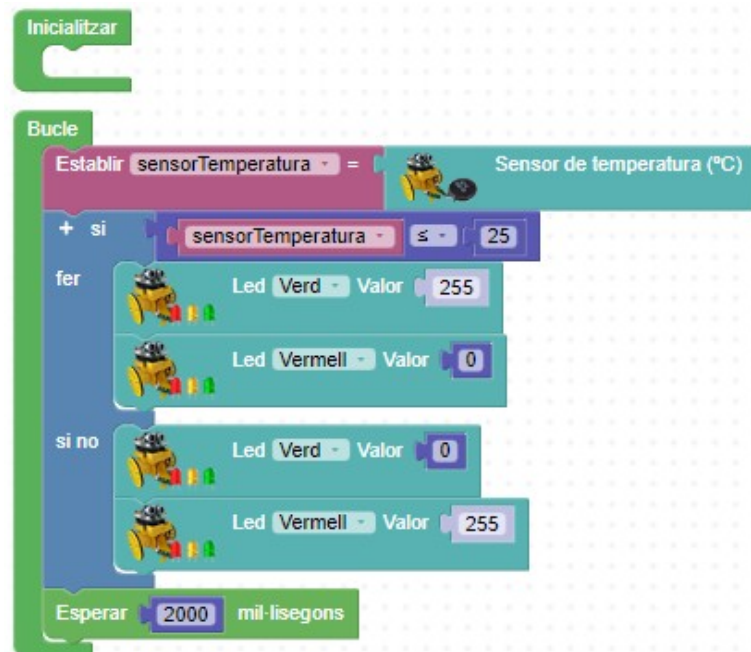
El sensor de temperatura ens permet saber la temperatura ambient. El bloc de programació per utilitzar el sensor el trobarem a l'agrupació de 3dBOT.



Es proposa un programa el qual controli la temperatura engegant el LED verd si la temperatura és inferior o igual a 25 graus i engegant el LED vermell si la temperatura és superior a 25 graus. A més a més, el control haurà d'actualitzar-se cada 2 segons i enviar pel port sèrie el següent: Temperatura: X °C (on X és el valor del sensor de llum).

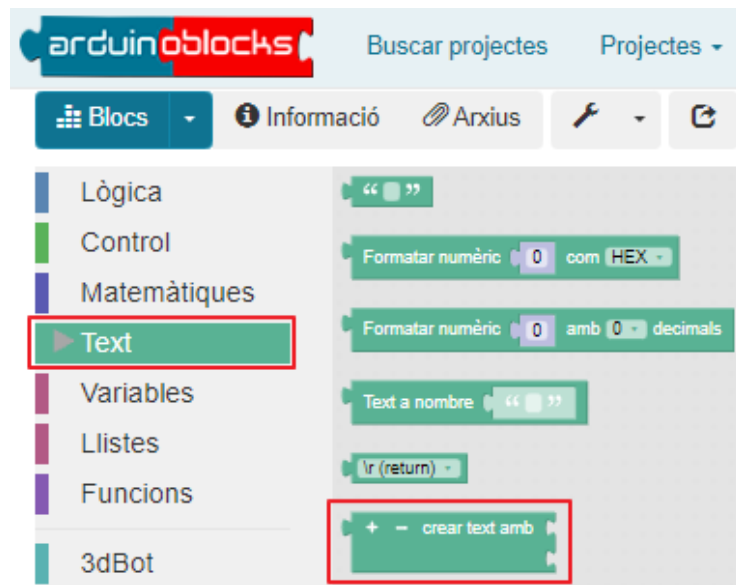
Primer farem el flux normal del programa i posteriorment hi afegirem les comunicacions.

Així doncs, primer de tot crearem una variable del sensor de temperatura i compararem si la temperatura és més petita o igual que 25. Si és així, engegarem el LED verd i apagarem el vermell. Per contra, si és més gran que 25, apagarem el LED verd i engegarem el LED vermell.



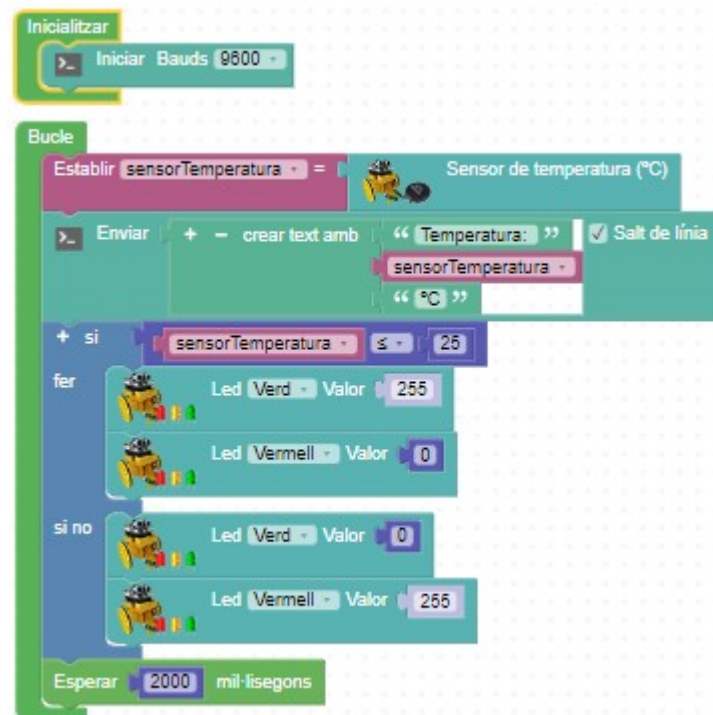
Finalment, falta afegir la comunicació amb la consola. Recordem que el que veiem per la pantalla ha de tenir el següent format: Temperatura: X°C (on X és un valor numèric). Per fer-ho, caldrà que abans que enviem el missatge a la consola creem conjuntament el missatge, ja que el missatge inclou tant text escrit com també el valor d'una variable.

Per fer-ho, podem usar un bloc de l'agrupació de Text que ens permet crear text a partir de diversos elements (fent clic al + augmenten opcions per afegir blocs i fent clic al – disminueixen).

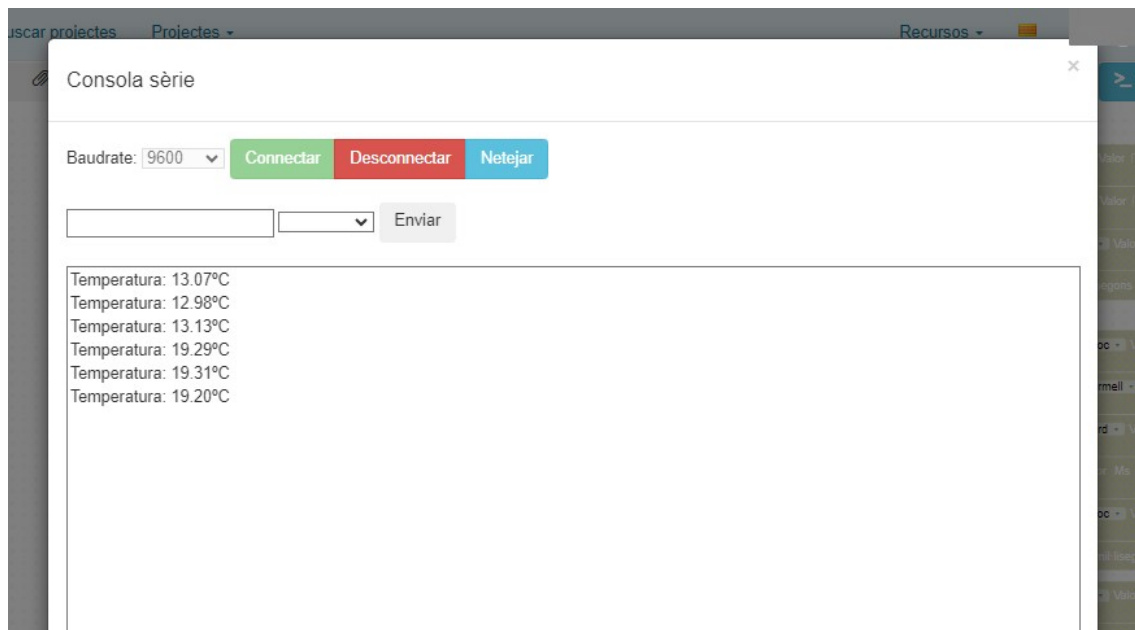


Aquest bloc l'haurem de posar dins l'espai buit del bloc que utilitzem per enviar a la consola, i per la dreta del bloc haurem d'introduir-hi un bloc de text on escriurem Temperatura: , un altre bloc que serà el valor de la variable del sensor de temperatura i un darrer bloc que serà un bloc de text on escriurem °C.

Finalment, amb la comunicació quedarà de la següent manera:



I a la consola hauríem de veure correctament el missatge escrit:



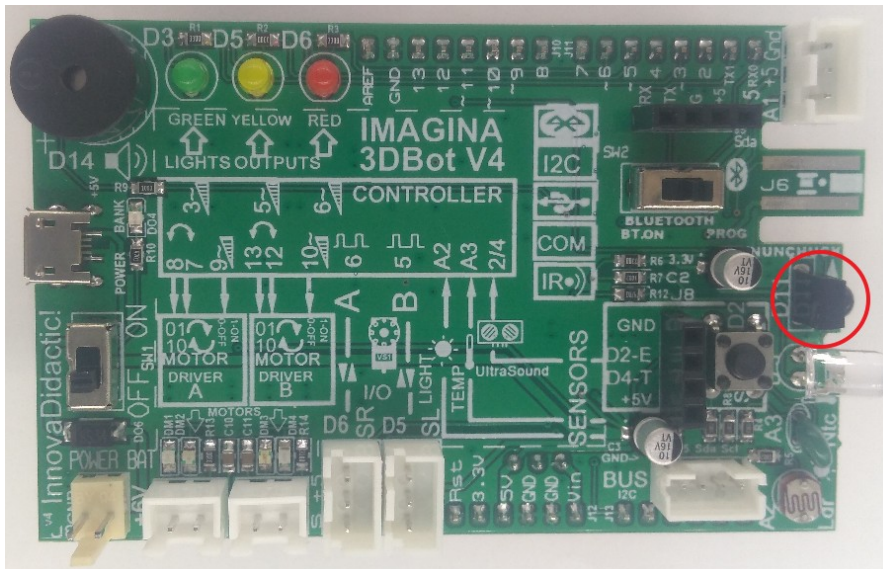
6.4.- Receptor IR

Una gran part dels electrodomèstics utilitzen comandaments a distància d'infraroigs, com les televisions o equips musicals.

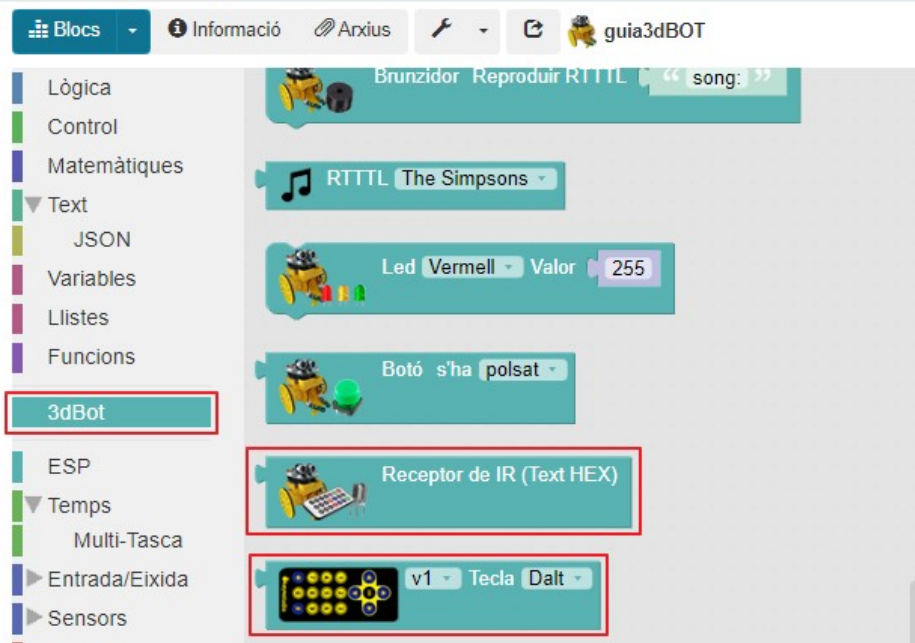
La placa Imagina 3dBOT té un receptor d'infraroigs (IR) que permet codificar els protocols de senyals de pulsos infraroigs utilitzats pels comandaments a distància. Els protocols detectats són: RC5, RC6, NEC, SONY, PANASONIC, JVC, SAMSUNG, WHYNTER, AIWA, LG, SANYO, MITSUBISHI i DENON. Això vol dir que detectaria qualsevol dels senyals emeses per algun d'aquests comandaments.

El comandament a distància conté un circuit intern, un processador i un LED que emet senyal infraroig.

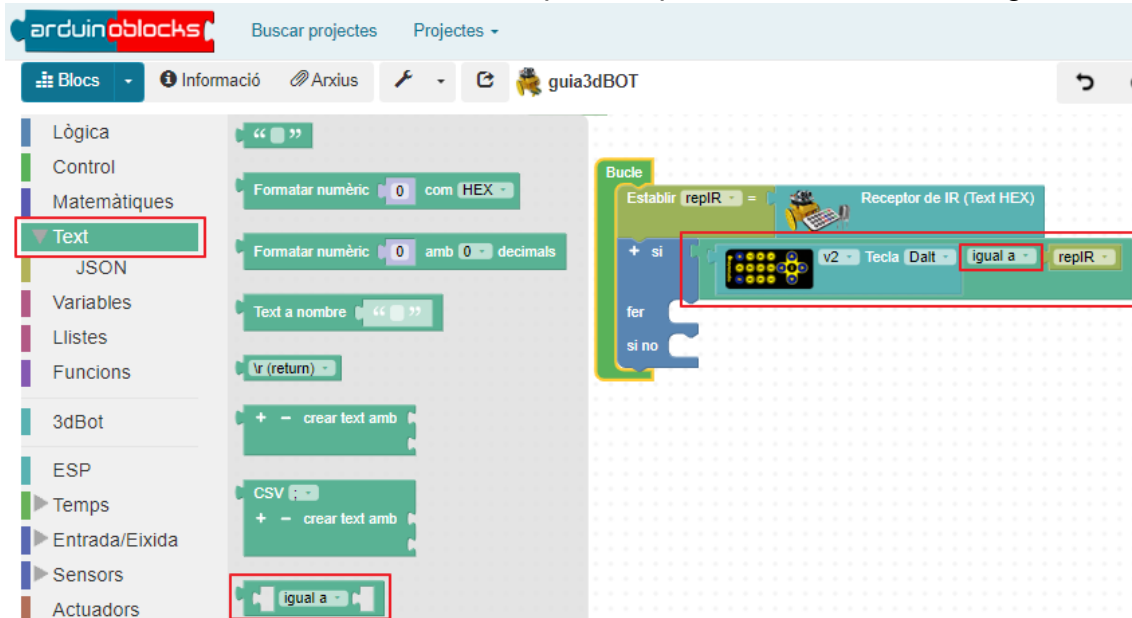




Amb la Imagina 3dBOT i ArduinoBlocks tenim simplificat el procés de manera que ja tenim blocs per identificar quina de les tecles del comandament s'ha premut. Per anar bé, haurem d'enfocar el comandament al receptor IR que conté la placa. Els blocs que utilitzarem serveixen el primer per assignar a una variable el valor que rebem del comandament i el segon per poder comparar amb el valor rebut del comandament quina tecla han premut. Utilitzarem la **V2** del comandament. El bloc també té un desplegable que permet seleccionar entre les diferents tecles.



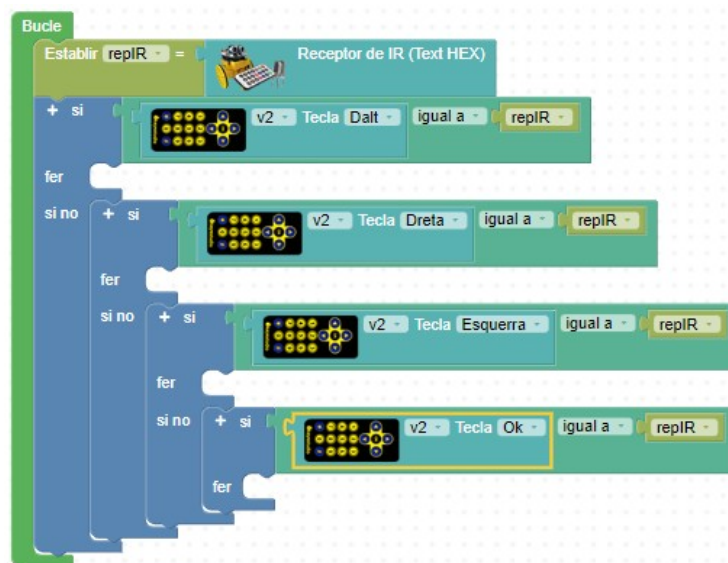
Abans de començar cal esmentar que la variable que crearem pel comandament no serà de tipus numèrica aquesta vegada, sinó que serà de tipus text. I que, per tant, per comparar el valor de la variable amb el bloc de la tecla seleccionada del comandament haurem d'utilitzar un bloc per comparar textos anomenat "igual a".



Podem posar d'exemple un programa ben senzill i que es pot fer tan complex com es vulgui utilitzant diferents funcions que hem anat explicant dels components de la placa. El programa el que haurà de fer és:

- Quan rebi que hem premut la tecla amunt del comandament s'haurà d'engegar el LED groc i apagar la resta de LEDs.
- Quan rebi que hem premut la tecla dreta del comandament s'haurà d'engegar el LED vermell i apagar la resta de LEDs.
- Quan rebi que hem premut la tecla esquerra del comandament s'haurà d'engegar el LED verd i apagar la resta de LEDs.
- Quan rebi que hem premut la tecla OK del comandament haurà de sonar la melodia del TETRIS i apagar els LEDs.

Primerament, haurem de crear i establir a la variable que reconeguem com a receptor d'IR el valor que rep el receptor IR. Posteriorment, caldrà que posem les condicions utilitzant els blocs de lògica pels si .. fer .. si no... i els blocs de text per comparar la variable del receptorIR amb la tecla del comandament corresponent.



I omplint amb les accions corresponents el codi queda de la següent manera:

