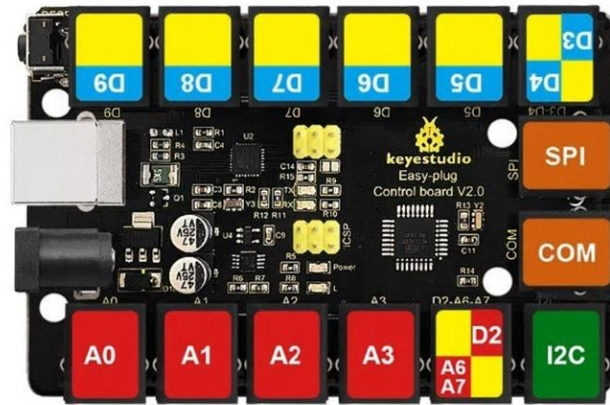


PROGRAMACIÓ DE LA PLACA EASYPLUG AMB ARDUINOBLOCKS



Elements i components EasyPlug

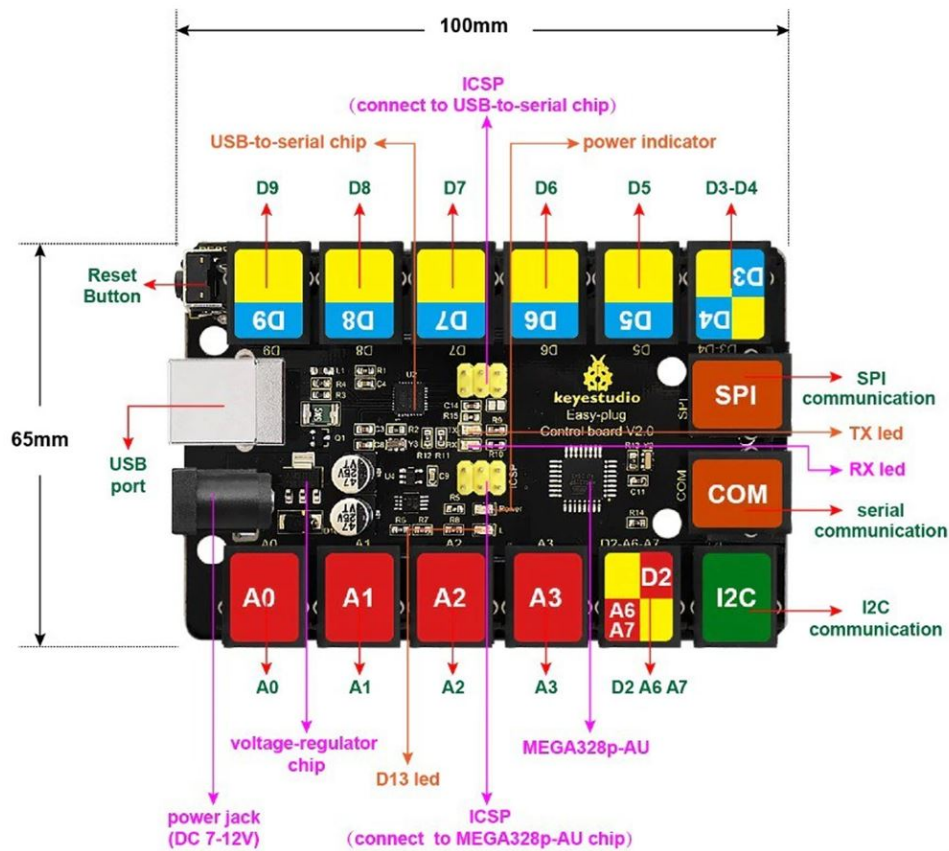
Què és la placa Easy Plug?

És una placa on podem connectar diferents sensors i actuadors de manera molt senzilla. A més, es poden crear diferents programacions, projectes i experiments molt interessants per les nenes i nens dels últims cursos d'Educació Primària com també, durant l'etapa d'educació secundària.

Quines característiques tècniques té la placa Easy Plug?

La placa Easy Plug, és compatible amb Arduino Uno. Té cinc ports digitals (d'entrada i sortida), quatre ports d'entrada analògica, un port SPI i un port I2C.

Els seus connectors són pins RJ-11 de quatre fils, fet que els fa molt més fàcils per connectar-los i evitar males connexions.



H:18n
only board weight : 55.

ArduinoBlocks: programa, accés i interfície

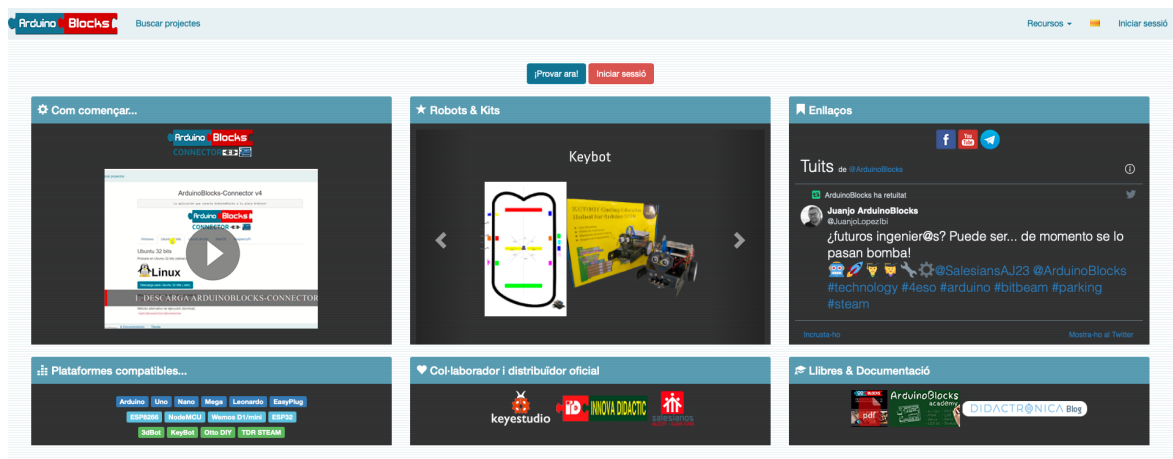
Què és Arduino?

Arduino és una plataforma de prototips de codi obert, basada en hardware i software flexibles i fàcils d'utilitzar. La seva finalitat és que tothom pugui generar projectes i entorns interactius mitjançant plaques i sensors compatibles amb Arduino.

Què és ArduinoBlocks?

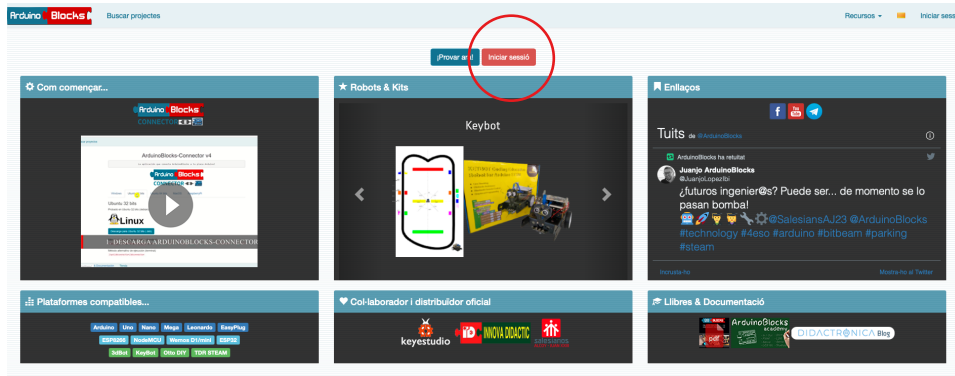
Arduino es programa amb llenguatge C++ i es necessita l'IDE (*Integrated Development Environment*), que permet escriure el codi. Programar amb C++ pot resultar complex i no és accessible per a tothom, per això, Juanjo López va crear ArduinoBlocks. Així doncs, ArduinoBlocks és un llenguatge de programació per blocs. Els diferents blocs de programació serveixen per llegir i escriure les entrades i sortides de la placa.

Trobem ArduinoBlocks, a la següent pàgina web: <http://www.arduinoblocks.com>. Ens podem registrar a la web, per tal de guardar els projectes, podem afegir informació als nostres projectes i veure projectes d'altres usuaris que hagin decidit compartir-ho de manera pública.

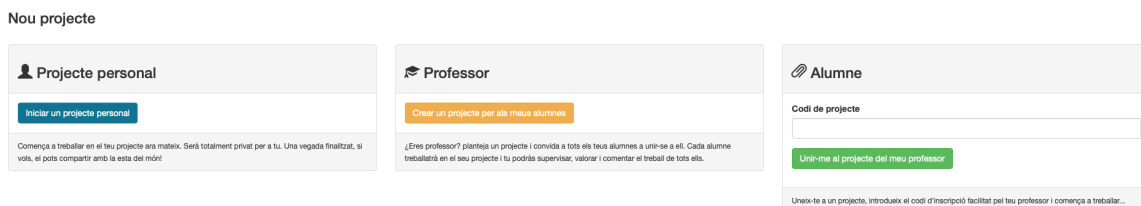


Com es treballa amb ArduinoBlocks?

Per a començar a treballar amb ArduinoBlocks, cal registrar-se i crear un nou usuari. Hem d'accedir al botó "Iniciar sessió" per a després, seleccionar l'opció de nou usuari.



Seguidament, cal seleccionar "Començar un projecte nou". Ens apareixen tres opcions: projecte personal, professor o alumne.



Si escollim el projecte personal només podrem accedir-hi nosaltres, però després el podem compartir si decidim fer-ho públic. En canvi, si seleccionem el projecte professor, no es comença un projecte sinó que es crea un codi perquè els alumnes puguin inscriure's en el projecte. D'aquesta manera, el professor pot supervisar les programacions dels seus estudiants. Finalment doncs, l'opció d'alumne és per unir-nos al projecte plantejat pel professor.

Com és la interfície de programació d'ArduinoBlocks?

Una vegada hem escollit l'opció de projecte personal, ens demana quina placa estem utilitzant i quin nom volem donar al nostre projecte. A la vegada, podem afegir informació, com: descripció del projecte, components que necessitem i altres especificacions a l'apartat de comentaris.

En el nostre cas escollirem: **Keyestudio EasyPlug**.

Nou projecte personal

Tipus de projecte: Keyestudio EasyPlug

Nom: Introducció

Descripció: Normal | A | B | I | U | S | | | | | | | |

Components: Normal | A | B | I | U | S | | | | | | | |

Comentaris: Normal | A | B | I | U | S | | | | | | | |

Nou projecte

Una vegada entrem al nou projecte, ens trobem amb la següent pantalla:

Eines per a programar

Espai per a programar el nostre projecte

Fer captura de pantalla

Guardar el projecte

Pujar el programa a la placa Arduino

Mostrar la consola serie.

ArduinoBlocks Connector

ArduinobBlocs genera el codi d'Arduino a partir dels blocs. El programa copil·la i puja la nostra programació a la placa gràcies a l'aplicació ArduinoBlocks Connector. Si no executem ArduinoBlocks Connector, podrem programar i accedir a la plataforma ArduionBlocks, però no podrem pujar el nostre programa a la placa. Així doncs, cal instal·lar-lo al nostre ordinador, i executar-lo quan fem ús de la plataforma.

Actualment, hi ha la versió 5 i està disponible per a diferents sistemes operatius: Windows, Ubuntu, MacOS, Chromebook i Raspberry Pi.

ArduinoBlocks Connector v5

L'aplicació que connecta ArduinoBlocks a la teua placa Arduino!



Sensors

Què és un sensor?

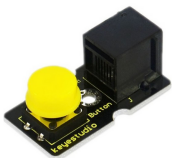
Els sensors permeten recollir informació perquè sigui processada per a la placa i s'accionin certes programacions segons la nostra finalitat.

Quins sensors hi ha?

Hi ha diversos tipus de sensors. Per exemple, si anem a l'apartat de "sensors" del menú d'eines d'EasyPlug, n'hi ha diversos. Alguns com:



Quins sensors programarem?



Polisador



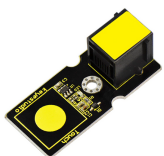
Nivell de llum (LDR)



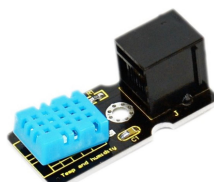
Detector de moviment (PIR)



Sensor de colp



Polisador tàctil



Temperatura i humitat (DHT-11)



Potenciòmetre

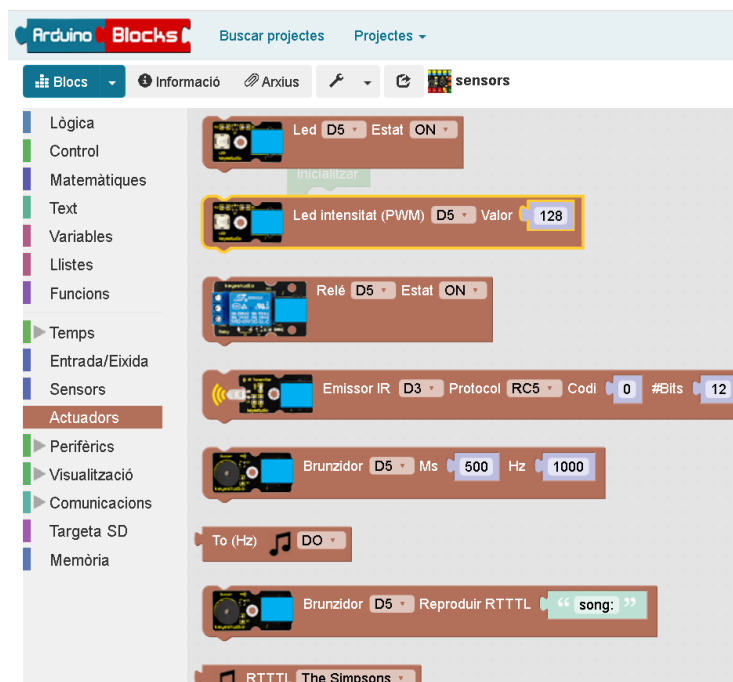
Actuadors

Què és un actuator?

Els actuadors s'accionen quan la placa processa la programació, que aquesta pot estar influïda per la recollida de dades d'algun sensor.

Quins actuadors hi ha?

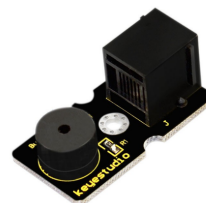
Hi ha diversos tipus d'actuadors. Per exemple, si anem a l'apartat d'"actuadors" del menú d'eines d'EasyPlug, n'hi ha diversos. Alguns com:



Quins actuadors programarem?



Led
Led Intensitat (PWM)



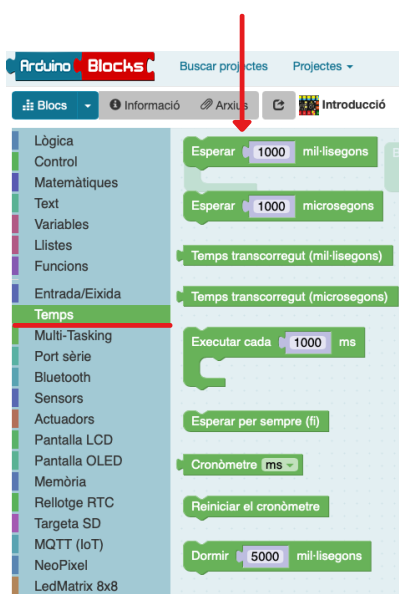
Brunzidor

A01: LED

En aquesta primera pràctica amb EasyPlug aprendrem a encendre un LED i a programar-lo perquè s'apagui i s'encengui durant un determinat nombre de temps.

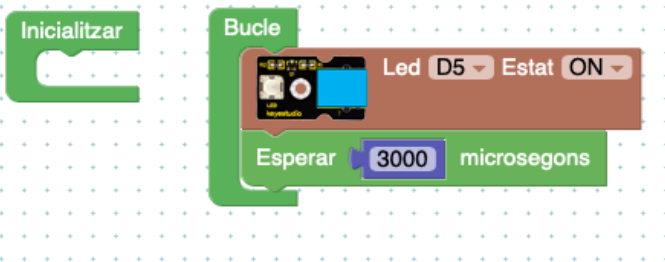


El LED és un actuator. Per tant, el trobarem a l'apartat d'actuadors. Si necessitem programar un temps, haurem d'anar a l'apartat de "Temps".



PRÀCTICA A01.1:

- Encendre un LED



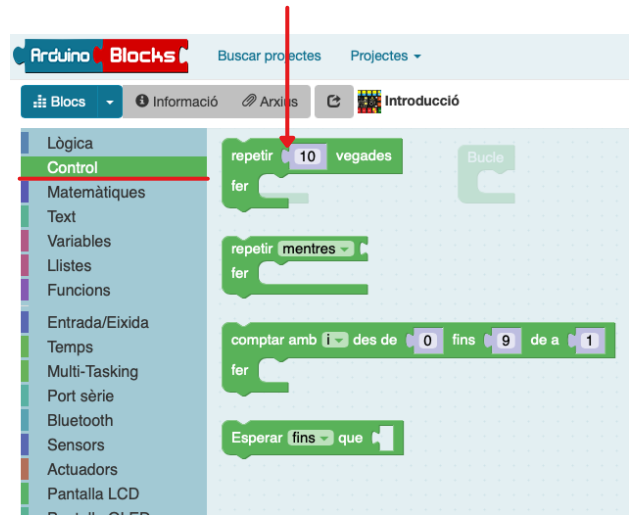
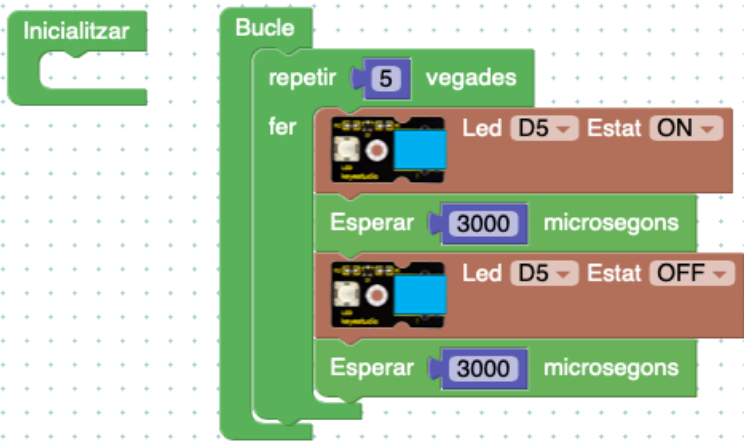
PRÀCTICA A01.2:

- Encendre un LED, esperar 3 segons i apagar el LED



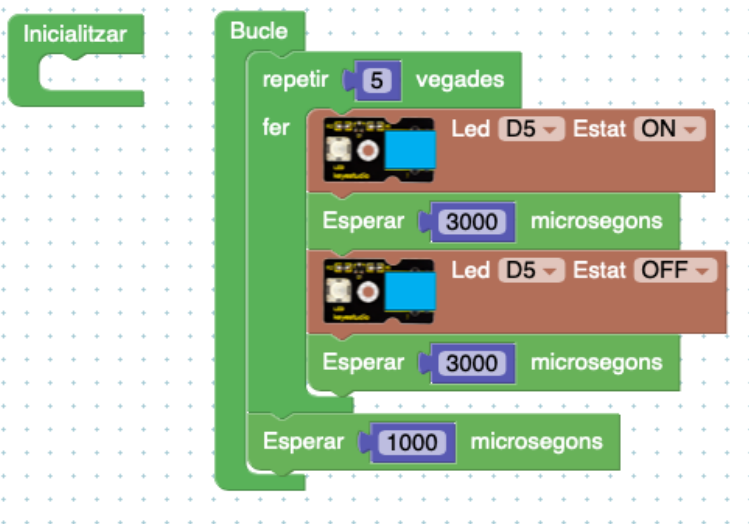
PRÀCTICA AO1.3:

- Repetir l'acció anterior 5 vegades



PRÀCTICA AO1.4:

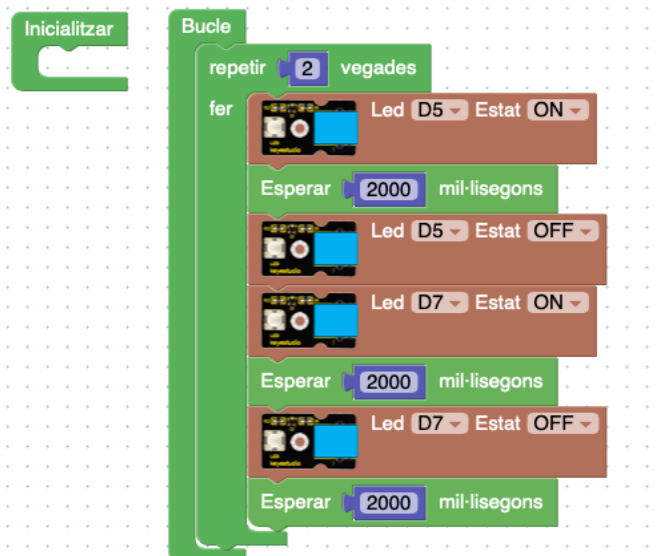
- Repetir l'acció anterior 5 vegades però esperar 1 segon entre repeticions.



PRÀCTICA AO1.5:

Connectem dos LEDS a la placa:

- LED 1: D5
- LED 2: D7
- S'encén LED 1 (D5) esperem 2 segons, s'apaga el LED 1 i s'encén el LED 2 (D7) esperem 2 segons i s'apaga. Aquesta acció es repeteix 3 vegades.



A02: Control de la intensitat del LED utilitzant el PWM

Continuant amb l'ús dels LEDs, controlarem la intensitat d'un LED utilitzant el PWM, que significa pulsewith modulation.

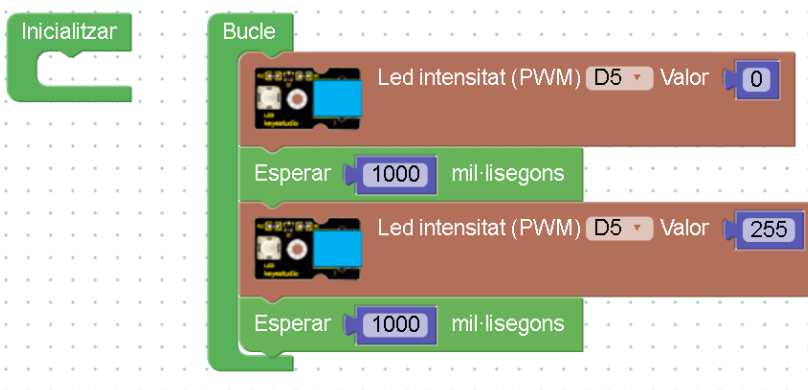
Les sortides de voltatge d'Arduino només tenen dos estats: ON-OFF, és a dir, una correspon a una sortida de 5 V (ON) i una de 0 V (OFF). Si només fem ús d'aquestes dues condicions, només podem fer activitats com encendre i apagar un LED. Gràcies al PWM podem programar un rang de valor de 0 a 255 que oscil·la entre els 0 V i els 5 V. D'aquesta manera podem controlar la intensitat d'un LED.

Troblem el bloc de programació a l'apartat d'actuadors:



PRÀCTICA A02.1:

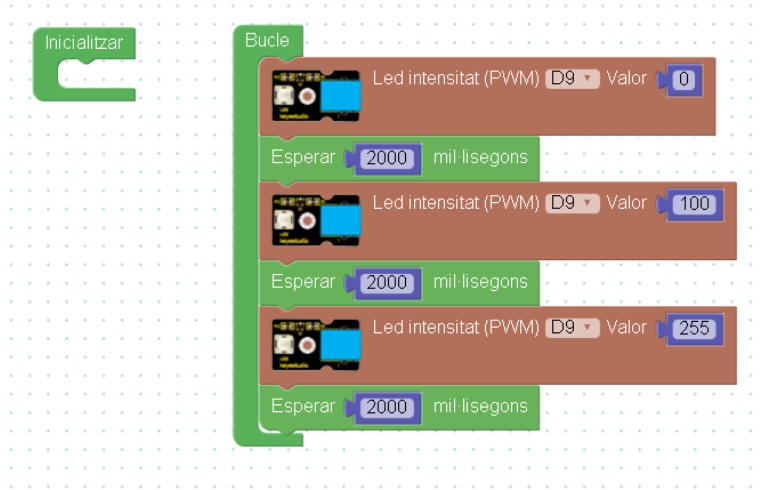
- Programem el LED en valor 255 perquè s'encengui i després d'un segon, s'apagui. És a dir, en valor 0.



PRÀCTICA A02.2:

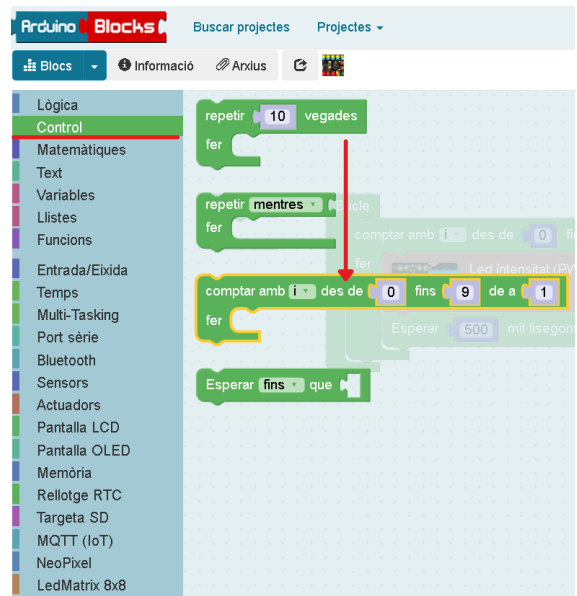
Ara farem que la intensitat del LED canviï.

- Cada 2000 mil·lisegons la intensitat del LED ha de canviar, augmentant de 0 a 100 i 255.



PRÀCTICA A02.3:

Per a poder programar d'una manera més pràctica i efectiva, podem fer ús del bloc situat a l'apartat de "Control". Aquest permet comptar un determinat número fins a un altre. A la vegada, podem decidir amb quina freqüència, és a dir, cada dos números o cada deu.



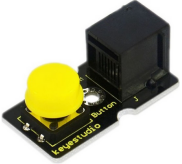
En aquest bloc hi ha un nou concepte que és molt utilitzat en la programació: **variables**. En aquesta activitat les farem servir de manera molt ràpida, però més endavant seran eines necessàries perquè la nostra programació tingui sentit. Justament, en el nostre bloc s'anomena "i".

- La intensitat del LED ha d'augmentar de 25 en 25, de 0 a 255.

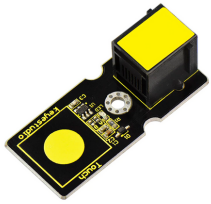


A03: Encendre un LED amb el pulsador

Continuant amb l'ús de LED, en les següents pràctiques combinarem dos pulsadors: el tàctil i el pulsador.

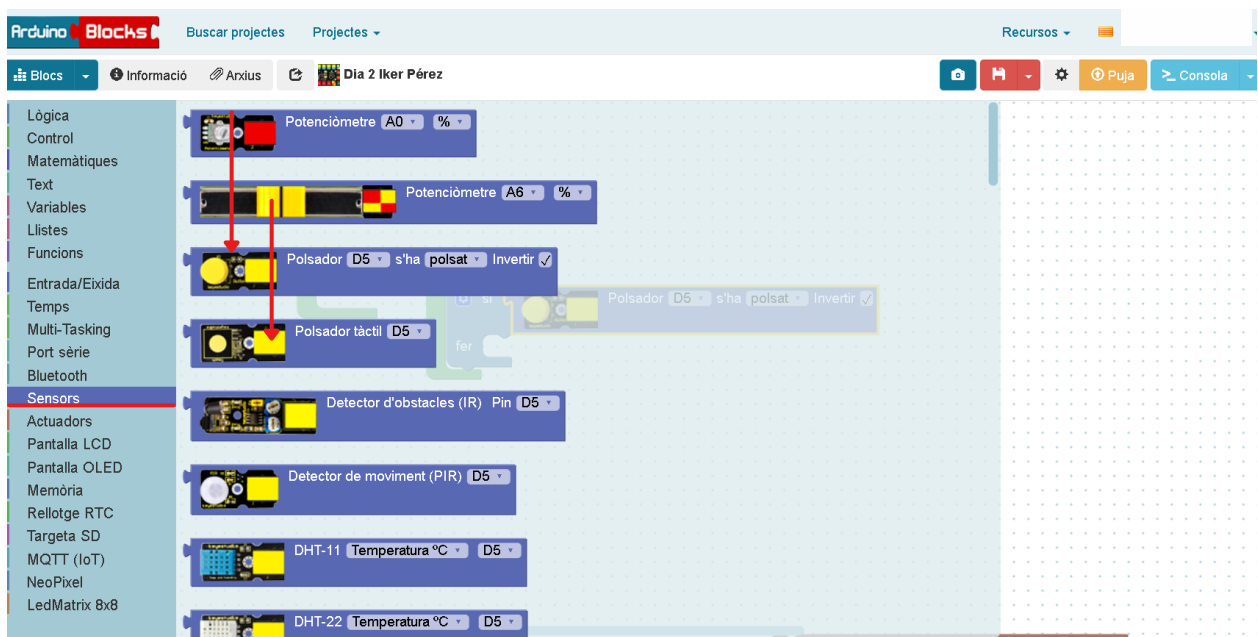


El pulsador és un sensor digital, que té dos estats. Quan es pressiona el botó, emet un senyal d'alt nivell, és a dir, 5 V. En canvi, quan es deixa anar el botó, emet un senyal de baix nivell: 0 V.

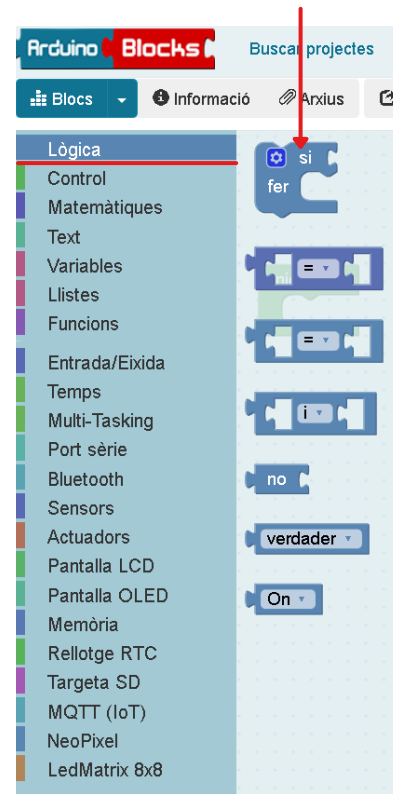


El pulsador tàctil és un sensor tàctil, tal com explica el seu nom pot "sentir" el tacte de manera molt sensible.

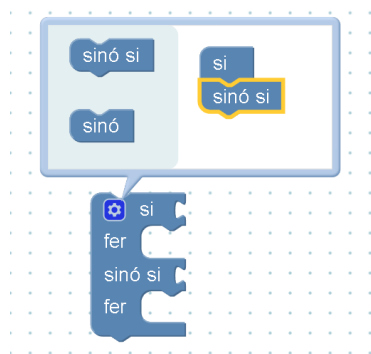
Els dos pulsadors són sensors, per tant, els trobarem a l'apartat "Sensors". En el moment de la programació és molt important tenir en compte a quin port l'hem connectat. Sempre es connecten els ports digitals, però cal programar el número correcte.



A la vegada també començarem a fer ús de les funcions de "Lògica", com la de condició: "si ... fer". Aquest bloc de programació és un dels pilars fonamentals en el món de la programació, ja que permet avaluar estats, i segons la condició, programar accions.



A l'apartat de condicions es pot introduir factors com: estat dels sensors, comparacions, igualtats, operacions matemàtiques. En canvi, a l'apartat d'accions, podem programar: encendre un LED, enviar missatges a la consola, escriure quelcom a alguna pantalla externa, etc.



També, es pot ampliar el bloc amb més condicionals. Fent clic al símbol d'engranatge, ens permet dues opcions:

- Sinó (és a dir, totes les accions que siguin diferents de la programada a la primera condició)
- Sinó si (una altra condició diferent de la primera programada)

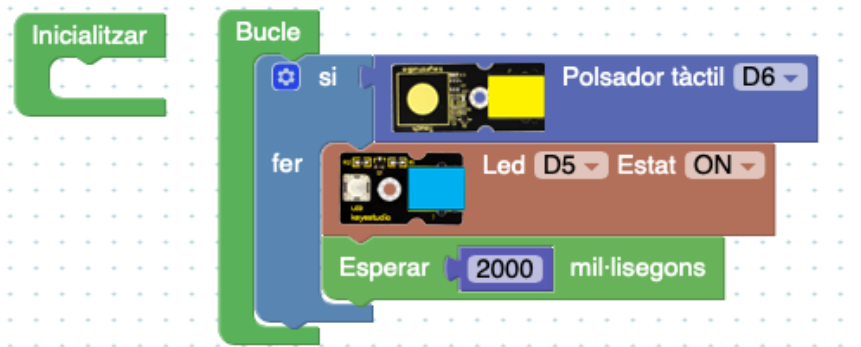
PRÀCTICA A03.1:

- Si fem clic al sensor polsador que el LED s'encengui durant 2 segons.



PRÀCTICA A03.2:

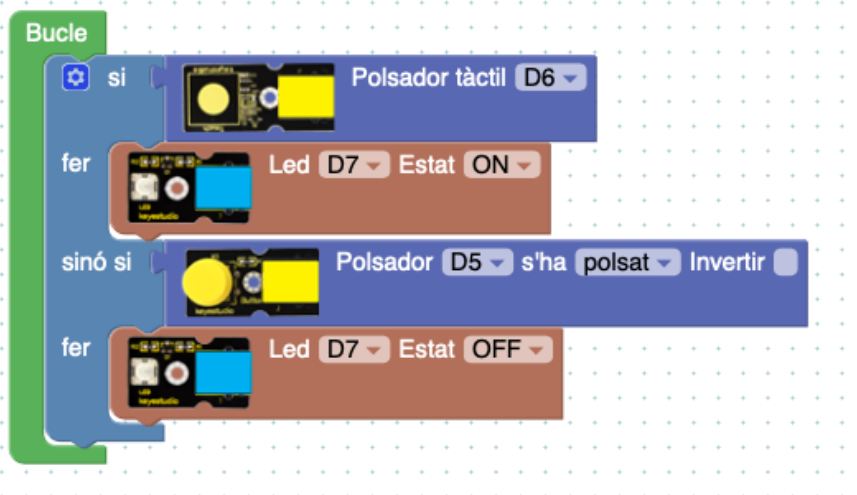
- Si fem clic al sensor polsador tàctil que el LED s'encengui durant 2 segons.



PRÀCTICA A03.3:

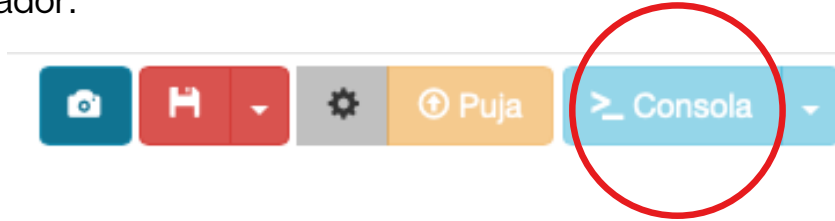
Ara utilitzarem els dos polsadors:

- Si fem clic al polsador tàctil s'encén el LED. En canvi, si fem clic a l'altre polsador, s'apaga.



A04: Consola

En aquesta pràctica treballarem amb la consola. És una funció del programa ArduinoBlocks que podem programar perquè s'envii informació a la pantalla del nostre ordinador.



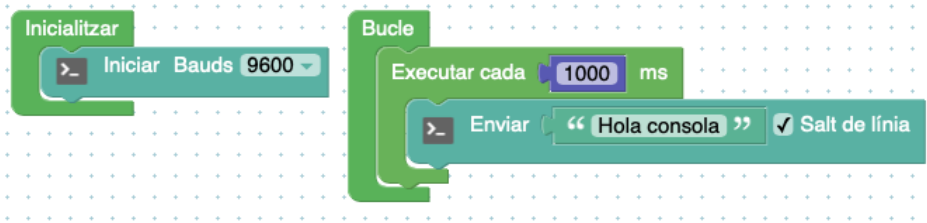
Per a poder "enviar" aquesta informació necessitem inicialitzar el "Port sèrie". Per tant, dins del bloc inicialitzar posarem: "Iniciar Bauds". En aquest cas, per enviar missatges a la consola amb el valor 9600 és suficient.

The screenshot shows the ArduinoBlocks IDE interface. On the left, a library panel lists various categories, with 'Port sèrie' (Serial) highlighted. A block 'Iniciar Bauds 9600' is selected from this category. The main workspace contains a sequence of blocks: 'Iniciar Bauds 9600', 'Fixar timeout 1000', 'Enviar' (with 'Salt de línia' checked), 'Enviar byte 0', 'Dades rebudes?', 'Rebre text' (with 'Fins salt de línia' checked), 'Rebre byte', 'Rebre com número' (with 'Fins salt de línia' checked), 'Esdeveniment al port Serial', and 'Plotter' (with 'Valor 0'). A 'Bucle' (Loop) block encloses the 'Rebre' blocks. A 'Inicialitzar' block contains the 'Iniciar Bauds 9600' block. In the bottom right, a terminal window titled 'ArduinoBlocks :: Consola sèrie' is open, showing a baudrate of 9600 and buttons for 'Connectar', 'Desconnectar', 'Netejar', and 'Enviar'.

PRÀCTICA A04.1:

Començarem aquesta primera activitat, enviant un missatge a la consola, que digui "Hola consola". Cal inicialitzar el Port Sèrie, fent servir els Bauds.

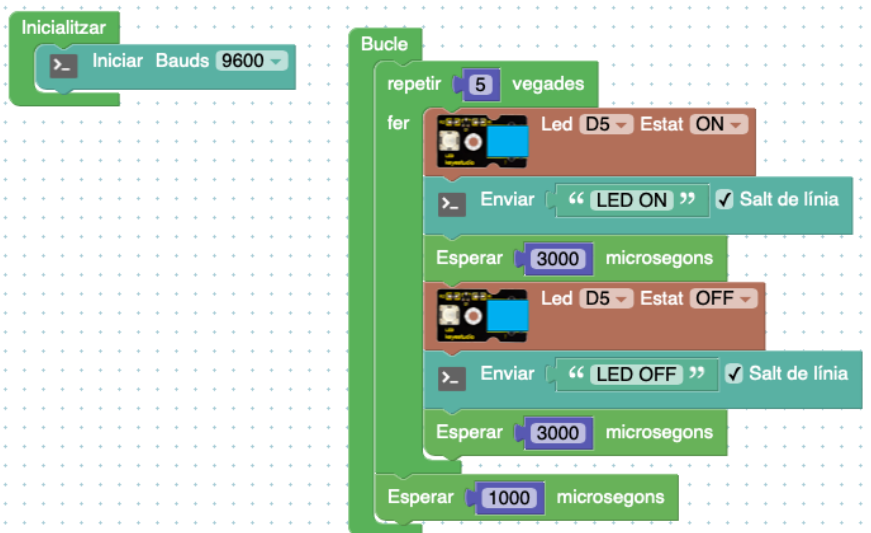
- Cada segon, a la consola ha d'aparèixer "Hola consola" amb salt de línia.



PRÀCTICA A04.2:

Ara que sabem enviar un missatge a la consola, combinarem diferents blocs de programació: control, actuadors, port sèrie i temps.

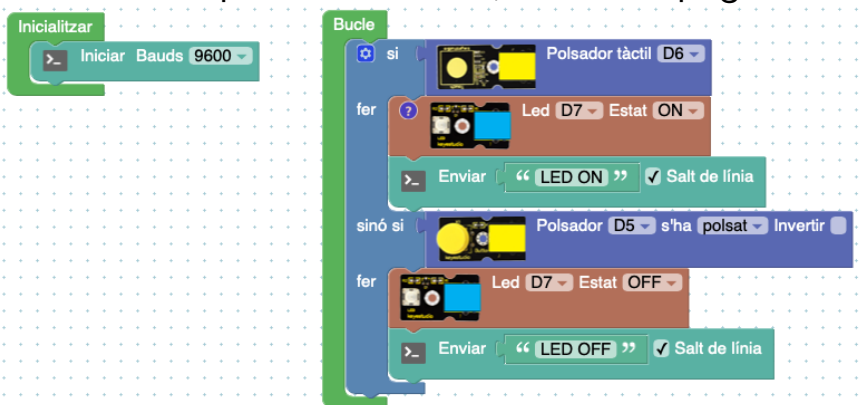
- Aquest programa es repeteix durant 5 vegades i entre repetició i repetició hi ha 1 segon d'espera: s'encén el LED i automàticament s'envia a la consola "LED ON", durant 3 segons. Després s'apaga, i a la consola apareix "LED OFF" durant 3 segons.



PRÀCTICA A04.2:

Ara programarem fent servir els blocs de lògica.

- Si fem clic al polsador tàctil, el LED s'encèn i a la consola apareix "LED ON". En canvi, si fem clic al polsador normal, el LED s'apaga i a la consola s'escriu: "LED OFF".



A05: Sensor LDR

En aquesta cinquena pràctica aprendrem a programar el Sensor d'intensitat de la llum o com s'anomena a ArduinoBlocks: Nivell de LLUM (LDR). Aquest sensor, cal connectar-ho a un port analògic.



El sensor Nivell de LLUM (LDR), també anomenat sensor Fotocèl·lula, és molt comú a la nostra vida quotidiana. Per exemple, quan s'encenen els fanals a la nit, les llums solars de jardí, els detectors de diners...

És un sensor analògic que obté valors entre 0 i 5 V, concretament de 0 a 1023.

En l'apartat de blocs de programació, es troba a "sensors".



PRÀCTICA A05.1:

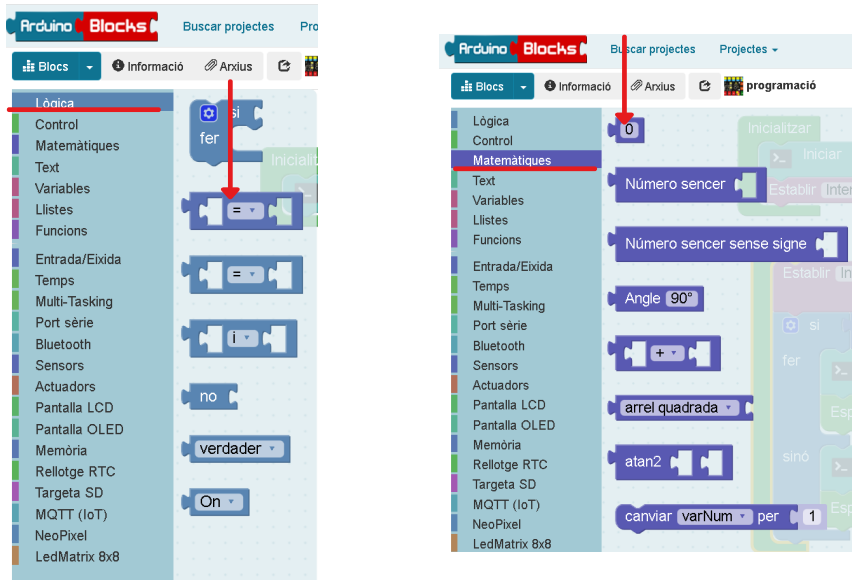
Abans de començar la programació, necessitem conèixer quins valors marca el sensor en el lloc on treballem. És a dir, no podem programar sense conèixer el % d'intensitat de llum que tenim al nostre entorn. Una vegada obtingut aquest valor, podrem començar les programacions més complexes. Així doncs:

- Visualitza a la consola el % d'intensitat cada 3 segons.



PRÀCTICA A05.2:

Per a realitzar aquesta segona pràctica, necessitem conèixer el bloc de programació que ens permet igualar o comparar si un número és més gran o més petit que un altre. A més, farem servir el bloc on ens permet escriure números. A les següents imatges podeu conèixer els diferents blocs de programació:

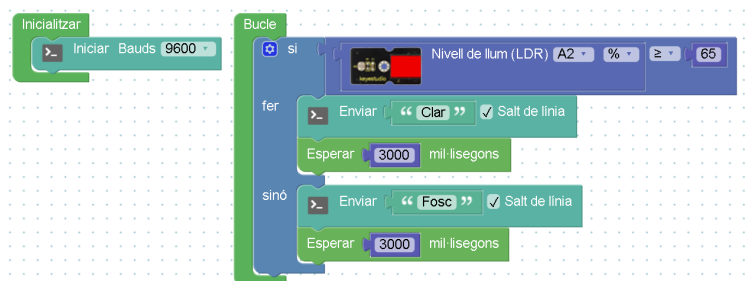
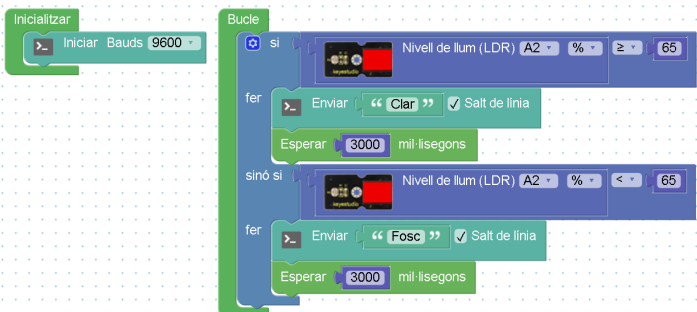


- Si la intensitat de llum és superior a X, en el nostre cas 65%, que a la consola aparegui "Clar".



PRÀCTICA

- Si la intensitat de llum és superior a X, en el nostre cas 65%, que la consola escrigui "Clar". En canvi, si és inferior al valor anterior, que la consola escrigui

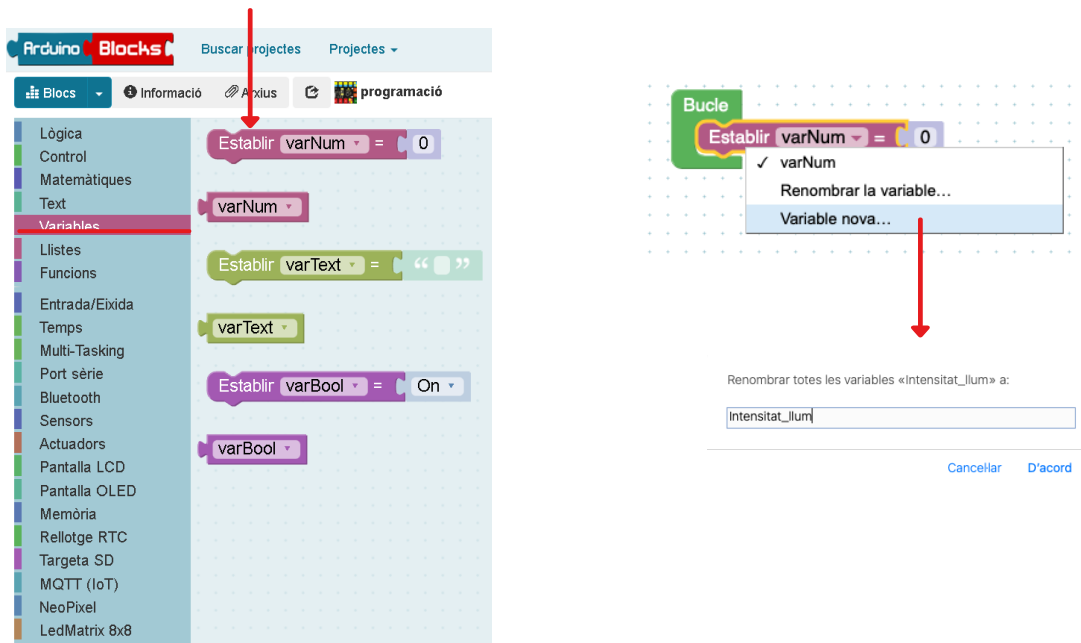


Fixa't que hi ha dues maneres de programar-ho, la de l'esquerra necessites especificar quina condició ha de succeir perquè la consola escrigui "Fosc". En canvi, la segona, s'entén que tot el que no sigui la primera condició, serà la segona.

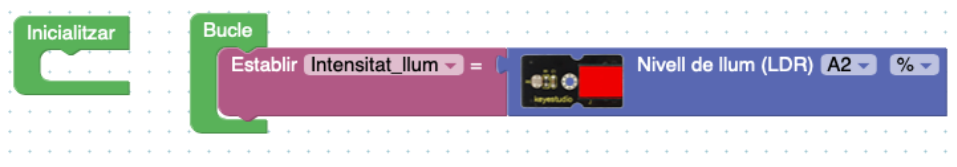
PRÀCTICA A05.4:

En aquesta quarta pràctica tornarem a treballar amb les variables. Les variables són elements molt comuns en programació. Quan creem una variable estem donant un nom a una dada o una lectura. Per exemple, quan el sensor d'intensitat de la llum detecta el % de llum de l'habitació, tota la mesura dels diferents valors els podem agrupar amb la variable nomenada "Intensitat de la llum". No és obligatori utilitzar variables, però és una manera molt més còmoda i entenedora.

Per a crear una nova variable, hem d'anar a l'apartat de variables i establir-ne una. És a dir, assignarem un nom a un conjunt de valors comuns.



Cal establir la variable a bucle, i associar-la a un sensor que ens pugui agrupar els valors:



Repetirem els mateixos exercicis que les anteriors pràctiques però utilitzant la variable en comptes del bloc del sensor.

- Envia a la consola el % d'intensitat de llum cada 3 segons. A la consola ha d'aparèixer escrit de la següent manera: Intensitat llum: 65,00

Per a fer-ho, podem utilitzar aquest nou bloc:



ArduinoBlocks :: Consola sèrie

Baudrate: 9600 Connectar Desconnectar Netejar

Enviar

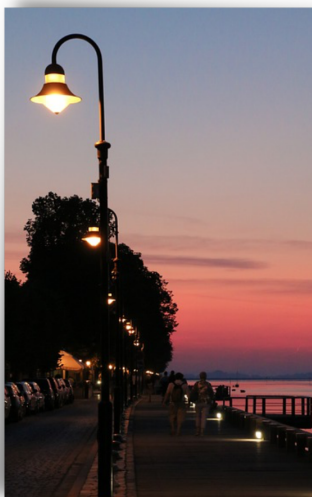
```

Intensitat llum:17.00
Intensitat llum:21.00
Intensitat llum:22.00
Intensitat llum:22.00
Intensitat llum:21.00
Intensitat llum:17.00
Intensitat llum:17.00
Intensitat llum:18.00
Intensitat llum:17.00
Intensitat llum:16.00
Intensitat llum:18.00
Intensitat llum:17.00
Intensitat llum:36.00
Intensitat llum:35.00
Intensitat llum:35.00
Intensitat llum:35.00
                    
```

PRÀCTICA A05.5:

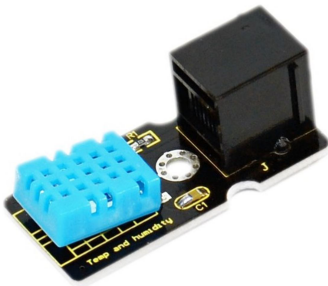
En aquest últim projecte fent servir el sensor de llum, imitarem què passa quan a la nit els fanals del carrer s'encenen. Per tant, utilitzarem un LED i el sensor.

- Exemplifica el que succeeix a la nit amb els fanals del carrer. És a dir, quan la intensitat de la llum sigui inferior o igual a 5%, que el LED s'encengui. En canvi, quan sigui superior a 5%, el LED s'apaga.



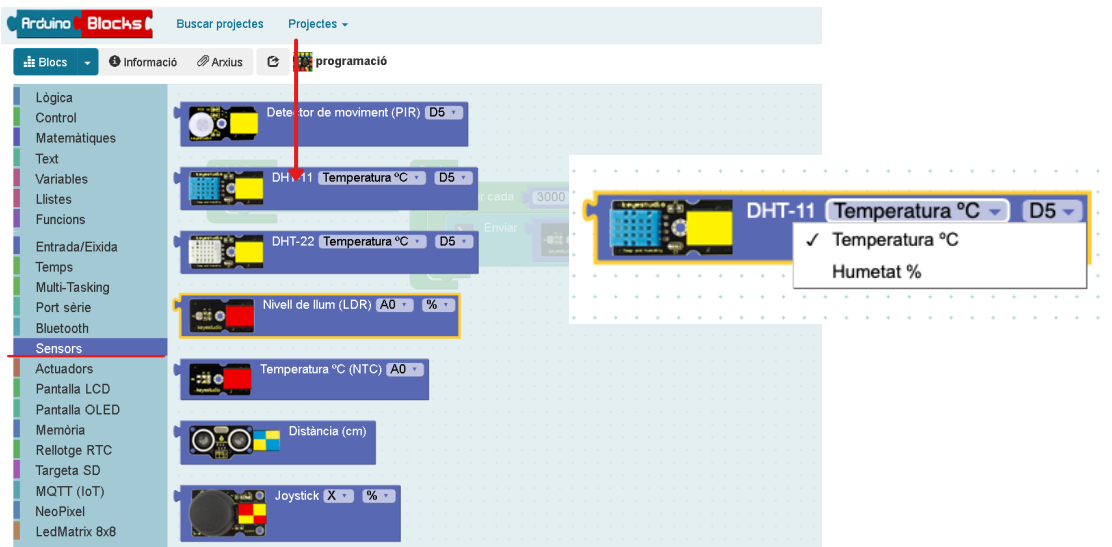
A06: Sensor DHT11

En aquesta sisena pràctica aprendrem a programar el Sensor de temperatura i humitat o com s'anomena a ArduinoBlocks: DHT11. Aquest sensor, cal connectar-ho a un port digital.

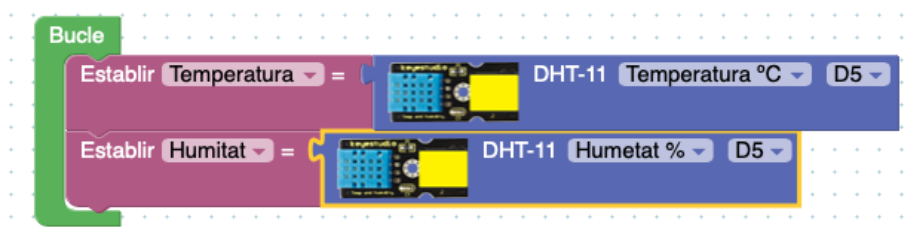


El sensor DHT11, realment és un sensor de temperatura i humitat. Té una sortida de senyal digital, que funciona en un rang de temperatures entre 0 i 50 °C, amb un error de 2°C i un rang d'humitat entre 20 i 90%, amb un error d'un 5%.

En l'apartat de blocs de programació, es troba a "sensors".

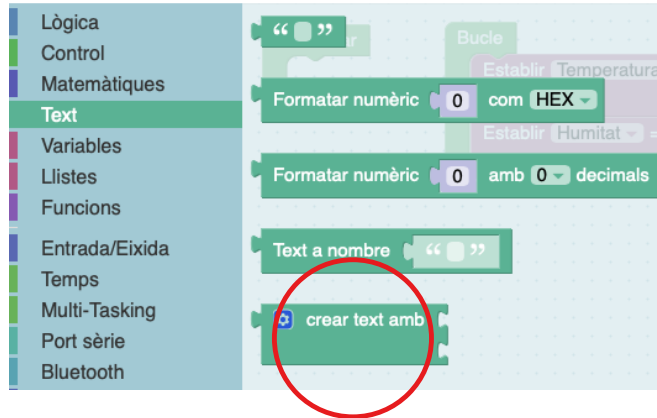


Abans de començar la pràctica, establim dues variables noves a bucle: una per a la temperatura i l'altra per a la humitat.

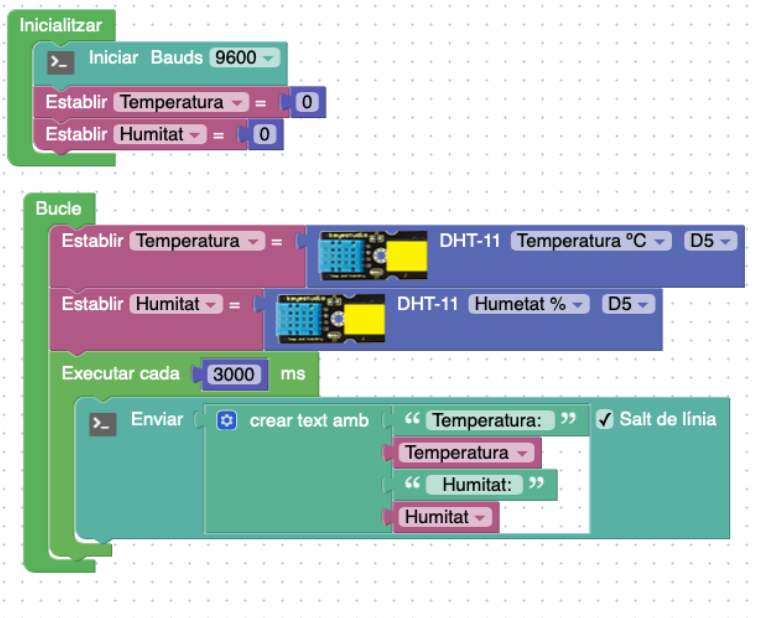


PRÀCTICA A06.1:

Aquesta primera pràctica enviarem els valors a la consola, és a dir, els graus centígrads i el tant per cent d'humitat de l'habitació on estem. Recorda i fes servir el bloc de programació "crear text amb", el qual et permet afegir diferents missatges en una mateixa línia de text.



- Envia a la consola els valors de temperatura i humitat cada 3 segons. A la consola hi ha d'aparèixer:
 - Temperatura: X Humitat: X



ArduinoBlocks :: Consola sèrie



```

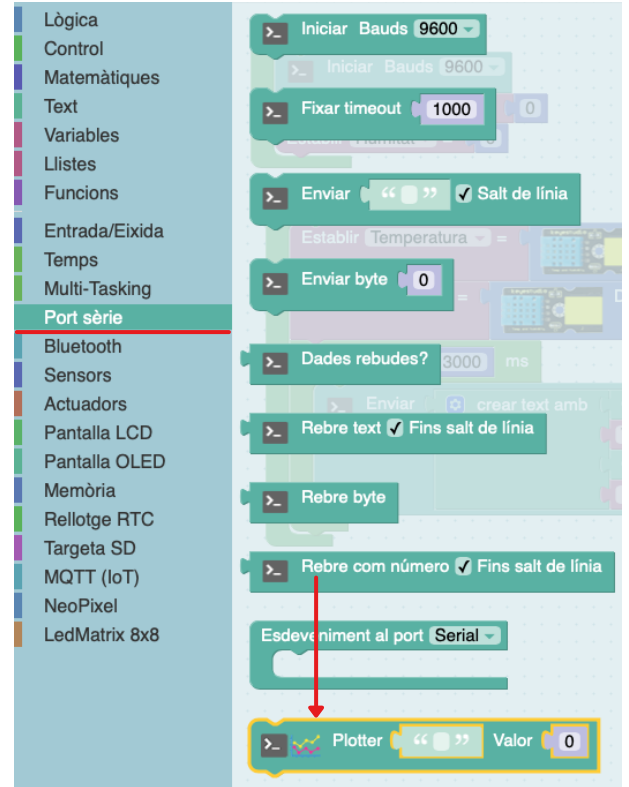
Temperatura: 18.00 Humitat: 87.00
Temperatura: 18.00 Humitat: 87.00
Temperatura: 18.00 Humitat: 86.00
Temperatura: 19.00 Humitat: 86.00
Temperatura: NAN Humitat: NAN
Temperatura: 19.00 Humitat: 93.00
Temperatura: 19.00 Humitat: 93.00
Temperatura: 19.00 Humitat: 93.00
Temperatura: 20.00 Humitat: 93.00
    
```

PRÀCTICA A06.2:

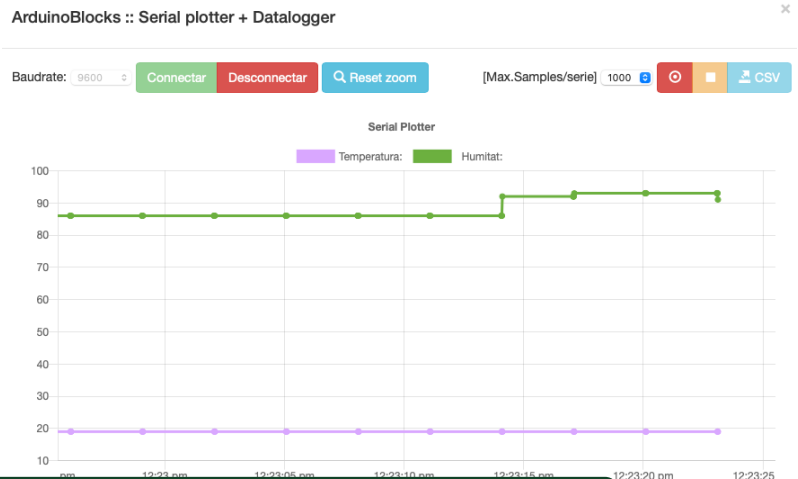
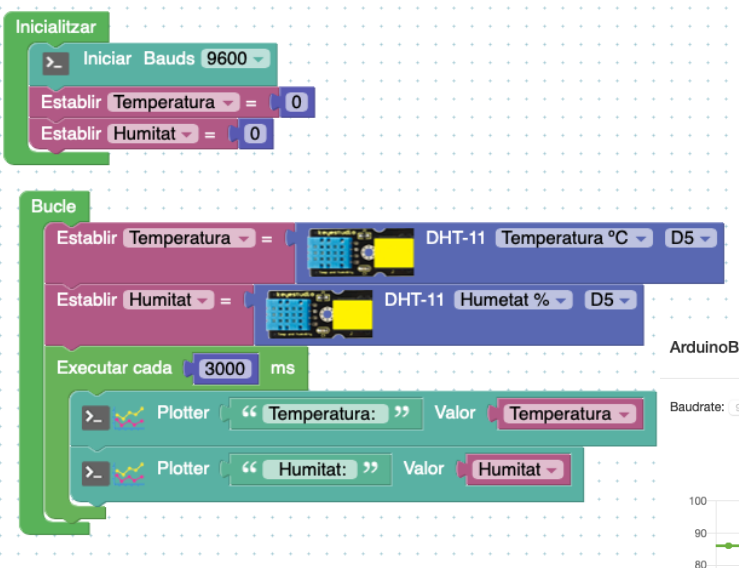
Seguidament, treballarem amb l'eina Plotter. Aquesta és semblant a la consola, de fet es troba al desplegable de consola. Aquesta eina crea a temps real una estadística dels valors obtinguts.



Per a programar-ho, ho trobarem a l'apartat de Port sèrie.



- Envia cada 3 segons a Serial Plotter els valors de temperatura i humitat.

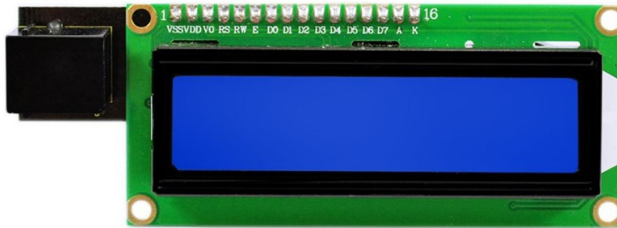


PRÀCTICA A06.3:

Una altra manera de poder visualitzar valors és fent ús de pantalles externes, és a dir, pantalles connectades a la placa Easy Plug.

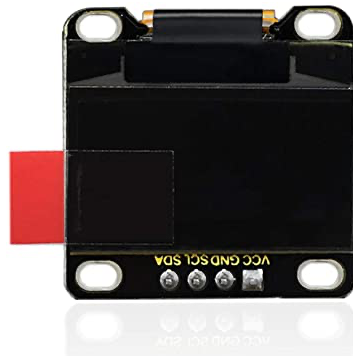
Tal com podem observar a l'apartat de programació tenim tres pantalles:

- Pantalla LCD: És una pantalla LCD de 16 caràcters per 2 línies amb fons blau i llum blanca. Es connecta al port I2C.

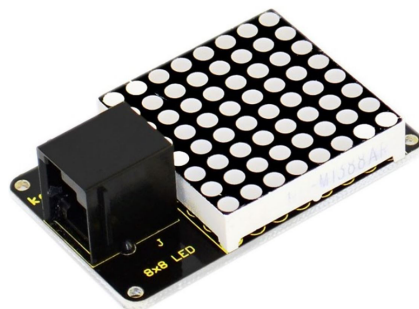


- Lògica
- Control
- Matemàtiques
- Text
- Variables
- Llistes
- Funcions
- Entrada/Eixida
- Temps
- Multi-Tasking
- Port sèrie
- Bluetooth
- Sensors
- Actuadors
- Pantalla LCD**
- Pantalla OLED
- Memòria
- Rellotge RTC
- Targeta SD
- MQTT (IoT)
- NeoPixel
- LedMatrix 8x8

- Pantalla OLED: El nom és una abreviatura a "Diodo Emisor de llum orgànic". Una pantalla OLED és una matriu de LEDs orgànics que s'il·luminen quan emeten energia.



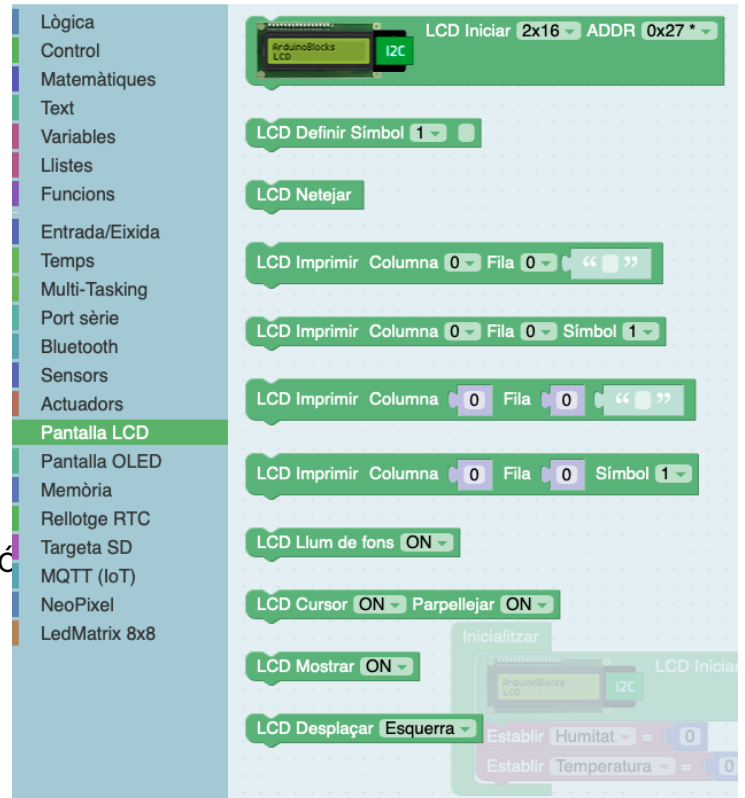
- LedMatrix 8x8: És una matriu de LEDs de 8x8.



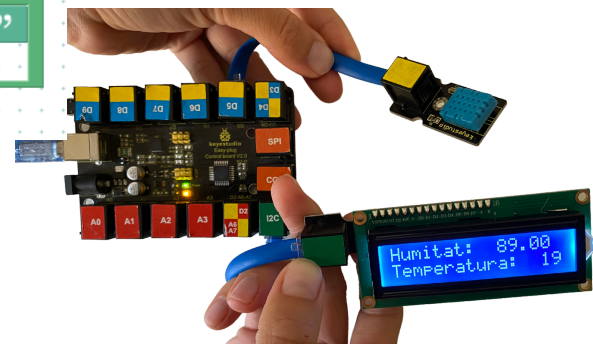
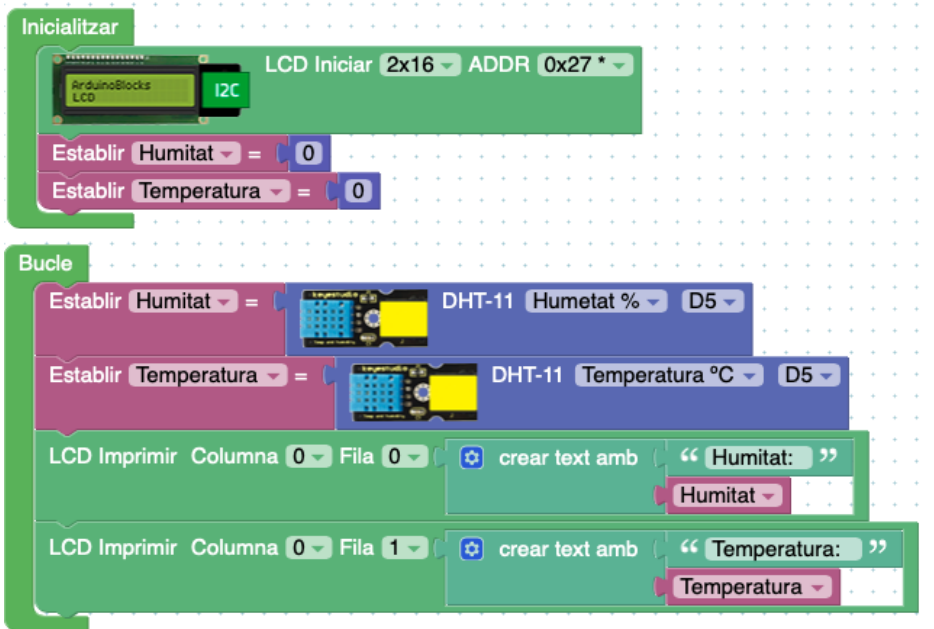
En aquesta pràctica A06.3. enviarem les mateixes dades que hem estat enviant a la Consola i el Serial Plotter a la nostra pantalla LCD. Així doncs, cal que connectis la placa al port I2C.

Els blocs de programació de la pantalla són els següents de la imatge. Cal seleccionar el primer, i col·locar-lo al bloc de "Inicialitza".

Els blocs que ens serviran per a enviar la informació a la placa són els següents. És important que estableixis quina fila vols que es trameti la informació. El número 0, ja compta com a una opció.

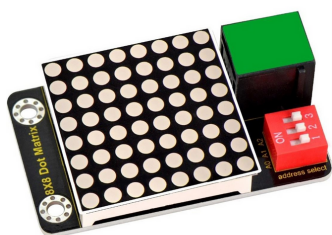


- Envia a la pantalla LCD els valors de temperatura i humitat cada 3 segons.



A07: Led Matrix 8x8

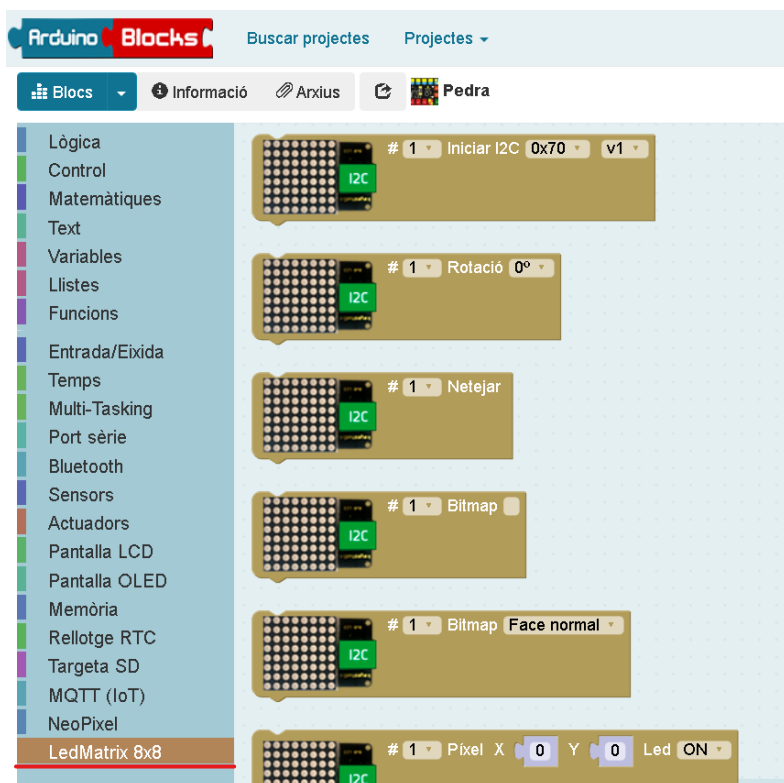
En aquesta pràctica l'objectiu és conèixer la LEDMatrix 8x8 o també anomenada Led de matrius.



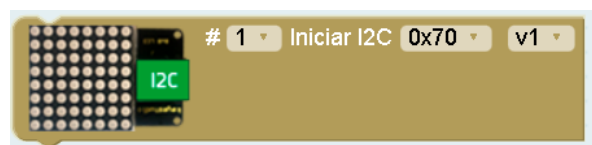
La LED de matrius, és una pantalla petita que té 64 LEDs petits. Es connecta al port de comunicació I2C.

En aquesta pantalla podem programar diferents símbols o elements, com: cares, icones, lletres... Hi ha opcions predissenyades des d'ArduinoBlocks i també, hi ha l'opció de crear-ne de personalitzats.

En l'apartat de blocs de programació, es troba a "LedMatrix 8x8". Hi ha diferents opcions de programació, segons el nostre objectiu:

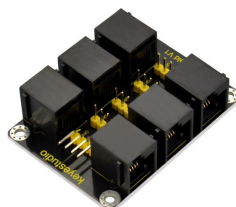


La primera tasca que hem de fer quan volem fer ús de la pantalla, és inicialitzar-la. El primer bloc, ja veiem que posa: inicialitzar I2C.



Hi ha una pestanya que ens deixa escollir diferents números. Això significa que podem utilitzar-ne quatre de diferents fent ús d'un Hub.

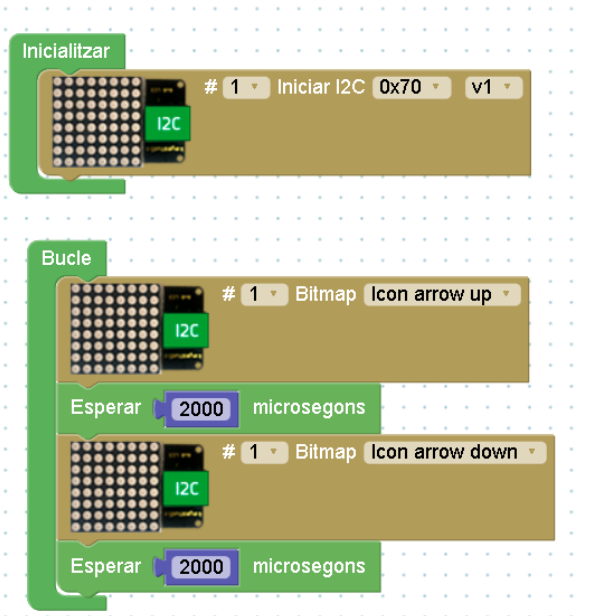
Un Hub seria com un "lladre", que ens permet connectar diversos dispositius alhora. Aquest es connecta al port I2C.



PRÀCTICA A07.1:

En aquesta pràctica introduïrem l'ús de la LedMatrix. El que farem és enviar un programa senzill:

- La pantalla LedMatrix mostra una fletxa cap amunt, i després de dos segons, mostra una fletxa cap avall.

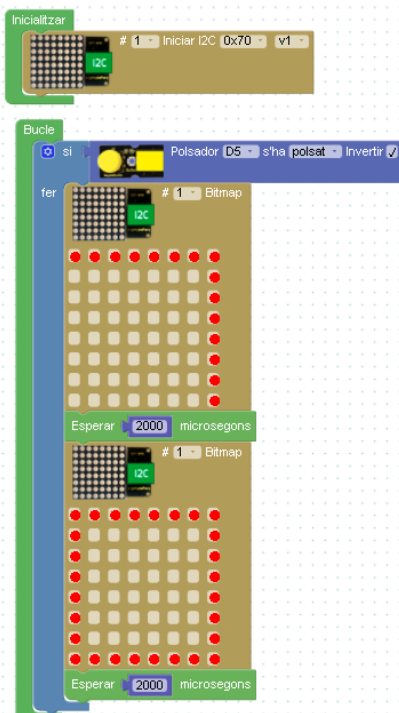


Recorda a inicialitzar la pantalla!

PRÀCTICA A07.2:

En aquesta pràctica introduïrem una altra manera de dissenyar símbols a la pantalla i també farem ús del polsador.

- Quan es premi el polsador, a la pantalla hi ha d'aparèixer un símbol personalitzat durant dos segons i després un altre diferent.

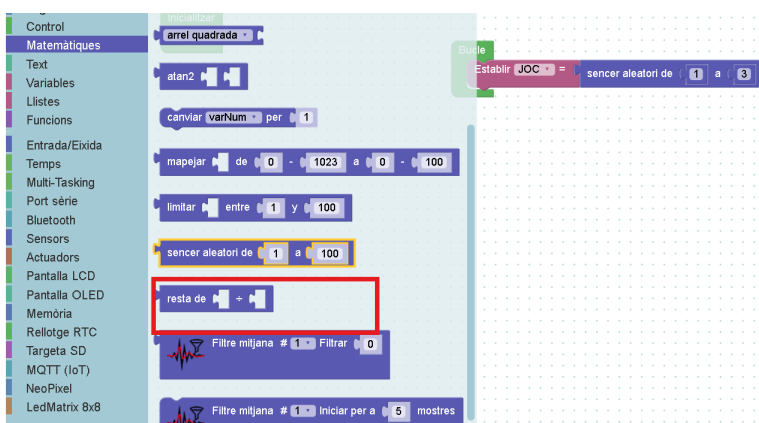


PRÀCTICA A07.3:

En aquesta tercera pràctica recrearem el famós i conegut joc "Pedra, paper i tisores". Aquesta vegada, aprendrem una altra manera de crear els nostres símbols personalitzats. També, utilitzarem el polsador per a activar el joc i aprendrem a crear una variable que esculli aleatòriament un dels tres símbols del joc.

- Quan es premi el polsador, la pantalla mostra, de manera aleatòria, un dels tres símbols: pedra, paper o tisores.

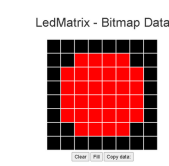
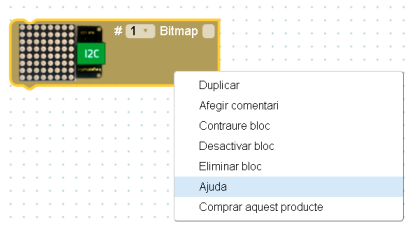
Per a fer aquesta pràctica, necessitem crear una variable que anomenarem "Joc". Aquesta variable, li assignem tres nombres aleatoris, de l'1 al 3. D'aquesta manera, assignarem a cada número un símbol. És a dir, que 1 és pedra, 2 és paper i 3 és tisores.



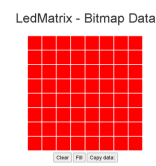
El bloc de programació: "sencer aleatori de x a x" el trobem a "Matemàtiques". Aquest ens permet escollir aleatòriament el rang de números que nosaltres establim. Així doncs, crearem la variable "joc" establint aquests valors d'1 a 3.

Seguidament, crearem els tres símbols: fent servir el "Bitmap". Per a seleccionar els LEDs que volem en estat ON, hem de fer els següents passos:

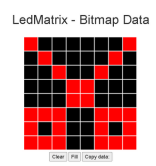
1. Fer clic al botó dret del ratolí sobre el bloc de programació.
2. Clicar l'opció "ajuda". Amb aquesta opció se'ns obre una altra pestanya on podem veure un simulador de la pantalla.
3. Seleccionem cada LED que volem encendre, així se'ns tornarà de color vermell.
4. Per últim, seleccionem "Copy data" i dins el quadradet que està el costat de "bitmap", fem Ctrl + V i se'ns enganxa en codi binari els LEDs que hem seleccionat.



Pedra



Paper



Tisores

Una vegada tenim la variable creada i els símbols, hem de seleccionar l'opció de condicionals dins el bloc de lògica, per establir que si la variable "joc" és igual a 1, sigui pedra. En canvi, si és igual a 2, sigui paper o si és igual a 3 sigui tisores. Seguidament, podeu veure la programació per a elaborar el joc.

```

Inicialitzar
# 1 Iniciar I2C 0x70 v1

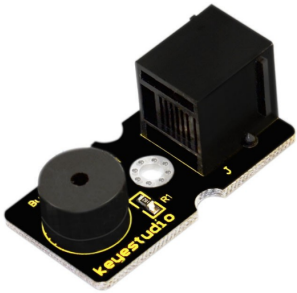
Bucle
si [Pulsador D5 s'ha polsat Invertir]
fer
  Establir [joc] = sencer aleatori de [1] a [3]
  si [joc] = 1
  fer
    # 1 Bitmap B10000001,B01000010,B00100100,B00011000,B0001100...
  sinó si [joc] = 2
  fer
    # 1 Bitmap B11111111,B11111111,B11111111,B11111111,B11111111...
  sinó si [joc] = 3
  fer
    # 1 Bitmap B00000000,B00111100,B01111110,B01111110,B01111111...
  
```

D'aquesta manera, cada vegada que el jugador premi el pulsador, el programa de manera aleatòria escollirà un número aleatori entre 1, 2 i 3. Quan l'esculli, la pantalla mostrarà el símbol que correspon al número seleccionat aleatòriament.



A08: Brunzidor

La següent activitat treballarem amb el brunzidor, el qual és un actuador connectat a un port digital.

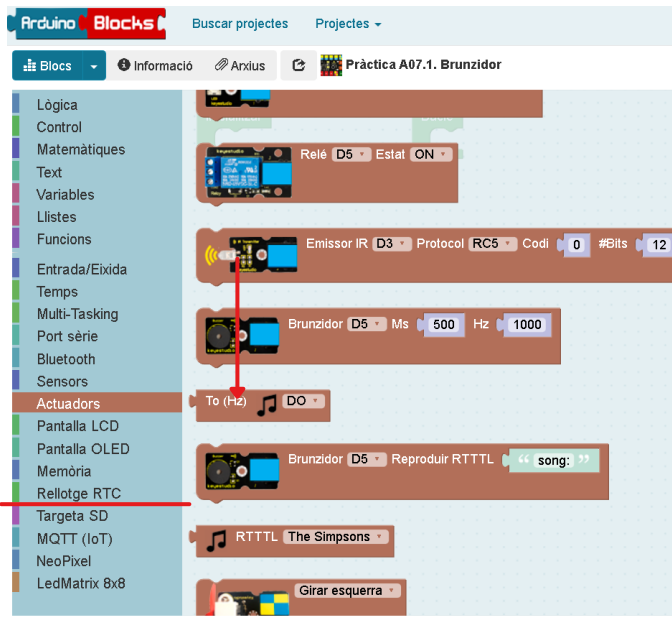


Aquest brunzidor és passiu, ja que no pot ser accionat per a ell mateix sinó per freqüències de pulsació externes.

No obstant això, es coneix com un petit "altaveu" que pot generar melodies musicals.

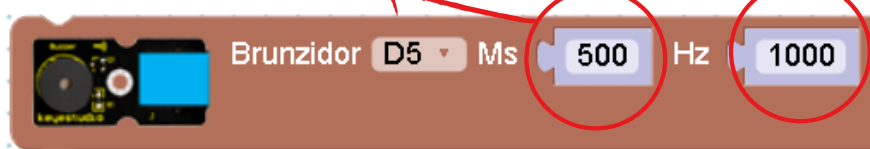
El so que emet el brunzidor depèn de la freqüència d'emissió del so. La freqüència s'entén com el nombre de repeticions per unitat de temps. El so es transmet en forma d'ona. Per tant, la freqüència d'un so és el nombre d'oscil·lacions per segon.

En l'apartat de blocs de programació, es troba a "actuadors". Hi ha diferents opcions de programació, concretament 4 blocs específics, des de seleccionar una cançó ja creada o decidir notes.



Temps que dura cada so en mil·lisegons

Freqüència que vibra la membrana del brunzidor per emetre el so



PRÀCTICA A08.1:

En aquesta pràctica crearem l'escala musical, des de Do. Per a fer-ho, pots tenir en compte la següent taula.

- Crea l'escala de "Do"

Nota	Freqüència
Do	261.6
Do#	277.2
Re#	293.7
Mi	329.6
Fa	349.2
Fa#	370
Sol	392
Sol#	415.3
La	440
La#	466.2
Si	493.2
Do	523.3

PRÀCTICA A08.2:

En aquesta pràctica enviarem al brunzidor, una música ja dissenyada pel programa.

- El brunzidor reproduïx una cançó, s'espera 5 segons i en reproduïx una altra.



Ara pots provar totes les cançons que vulguis de la plataforma ArduinoBlocks!

Pots buscar més cançons fent clic al botó dret i seleccionant "ajuda".



Seguidament, t'apareix diferents webs on pots trobar aquests sons.

A09: Sensor PIR

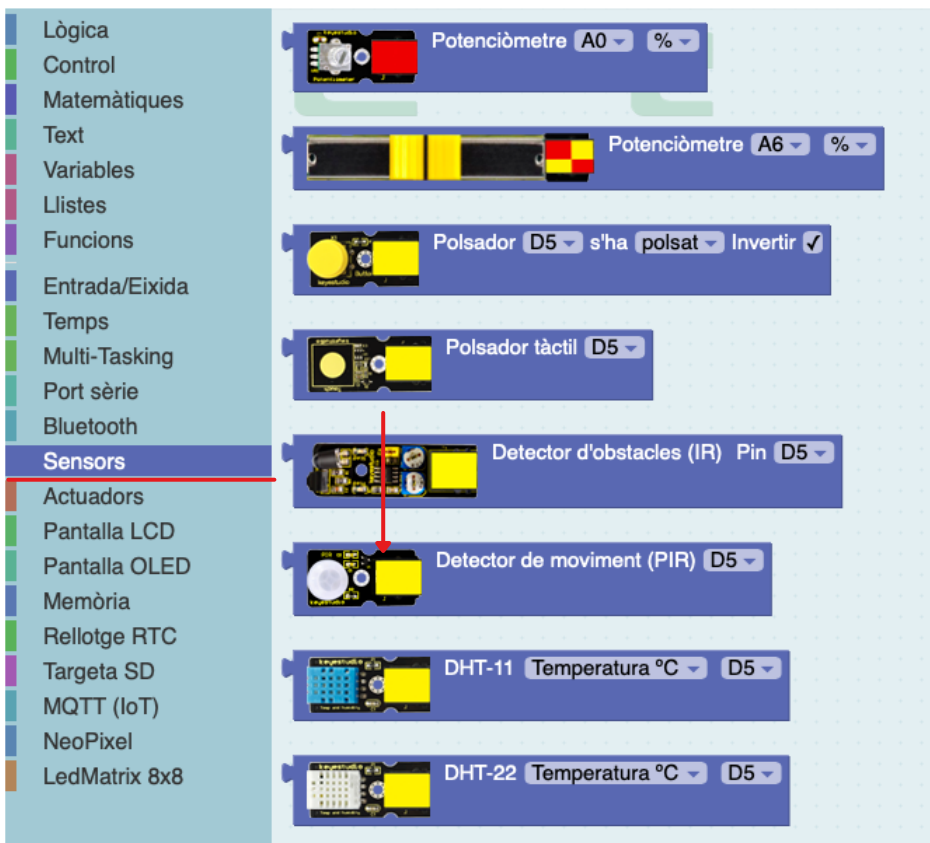
En aquesta pràctica l'objectiu és conèixer el sensor de moviment, o també anomenat PIR.



El sensor PIR és un sensor de moviment, que tal com diu el seu nom pot detectar senyals infraroigs provinents d'una persona, animal o objecte en moviment.

Té un funcionament similar a una alarma.

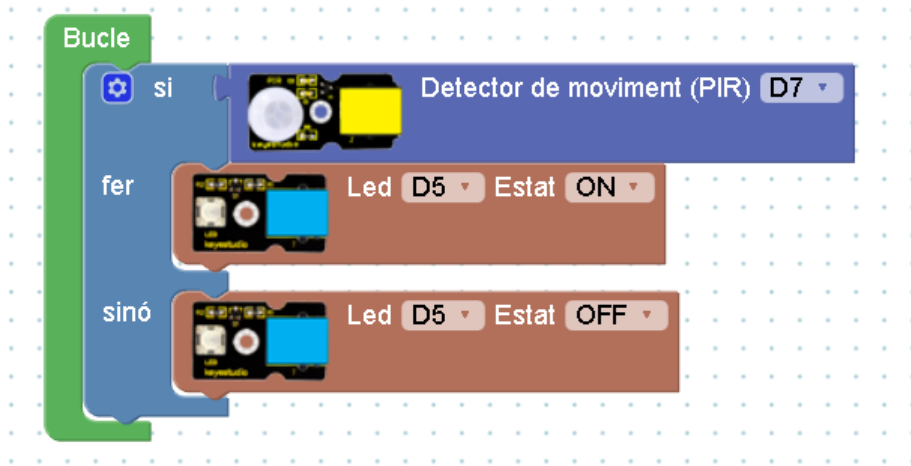
En l'apartat de blocs de programació, es troba a "Sensors".



PRÀCTICA A09.1:

En aquesta pràctica farem servir el sensor PIR i un LED.

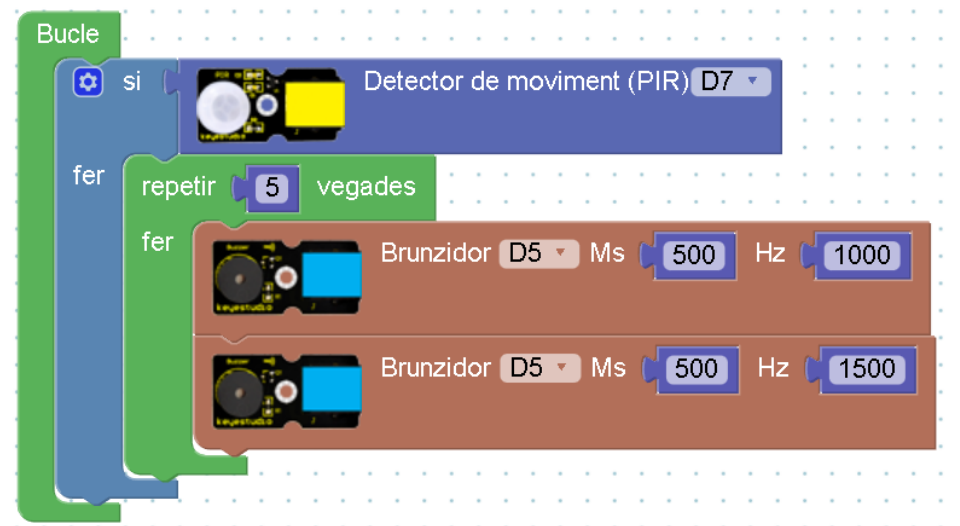
- Si detecta moviment que s'encengui el LED si no que s'apagui.



PRÀCTICA A09.2:

En aquesta pràctica reproduïrem una alarma. Així doncs, farem servir el sensor PIR, i el bronzidor.

- Si detecta moviment que el bronzidor emeti un so similar a una alarma. Per a fer-ho, caldrà que emeti aquest dos tipus de so durant 500 mil·lisegons cadascun, repetint certes vegades.



A10: Potenciòmetre

En aquesta pràctica l'objectiu és conèixer el potenciòmetre.



El potenciòmetre és un sensor de rotació analògic. El seu voltatge es pot subdividir en 1024 o també pot donar valors en tant per cent.

En l'apartat de blocs de programació, es troba a "Sensors".

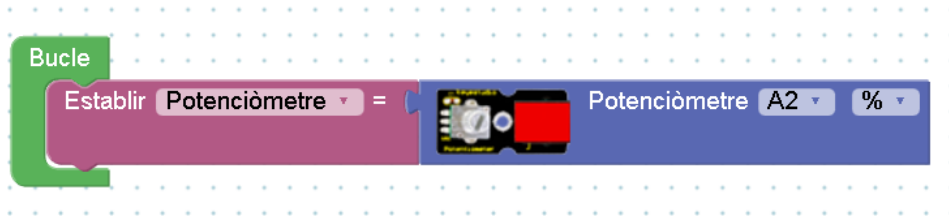
The screenshot shows the 'Sensors' category selected in the left sidebar. The main workspace contains several sensor blocks:

- Potenciòmetre** A0 % (highlighted with a red arrow)
- Potenciòmetre** A6 %
- Pulsador** D5 s'ha polsat Invertir
- Pulsador tàctil** D5
- Detector d'obstacles (IR)** Pin D5
- Detector de moviment (PIR)** D5
- DHT-11** Temperatura °C D5
- DHT-22** Temperatura °C D5

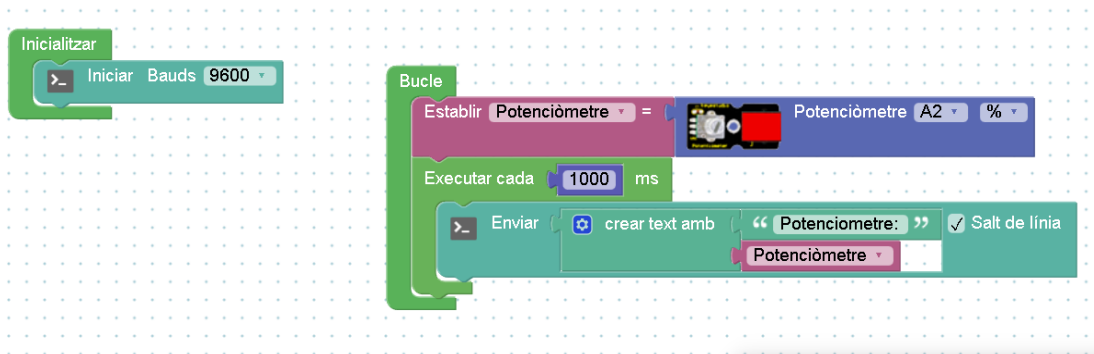
PRÀCTICA A10.1:

En aquesta pràctica coneixerem com funciona el potenciòmetre. Els valors que recull es poden calcular en % o en numeració fins a 1023 (combinació binària). En aquesta pràctica observarem els paràmetres i el moviment del potenciòmetre a través de la consola.

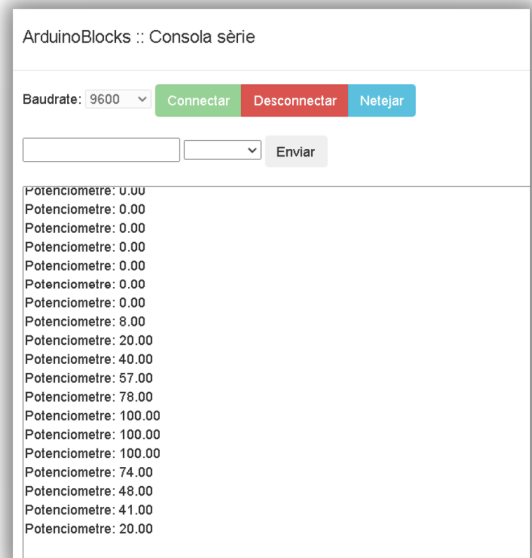
Per tant, primer de tot crearem una variable que s'anomeni "potenciòmetre" i l'associarem al sensor potenciòmetre.



- Que la consola mostri cada segon quin és el valor del potenciòmetre.



Aquests valors són els que recull la consola, essent 0 a un extrem del potenciòmetre i 100 a un altre.



PRÀCTICA A10.2:

Ara que coneixem com descriure el valor que recull el potenciòmetre, ara el programarem combinant amb un brunzidor:

- Si el potenciòmetre està entre 0 i 49% emeti un so, en canvi, si està entre 50% i 100% n'emeti un altre.

Per tant, encara farem ús de la variable i haurem de fer servir el bloc de lògica de la "i" i combinar-ho amb el de signes de més gran i petit. És a dir, aquests dos blocs:



Per tant, per a programar l'interval entre 0 i 49% haurem d'assignar al programa el següent bloc de lògica:

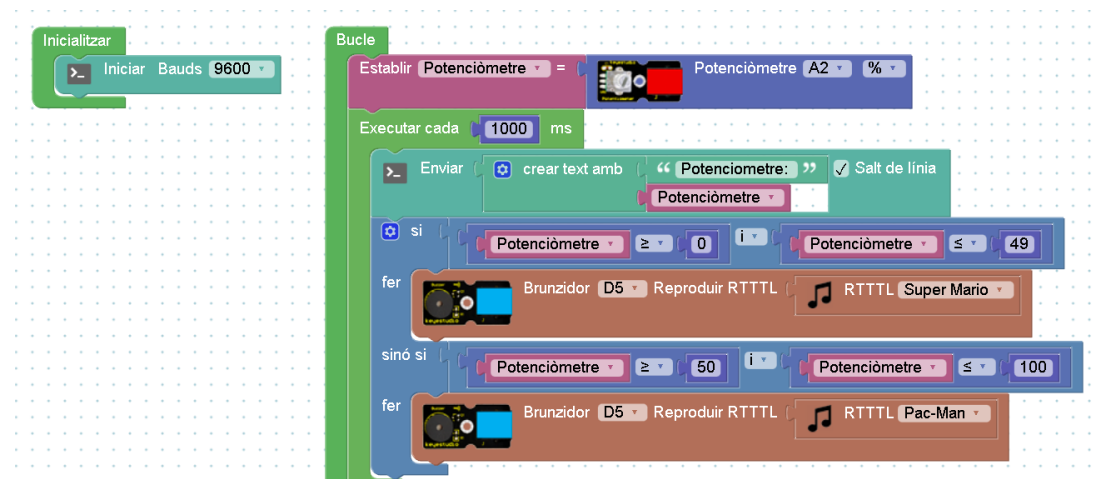
- Si l'angle és igual o més gran de 0 i igual o més petit de 49.



En canvi, per a programar l'interval que comprèn entre 50% i 100%, és:



En aquesta programació és interessant mantenir la consola, perquè així podem visualitzar el canvi de percentatge. La programació final seria la següent:

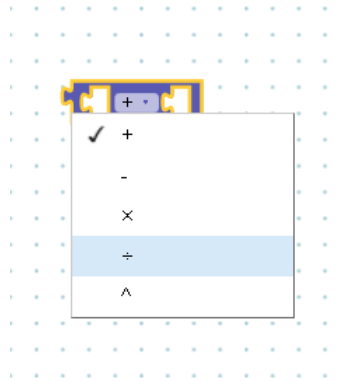


PRÀCTICA A10.3:

En aquesta tercera pràctica utilitzarem el potenciòmetre per a regular la intensitat del LED. Recordem que a la pràctica 2, vam estar treballant el PWR, és a dir, la intensitat del LED. Durant aquelles pràctiques vam descobrir que es mesurava el seu rang, entre els valors 0 i 255, és a dir, un total de 256 valors.

A la vegada, sabem que el potenciòmetre ens pot donar el rang de valors de 0 a 100% o també de 0 a 1023. Escollirem aquesta última condició i crearem una variable seleccionant el potenciòmetre. Però, aquesta vegada dividirem els valors que reculli entre 4, ja que si dividim 1023 entre 4, ens surt 255. D'aquesta manera podem veure la similitud: 255 (màxima intensitat del LED) i 1023 (màxim valor del potenciòmetre).

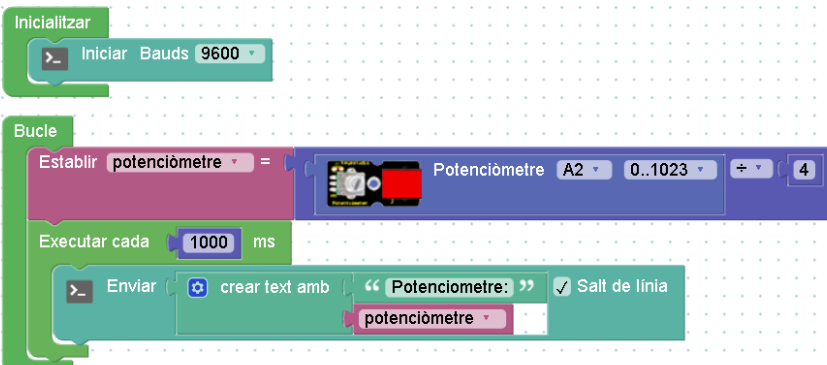
Comencem creant la variable, per a fer-ho, hem d'anar a l'apartat de "Matemàtiques" i seleccionar el d'operadors. Aquest té un desplegable i hem d'escollir la divisió:



La variable que ens ha de quedar aquesta vegada és la seqüent:

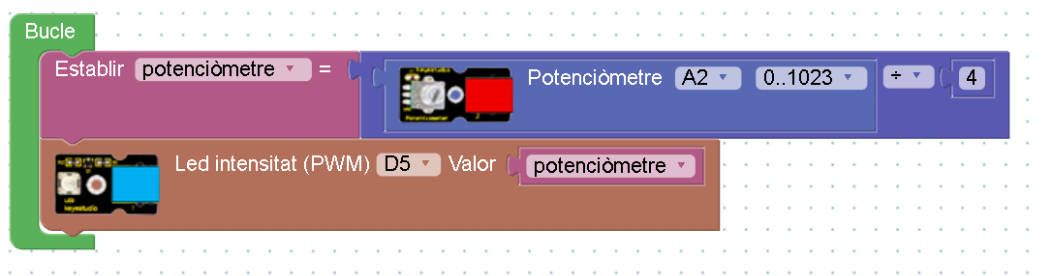


Podem comprovar aquesta nova variable, enviant valors a la consola.



- Que a mesura que anem "girant" el potenciòmetre, la intensitat del LED vagi pujant.

Per aconseguir aquesta programació, hem de col·locar la variable que hem creat dins l'espai de valor que contempla el LED PWM.



A11: Sensor de colp

Al llarg d'aquesta pràctica aprendrem a programar el sensor de colp.



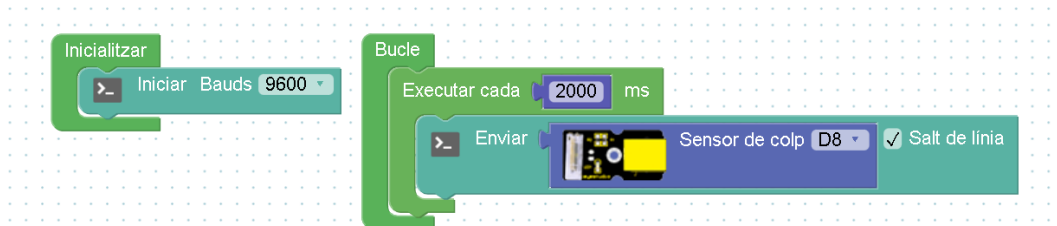
És un sensor de detector de colps. Per tant, quan el colpeges pot enviar un senyal instantani. Es pot fer servir per a substituir el polsador per engegar un LED o quan el colpegem emetre un so.

En l'apartat de blocs de programació, es troba a "Sensors".

The screenshot shows the Arduino Blocks environment. On the left, a sidebar lists various categories: Lògica, Control, Matemàtiques, Text, Variables, Llistes, Funcions, Entrada/Eixida, Temps, Multi-Tasking, Port sèrie, Bluetooth, **Sensors** (highlighted), Actuadors, Pantalla LCD, Pantalla OLED, Memòria, Relotge RTC, Targeta SD, MQTT (IoT), NeoPixel, and LedMatrix 8x8. The main workspace shows a project titled 'A11. Sensor de cop'. A list of sensor blocks is displayed, including 'Sensor de camp magnètic', 'Temperatura °C (LM35)', 'Sensor de xoc', 'Sensor de colp', 'Sensor d'orientació', 'Detector de flama', 'Sensor de vibració', and 'Interruptor magnètic'. A red arrow points to the 'Sensor de colp' block, which is highlighted.

PRÀCTICA A11.1:

Com que és un sensor nou, primer de tot el "llegirem". És a dir, enviarem els seus valors a la consola.



Podràs observar que la consola sempre ens envia el mateix valor: 1. Això significa que el sensor de cop es mesura en 0 i en 1. El "problema" és que és un senyal tan ràpid i petit que no podem visualitzar el valor 0.

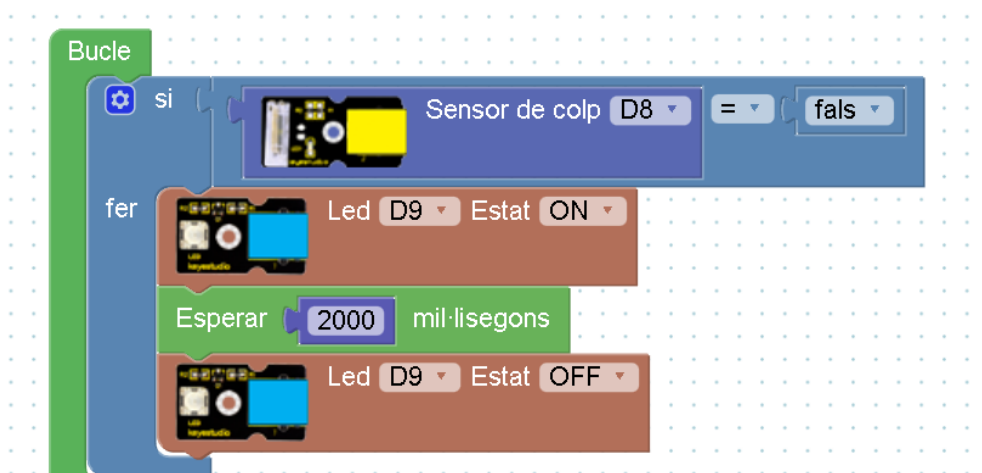
PRÀCTICA A11.2:

En aquesta pràctica aprendrem a engegar el LED fent servir el sensor de colp. Connectem tant el LED com el sensor de colp a dos ports digitals.

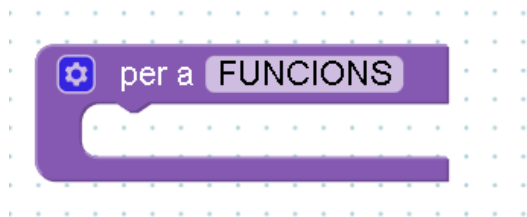
- Quan fem un colp, que el LED s'encengui durant 2 segons i que després s'apagui.

Sabem que 1 és quan no hi ha "colp" i 0 quan hi ha el "colp". Per tant, els valors d'aquest sensor es mou entre dues variables: sí/no, veritat/fals...

Així doncs, en aquesta programació hem d'igualar que si el sensor de colp és fals, és a dir, que està a 0 s'encengui el LED. La programació és la següent:



Seguint amb aquesta pràctica, farem una petita modificació. Introduïrem el concepte de "funcions".



ArduinoBlocks té un apartat que s'anomena "funcions".

Per entendre què és una funció, podem descriure-ho com un conjunt d'instruccions que les agrupem sota un nom. Quan creem una funció, automàticament es genera un bloc d'aquesta funció i es pot inserir en qualsevol moment del programa. És una manera d'agilitzar la programació i perquè en programacions més "grans" ens quedi tot una mica més ordenat.

En aquest cas, doncs, agruparem la programació "LED ON, esperar 2000 mil·lisegons, LED OFF" en una funció anomenada LLUM ON.

Lògica
Control
Matemàtiques
Text
Variables
Llistes
Funcions
Entrada/Eixida
Temps
Multi-Tasking
Port sèrie
Bluetooth
Sensors
Actuadors
Pantalla LCD
Pantalla OLED
Memòria
Relotge RTC
Targeta SD
MQTT (IoT)
NeoPixel
LedMatrix 8x8

