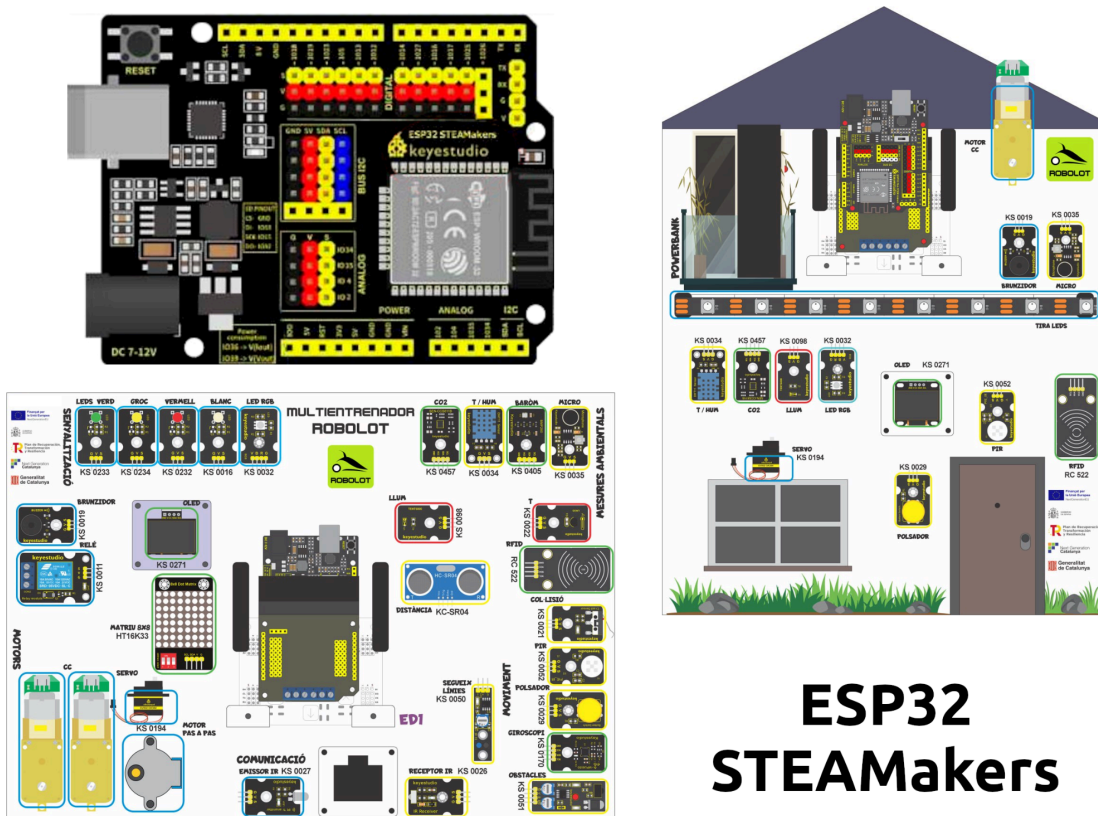


MANUAL DE PRÀCTIQUES amb ESP32 STEAMakers i arduinoblocks



ESP32 STEAMakers

Elaborat per a la Formació:

Reptes de robòtica amb la placa ESP32 STEAMakers

Versió 2.0 - Novembre 2024

Autoria: Cristina Blaya Góngora, Andreu Bonet Beltran, Josep Maria Filella Agulló, Sergi Horrillo Garcia, Toni Hortal Martínez, Miguel Pérez Manso, Marta Simon Bonet, Dani Soldevila Guitart i Alan Torrens Berenguer.



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-CompartirIgual 4.0 Internacional de Creative Commons](https://creativecommons.org/licenses/by-nc-sa/4.0/)

ÍNDEX

ÍNDEX	2
Introducció	3
Funcionament d'un sistema de control programat	4
Elements dels conjunts de sensors i actuadors	6
Placa de control ESP32 STEAMakers	7
Programació amb arduinoblocks	9
P01. Engegar un LED	13
P02. Engegar un LED amb un polsador (sensor digital)	17
P03. Sensor de llum (sensor analògic)	24
P04. LED RGB	29
P05. Mesurar la temperatura i humitat amb DHT-11	34
P06. Pantalla OLED	40
P07. Matriu LED 8X8	47
P08. Llista de reproducció de música amb el brunzidor	53
P09. Mesurar distàncies amb el sensor d'ultrasons HC-SR04	59
P10. Controlar un servomotor	63
P11. Tira de leds NeoPixel	66
P12. Giroscopi i acceleròmetre	72
P13. Bomba d'aigua	79
P14. Condicions ambientals	84
P15. Motors CC	89
P16. Infrarojos + Comandament	97
P17. Emissor Infrarojos	102
P18. Detector de moviment (PIR)	105
P19. Mòdul RFID	110
P20. Seguidor de línies.	115
P21. Sensors integrats i consum d'energia	119
Glossari de termes de programació	125
Reconeixements	126

Introducció

El present manual sorgeix del treball cooperatiu d'un equip de professors i professores de secundària d'arreu de Catalunya que durant l'estiu de 2024 es van reunir a la Garrotxa, gràcies al finançament del fons Next Generation, amb l'objectiu d'investigar i treure'n profit educatiu als materials de robòtica que havien arribat recentment als centres educatius. Aquestes persones, unides per la passió per portar la programació i la robòtica educativa a les aules, vam analitzar les necessitats de formació docent que calia oferir per tal d'acompanyar el procés d'implantació de la dotació esmentada. Una inversió tan gran en materials d'última generació no pot quedar oblidada als armaris dels instituts per una manca de coneixement del professorat. Dissortadament els assessoraments tècnics oferts pel Departament d'Educació als claustres a finals del curs passat no han cobert les necessitats del professorat. És per això que des de l'Associació Robolot Team s'han preparat una sèrie de formacions al professorat pel curs 2024/25. Aquest manual és la guia pràctica d'una d'aquestes formacions.

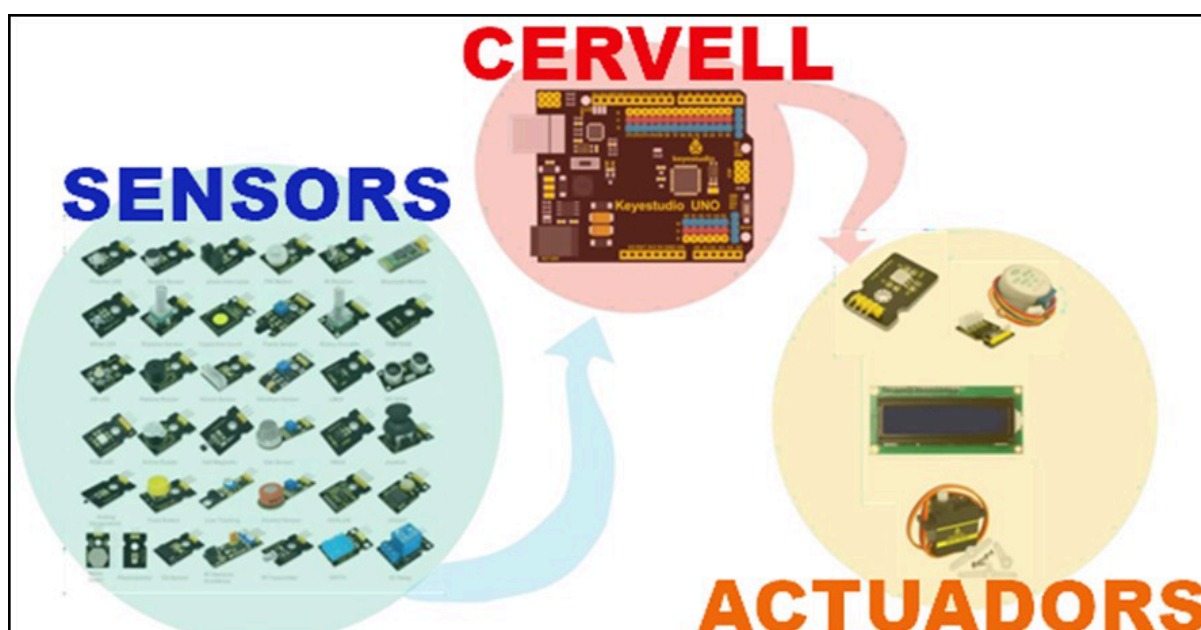
Aquest document és una recopilació no exhaustiva de pràctiques per aprendre a fer servir alguns dels materials de la dotació de robòtica educativa que van arribar als centres el curs 2023/24. En concret treballa amb la placa electrònica de control ESP32 STEAMakers i els conjunts de sensors i actuadors secundària bàsic i avançat (maleta verd i maleta vermella). Totes les pràctiques recollides en el present document estan pensades per a programar-se amb arduinoblocks. Aquest entorn de programació en blocs ha estat desenvolupat pel professor Juanjo López i permet aprendre a programar d'una manera visual i sencilla.

Les pràctiques aquí desenvolupades, així com els reptes d'ampliació, estan orientades a l'aplicació directa a les aules de secundària. La filosofia de l'*Aprenentatge Basat en Reptes* vol superar el model de classe magistral posant a l'alumne al centre, i fent-lo protagonista d'un aprenentatge autoregulat, basat en les errades com a part del procés.

El material recopilat en aquest manual contribueix a l'adquisició de la competència en pensament computacional per part de l'alumnat. De forma paral·lela potencia la relació entre les matèries de l'àmbit STEAM i promou la cultura maker entre el professorat.

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 aquesta 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 o els músculs serien les sortides.

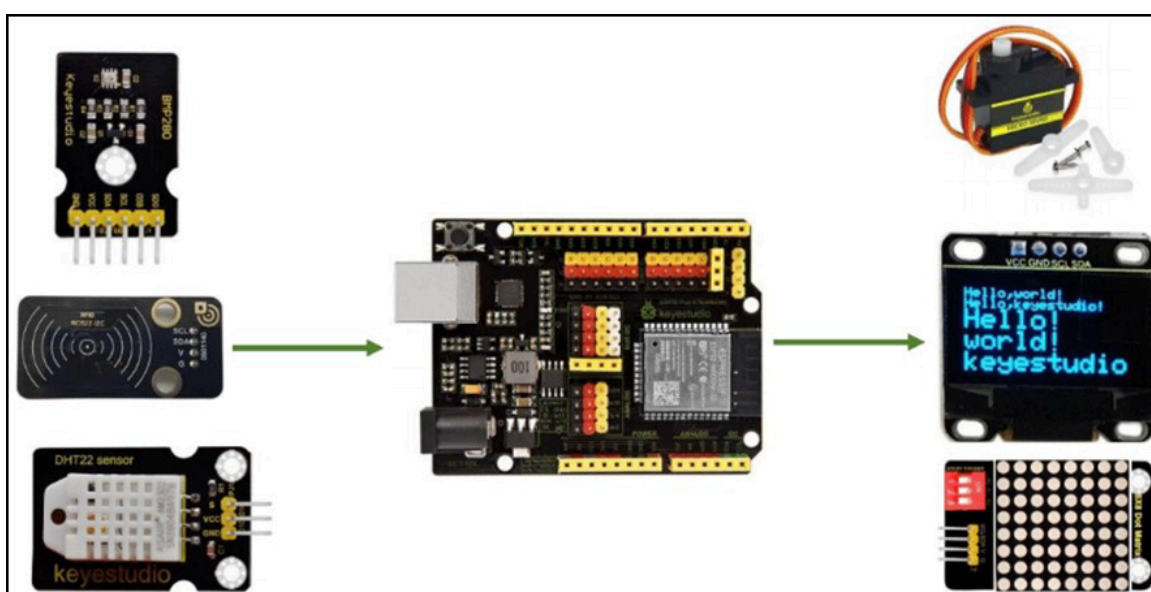
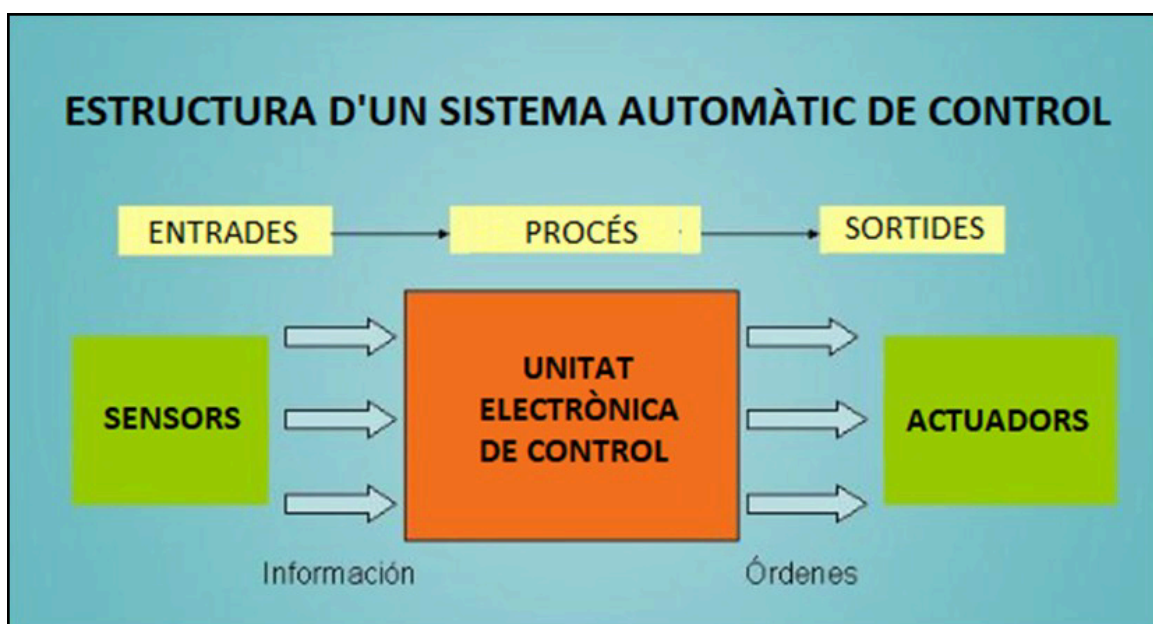


En el cas d'un sistema de control programat, un microcontrolador fa la funció de cervell. Aquest component electrònic unes entrades d'informació on es connecten els sensors de llum (LDR), temperatura (NTC), so, i també té sortides, on es connecten els motors, LEDs, bronzidors, etc.



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 elèctrics que seran convertides en els actuadors (sortides) en altres tipus de magnituds.



Elements dels conjunts de sensors i actuadors



Sensors i actuadors ESO bàsics



[Fitxa resum](#)



[Presentació](#)

- Sensor de llum
- Sensor d'humitat del terra
- Sensor de temperatura
- Pantalla OLED
- Sensor de col·lisió o fi de cursa
- Sensor infraroig x3
- Sensor PIR
- Sensor de distància per ultrasons
- Tira de 30 leds (1 m)
- Motor amb roda x2
- Servomotor x2
- Cables Dupont x30
- Cables GVS x5

Sensors i actuadors ESO avançats



[Fitxa resum](#)



[Presentació](#)

- Sensor de so / micròfon
- Giroscopi i acceleròmetre
- Sensor CO2
- Sensor d'humitat i temperatura
- Sensor de pressió baromètrica
- LED blanc
- LED groc
- LED verd
- LED vermell
- LED RGB
- Brunzidor
- Relé
- Bomba d'aigua
- Receptor IR i comandament
- Emissor IR
- Polsador
- Cables Dupont x30
- Cables GVS x5

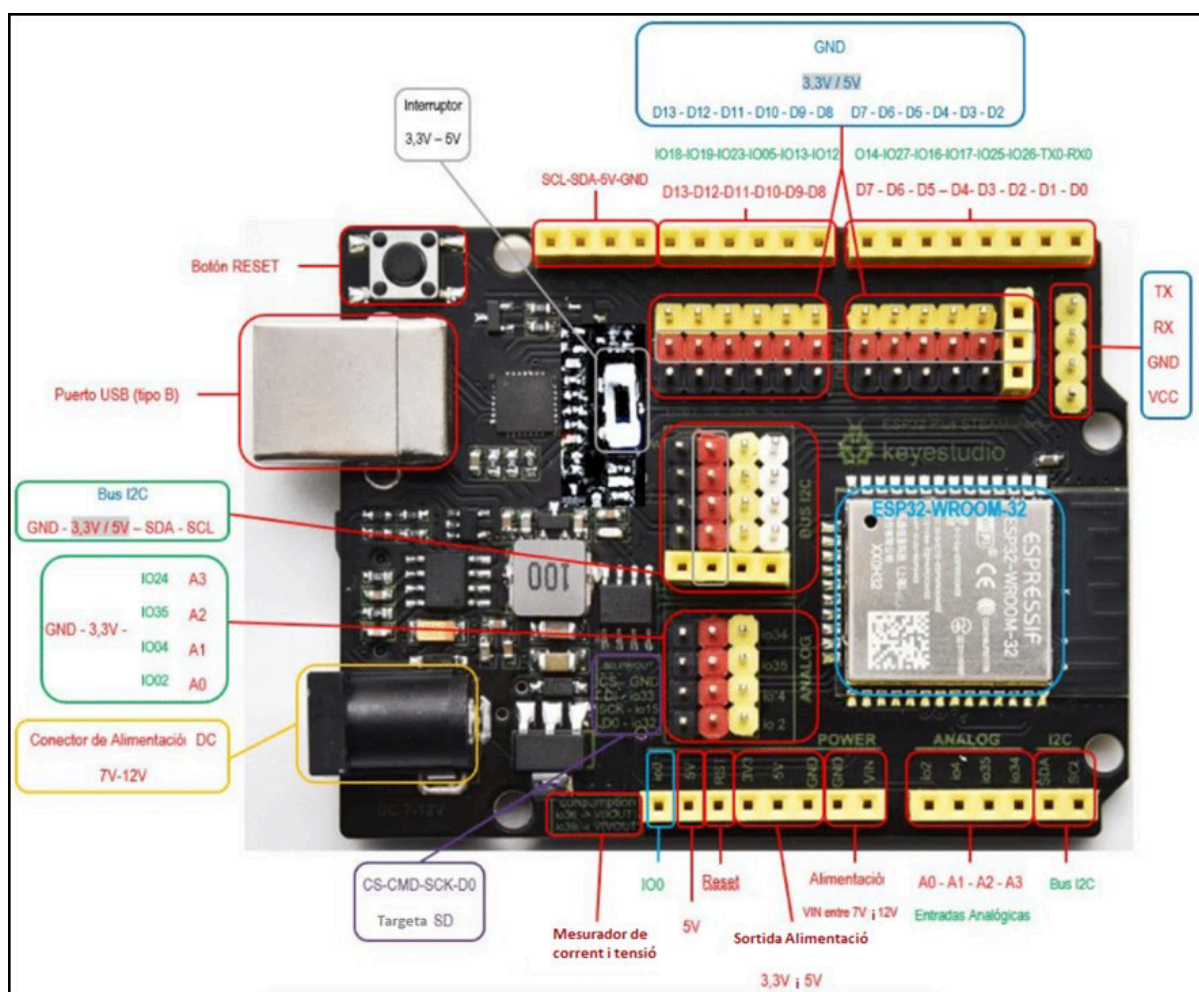
Placa de control ESP32 STEAMakers

La placa està basada en una placa Arduino UNO. Arduino és una plataforma de prototips electrònics de codi obert (Open-Source) basat en hardware i software lliure, flexible i fàcil d'usar, amb una comunitat molt gran que genera moltíssima informació.

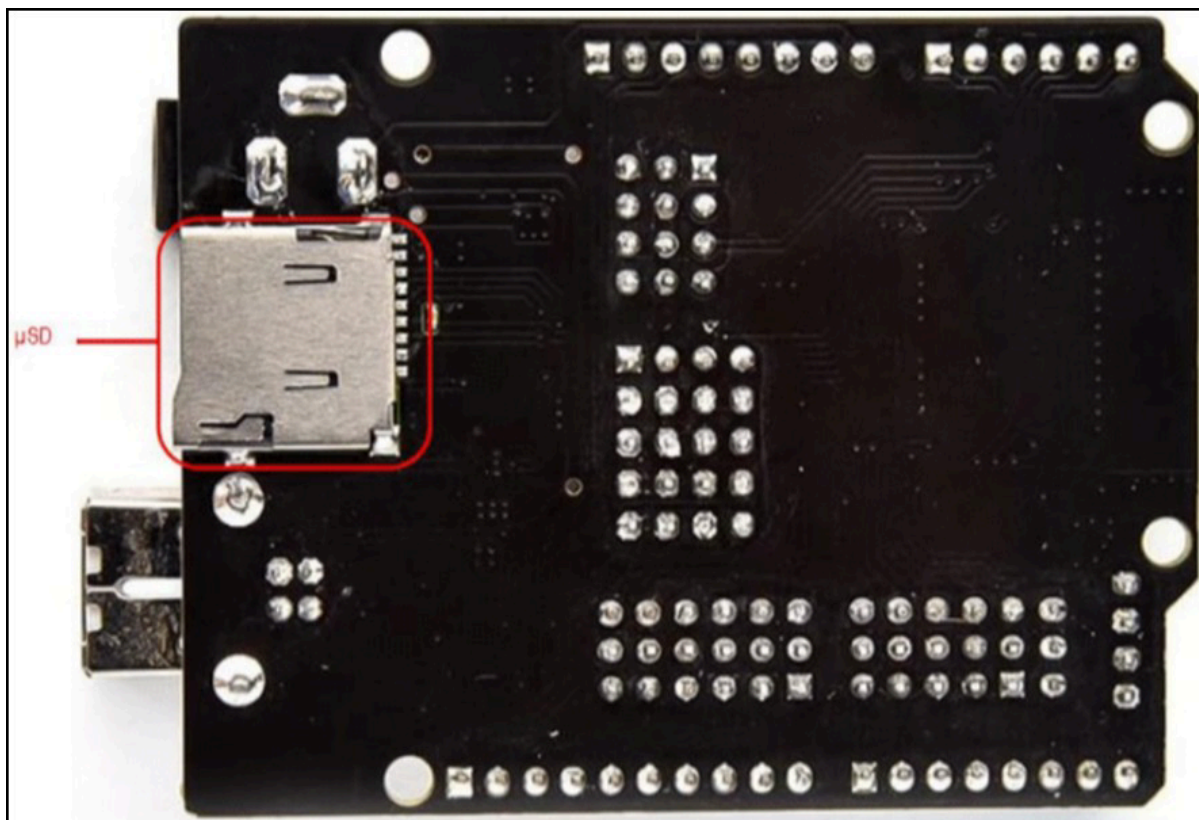
Aquesta plataforma permet crear diferents tipus de sistemes per diferents usos. Pot ser utilitzada per artistes dissenyadors, enginyers, professors, alumnes, etc., i en general, qualsevol persona que estigui interessada a crear objectes o entorns interactius, prototips, sistemes robòtics, etc.

La placa té els següents elements.

Per la part del davant:



Per la part d'endarrere de la placa:



El fet de ser Arduino un hardware lliure, existeixen multitud de fabricants que han desenvolupat versions basades en Arduino. Un d'aquests fabricants és **Keystudio**, que ha desenvolupat juntament amb l'equip Innova Didàctic una placa compatible amb Arduino, però basada en una ESP32. La nova placa ESP32 Plus STEAMakers.

Programació amb arduinoblocks

Arduinoblocks és una plataforma web en línia, creada pel professor Juanjo López, on podeu programar la vostra placa de forma visual (consulteu la llista adjunta de plaques suportades). Els diferents blocs permeten llegir diferents entrades i enviar valors a les sortides de la placa, així com programar funcions lògiques, de control, etc.



La programació en arduinoblocks es fa amb blocs a l'estil de MicroBlocks. No heu d'escriure línies de codi i no està permès unir blocs incompatibles, evitant així possibles errors de sintaxi.

La plataforma Arduinoblocks genera, compila i envia el programa a la placa per mitjà de la connexió USB

Depenent del sistema operatiu, caldrà disposar dels corresponents drivers instal·lats i, sempre, executar AB-connector per a poder enviar el programa a la placa.

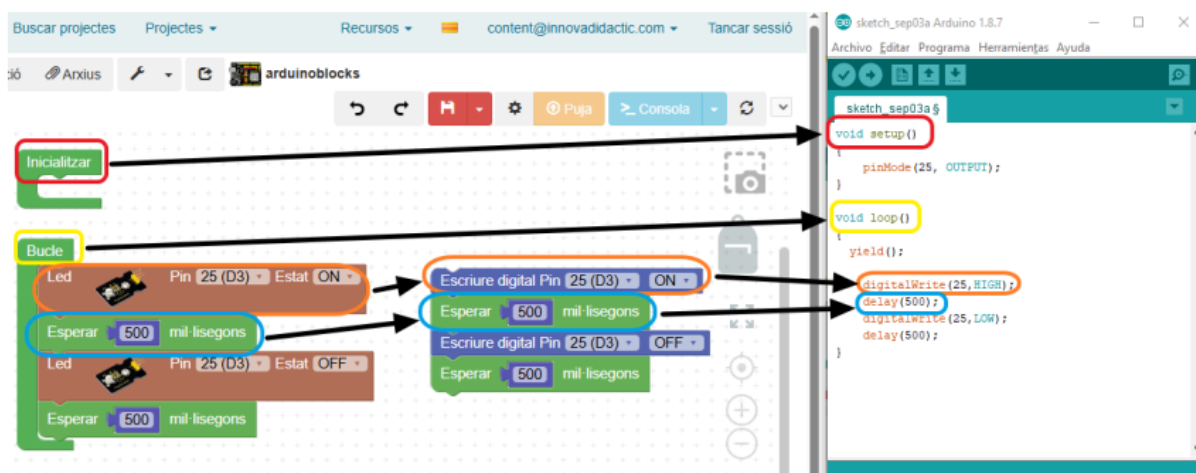
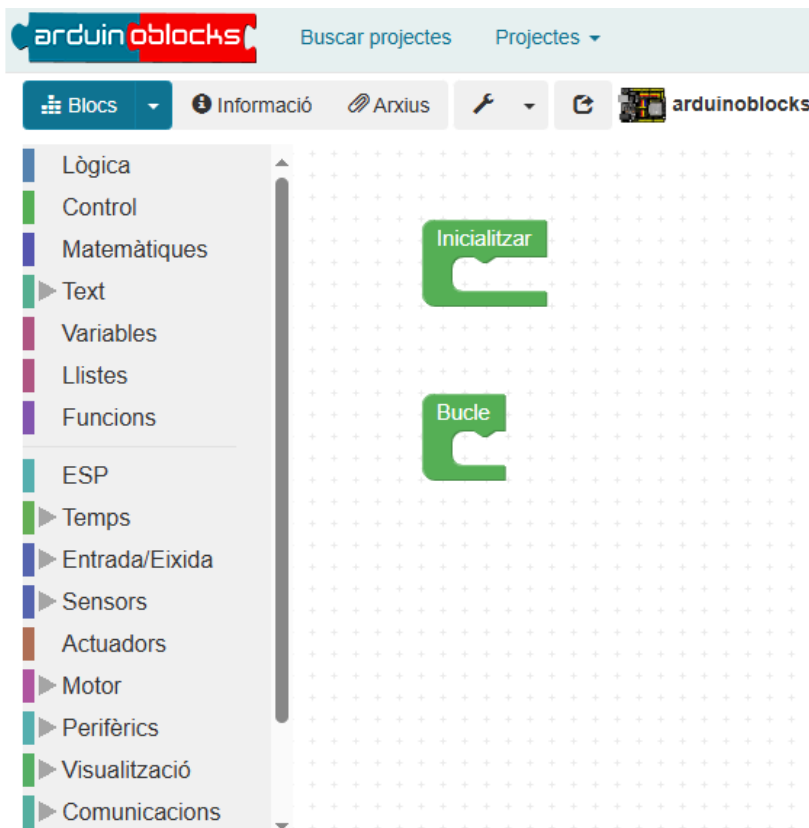
Un cop pujat el programa a la placa, no caldrà connexió lògica amb l'ordinador per a funcionar. Podeu alimentar-la amb bateries, la mateixa connexió USB o una font d'alimentació perquè funcioni de forma autònoma.

Arduinoblocks actualment funciona amb tots els navegadors d'última generació: Google Chrome, Microsoft Edge, Mozilla Firefox, Opera, Safari, etc.

Plaques suportades

- UNO
- NANO / ATmega328
- NANO / ATmega328 (new bootloader)
- MEGA / 2560
- Leonardo
- UNO + Imagina TdRSTEAM
- UNO + Imagina 3DBot
- Keystudio EasyPlug
- Keystudio KeyBot
- Keystudio KidsIoT
- ESP32 STEAMakers
- ESP32 STEAMakers + Imagina TdR STEAM
- ESP32 STEAMakers + Imagina 3DBot
- ESP32 micro:STEAMakers
- ESP32 / WROOM
- ESP8266 / NodeMCU v2
- ESP8266 / WeMos D1
- Otto DIY / Nano
- Otto DIY / Nano (new bootloader)

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



INSTAL·LACIÓ D'ARDUINOBLOCKS

Arduinoblocks treballa en línia, funciona amb tots els navegadors d'última generació: Google Chrome, Microsoft Edge, Opera, Safari, etc. **La programació no requereix de cap instal·lació específica**, però sí que es necessita un programa per poder enviar-lo al microprocessador de la placa. Aquest programa s'anomena **AB-Connector**.

- Per a ordinadors del departament:

Entrar al portal company i buscar AB connector

- Per a ordinadors personals:

[Enllaç](#)



La icona que es veu després de la instal·lació és la següent:

Haurem de tenir sempre obert aquest programa quan enviem el codi a la placa o quan vulguem fer comunicacions per la consola o el Serial Plotter d'arduinoblocks.

Primer de tot haurem de crear un compte amb arduinoblocks i després instal·lar el programari AB-Connector per poder comunicar l'ordinador amb la placa a programar.

Obrim el AB-Connector i veiem la següent pantalla:



```

AB-Connector v5.3
arduinoblocks Log file...
13:30:35> AB-Connector v5.3
13:30:35> Path: C:\Program Files\abconnector\bin
13:30:35> Port: 9987
13:30:35> Arduino-CLI: 0.35.3
13:30:38> ['arduino:avr', 'esp32:esp32', 'esp8266:esp8266']
13:30:38> Checking/updating libs...
13:30:38> Libraries version: 60
  
```

CREAR UN COMPTE



arduinooblocks Buscar projectes

Iniciar sessió

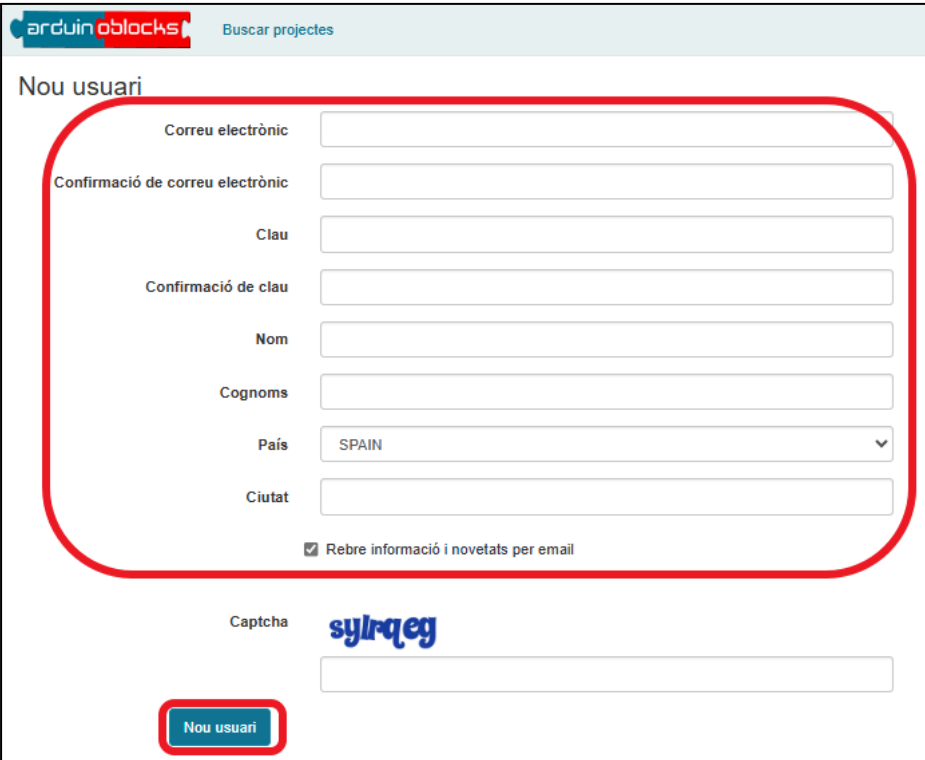
Correu electrònic

Password

Iniciar sessió

Nou usuari

No recorde la meua clau



arduinooblocks Buscar projectes

Nou usuari

Correu electrònic

Confirmació de correu electrònic

Clau

Confirmació de clau

Nom

Cognoms

País

Ciutat

Rebre informació i novelats per email

Captcha **syjrqeg**

Nou usuari



P01. Engegar un LED



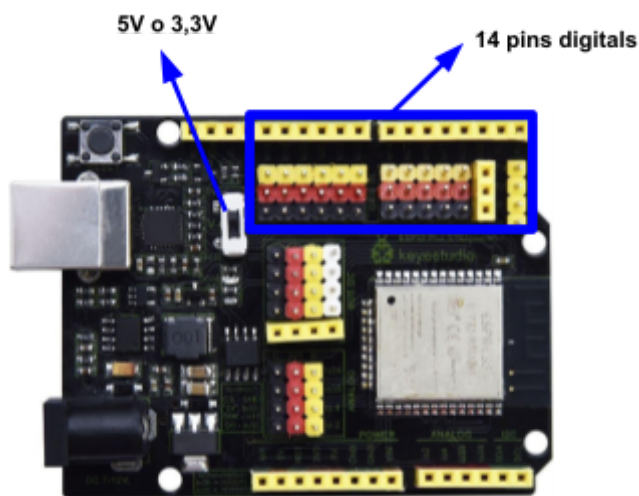
DESCRIPCIÓ

- Engegar i apagar un LED de forma intermitent.
- Regular la intensitat d'un LED

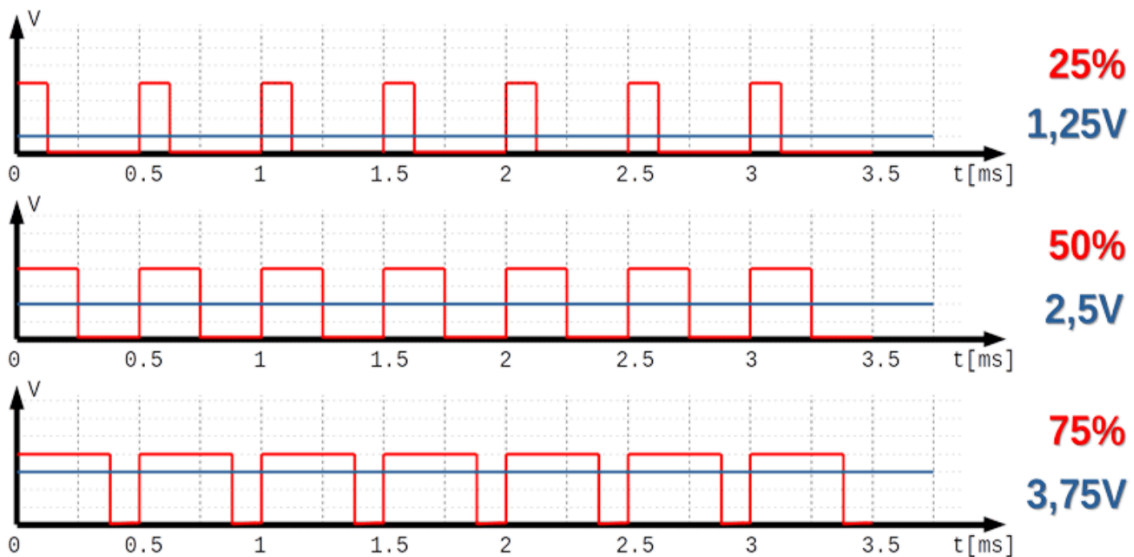
SABERS

- Sortida digital: És un pin del microcontrolador que pot ser configurat per enviar senyals de sortida binària, és a dir, senyals que només poden tenir dos estats: alt (HIGH) o baix (LOW).


La placa té un interruptor per configurar el nivell ALT a 5V o 3,3V, i el nivell baix són 0V.



- Sortida analògica: Podem configurar alguns pins per generar senyals analògics utilitzant la tècnica PWM (Modulació per Amplada de Pols). El PWM permet simular una sortida analògica amb un pin digital canviant la quantitat de temps que el pin està engegat (HIGH) o apagat (LOW) durant un cicle. El cicle de treball és el percentatge de temps que el senyal està engegat. Això permet controlar la brillantor dels leds o la velocitat dels motors ajustant l'energia que es lliura al dispositiu.

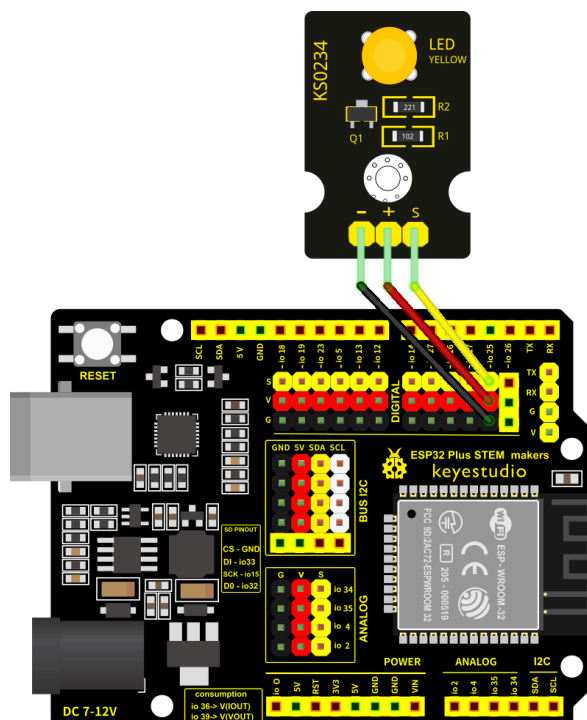


LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Leds

MUNTATGE

El muntatge és comú per a les dues parts de la pràctica.



fritzing

PROGRAMACIÓ 1.1

Inicialitzar



Bucle



El LED s'haurà d'engegar i apagar amb una cadència d'un segon.

Prova de canviar el temps d'espera que com veus està mesurat en mil·lisegons.

PROGRAMACIÓ 1.2

Inicialitzar




El LED s'haurà d'engegar progressivament. El valor que enviem a la sortida PWM ("analògica") és un valor de 8 bits i, per tant, pot prendre valors decimals d'entre el 0 i el 255 decimal. ($2^8=256$ valors possibles). Un valor de 0 equival a 0 V i un valor 255 equival a un valor de 5 V o 3,3 V (depenent del valor amb el que treballem).

REPTES AMPLIACIÓ

- **Semàfor:** Simula el funcionament d'un semàfor de trànsit utilitzant els leds vermell, groc i verd. Programa els leds per encendre'ls en seqüència (verd, groc, vermell) amb intervals de temps adequats.

P02. Engegar un LED amb un pulsador



DESCRIPCIÓ


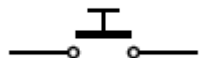
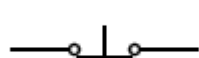
- Engegar i apagar un LED mitjançant un pulsador digital

SABERS

En el següent repte utilitzarem el pulsador. Prèviament, hem de recordar la diferència entre un pulsador i un interruptor. Un interruptor és un dispositiu que obre o tanca en pas del corrent elèctric, per exemple, els interruptors de la llum de les nostres cases, cada vegada que els premem canvien d'estat i romanen en ell fins a ser premuts de nou. No obstant això, un pulsador només s'activa mentre duri la pulsació tornant al seu estat inicial en el moment en el qual es deixa de prémer.




Hi ha dos tipus de pulsadors:

- NO (Normalment Obert): Quan no es prem el pulsador, el circuit està obert (sense connexió). En prémer-lo, el circuit es tanca i activa la funció o acció associada.
- NC (Normalment Tancat): Quan no es prem el pulsador, el circuit està tancat (amb connexió). En prémer-lo, el circuit s'obre i interromp la funció o acció associada.

		
símbol interruptor	símbol pulsador NO	símbol pulsador NC

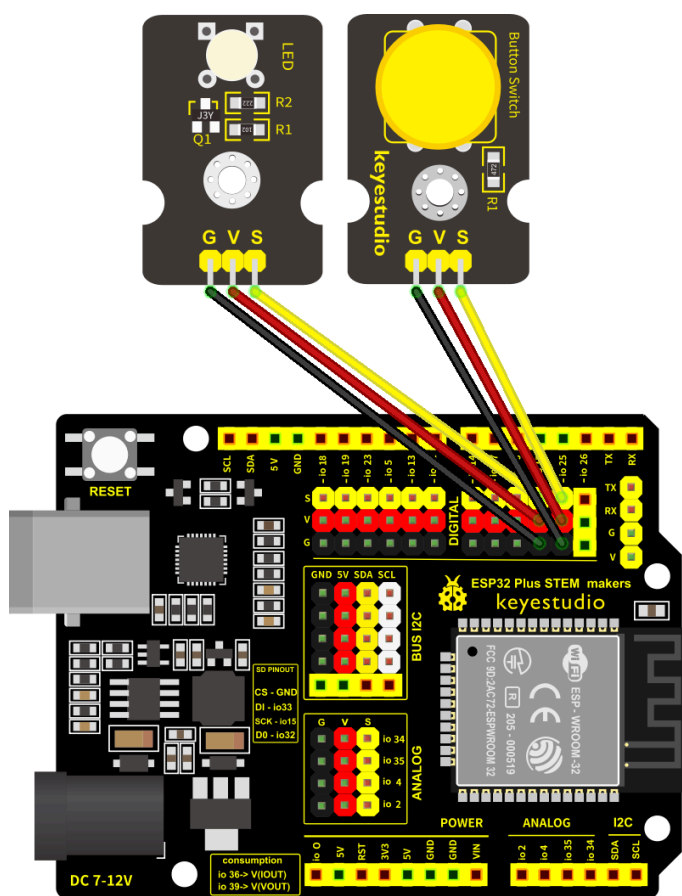
Al multientrenador trobem pulsadors digitals que es poden programar per comportar-se com ho farien aquests tres components elèctrics.

LLISTA D'ELEMENTS

- 1 Placa ESP32
- LED 
- Polsador digital 
- 2 cables Dupont RBL NN 

MUNTATGE

Connectem el polsador digital al pin IO25 (D3) i el LED al IO17 (D4)



fritzing

PROGRAMACIÓ 2.1: POLSADOR

Ara realitzarem un programa en el qual en prémer sobre el polsador s'engegui el LED i s'apagui quan el deixem de prémer, és a dir volem programar el polsador digital per tal de que es comporti com un polsador NO.

Al menú d'arduinoblocks trobem 2 blocs diferents de polsador a l'apartat de sensors. Començarem amb el primer bloc.



Per a realitzar aquest programa necessitem conèixer unes de les funcions més utilitzades en programació. Les funcions del menú Lògica amb els blocs de condició (condicionals).

- Condicional:

Un condicional és una estructura de programació que permet que el codi prengui decisions basades en certes condicions. Això vol dir que el programa pot executar diferents blocs de codi segons si una condició és certa o falsa. És com si el programa preguntés: "Si passa això, què he de fer?"

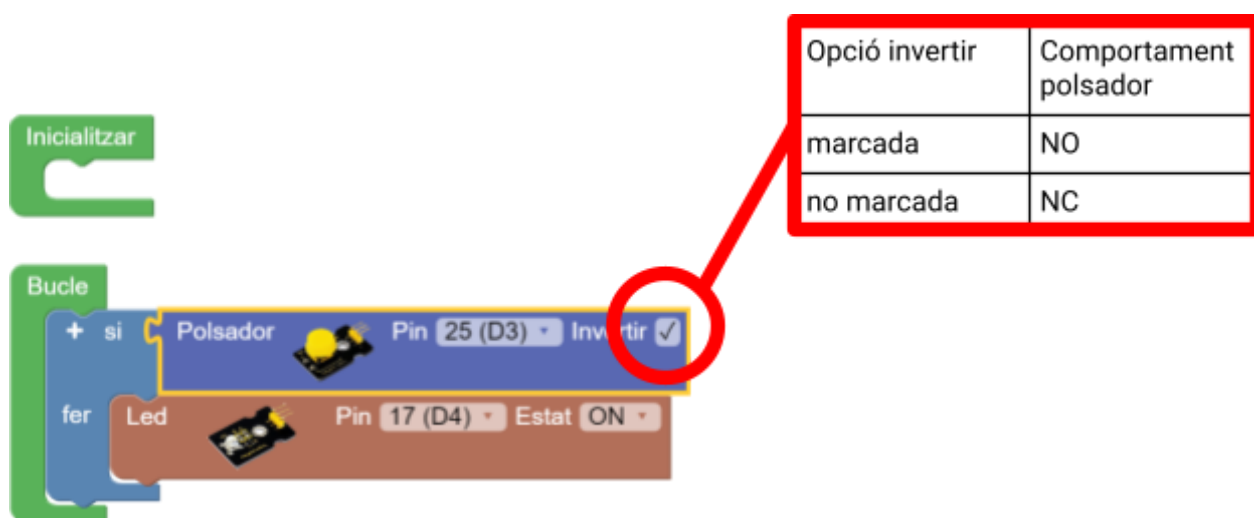
Bloc condicional simple	Bloc de condicional complet

El funcionament amb el bloc condicional simple és el següent: si es compleix la condició inclosa en el seu primer apartat, llavors es realitza l'acció inclosa en el seu segon apartat. En cas contrari, no es fa res.

En canvi, amb el bloc condicional complet, en cas que no es compleixi la condició es fa una acció diferent.

En l'apartat de condicions es poden introduir multitud de factors: estat de sensors (analògics o digitals), comparacions, igualtats, operacions matemàtiques, etc. Hem de tenir en compte que sempre estem treballant amb condicions cert/fals.

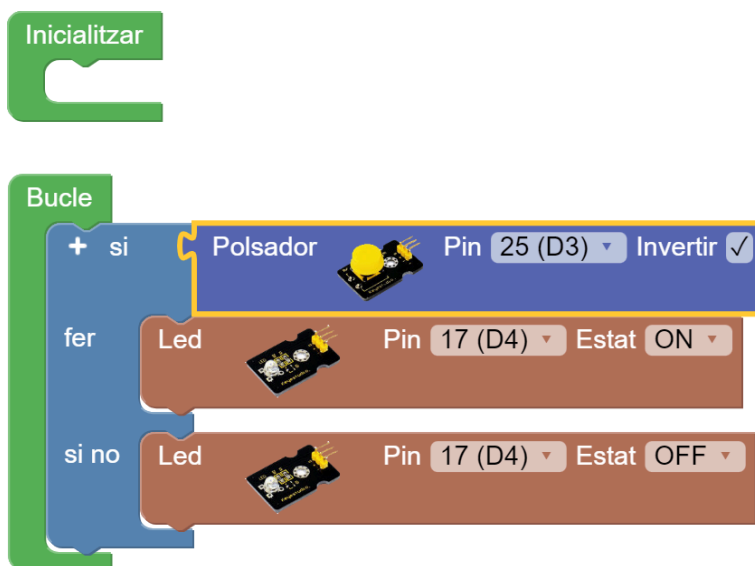
Per la pràctica que ens ocupa fem servir el primer bloc del polsador dins del bloc lògic condicional Si.... fer... i el programa quedaria com la imatge.



Opció invertir	Comportament pulsador
marcada	NO
no marcada	NC

Si fem això no s'apaga el LED mai, això no és el que volem, el problema és que en cap moment del programa diem que el LED hagi d'estar en la posició OFF.

Farem servir un altre bloc condicional més complet que inclou el comportament en cas de que no es doni la condició inicial.



El LED s'engegarà mentre premem el polsador i s'apagarà en deixar-lo anar, comportament normalment obert, NO. Si provem el mateix programa però sense marcar l'opció invertir veurem que es comporta com un polsador normalment tancat, NC.

PROGRAMACIÓ 2.2: INTERRUPTOR

Ara realitzarem un programa que engegui un LED quan premem el polsador i es mantindrà així fins que el tornem a prémer, moment en que s'apagarà, és a dir programarem el polsador digital per tal de que es comporti com un interruptor elèctric.

Per programar aquesta pràctica necessitem fer servir variables.

- Variables:

Una variable en programació és un element que utilitzem per emmagatzemar informació que pot canviar mentre el programa s'executa.

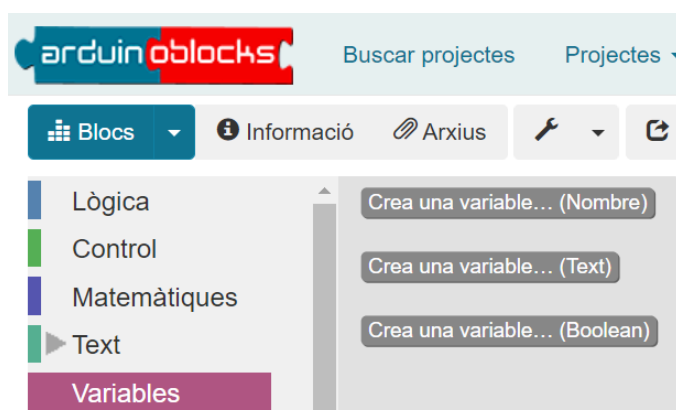
Pensa en una variable com una caixa amb una etiqueta: pots posar-hi a dins diferents valors i canviar aquests valors sempre que vulguis.

Per exemple, si estem programant un joc, podries tenir una variable anomenada "punts" per guardar la puntuació del jugador. Cada vegada que el jugador guanya punts, el valor de la variable "punts" canvia per reflectir la nova puntuació. Així, una

variable és una manera de guardar i gestionar dades que poden variar durant l'execució d'un programa.

Per definir una variable a arduinoblocks s'ha de seleccionar el menú "variables", on se'n poden crear de tres tipus:

- Nombre: emmagatzema valors numèrics
- Text: emmagatzema text
- Boolean: emmagatzema un estat binari ON o OFF (cert o fals)



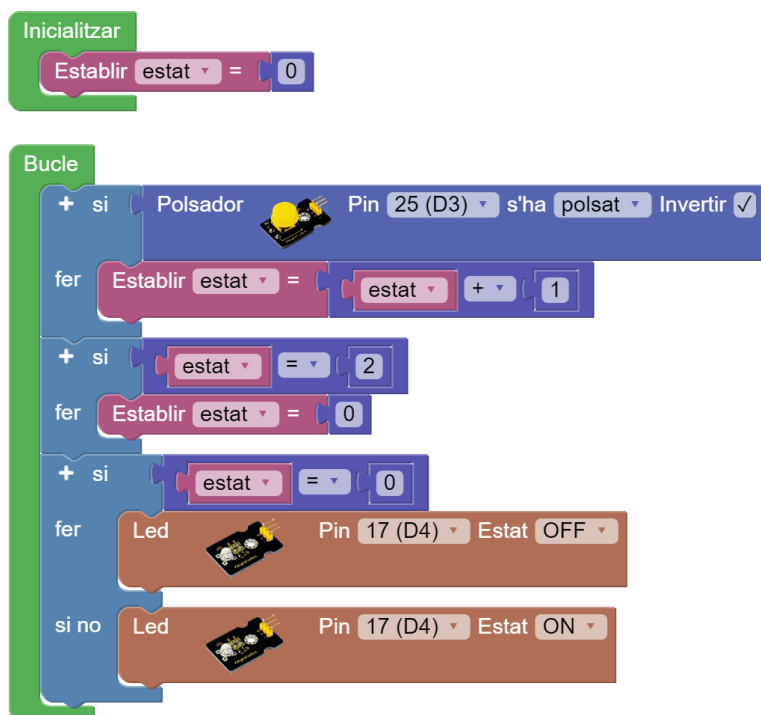
El sistema ens permet assignar a les variables un nom simbòlic, com per exemple "temperatura_exterior", "velocitat", "posició_servo_1",... per a facilitar el seu ús. (Tot i que arduinoblocks permet crear variables on el seu nom pot contenir espais en blanc, cal acostumar-nos a no fer-ne ús, de la mateixa manera que altres caràcters problemàtics com per exemple la "ç" o la "ñ", ja que en programació per codi no es pot fer).

En el cas que ens ocupa, definim una variable de tipus numèric: "estat" que emmagatzemarà l'estat del LED. Si el valor és 0 el LED ha d'estar apagat, si és 1, encès. Fem servir una operació de suma per canviar el valor de la variable cada vegada que es premi el polsador. Com que només treballarem amb el valor 0 i el 1, quan arribi a 2 li tornem a donar el valor 0.

En el programa que farem a continuació necessitem assegurar-nos que cada vegada que accionem el polsador sigui detectat com un sol senyal ja que realitzarem un comptador. Per tant hem de fer servir el segon bloc de polsador, que agafa el senyal "filtrat", és a dir, només detecta una pulsació bé en ser premut o en ser deixat anar.



Aquest seria el programa resultant:



REPTES AMPLIACIÓ

- Prova de fer el programa de la pràctica 2.1 amb l'altre bloc del pulsador, fixa't que en aquest hi han dues possibles accions s'ha polsat i s'ha deixat anar.
- Canvi de color: programa el pulsador de manera que cada vegada que el premis s'encengui un LED diferent en una sèrie repetitiva. Per fer aquesta ampliació has de connectar a la placa electrònica la resta de leds del multientrenador.

P03. Sensor de llum (sensor analògic)



DESCRIPCIÓ

- Mostrar el valor del sensor de llum a través de la consola sèrie.
- Engegar un LED automàticament quan es faci fosc (interruptor crepuscular)

SABERS

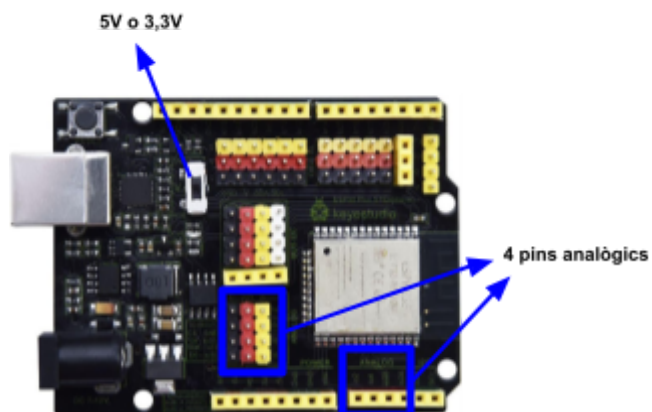
- Entrada analògica:

Imagina un control de volum a casa. Pots girar el botó per fer sonar la música més forta o més baixa. A diferència d'un interruptor que només té dos estats (engegat i apagat), el control de volum permet ajustar el so a molts nivells diferents. Els pins d'entrada analògica de la placa ESP32 STEAMakers funcionen de manera semblant.



Un pin d'entrada analògica llegeix voltatges que poden tenir molts valors diferents entre 0V i 5V (o 3,3V). Aquest pin converteix el voltatge analògic en un número digital que pot entendre la placa ESP32 STEAMakers. Aquest número digital va des de 0 fins a 4095. Aquests valors son deguts a que les entrades analògiques de la placa son transformades a senyal digital gràcies a conversors analògic-digital (ADC) de 12 bits de resolució. Amb 12 bits, el rang de valors és $2^{12}=4096$. Aquests valors van de 0 a 4095 (4096 nivells en total).

Així, quan el pin llegeix 0V, el número digital serà 0. Quan llegeix 5V (o 3,3V), el número digital serà 4095. Qualsevol valor entre 0V i 5V (o 3,3V) es convertirà en un número entre 0 i 4095. Aquesta funció ens permet llegir amb molta precisió la intensitat de la llum, la temperatura, la humitat, o altres senyals analògics.

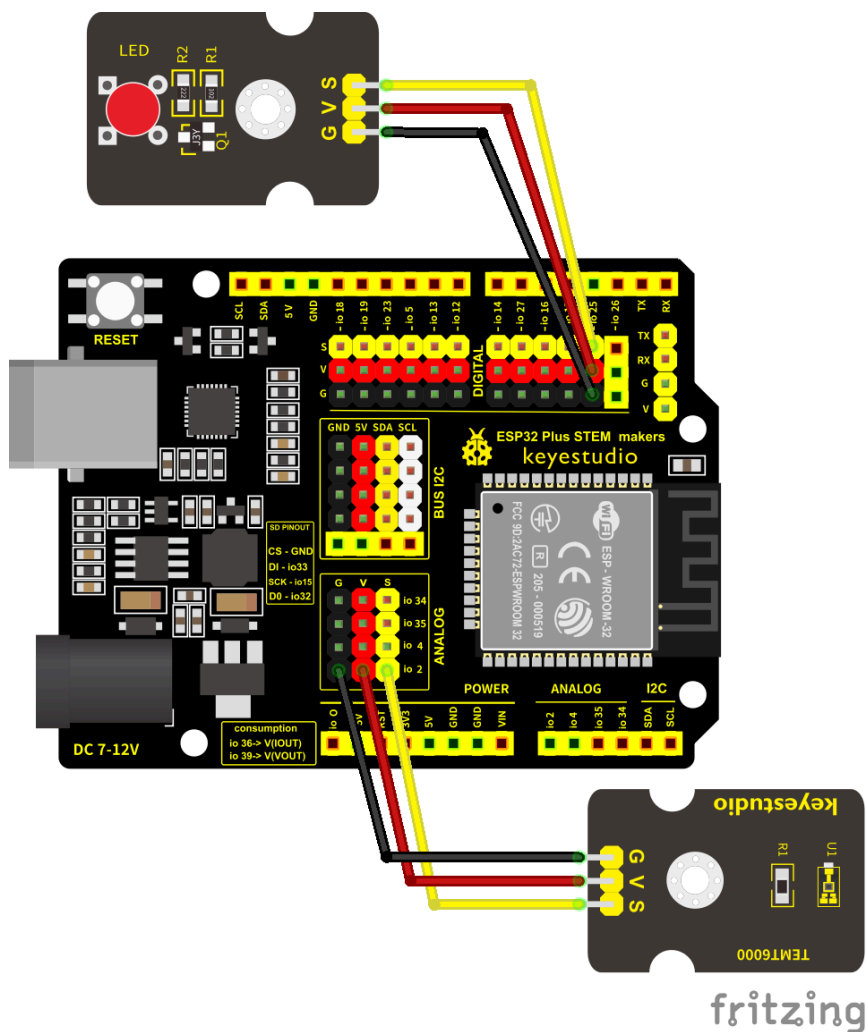
En resum, un pin d'entrada analògica és com un control de volum que pot detectar molts nivells, no només engegat i apagat. Converteix un voltatge entre 0V i 5V en un número digital entre 0 i 4095 perquè la placa ESP32 STEAMakers ho pugui entendre i utilitzar.



LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Sensor de llum (LDR) TEMT6000
-  LED (2a. pràctica)

MUNTATGE:



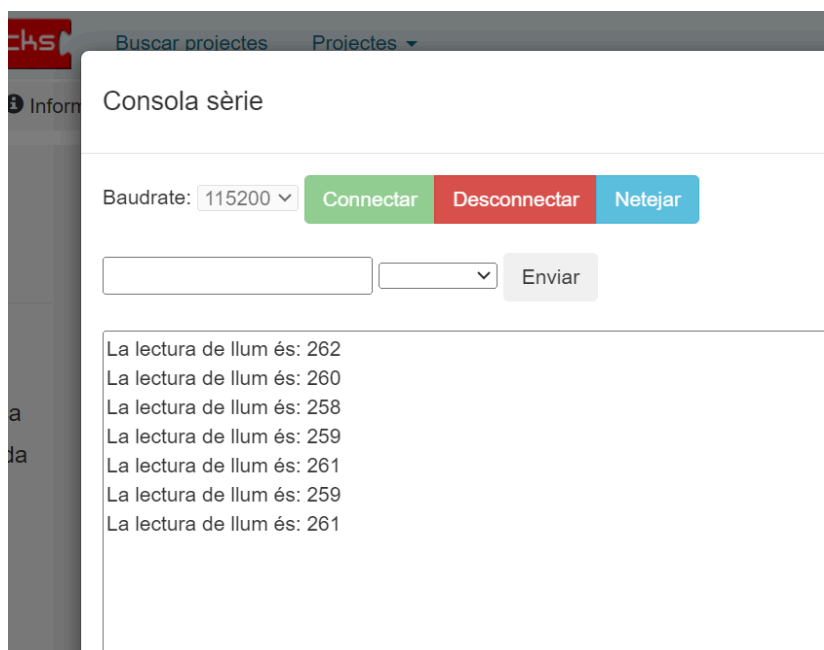
PROGRAMACIÓ 3.1



Després de carregar el programa hauràs d'obrir la consola de comunicació sèrie a una velocitat de 115200. Això farà que la informació del sensor s'envii.



Llavors podràs visualitzar en el teu ordinador els valors que envia el sensor.



percentatge.

També pots veure aquesta informació en



PROGRAMACIÓ 3.2

El LED s'engegarà automàticament quan es fa fosc. Haureu d'establir el valor llindar adequat a les vostres condicions ambientals (en el programa 5%).



REPTES AMPLIACIÓ

- Joc de reflexos: Crea un joc on els jugadors han de reaccionar ràpidament posant la mà a sobre del sensor LDR quan un LED s'encén. El joc encén el LED de manera aleatòria, i els jugadors han de cobrir el sensor de llum amb la mà tan ràpidament com sigui possible. El temps de reacció es pot mesurar i mostrar a la consola sèrie per veure qui ha estat el més ràpid.
- Alarma de llum: Activa un brunzidor quan el nivell de llum cau per sota d'un cert llindar.
- Detector de pas: Comptar el nombre d'elements que creuen el feig de llum. S'entén que en funció de les condicions particulars de la llum ambiental s'haurà d'anar manipulant el llindar.
- Estalvi d'Energia en il·luminació: Redueix la intensitat dels llums quan hi ha prou llum natural, utilitzant el sensor LDR per detectar la llum ambiental.

P04. LED RGB



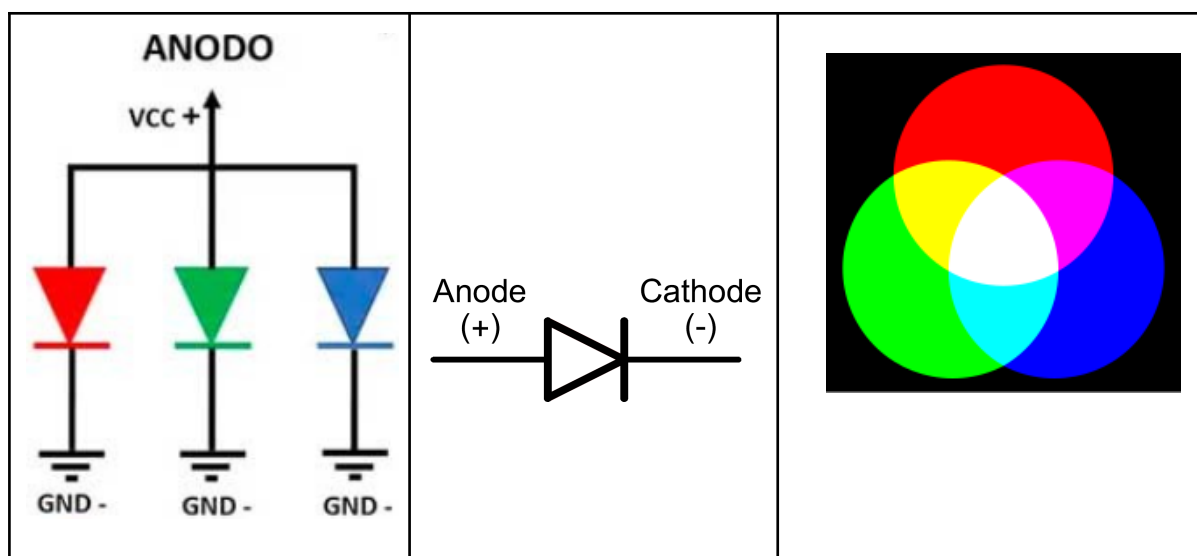
OBJECTIUS

- Engegar un LED RGB amb diferents colors.
- Fer diferents combinacions de colors

SABERS



- LED RGB:

Un led RGB és un led que incorpora al seu mateix encapsulat tres leds. Les sigles RGB corresponen a: R (Red=vermell), G (Green=verd) i B (Blue=blau). Amb aquests tres colors, en òptica, es pot formar qualsevol altre color, ajustant de manera individual la intensitat de cadascun.

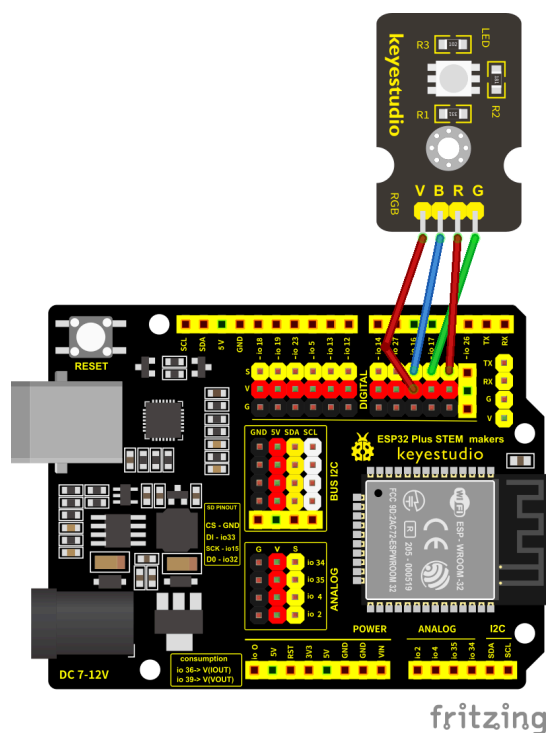


A Arduino, cadascun d'aquests leds podria prendre 256 colors diferents, és a dir, el Vermell podria anar des de 0 fins a 255, el Verd de 0 a 255 i el Blau de 0 a 255, en total un led RGB podria donar més de 16,5 milions de colors diferents.

LLISTA D'ELEMENTS

- 1 Placa ESP32
-  LED RGB
-  Polsador

MUNTATGE



fritzing

PROGRAMACIÓ 4.1

Inicialitzar



Bucle



Podeu provar aquest programa per veure com només apareix el color vermell.

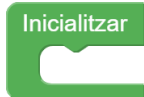
Reviseu que està amb ànode comú.

Podeu fer diferents combinacions per trobar colors diferents.

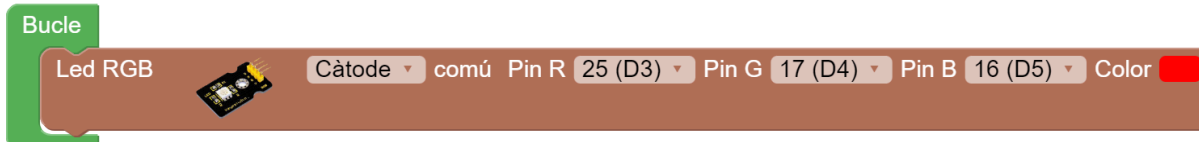
Observeu [aquest web](#) i trobareu algunes d'aquestes combinacions.

També hi ha un bloc d'arduinoblocks que permet agafar directament el color d'una taula.

Inicialitzar



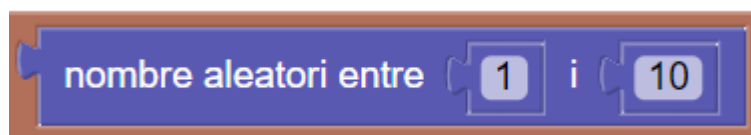
Bucle



SABERS

- Nombre aleatori entre dos valors:

El bloc de nombre aleatori entre dos valors és una eina de programació que genera un nombre de manera aleatòria dins d'un rang específic. Aquest rang es defineix per dos valors: un valor mínim i un valor màxim. Cada vegada que s'utilitza aquest bloc, retorna un nombre diferent dins del rang indicat.



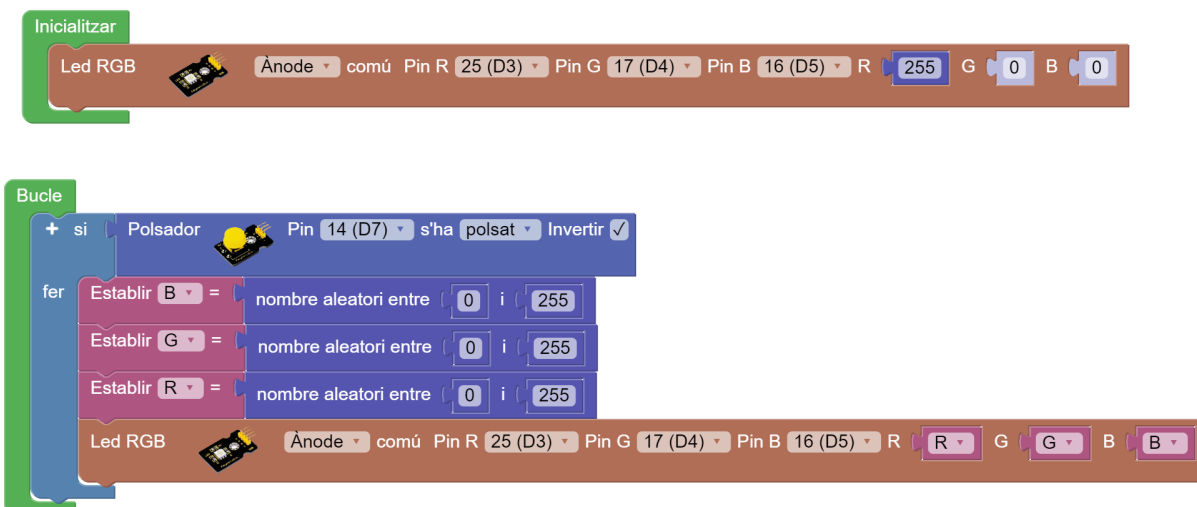
Exemple: Si configurem el bloc per generar un nombre aleatori entre 1 i 10, cada vegada que el programa s'executi, el bloc pot retornar qualsevol nombre enter entre 1 i 10, com ara 3, 7 o 10. Aquest bloc és molt útil per a jocs, simulacions i altres aplicacions on es necessiti variabilitat o imprevisibilitat.

PROGRAMACIÓ 4.2

Provarem de fer un programa per mostrar un color aleatori cada vegada que es premi el polsador.

S'han de crear tres variables, una per a cada LED.

Haureu observat que al menú no apareixen totes les variables que heu definit, es queda amb l'última variable en pantalla, però les altres estan guardades, només cal fer click al desplegable.



Per poder observar correctament els colors, podeu posar un full en blanc sobre el RGB.



REPTES D'AMPLIACIÓ

- **Atenuació Suau de Colors:** Crea un efecte de transició suau entre colors amb el LED RGB. Programa el LED RGB perquè canviï de color de manera gradual i suau en un cicle continu. Com si passes pels diferents colors de l'Arc de Sant Martí.
- **Semàfor:** Simula el funcionament d'un semàfor utilitzant els colors vermell, groc i verd. Programa el LED RGB perquè canviï de color en seqüència amb intervals de temps específics.
- **Joc de Memòria de Colors:** Crea un joc on els jugadors han de recordar i repetir una seqüència de colors. Programa el LED RGB per mostrar una seqüència de colors que els jugadors han de repetir prement botons.
- **Codi Morse:** Fes servir el LED RGB per transmetre missatges en codi Morse canviant els colors. Programa el LED RGB per mostrar missatges en codi Morse amb diferents colors per punts i ratlles.

P05. Mesurar la temperatura i humitat amb DHT-11

OBJECTIUS

- Llegir valors de temperatura i humitat utilitzant el sensor DHT11

SABERS

El sensor DHT11 permet fer la lectura de la temperatura i de la humitat utilitzant el mateix pin digital.

Aquest sensor mesura temperatures en un rang d'acció de 0°C a +50°C amb un error de +/- 2°C i la humitat relativa entre 20% i 90% amb un error de +/-5%. No és un sensor amb una gran sensibilitat, però compleix els nostres objectius sobradament.

El sensor de temperatura és un termistor tipus NTC. Un termistor és un tipus de resistència (component electrònic) el valor del qual varia en funció de la temperatura d'una manera més acusada que una resistència comuna.



El seu funcionament es basa en la variació de la resistivitat que presenta un semiconductor amb la temperatura.

El terme prové de l'anglès "thermistor", el qual és un acrònim de les paraules Thermally Sensitive Resistor (resistència sensible a la temperatura).

Hi ha dos tipus fonamentals de termistors:

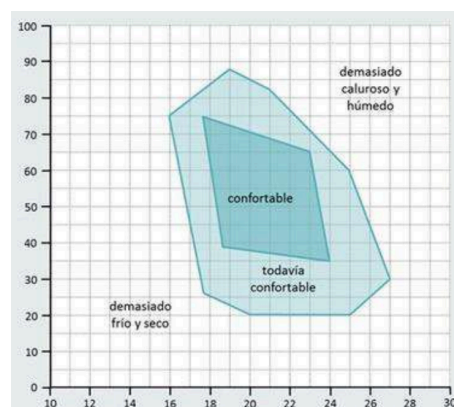
- Els que tenen un coeficient de temperatura negatiu (en anglès Negative Temperature Coefficient o NTC), els quals decreixen la seva resistència a mesura que augmenta la temperatura.

- Els que tenen un coeficient de temperatura positiu (en anglès Positive Temperature Coefficient o PTC), els quals incrementen la seva resistència a mesura que augmenta la temperatura.


Zona de confort higrotèrmic

Es pot definir confort tèrmic, o més pròpiament confort higrotèrmic, com l'absència de malestar tèrmic. En fisiologia, es diu que hi ha confort higrotèrmic quan no han d'intervenir els mecanismes termoreguladors del cos per a una activitat sedentària i amb una indumentària lleugera. Aquesta situació es pot registrar mitjançant índexs que no han de ser sobrepassats perquè no es posin en funcionament els sistemes termoreguladors (metabolisme, sudoració i altres).

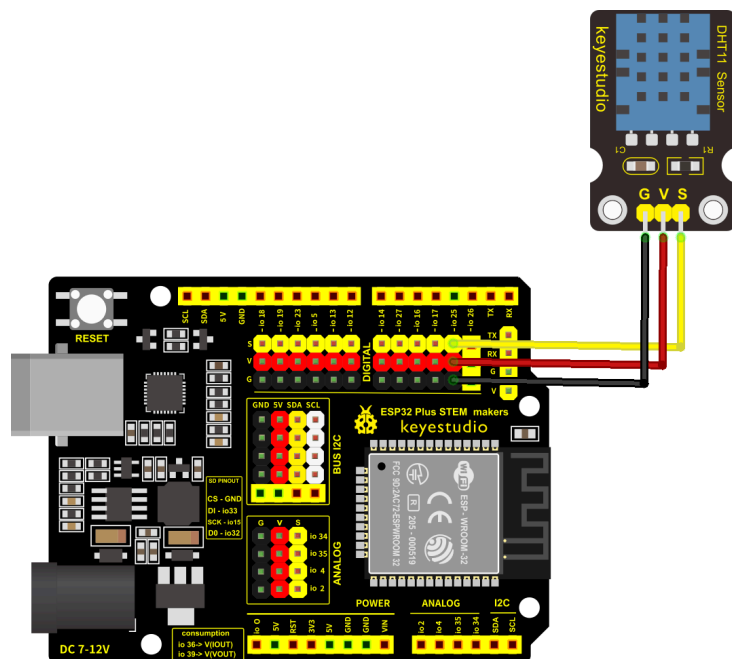
Segons la imatge adjunta anem a marcar uns punts de temperatura i humitat als que estarem dins de la zona de confort tèrmic, dins d'una zona de mig confort i fora de la zona de confort.



LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Sensor DHT11

MUNTATGE



fritzing

PROGRAMACIÓ 5.1

En aquesta activitat controlarem la temperatura i la humitat amb el sensor DHT11 connectat el pin IO25(D3). Mostrarem les dades per consola:

```

Inicialitzar
  Iniciar Bauds 115200

Bucle
  Establir temp = DHT-11 Temperatura °C Pin 25 (D3)
  Establir hum = DHT-11 Humitat % Pin 25 (D3)
  Executar cada 500 ms
    Enviar " Temperatura: " Salt de línia
    Enviar temp Salt de línia
    Enviar " Humitat: " Salt de línia
    Enviar hum Salt de línia
  
```

PROGRAMACIÓ 5.2

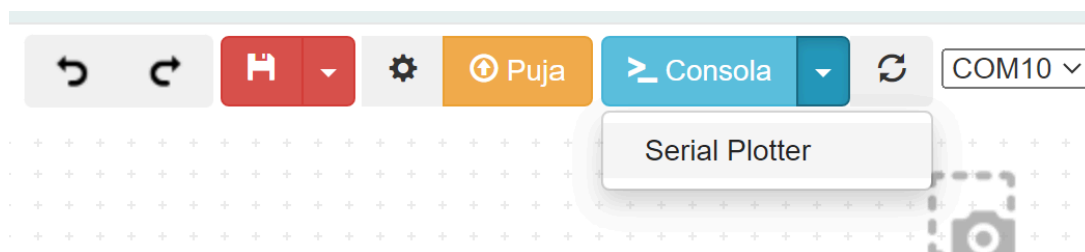
Mostrar les dades de temperatura i humitat al plotter:

```

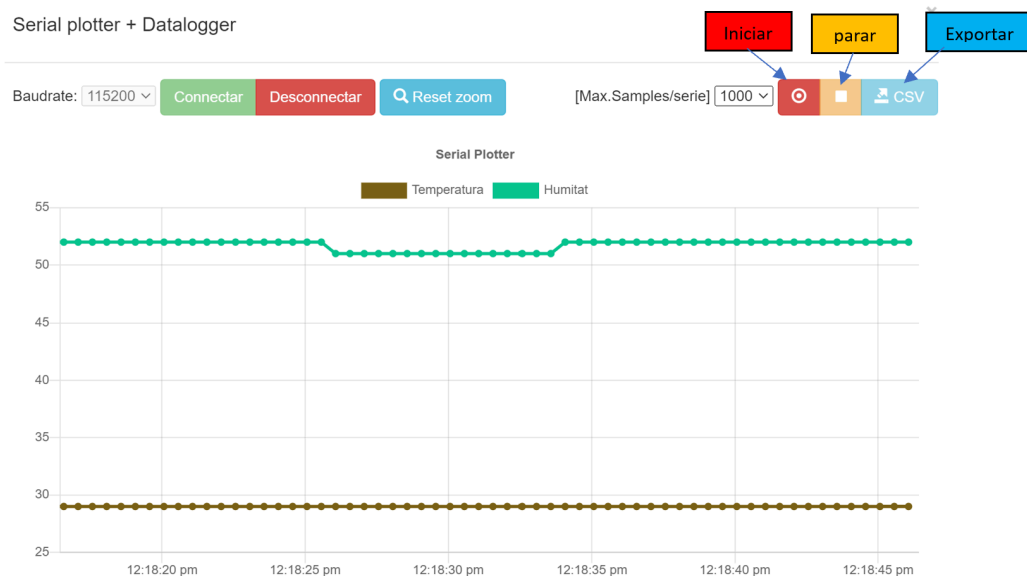
Inicialitzar
  Iniciar Bauds 115200

Bucle
  Establir temp = DHT-11 Temperatura °C Pin 25 (D3)
  Establir hum = DHT-11 Humitat % Pin 25 (D3)
  Executar cada 500 ms
    Plotter " Temperatura " Valor temp
    Plotter " Humitat " Valor hum
  
```

Enviem el programa a la placa i activem el Serial Plotter a través de la pestanya que hi ha al costat de Consola.



Podem enregistrar les dades en format .csv. Per fer-ho hem d'iniciar la captura de dades i finalitzar-la per poder generar els fitxers .csv:



Es crearà un arxiu .csv per a cada variable que enregistrem. En aquest exemple es crearan 2 arxius .csv, un arxiu amb les dades enregistrades de temperatura i un altre amb les dades enregistrades d'humitat.

	A	B	C
1	Plotter + Datalogger		
2	DateTime	Humitat	
3	15/7/2024 12:10	51	
4	15/7/2024 12:10	51	
5	15/7/2024 12:10	51	
6	15/7/2024 12:10	51	
7	15/7/2024 12:10	51	
8	15/7/2024 12:10	51	
9	15/7/2024 12:10	51	
10	15/7/2024 12:10	51	

REPTES AMPLIACIÓ

- Semàfor de zona de confort: Definir un rang de confort i dissenyar els límits amb un led RGB. Amb el Led RGB indicarem aquestes zones:
 - Led RGB en VERMELL; fora de la zona de confort.
 - Led RGB en VERD; a la zona de confort.

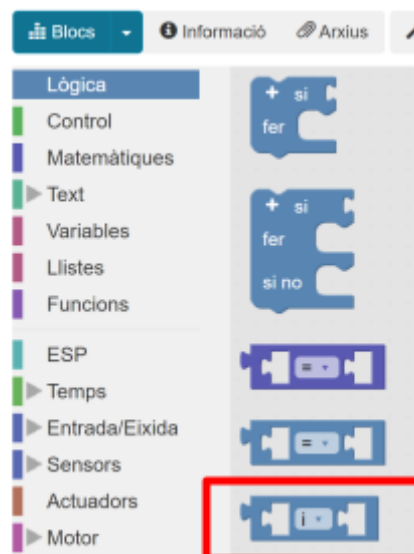
Zona VERDA:

- Humitat entre 40% i el 65%.
- Temperatura entre 18°C i 24°C.

El led brillarà a VERD dins dels paràmetres de la Zona VERDA, per a la resta el led estarà parpellejant en color VERMELL.

Per realitzar aquest programa necessitarem diversos blocs del menú de Lògica. Necessitarem avaluar una condició Lògica i utilitzar conjuncions i disjuncions.

- I: es compleix si els dos operands són veritat.
- O: es compleix si algun dels dos operands és veritat



P06. Pantalla OLED

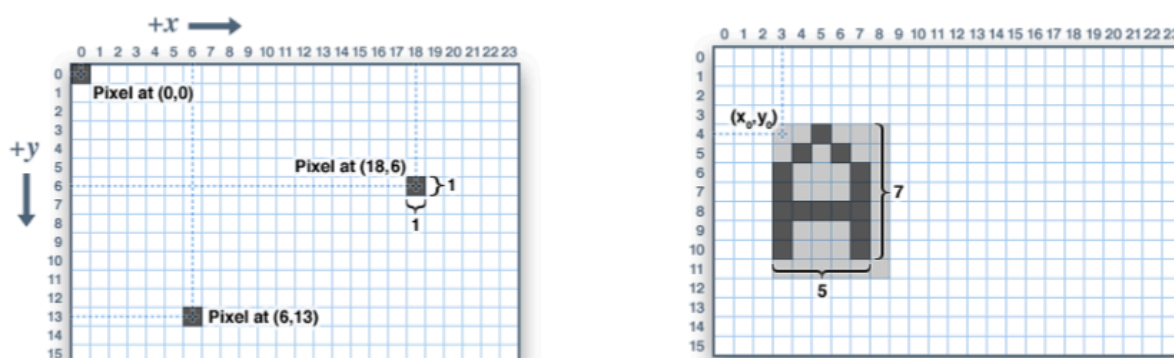
OBJECTIUS

- Mostrar informació per la pantalla OLED en temps real.

SABERS

La pantalla OLED la connectarem al port de comunicacions I2C respectant les connexions.

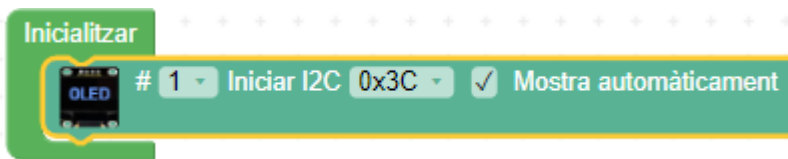
La pantalla que utilitzarem és una pantalla OLED de 0,96 polzades de 128x64 pixels.



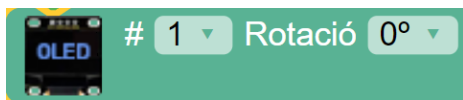
OLED és l'abreviatura de díode emissor de llum orgànic. A escala microscòpica, una pantalla OLED és una matriu de leds que s'il·luminen quan emeten energia. La nova tecnologia OLED sols utilitza electricitat per a cada píxel que vulguem engegar. Això fa que la tecnologia OLED sigui molt eficient. A més, la forma en què es construeixen aquests tipus de OLED permet que siguin molt primes en comparació amb l'antiga pantalla LCD.

Existeixen una sèrie de blocs específics per a controlar la pantalla.

- Primer configurarem l'adreça I2C de la pantalla OLED. En el cas de la pantalla utilitzada, haurem de seleccionar 0x3C.



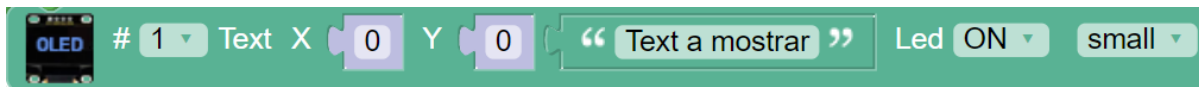
- Rotació: Permet indicar en quina posició està situada la pantalla.



- Netejar: Elimina tot el contingut de la pantalla.



- Text: Mostra un text en la posició indicada. Donat que la pantalla és monocroma, podem indicar si el text s'ha de mostrar en positiu (engegant pixels a la pantalla) o en negatiu (apagant pixels). A més podem especificar 3 mides de lletra a la pantalla.



- Per treure una imatge per la pantalla ho podem fer a través d'un mapa de bits, que podem carregar des d'un arxiu o editar-lo des de l'editor de mapes de bits.



Obrim l'editor de mapes de bits.

- Per a dibuixar un rectangle li has de donar el primer vèrtex, amplada i alçada del rectangle.



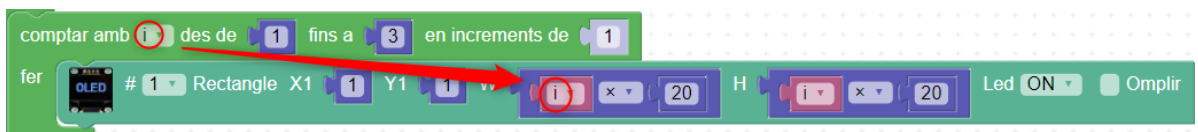
- Per a dibuixar un cercle li has de donar el centre i el radi.




- Per a dibuixar una línia li has d'indicar el punt inicial i el final.



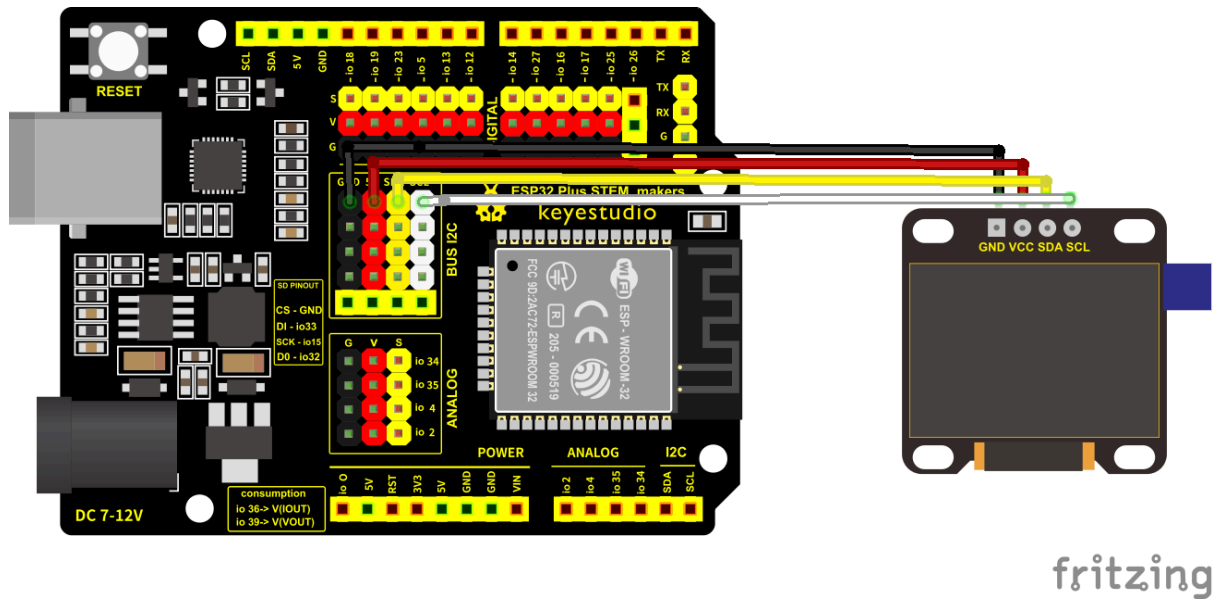
- Per a dibuixar diferents figures amb una raó de proporcionalitat podem utilitzar un comptador com a variable.



LLISTA D'ELEMENTS

- Placa ESP32 STEAMakers.
-  Pantalla OLED

MUNTATGE



PROGRAMACIÓ 6.1: MOSTRAR UN MISSATGE

Inicialitzar

1 Iniciar I2C 0x3C Mostra automàticament

Bucle

1 Text X 0 Y 5 + - crear text amb "Hola Món" Led ON medium

PROGRAMACIÓ 6.3: DIBUIXAR FIGURES PER COORDENADES.

- Dibuixarem 3 quadrats de mides diferents que comparteixen un vèrtex, 3 cercles concèntrics de diferent radi i un triangle donats els seus vèrtexs.

Inicialitzar

1 Iniciar I2C 0x3C ✓ Mostra automàticament

Bucle

comptar amb i des de 1 fins a 3 en increments de 1 **3 quadrats amb vèrtex comú**

fer # 1 Rectangle X1 1 Y1 1 W 20 H 20 Led ON Omplir

Esperar 3000 mil·lisegons

1 Netejar

comptar amb i des de 1 fins a 3 en increments de 1 **3 cercles concèntrics**

fer # 1 Cercle X 64 Y 32 R 10 Led ON Omplir

Esperar 3000 mil·lisegons

1 Netejar

Un triangle a partir dels seus vèrtexs

1 Línia X1 60 Y1 62 X2 120 Y2 62 Led ON

1 Línia X1 120 Y1 62 X2 90 Y2 2 Led ON

1 Línia X1 90 Y1 2 X2 60 Y2 62 Led ON

Esperar 3000 mil·lisegons

1 Netejar

REPTES AMPLIACIÓ

- Amb polsadors, seleccionar la informació que volem veure per pantalla.
- Mostrar informació, utilitzant diferents mides de text.
- Carregar una imatge Bitmap.
- Dibuixar línies dirigides amb polsadors (o amb el joystick)
- Mostrar T i H en la pantalla OLED
- Fer torn d'una botiga (augment un quan es pren un polsador) i mostrar-ho per la pantalla.

P07. Matriu LED 8X8

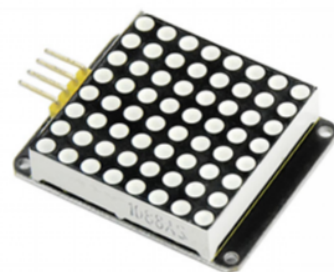


OBJECTIUS

- Mostrar informació per la matriu de 8x8 leds en temps real.

SABERS

Les matrius de leds permeten controlar 64 leds monocroms (normalment vermells, blaus o verds) disposats en forma de matriu de 8 files i 8 columnes. Per a controlar els leds la matriu té un xip controlador (HT16K33) que gestiona els leds i es comunica amb la placa controladora mitjançant el bus I2C.

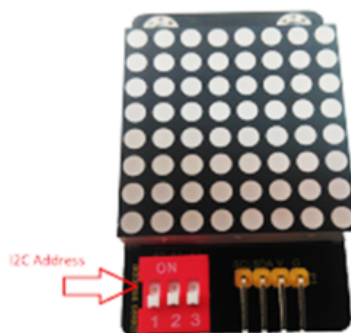


Podem connectar diverses matrius de leds al mateix bus I2C, per a controlar-les individualment hem de canviar la direcció de cadascuna d'elles.

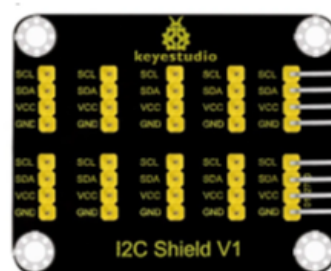
Configuració de la direcció I2C mitjançant pins soldats



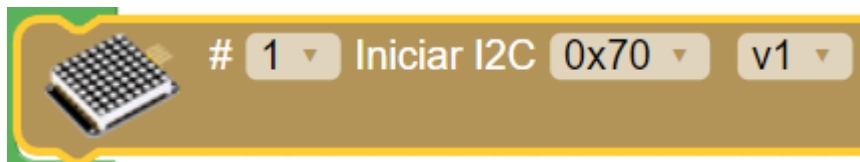
Configuració de la direcció I2C mitjançant micro-switchs



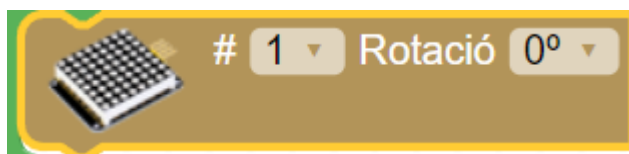
Hub I2C per a ramificar i connectar diversos dispositius al mateix bus I2C



- Iniciar: Inicialitza una matriu en una direcció en concret. Posteriorment podrem referir-nos a ella mitjançant el ID #



- Rotació: Permet ajustar l'orientació de la matriu led.



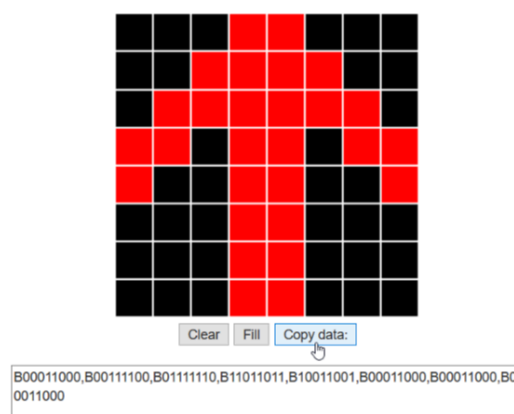
- Netejar: Apaga tots els leds de la matriu.



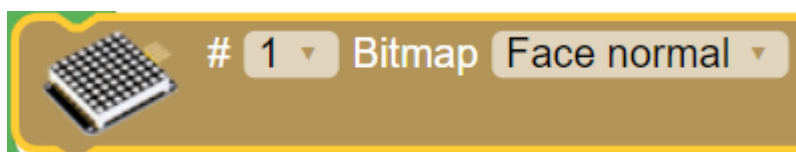
- Bitmap: Permet establir un mapa de bits per a dibuixar a la matriu led. Mitjançant l'opció "Ajuda" del bloc podem accedir a l'editor de mapa de bits.



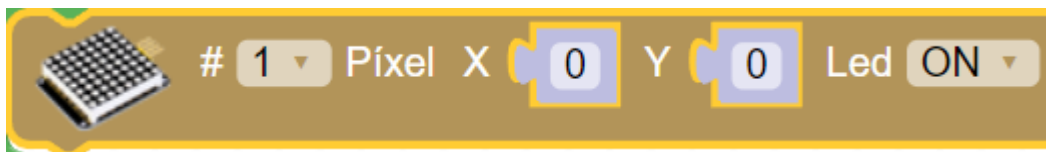
LedMatrix - Bitmap Data



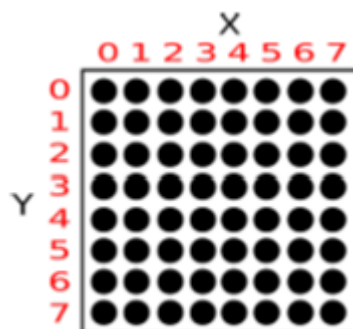
- Bitmap (predefinit): Mitjançant aquest bloc podem carregar bitmaps predefinitos d'una forma molt senzilla.



- Píxel: Encèn o apaga un led en concret indicant-li les seves coordenades.



Sistema de coordenades de la matriu led 8x8



- Línia: Permet dibuixar una línia recta entre dues coordenades a la matriu led.
Punt origen: X1, Y1

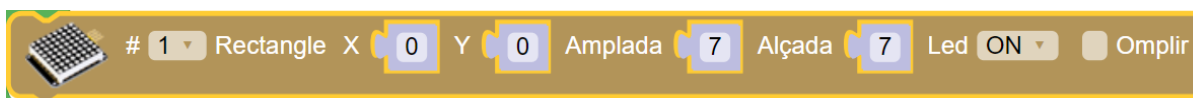
Punt destinació: X2, Y2



- Rectangle: Dibuixa un rectangle indicant les coordenades d'una cantonada i la contrària.

Cantonada 1: X1, Y1

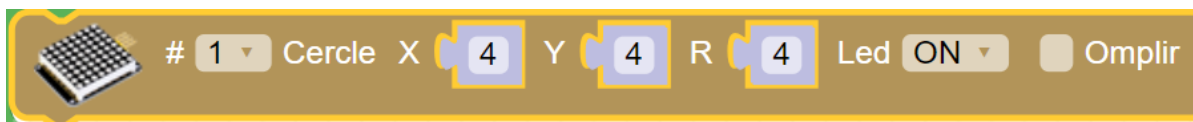
Cantonada 2: X2, Y2



- Cercle: Dibuixa un cercle indicant les coordenades del centre i el nombre de píxels de radi.

Centre: X, Y

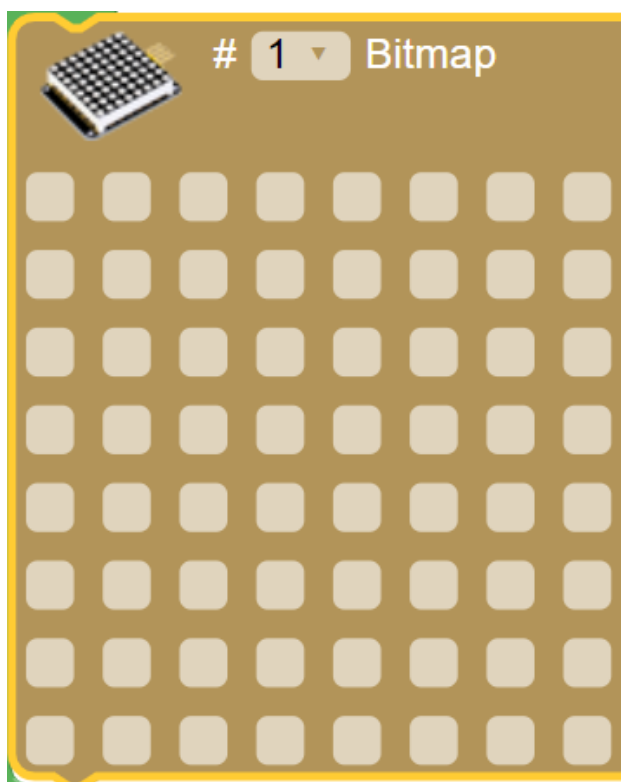
Radi: R



- Text: Dibuixa un text a la matriu led. Utilitzant coordenades fora de les visibles podem realitzar textos amb desplaçament amb una única matriu led.



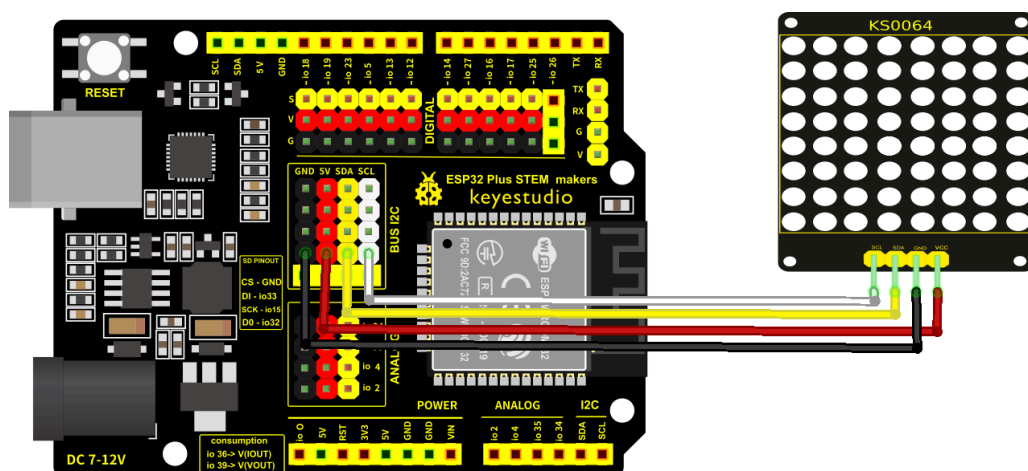
- Bitmap: Permet definir un mapa de bits directament marcant els píxels encesos i apagats al propi bloc d'una forma visual.



LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Matriu LED 8x8
-  Polsador

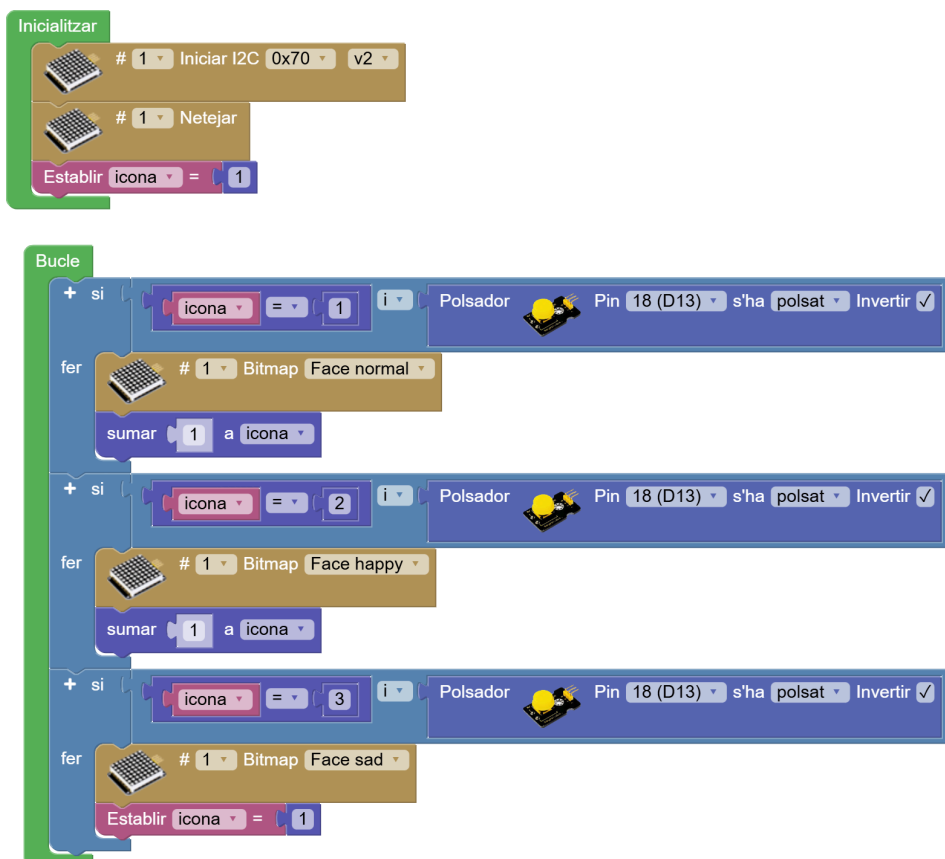
MUNTATGE



fritzing

PROGRAMACIÓ 7.1

Farem un programa que vagi mostrant diferents icones. Cada vegada que premem el polsador la matriu LED mostra una icona diferent.



REPTES D'AMPLIACIÓ

- Dibuixar figures senzilles.
- Mostrar un missatge de text en moviment

P08. Llista de reproducció amb el brunzidor

OBJECTIUS

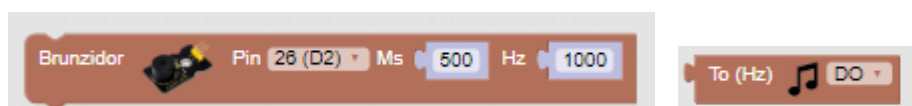
Crear una llista de reproducció de música.

SABERS

- El brunzidor (*Buzzer* en anglès) és un transductor electroacústic que produeix un so o brunzit continu o intermitent. En funció de si es tracta d'un brunzidor Actiu o Passiu, aquest brunzit serà del mateix to o el podem variar. Serveix com a mecanisme de senyalització o avís i s'utilitza en múltiples sistemes, com en automòbils o en electrodomèstics.

Per controlar el brunzidor passiu i fer que sonin diferents notes, necessitem introduir ones quadrades de diferents freqüències al pol positiu del component i posar el pol negatiu a terra, per fer-ho el connectarem a un pin digital de la placa.

El podem controlar reproduint un so a una certa freqüència i durant un cert temps, o bé, mitjançant el bloc de reproduir una melodia RTTTL:



- Les llistes de dades ens permeten emmagatzemar un llistat de valors i accedir a ells per la seva posició en la llista. Les llistes poden ser de tipus numèriques o de text. La creem donant-li un nom i llavors podem anar afegint elements en qualsevol moment.

Inicialitzar

Llista [exemple llista] Afegir Valor 24
 Llista [exemple llista] Afegir Valor 38
 Llista [exemple llista] Afegir Valor 51

posició	0	1	2
valor	24	38	51

Aquí hem creat una llista de tipus numèrica amb 3 valors, ordenats com es veu a la taula. Amb les llistes dinàmiques podem afegir elements amb format CSV separats per comes:

Inicialitzar

Llista [exemple llista] Afegir Valor 24
 Llista [exemple llista] Afegir Valor 38
 Llista [exemple llista] Afegir Valor 51
 Llista [exemple llista] Afegir CSV "23,68,45,87" Separador " ,"

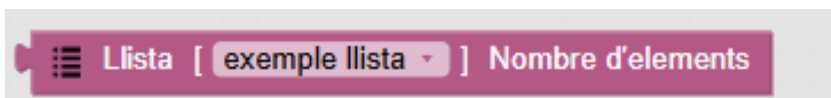
Així la llista anterior d'exemple quedaria de la següent manera:

posició	0	1	2	3	4	5	6
valor	24	38	51	23	68	45	87

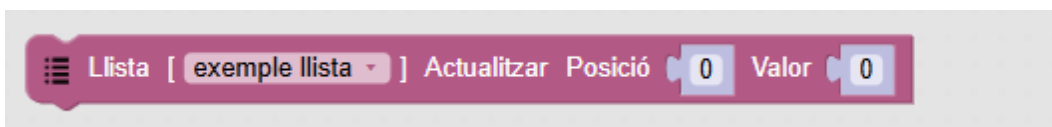
En una llista podem obtenir el valor d'una posició (des de la 0 fins al nombre d'elements -1 en la llista) amb el següent bloc (a l'exemple tenim 7 elements, podem accedir a ells del 0 al 6):

Llista [exemple llista] Obtenir Posició 0

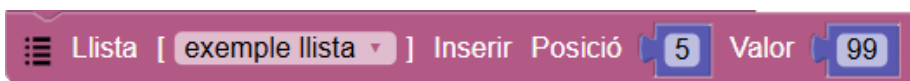
Per a saber el nombre d'elements que tenim en una llista podem usar el bloc "Nombre d'elements".



Per modificar un element de la llista sense alterar la resta:

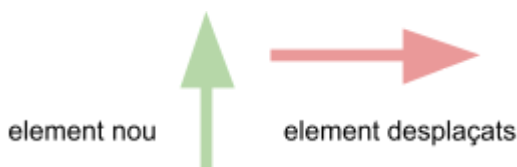


Per afegir un element en un punt determinat de la llista:



Amb aquest últim bloc es desplacen els elements a partir del nou una posició cap amunt. Veiem com quedaria la llista d'exemple:

posició	0	1	2	3	4	5	6	7
valor	24	38	51	23	68	99	45	87



I l'acció contrària que seria eliminar un element de la llista, amb el conseqüent desplaçament cap a baix dels elements següents a la posició eliminada.

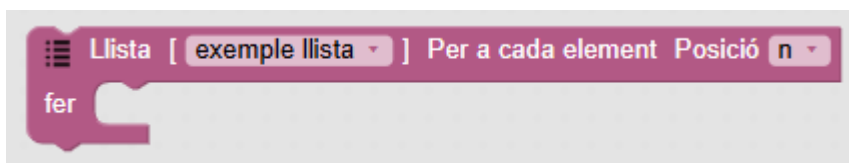


Veiem com quedaria la llista d'exemple si eliminem el element amb índex 2, el 51:

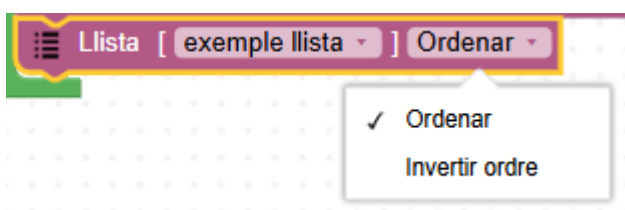
posició	0	1	2	3	4	5	6
valor	24	38	23	68	99	45	87



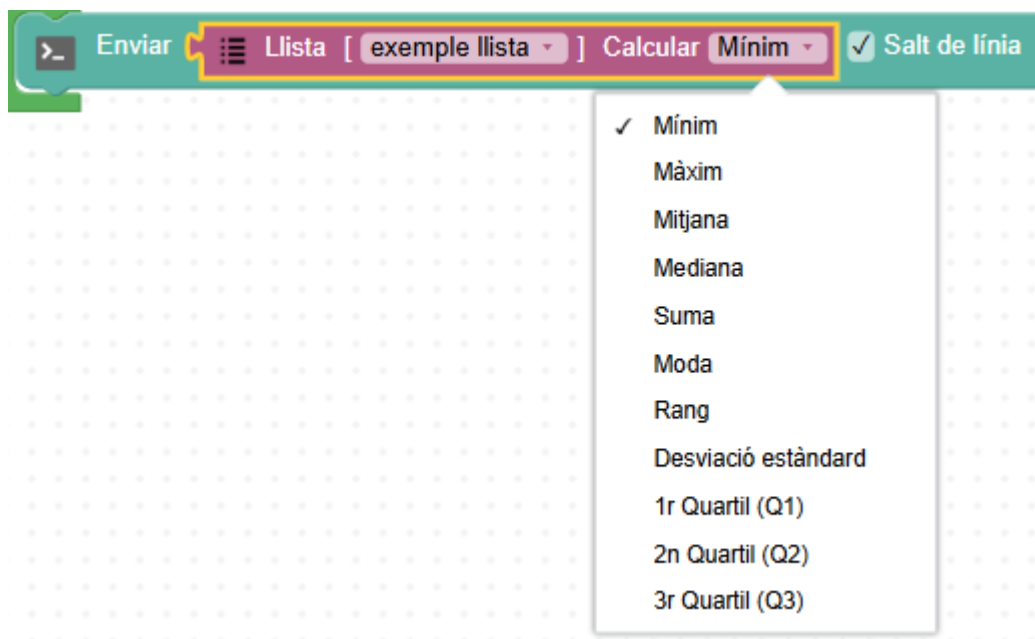
Per recórrer una llista podem fer servir aquest bloc que crea un bucle local:



Amb aquest bloc podem ordenar els elements de la llista de menor a major, o a la inversa:



També tenim la possibilitat d'obtenir alguns valors estadístics de la llista creada amb el bloc "calcular" i les seves opcions:




Per últim, podem eliminar tota la llista amb el següent bloc:



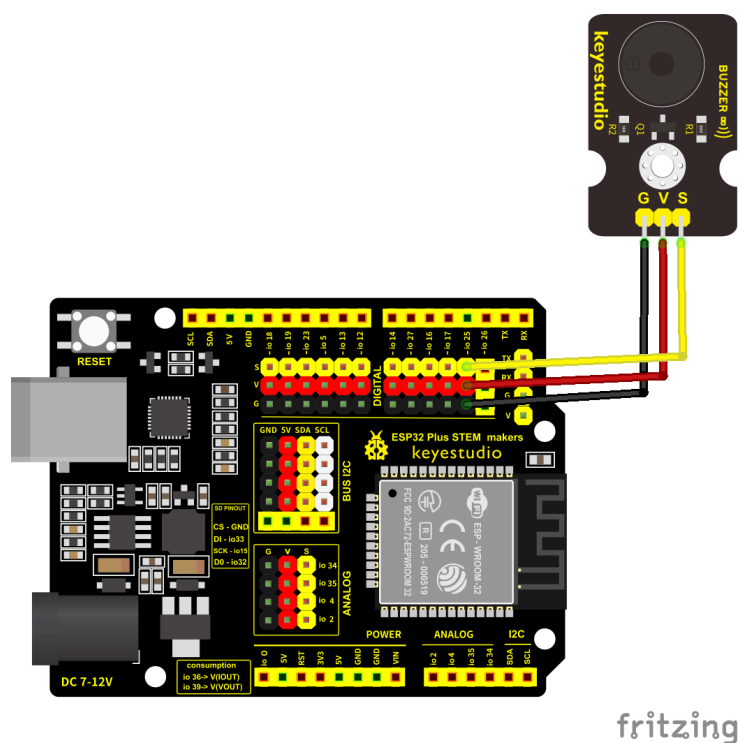
Això resulta important per tal d'alliberar l'espai de memòria de la placa quan no sigui estrictament necessari.

Aquests blocs també estan disponibles per les llistes de text, en color verd, amb algunes funcions específiques diferents com ara ordenar alfabèticament enlloc d'ordenar de major a menor, i sense la part estadística.

LLISTA D'ELEMENTS

- 1 Placa ESP32 STEAMakers
-  Brunzidor

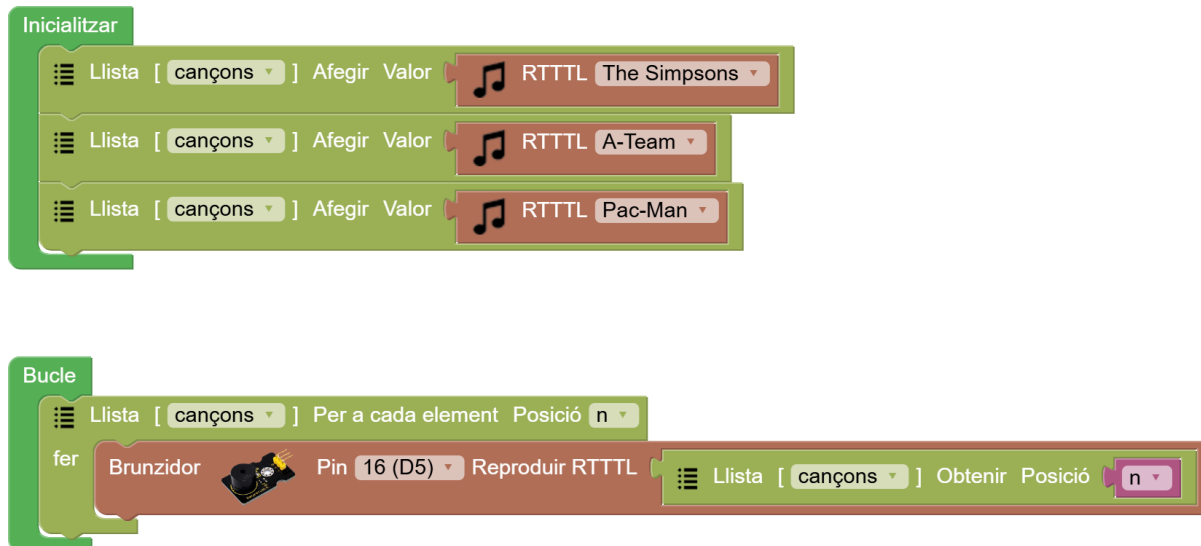
MUNTATGE



fritzing

PROGRAMACIÓ 8.1

Volem programar la placa i el bronzidor per tal de que reproduïxin una llista de cançons prèviament creada:



REPTES AMPLIACIÓ

- Fer que no soni la cançó, o la cançó següent, fins que es premi el polsador.
- Que surti per la pantalla OLED, el nom de la cançó

P09. Mesurar distàncies amb el sensor d'ultrasons



OBJECTIUS

- Mostrar la distància des del sensor fins a un obstacle a la consola sèrie.
- Mostrar la distància des del sensor fins a un obstacle a una pantalla OLED.
- Activar una alarma quan es detecta un objecte a menys de 10 cm.

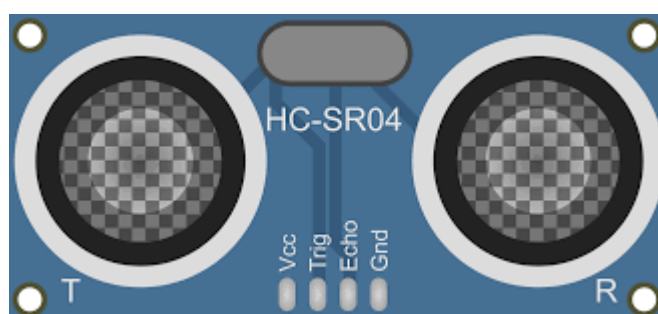
SABERS

- Sensor de distància per ultrasons:



El sensor de distància HC-SR04 utilitza ultrasons per mesurar la distància a un objecte.

Funciona de la següent manera:

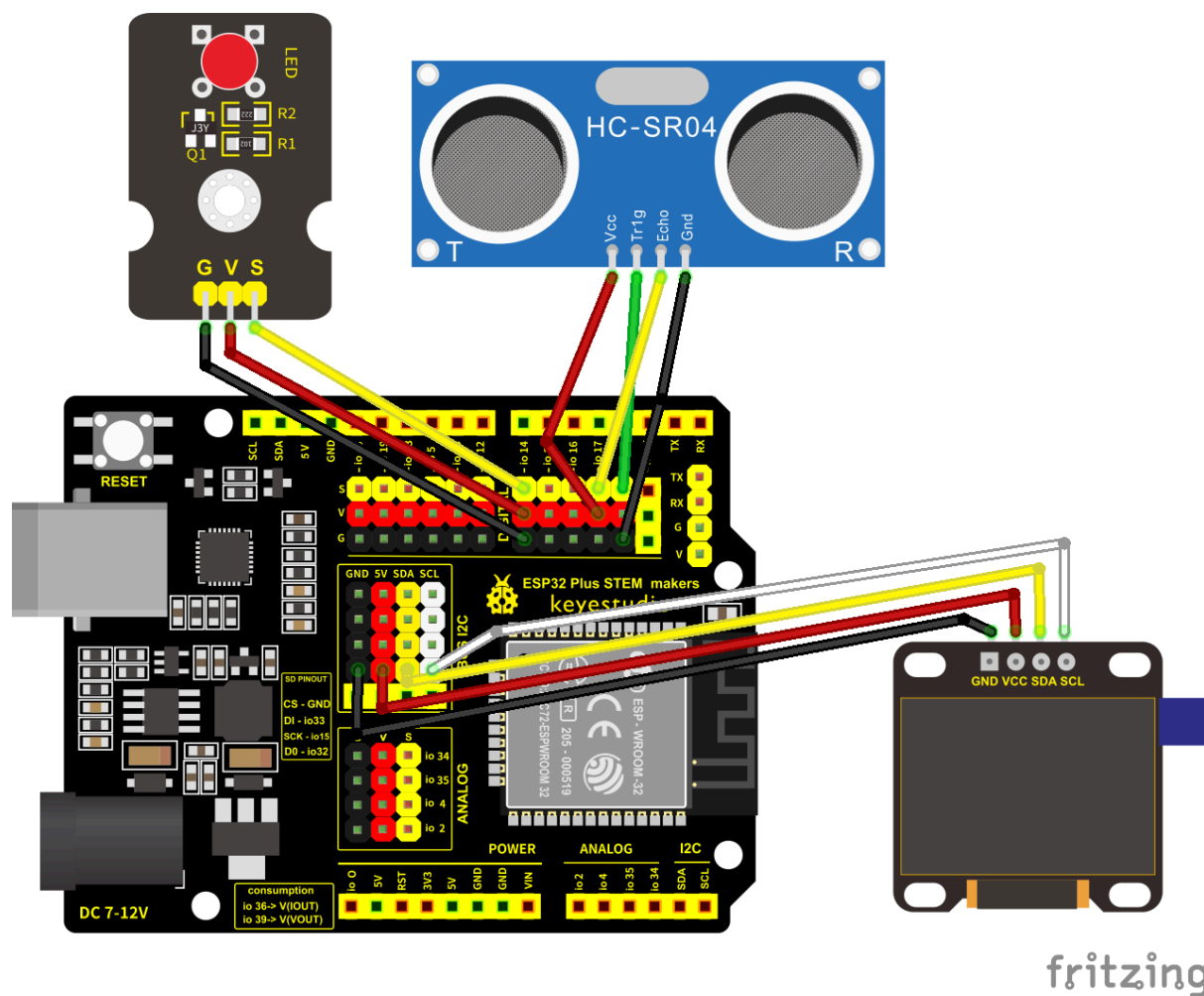
1. **Enviar el Senyal:** El sensor envia un pols ultrasònic mitjançant el pin Trig (trigger, que vol dir dispar).
2. **Rebre l'Eco:** Quan el pols colpeja un objecte, es reflecteix i torna al sensor. El pin Echo detecta aquest eco.
3. **Mesurar el Temps:** El sensor mesura el temps que triga l'eco a tornar.
4. **Calcular la Distància:** Utilitza el temps mesurat per calcular la distància a l'objecte.



LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Sensor d'ultrasons HC-SR04
-  LED

MUNTATGE



PROGRAMACIÓ 9.1

Inicialitzar

Iniciar Bauds 115200

Bucle

Enviar crear text amb " L'obstacle es troba a : "

Distància (cm) [Trigger] 25 (D3) [Echo] 17 (D4)

" cm. "

Esperar 1000 mil·lisegons

Salt de línia

Haureu d'obrir la consola i podeu verificar la distància a la qual es troba l'obstacle. El màxim abast són 152 cm aproximadament.

PROGRAMACIÓ 9.2

Per presentar la informació a la pantalla OLED haurem d'iniciar-la. Recordem que sols accepta missatges de text en anglès, per la qual cosa no accepta accents. Si voleu que la lletra sigui més gran haureu de jugar amb els valors de coordenades de la pantalla OLED.

Inicialitzar

1 Iniciar I2C 0x3C Mostra automàticament

1 Netejar

Bucle

1 Netejar

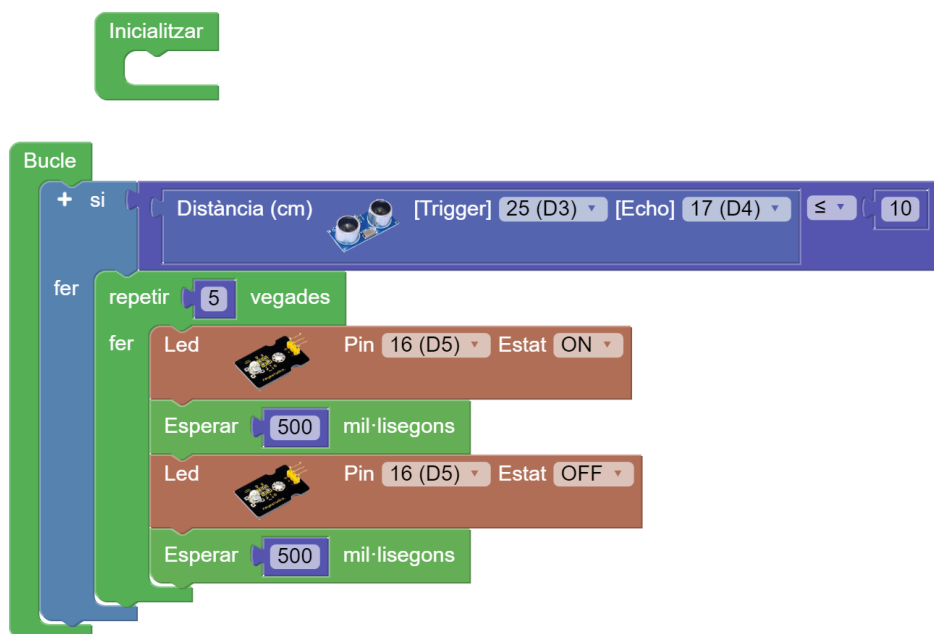
1 Text X 0 Y 0 " La distancia es: " Led ON small

1 Text X 10 Y 10 Distància (cm) [Trigger] 25 (D3) [Echo] 17 (D4) Led ON small

1 Text X 40 Y 10 " cm. " Led ON small

Esperar 500 mil·lisegons

PROGRAMACIÓ 9.3



REPTES AMPLIACIÓ

- **Alarma de proximitat acústica:** Crear una alarma que s'activi quan algú s'apropa massa al sensor d'ultrasons. L'alarma emetrà un so que augmentarà de freqüència a mesura que la persona s'acosti més al sensor, advertint així de la proximitat. Seria el típic exemple de quan t'acostes molt a una obra d'art en un museu i es genera una alarma sonora per proximitat.
- **Theremin:** Crear un Theremin senzill utilitzant un sensor d'ultrasons i un brunzidor. El Theremin és un instrument musical electrònic que es toca sense contacte físic, i en aquest projecte la distància entre la mà i el sensor d'ultrasons determina la freqüència del so produït pel brunzidor.
- **Mesurador d'alçada:** Mesurar de manera estimativa als alumnes posicionant el sensor d'ultrasons a uns 90 cm.
- **Identificador d'alçades:** imagineu que voleu saber qui està davant de la porta, si trobeu una persona d'estatura petita que el sistema posi una cançó infantil, si trobeu una persona de major alçada que posi una cançó diferent.

P10. Controlar un servomotor

OBJECTIUS

- Controlar el moviment d'un servomotor.

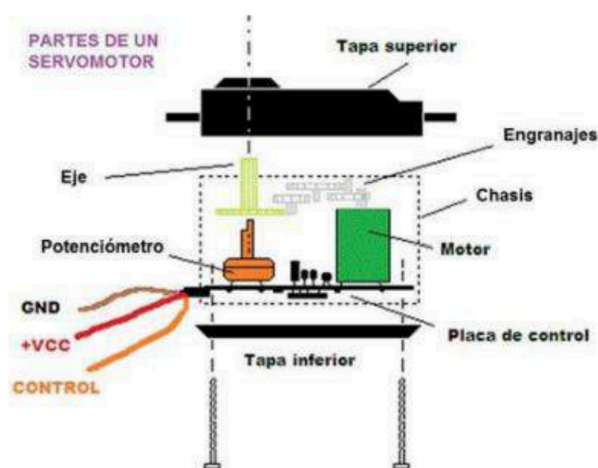
SABERS

Un SERVO és un tipus de motor utilitzat normalment en electrònica. Són còmodes d'utilitzar, ja que podem indicar directament l'angle de rotació desitjat (acostumen a tenir un angle de gir entre 0 a 180°).

- D'una banda, ens permet mantenir una posició que indiquem, sempre que estigui dins del rang d'operació del dispositiu.
- D'altra banda ens permet controlar la velocitat de gir, podem fer que abans que es mogui a la següent posició esperi un cert temps.

Internament, un servo està constituït per un motor de corrent continu, acoblat a un reductor per reduir la velocitat de gir, juntament amb l'electrònica necessària per controlar-ne la posició.

Normalment, es disposa d'un potenciòmetre unit a l'eix del servo, que permet al servo conèixer la posició de l'eix. Aquesta informació la tracta un controlador integrat que s'encarrega d'ajustar i actuar sobre el motor per assolir la posició desitjada.

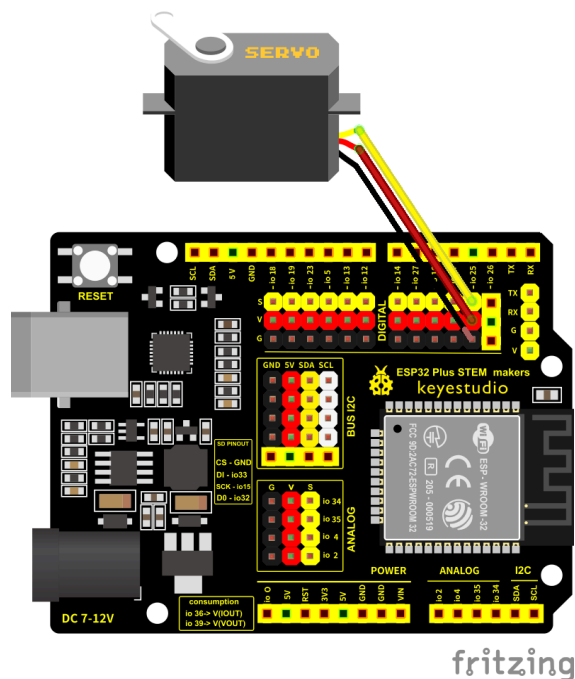


Connectar un servo és senzill. El servo disposa de tres cables, dos d'alimentació (GND i +VCC) i un de senyal de control.

LLISTA D'ELEMENTS

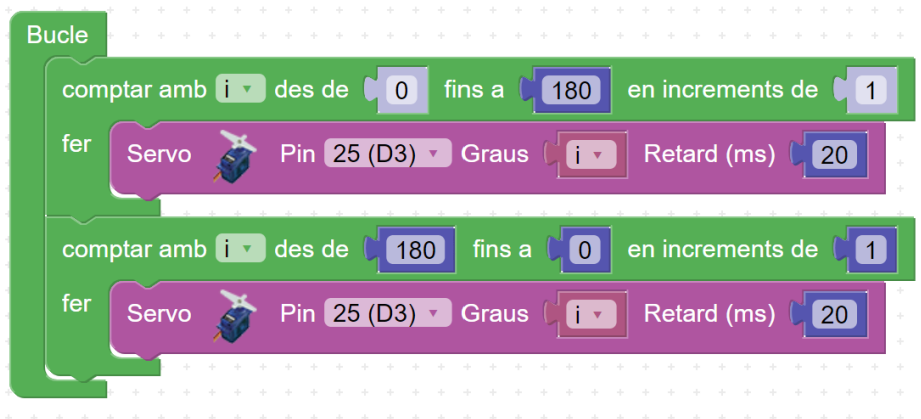
- 1 Placa ESP32
- 1 servomotor

MUNTATGE



PROGRAMACIÓ

Generar un moviment continu del servo des de la posició 0° fins a 180° i tornar de la mateixa manera a la posició 0°



REPTES AMPLIACIÓ

Moure el servomotor 10° cada vegada que premem el polsador.

- Si el motor arriba a 180°, cada vegada que premem el polsador disminuïrem 10 graus la posició.
- Si el motor arriba a 0°, cada vegada que premem el polsador augmentarem 10 graus la posició.

P11. Tira de leds NeoPixel

OBJECTIUS

- Engegar progressivament els leds de la tira
- Crear l'efecte del cotxe fantàstic
- Programació: llistes, bucles i comptadors.

SABERS

Els NeoPixel són leds RGB intel·ligents que incorporen el seu propi microcontrolador dins de cada LED. Normalment, es troben en forma de tira de leds connectats un darrere l'altre, podent haver tires de desenes o centenars de leds.

Un LED NeoPixel té 4 pins de contacte:

VCC: alimentació a 3,3 V o 5 V

GND: 0 V

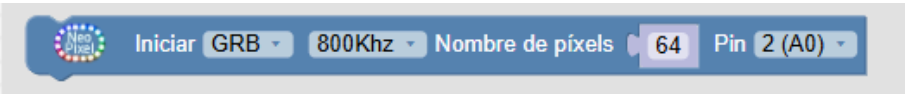
DIN (DI): Entrada d'informació. Connectat a un pin de sortida digital.

DOUT (DO): Sortida d'informació. Connectat al següent LED o tira de leds.

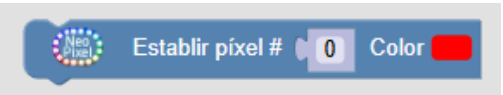
Els blocs de control es troben a l'apartat de "Visualització" → NeoPixel



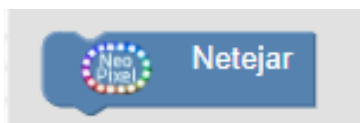
Per inicialitzar la tira, disposem del bloc on es defineix el nombre de LEDs (píxels) i el pin on la connectem:



Definim el color d'un píxel indicant el número de LED segons la seva posició. De 0 en endavant:



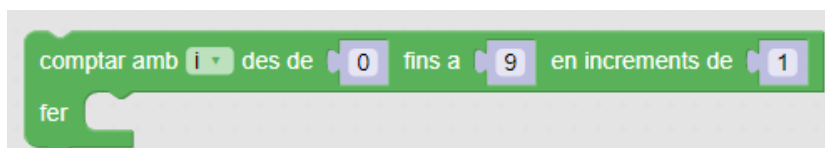
Per apagar tots els píxels que estiguin connectats:




Cal actualitzar les dades enviades als NeoPixel, ja que les operacions no es reflecteixen fins que s'executa el bloc "mostrar".



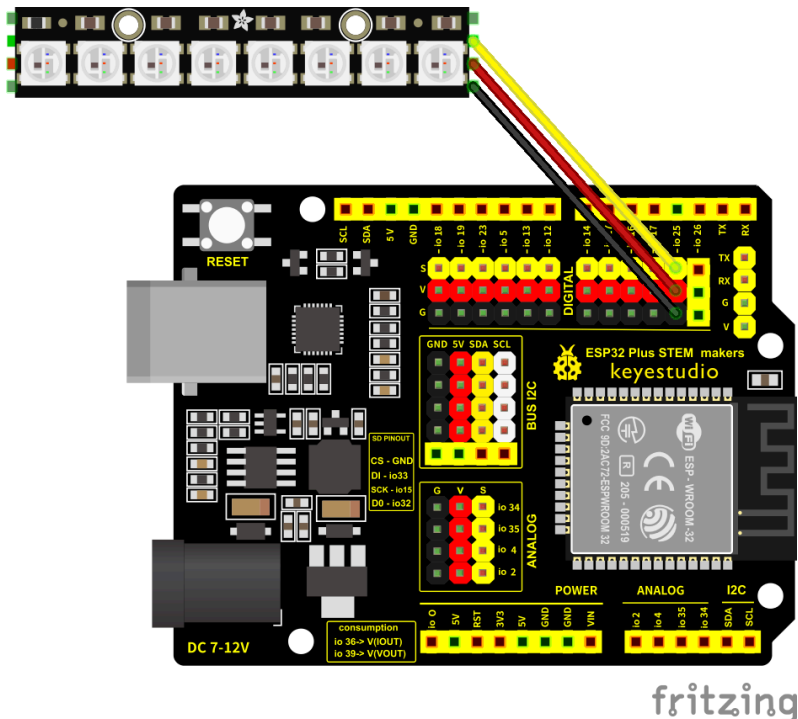
- Comptar: Realitza un bucle comptant amb una variable índex "i". Es defineix un valor d'inici, un valor de fi i els increments que es realitzaran en cada iteració del bucle. Dins del bucle podem usar aquesta variable "i", que arduinoblocks crea automàticament.



LLISTA D'ELEMENTS

- 1 Placa ESP32 STEAMakers
-  Tira de leds NeoPixel

MUNTATGE



PROGRAMACIÓ 11.1

Provem d'engegar els leds de la tira NeoPixel de manera seqüencial fent servir un comptador.

Inicialitzar

- NeoPixel Iniciar GRB 800Khz Nombre de píxels 8 Pin 27 (D6)
- NeoPixel Netejar
- NeoPixel Mostrar

Bucle


- comptar amb i des de 0 fins a 7 en increments de 1
- fer
 - NeoPixel Establir píxel # i Color
 - NeoPixel Mostrar
 - Esperar 500 mil·lsegons


ROBOLOT 2024-2025 Manual de pràctiques - ESP32 STEAMakers


PROGRAMACIÓ 11.2

Simular la tira de llumetes del cotxe fantàstic.


Inicialitzar










Bucle





fer



















fer







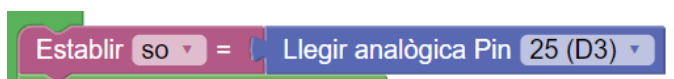






REPTES AMPLIACIÓ

- Engegar els leds, començant pel central i anar cap als extrems
- Semàfor de soroll: Fer que la tira de leds s'engegui d'un color o un altre, en funció del soroll. Utilitzarem el sensor de so, que cal connectar-lo a una entrada analògica, per exemple el IO25/D3, i funciona amb el bloc de llegir l'entrada analògica.



P12. Giroscopi i acceleròmetre



OBJECTIUS

- Representació gràfica de l'angle de gir.
- Control dels leds d'una tira en funció de l'angle d'inclinació del dispositiu.
Segons la inclinació del panell que s'engeguin els leds de la tira.
- Programació:
 - Funcions
 - Mapejar

SABERS

- Un giroscopi és un element que mesura els girs que es fan en cadascun dels eixos.

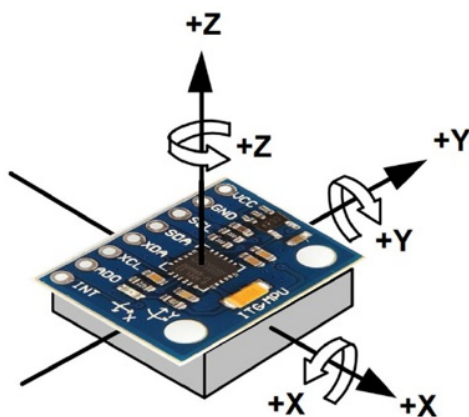


Fig.01 Funcionament d'un giroscopi

- Acceleròmetre: és un sensor que mesura l'acceleració. Aquesta acceleració pot ser la causada pel moviment o la gravetat. En altres paraules, un acceleròmetre pot dir-te com de ràpid alguna cosa està canviant la seva velocitat o direcció, així com l'orientació de l'objecte respecte a la Terra.

- **Funció:**

Una funció és un bloc de codi que executa una tasca específica i que pot ser reutilitzat diverses vegades dins del programa. Les funcions ajuden a organitzar el codi en parts més petites i manejables, permetent així una programació més clara i eficient.



Canviem el nom de la funció. I ens apareixerà sota aquests dos blocs la funció que acabem de declarar.

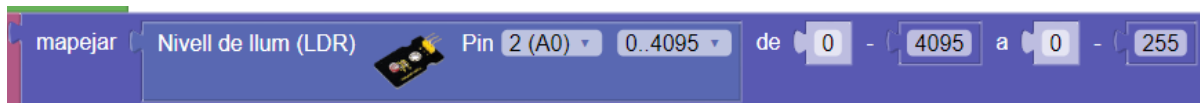
- **Bloc mapejar:**

El bloc mapejar converteix un valor d'un rang a un altre rang. Això és útil per ajustar els valors de sensors i actuadors perquè treballin correctament junts.



Paràmetres del Bloc:

- Valor: El valor que vols convertir.
- De Min: El valor mínim del rang original.
- De Max: El valor màxim del rang original.
- A Min: El valor mínim del nou rang.
- A Max: El valor màxim del nou rang.

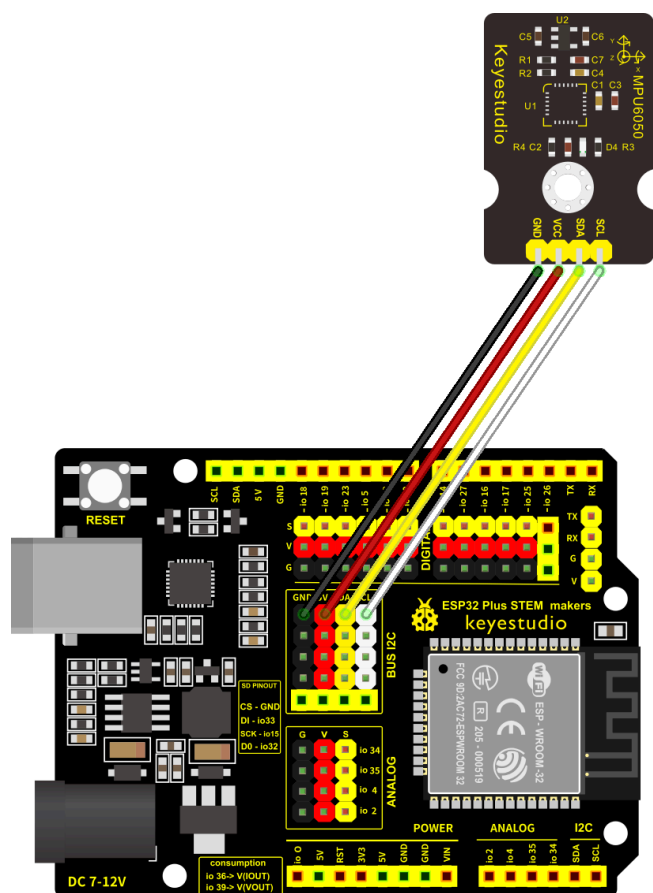
Exemple: Si un sensor llegeix valors entre 0 i 4095 i vols controlar un LED que accepta valors entre 0 i 255, pots fer servir el bloc mapejar per convertir els valors.



LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Giroscopi i acceleròmetre
-  Tira de 30 leds (NeoPixel)

MUNTATGE



fritzing

PROGRAMACIÓ 12.1

Aquesta primera pràctica ens permet entendre el funcionament del giroscopi. Observem que quan girem la placa aquesta canvia i genera un pic positiu o negatiu depenent de l'angle de gir de la placa, i ràpidament, s'estabilitza. És molt important que el temps d'espera sigui petit per poder captar bé el senyal.

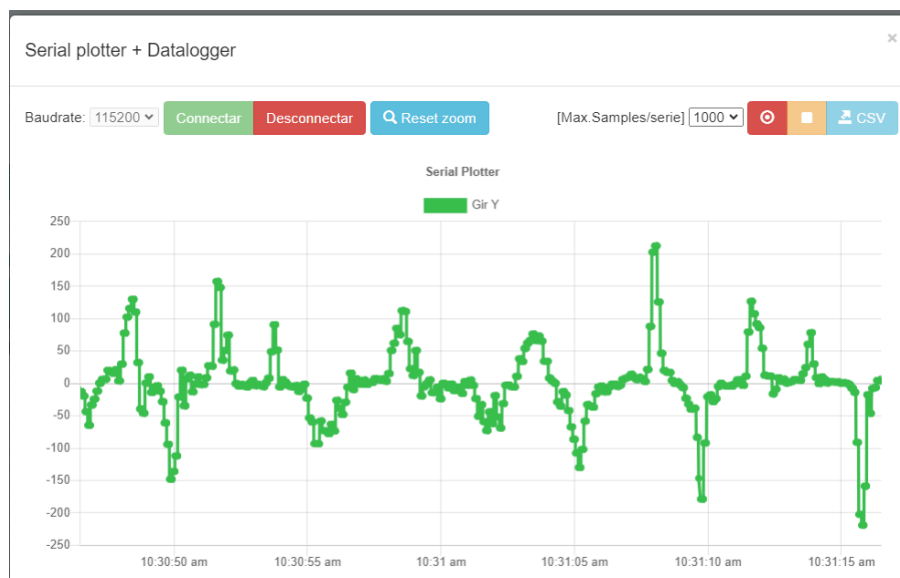
Inicialitzar

> Iniciar Bauds 115200

Bucle

> Plotter "Gir Y" Valor Accel+Gyro (MPU6050) Giro-Y (deg/s)

Esperar 100 mil·lisegons



PROGRAMACIÓ 12.2

Aquest programa introdueix les funcions i el mapejat. Obtenim del giroscopi el valor de la inclinació en l'eix Y. Per verificar el seu funcionament hem fet servir el port sèrie que ens permet veure les dades que està captant, per fer-nos idea dels valors. A conseqüència dels seus valors observem que fora bo fer un mapejat per obtenir uns valors més adients per posar-los a la tira de NeoPixels. El marge de treball és de -90

a +90 graus en l'entrenador, com disposem d'una tira de 30 NeoPixels, ho escalem a aquesta mesura. Si disposeu d'una tira de NeoPixels sense tallar, poseu sobre el 14 o 15 el centre per provar la seva estabilitat.


Inicialitzar

- Iniciar NeoPixel: GRB, 800Khz, Nombre de píxels: 30, Pin: 25 (D3)
- Iniciar Bauds: 115200
- Establir LED = 14
- encendre LED

Bucle

- Establir Inclincio Y = Accel+Gyro (MPU6050) Giro-Y (deg/s)
- Enviar Inclincio Y, Salt de línia
- Establir LED = mapejar Inclincio Y de -90 - 90 a 0 - 29
- encendre LED
- Esperar 500 mil·lisegons

+ a encendre LED

- Netejar
- Establir píxel # LED Color 
- Mostrar

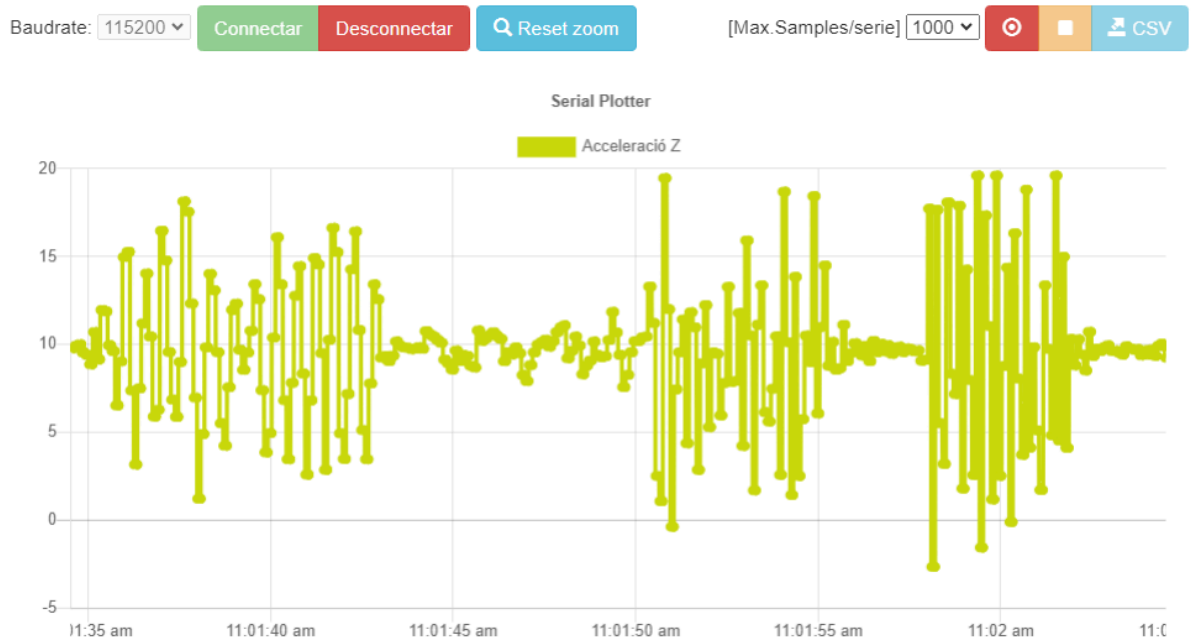
PROGRAMACIÓ 12.3

Per representar gràficament els valors de l'acceleròmetre en l'eix Z (amunt i avall) fem servir la consola del port sèrie.

The image shows a Scratch script with the following blocks:

- Inicialitzar:** A block labeled "Iniciar Bauds" with a dropdown menu set to "115200".
- Bucle:** A loop containing:
 - A "Plotter" block with the text "Acceleració Z" and a "Valor" block set to "Accel+Gyro (MPU6050)". A dropdown menu next to it is set to "Accel-Z (m/s²)".
 - An "Esperar" block set to "100 mil·lisegons".

Serial plotter + Datalogger



REPTES AMPLIACIÓ

- Detectar salts: per fer una carrera de salts en un temps determinat, cada salt permet fer servir la tira de leds per visualitzar el total de salts.
- Podòmetre senzill que compti els passos
- Detector de caiguda
- Control motor: Fer servir el giroscopi per controlar un motor, que podria servir per controlar la direcció (volant) d'un cotxe.
- Detector de moviment d'un objecte o d'una persona.

P13. Bomba d'aigua



OBJECTIUS

- Connectar la bomba d'aigua depenent de la humitat del terra.

SABERS

La bomba d'aigua és un motor estanc de 3.3V que normalment es combina amb el relé que l'obre o el tanca segons les condicions donades.



Un relé és un aparell o dispositiu destinat a controlar un circuit determinat d'una certa potència (generalment tancar o obrir el circuit) mitjançant un circuit auxiliar de petita potència i, per tant, amb un corrent molt feble.



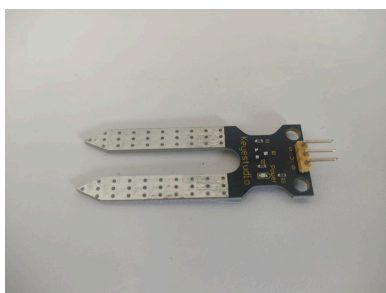
El circuit té tres pins NO (normally open), COM (common contact) i NC (normally closed):

1. NO: aquest pin està normalment obert, i es connecta a la placa ESP32 al pin de corrent de 3.3V
2. COM: aquest pin és el que es connecta amb els altres mòduls que volem controlar, en el nostre cas, al vermell de la bomba d'aigua.
3. NC: aquest pin normalment està tancat i és el que utilitzarem amb l'ESP32 per tancar el circuit.



El circuit auxiliar és el que connectem a la placa ESP32 per transmetre l'obertura/tancament del circuit.

El sensor d'humitat s'utilitza per mesurar el contingut d'aigua al sòl volumètric, controlar la humitat del sòl, regar els cultius i protegir els boscos. És un sensor de tipus analògic.



LLISTA D'ELEMENTS

- 1 Placa ESP32
- 1 cable mascle-mascle per la connexió + del relé.
- ■ Relé (relay Module)
- ■ Bomba d'aigua
- ■ Sensor d'humitat del terra

REPTES AMPLIACIÓ

Sistema de Reg amb Previsions Meteorològiques

- Sistema de Reg Temporitzat: Activar la bomba d'aigua a intervals regulars programats, independentment de les lectures dels sensors. Per exemple, la bomba s'activa durant 10 minuts cada 2 hores.

P14. Condicions ambientals









OBJECTIUS

- Conèixer els elements que poden conformar una estació ambiental, per recollir dades: pressió atm, altitud, qualitat de l'aire, llum, temperatura, i humitat.
- Enviar les dades a través de la consola sèrie.
- Gravar dades en una targeta MicroSD.
- Recuperar dades d'una targeta MicroSD.

SABERS

Reconèixer el funcionament dels diferents sensors.

LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Sensor de pressió baromètrica BMP280
-  Sensor de temperatura i humitat DHT-11
-  Sensor de temperatura LM35
-  Sensor de llum TEMT 6000
-  Sensor de CO₂ CCSS811B
-  Targeta MicroSD

PROGRAMACIÓ 14.1

Per llegir els valors dels sensors enviem les seves dades a la consola.

Inicialitzar

Iniciar Bauds 115200

Bucle

Enviar crear text amb " DHT11 --> Temperatura: " Salt de línia

DHT-11 Temperatura °C Pin 25 (D3)

" Humitat: "

DHT-11 Humitat % Pin 25 (D3)

Enviar crear text amb " BMP280 --> Pressió atmosfèrica: " Salt de línia

Baromètric BMP280 Pressió (mb)

" Altitud: "

Baromètric BMP280 Altitud (m)

" Temperatura: "

Baromètric BMP280 Temperatura °C

Enviar crear text amb " 811 --> Qualitat de l'aire: " Salt de línia

Sensor CO2/TVOC (CCS811) CO2 (ppm)

Enviar crear text amb " Sensor de llum --> Nivell de llum (0-100): " Salt de línia

Nivell de llum (TEMT6000) Pin 34 (A3) *WiFi/BT %

Enviar crear text amb " LM35 --> Temperatura: " Salt de línia

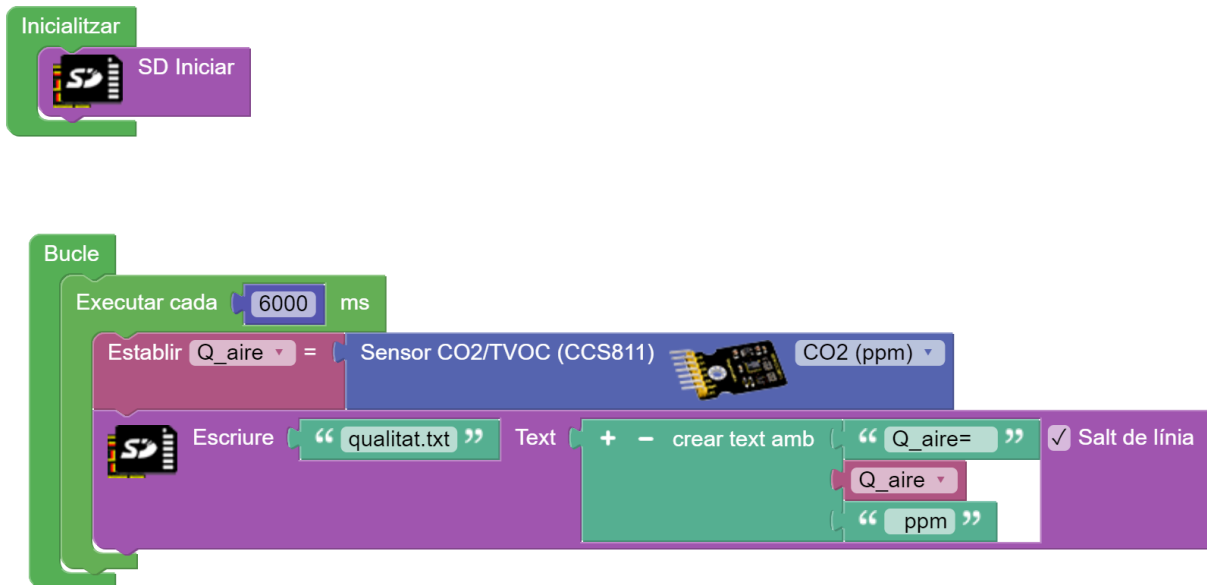
Temperatura °C (LM35) Pin 13 (D9)

Enviar " " Salt de línia

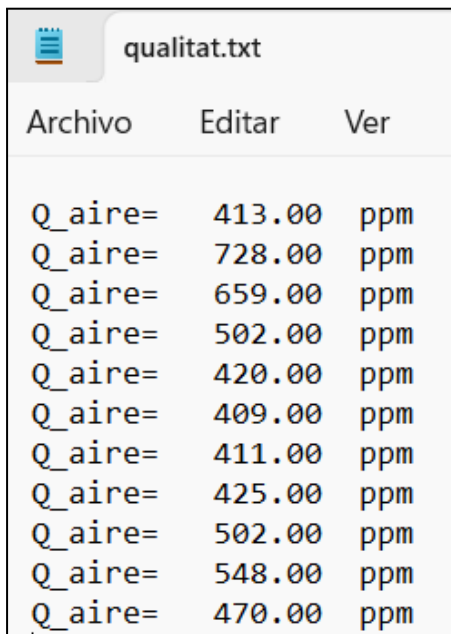
Esperar 5000 mil·lisegons

PROGRAMACIÓ 14.2

Gravar els resultats a una targeta SD.

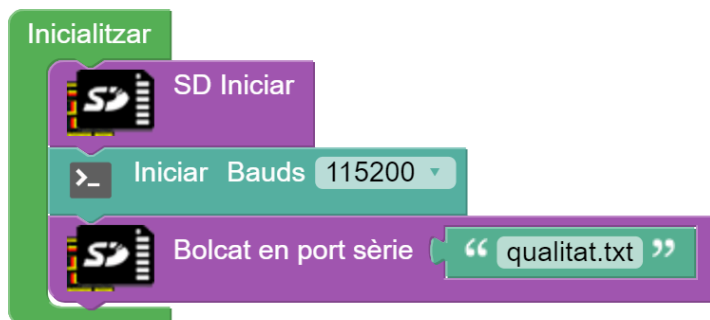


A continuació hem tret la targeta de la placa i hem llegit el valor en l'ordinador:



PROGRAMACIÓ 14.3

Si volem bolcar les dades al port sèrie i captar-les mitjançant la consola.



REPTES AMPLIACIÓ

- Seguiment solar: Ajusta la posició d'un panell solar per maximitzar la llum solar rebuda durant el dia.
- Monitoratge de soroll ambiental: Registra els nivells de soroll en diferents períodes del dia.
- Reg automàtic: Activa un sistema de reg quan es detecta que el sòl està sec.
- Monitoratge de plantes: Envia notificacions quan les plantes necessiten ser regades.
- Robot autònom avançat que detecti focs: Que utilitzi sensors de distància i temperatura per navegar i reaccionar de manera intel·ligent en diferents situacions.
- Detector de febre.


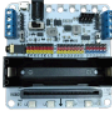
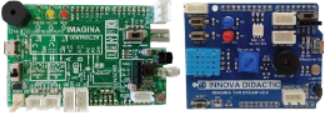
P15. Motors CC



OBJECTIUS

- Controlar el funcionament d'1 o 2 motors de corrent continu.

SABERS

	<p>Motor amb roda</p> <p>Motor amb reductora, encapsulat amb protecció, amb cable polaritzat i roda adaptable.</p> <p>Voltatge 3.0V a 12.0 V DC</p> <p>Un eix (reductora 1:120, 90 rpm), amb placa.</p> <p>A 6V gira aprox. a 55 rpm.</p> <p>Diàmetre de l'eix: 5.2 mm.</p> <p>Roda: diàmetre de 68 mm i amplada de 25 mm.</p>	<p>Actuador</p> <p>Compatible amb:</p>  <p>Compatible amb:</p> 
	<p>Voltatge</p> <p>3-6 V</p>	<p>Tipus de senyal</p> <p>Digital</p>
	<p>Tipus de connexió</p> <p>Clavilla JST. Per a la micro:shield, s'ha de tallar la clavilla o afegir-hi cables de connexió mascle-masclle.</p>	

Els motors de corrent continu tenen les següents característiques:

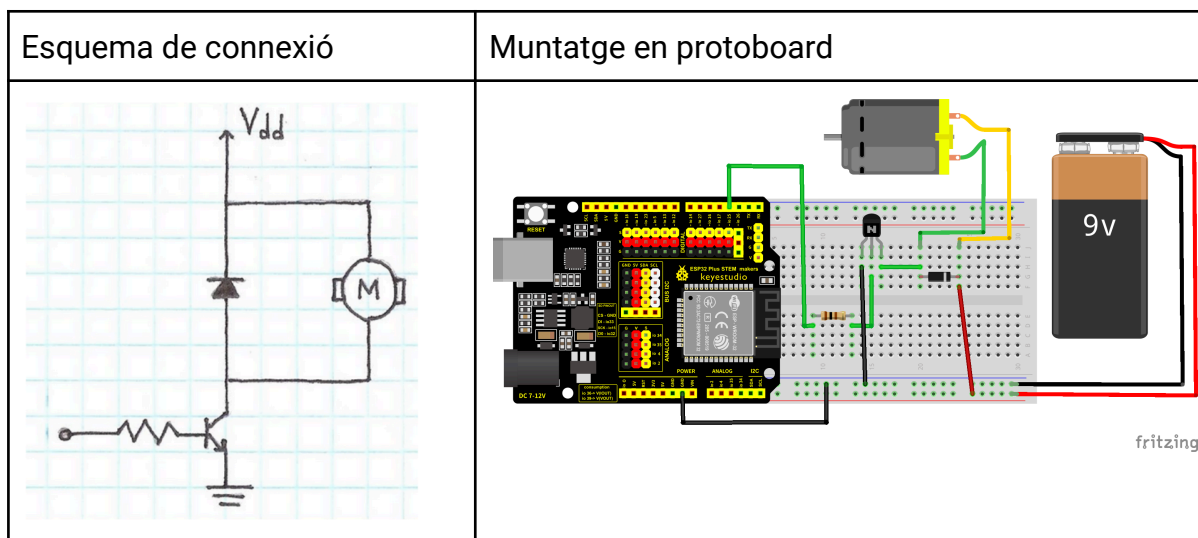
Les sortides de la placa STEAMakers no proporcionen suficient corrent per a controlar un motor de corrent continu (proporciona uns 50 mA i un motor pot consumir uns 1000 mA) per la qual cosa necessitarem realitzar un petit circuit bé amb un transistor, bé amb un driver per a controlar un corrent molt més gran.

LLISTA D'ELEMENTS

- 1 Placa ESP32
- ■ 2 Motors de CC amb rodes
- ■ Placa protoboard amb transistor NPN + pila 9V
- ■ controlador de motors L298N/L293D + pila 9V
- ■ Shield IMAGINA Arduino

Muntatge amb transistor NPN.

Podem utilitzar un transistor NPN en mode tall/saturació que permeti, com si fos un interruptor, el pas d'una intensitat de corrent molt superior des d'una font d'alimentació auxiliar.



PROGRAMACIÓ 15.1

Activar el gir del motor:



Activar gir controlant velocitat:



Augment progressiu de la velocitat:



Muntatge amb controlador de motors L298N/L293D.

Per tal de poder controlar els motors CC podem utilitzar també el controlador L298N per la seva facilitat d'utilització. El controlador L298N pot treballar amb tensions de fins a 35V i una intensitat de fins a 2A per cada canal, és a dir, pot controlar fins a 4A en total i uns 25W.

Aquest mòdul té una altra característica molt important i és que ens permet controlar la direcció de gir dels motors, és a dir, podem fer que girin en tots dos sentits.

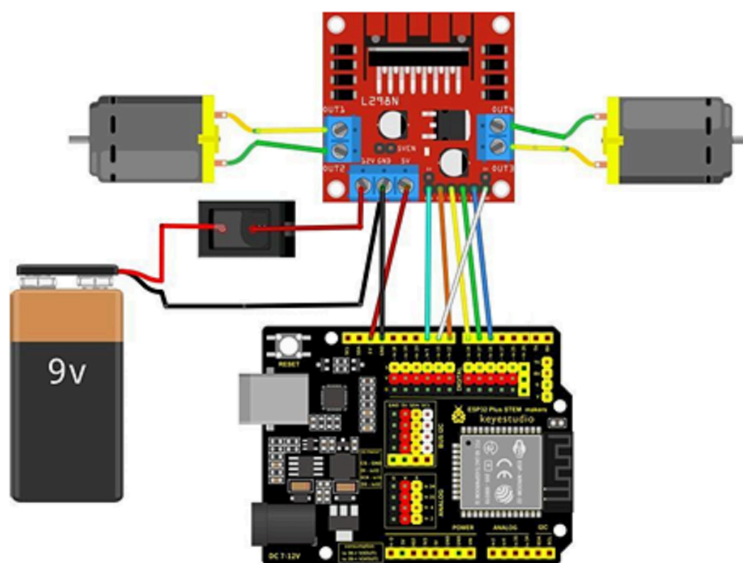
A més a més, ens permet obtenir una tensió del mòdul de 5V, perfecta per alimentar la nostra placa ESP32 sense utilitzar cap altra font addicional. Això sí, aquest regulador només funciona si alimentem el mòdul amb una tensió màxima de 12V.

<i>Esquema de un puente en H para controlar la dirección de giro de un motor</i>	<i>Módulo típico con configuración en puente H para control de motores de C.C.</i>

L'entrada Vin admet tensions entre 3V i 35V, i just a la seva dreta a la imatge tenim el pin que hem de connectar a GND. La tercera connexió d'aquest grup és V logic que subministra 5V si el jumper del regulador està tancat.

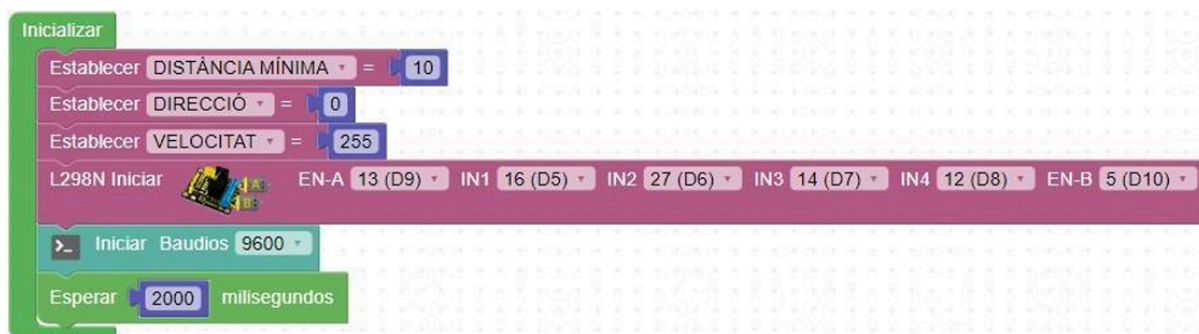
Es pins IN1 fins a IN4 ens serveixen per controlar el sentit de gir dels motors (1 i 2 per al motor de la sortida A i 3 i 4 per al motor de la sortida B). Funcionen de manera que si IN1 està HIGH i IN2 està LOW, el motor A gira en un sentit i si estan al revés el motor canvia el sentit de gir.

Per a controlar la velocitat de ir dels motors hem de treure els ponts que hi ha als pins ENA i ENB i connectar aquests a dos sortides PWM de la placa. D'aquesta manera, enviem un valor entre 0 i 255 que controlarà la velocitat de gir dels motors. Si tenim els ponts muntats, girarà sempre a la mateixa velocitat.



PROGRAMACIÓ 15.2

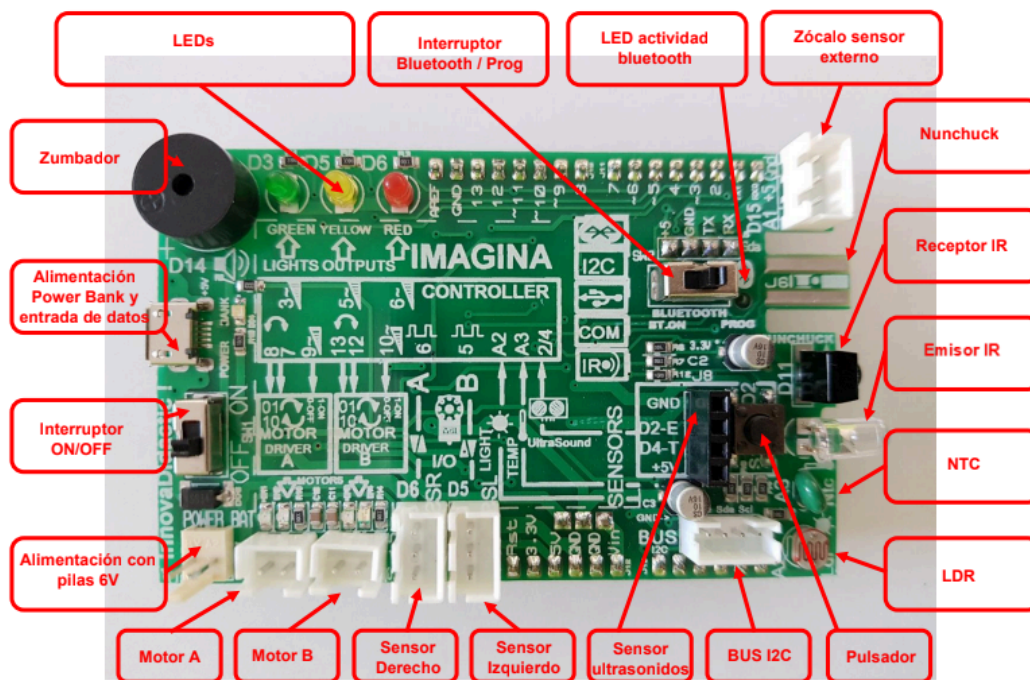
En Inicialitzar, iniciem el controlador L298N indicant els pins de la placa ESP32 als quals està connectat, així com la consola interna de la nostra placa. La consola interna la iniciem per tal de poder veure alguna lectura del sensor d'ultrasons i poder ajustar la distància a la qual el nostre cotxe haurà de girar. També es poden crear tres variables que després ens seran útils durant el programa.



Es poden crear també les quatre funcions principals del nostre programa per tal d'evitar-nos escriure cada cop totes les ordres. Aquestes funcions són quatre: avançar, parar, girar a l'esquerra i girar a la dreta. Aquestes funcions es munten a partir d'instruccions del controlador L298N. Podem triar quin dels dos motors estem comandant, si volem que giri avant o enrere i a quina velocitat volem que giri.

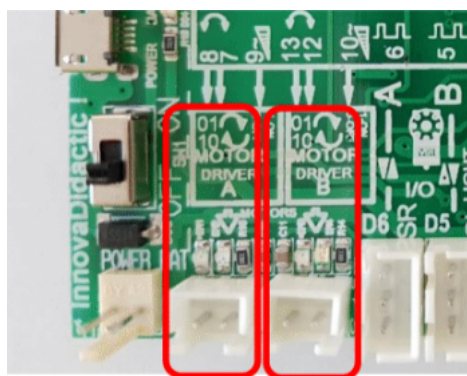


Muntatge amb el shield Imagina Arduino.



Ubicación de componentes

La placa Imagina Arduino ja porta incorporat un " driver" o controlador de motors de CC, amb ell podem realitzar el control de dos motors de corrent continu (CC) de fins a 2A (Ampers).

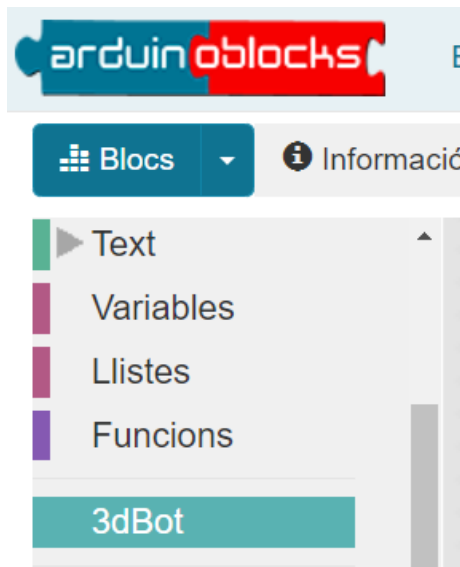


Motor A Motor B

Utilitzarem el shield Imagina Arduino connectat a la placa ESP32 STEAMakers. Aquest shield disposa, entre altres connexions, de 2 connectors JST.

Per ampliar informació sobre aquesta placa shield podeu consultar el següent [enllaç](#).

PROGRAMACIÓ 15.3



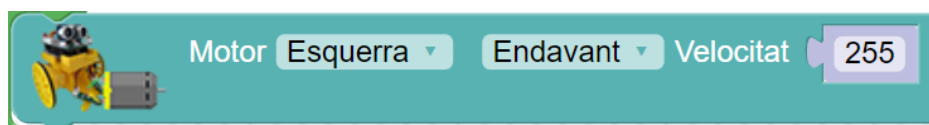
Creem un nou projecte del tipus ESP32 STEAMakers + Imagina 3DBot

Al submenú 3dBot d'arduinoblocks podem observar que existeixen diferents instruccions específiques per a controlar el motor A (o esquerre) on podem escollir quina acció farà si girar endavant o endarrere.

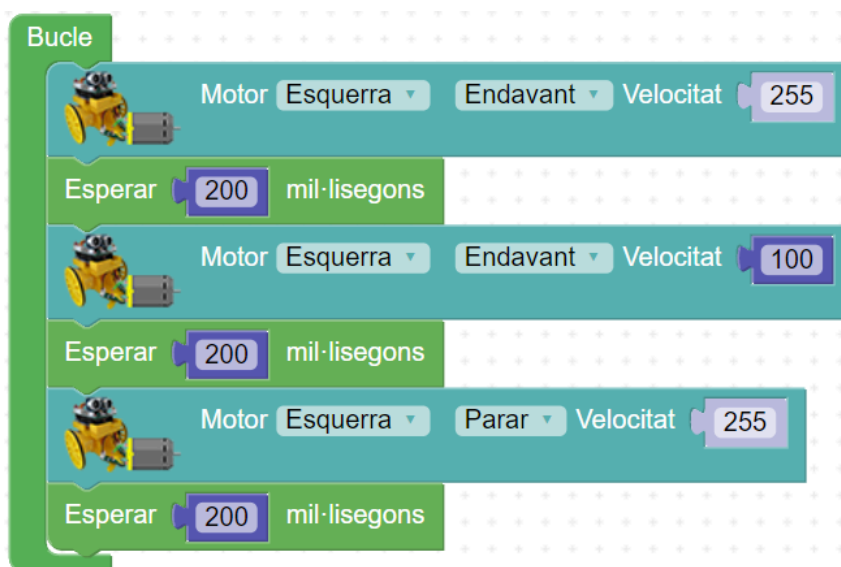
Canviant la selecció del motor, també serveix per a controlar les mateixes accions per al motor B (o dret).

En total podem fer funcionar qualsevol dels dos motors, en qualsevol dels sentits, parar i fins i tot podem ajustar la velocitat de rotació.

A "Velocitat", valors molt petits no són capaços de posar en marxa el motor.



Si introduïm aquest programa observarem que el motor A va cap endavant a una velocitat ràpida durant 0.2 segons, després a una velocitat més lenta uns altres 0.2 segons i finalment estarà parat 0.2 segons i repetirà el procés.





REPTES D'AMPLIACIÓ

- **Fer que camini el multientrenador:** Muntar els motors als laterals del multientrenador i programar-lo per a que avanci com un robot
- **Relloige:** crear un rellotge amb els dos motors, un que faci d'agulla minutera i l'altre de les hores.

P16. Infrarojos + Comandament



OBJECTIUS

- Engegar i parar LEDs amb el comandament
 - 1 → LED GROC
 - 2 → LED VERMELL
 - 3 → LED VERD
 - 4 → LED BLANC
- Comandar motors amb l'infraroig

SABERS

Una gran part dels electrodomèstics utilitzen comandaments a distància d'infrarojos, com els televisors o els equips musicals. El sensor infraroig és un dispositiu optoelectrònic capaç de mesurar la radiació electromagnètica infraroja dels cossos en el seu camp de visió. Tots els cossos emeten una certa quantitat de radiació, aquesta resulta invisible per als nostres ulls, però no per a aquests aparells electrònics, ja que es troben en el rang de l'espectre just per sota de la llum visible.

En el cas del receptor d'infrarojos (IR) permet codificar els protocols de senyals de polsos infrarojos utilitzats pels comandaments a distància.

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

El senyal infraroig transmet el codi corresponent al botó premut del comandament a distància i el transmet al dispositiu en forma d'una sèrie d'impulsos de llum infraroja. El receptor IR els rep i els passa a un processador que descodifica la sèrie de 0 i 1 en una dada de text, que processarem en el nostre programa.




En el menú sensors → Receptor IR, trobem els blocs que permeten llegir la dada que ens arriba amb el senyal del comandament. S'ha de guardar en una variable de text.



Per comprovar el codi que s'està rebent de cada tecla del comandament, podem enviar aquesta dada a la consola.

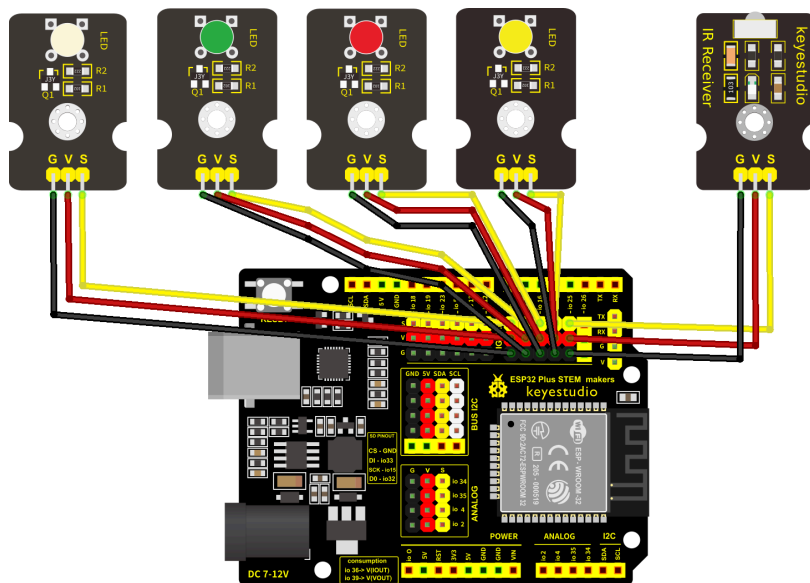
Un cop sabem el codi rebut, podem programar les diferents funcionalitats.

LLISTA D'ELEMENTS

- 1 Placa ESP32
-  Comandament a distància
-  Receptor de IR
-  Pila botó CR2025 / 2032

MUNTATGE

LED	ENTRADA ESP32 STEAMakers
SENSOR IR	D3/IO23
GROC	D4/IO17
VERMELL	D5/IO16
VERD	D6/IO27
BLANC	D7/IO14



fritzing

PROGRAMACIÓ

Inicialitzar

Iniciar Bauds 115200

Bucle

Establir **codi** = Receptor de IR (Text HEX) Pin 25 (D3)

+ si **codi** diferent de " "

fer >_ Enviar **codi** Salt de línia

+ si **codi** igual a " 00FF6897 "

fer Led Pin 17 (D4) Estat ON

Esperar 1000 mil·lisegons

+ si **codi** igual a " 00FF9867 "

fer Led Pin 16 (D5) Estat ON

Esperar 1000 mil·lisegons

+ si **codi** igual a " 00FFB04F "

fer Led Pin 27 (D6) Estat ON

Esperar 1000 mil·lisegons

+ si **codi** igual a " 00FF30CF "

fer Led Pin 14 (D7) Estat ON

Esperar 1000 mil·lisegons

+ si **codi** igual a " "

fer Led Pin 17 (D4) Estat OFF

Led Pin 16 (D5) Estat OFF

Led Pin 27 (D6) Estat OFF

Led Pin 14 (D7) Estat OFF

REPTES AMPLIACIÓ

- **Comandar motors amb IR**

Podem simular que els motors de corrent continu que tenim són els d'un robot. Podem dissenyar funcions de moviment del robot, que s'activen amb les tecles del comandament.

Connectem els motors amb la shield Imagina Arduino i obrim un projecte de l'arduinoblocks, tipus: ESP32 STEAMakers + Imagina 3DBot.

El comandament a la placa Imagina Arduino 3DBot, està connectat a l'entrada D11/I023 de la placa, i el projecte ja disposa dels blocs del 3DBot, que controlen els moviments dels motors.

- **Desxifrar senyals d'altres comandaments i reutilitzar-los**

Agafem qualsevol comandament de llums LED que tinguem per casa i desxifrem els codis que envia cada botó. Reutilitzem aquest comandament per governar els actuadors del nostre Kit.

P17. Emissor Infrarojos

OBJECTIUS

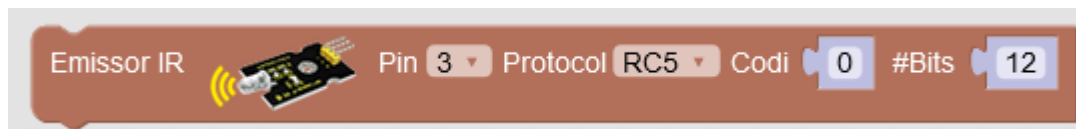
- Copiar un botó del comandament des d'un arduino UNO i que el rebem des d'una STEAMakers, per tal d'engegar un LED.

SABERS

Mitjançant el mòdul emissor IR, podem emetre codis d'infrarojos com un comandament a distància.

Farem un programa per la placa emisora, en aquest cas, la Keyestudio UNO, utilitzant el bloc que està al menú Actuadors.

Cal connectar l'emissor al pin 3 de la placa, escriure el protocol del comandament que volem clonar i el nombre de bits que utilitza.



El comandament utilitza un protocol NEC amb un número decimal i 32 bits. Per tant, nosaltres enviarem un nombre en decimal, que el receptor ens codifica en Hexadecimal.

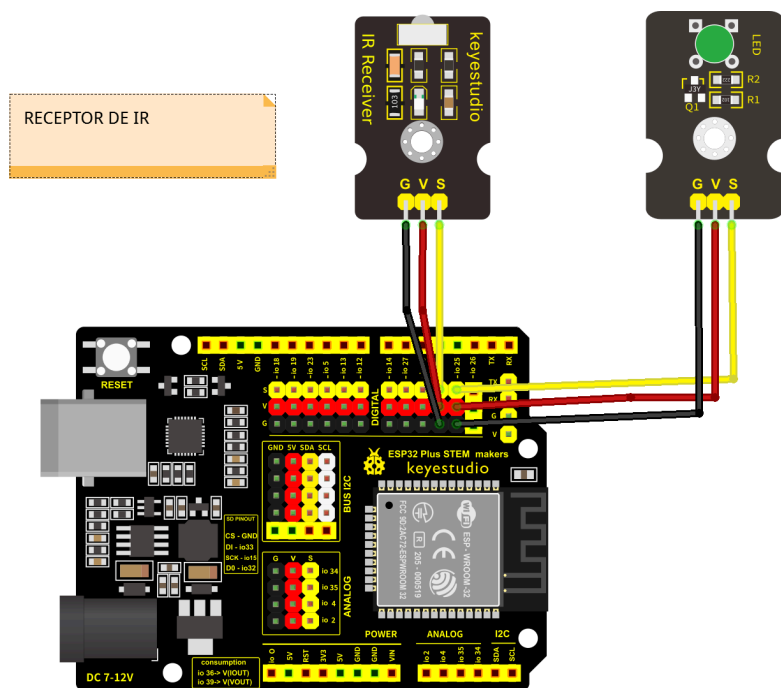
El sistema hexadecimal (abreujat hex) és un sistema numèric amb base 16. Es representa normalment utilitzant els símbols del 0 al 9 i de l'A a la F. Per exemple, el nombre decimal 79, la representació del qual en sistema binari és 01001111, es pot escriure com 4F en hexadecimal (4 = 0100, F = 1111).

Si el codi del comandament en Hexadecimal és 00FF6897, caldrà que enviem el número 16738455 en decimal.

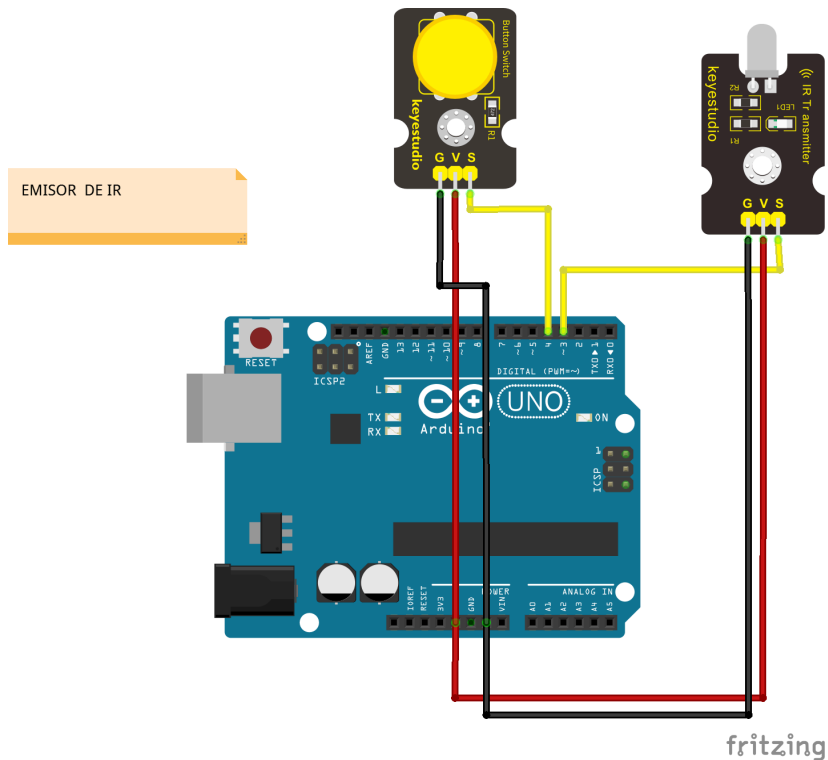
LLISTA D'ELEMENTS

- Placa ESP32 STEAMakers
- Placa Keystudio UNO
- Emissor de IR ■
- Receptor de IR ■
- Polsador ■
- LED ■

MUNTATGE



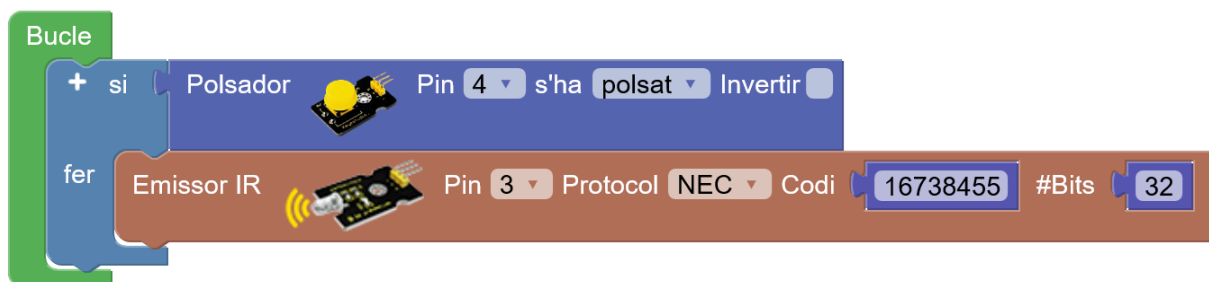
fritzing



Nota: El muntatge de l'emissor es pot fer amb una placa Keystudio UNO

PROGRAMACIÓ

Programació per l'emissor (Keystudio UNO)



Programació pel receptor (ESP32 STEAMakers)

Inicialitzar

>_ Iniciar Bauds 115200

Bucle

Establir codi = Receptor de IR (Text HEX) Pin 25 (D3)

+ si codi diferent de " "

fer >_ Enviar codi Salt de línia

+ si codi igual a "00FF6897"

fer Led Pin 17 (D4) Estat ON

Esperar 1000 mil·lisegons

REPTES AMPLIACIÓ

- **Robot controlat a distància:** Controlar els moviments del 3dBot a través de diferents sensors del panell de control.

P18. Detector de moviment (PIR)

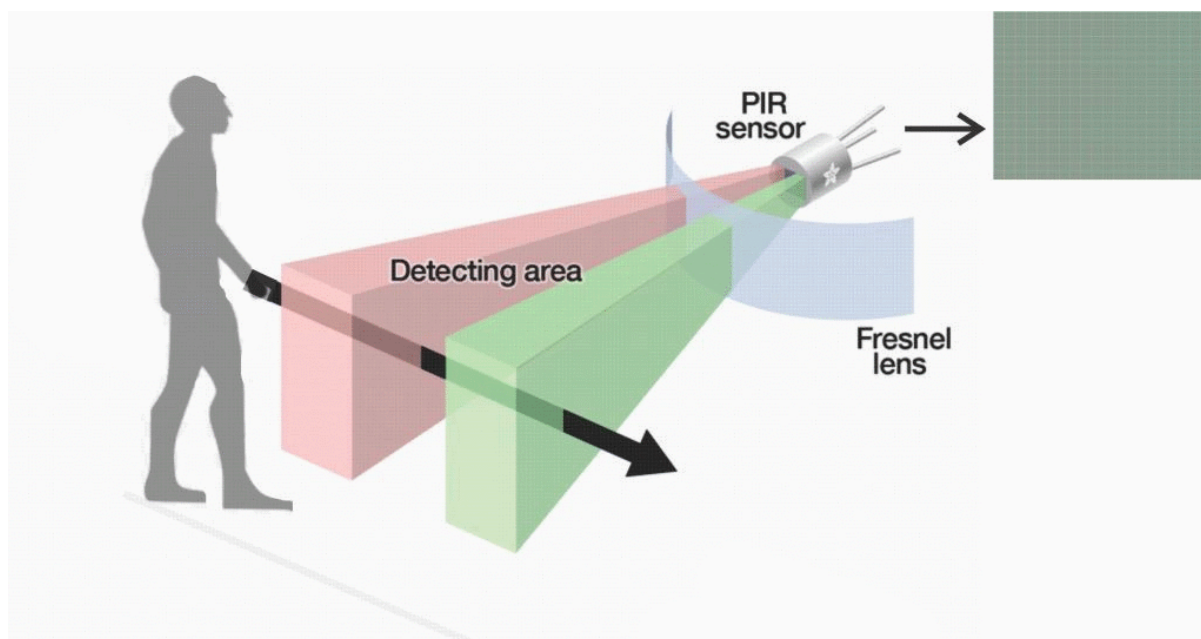


OBJECTIUS

- Engegar i apagar un LED mitjançant el detector de moviment PIR

SABERS

En el següent repte utilitzarem el detector de moviment, sensor d'infrarojos passiu, (PIR sigles en anglès). Aquest sensor funciona detectant les emissions infraroges dels cossos que emeten persones o animals degudes a la calor corporal, a major calor major radiació.



El funcionament intern del dispositiu es basa en un sensor piroelectric capaç de detectar les radiacions infraroges i transformar-les en un senyal elèctric. En realitat cada sensor PIR es divideix en dos camps, i el que es detecta és la diferència entre ells. Si la senyal rebuda pels dos camps és diferent es produeix corrent elèctric i si és igual, no.




L'altra part del conjunt és l'òptica, necessària per a que tot funcioni, es tracta d'una cúpula de polietilè, coneguda com a lens de fresnel, que concentra la llum en un punt central o focus per facilitar la lectura per part del sensor.

S'ha de tenir en compte que el rang de detecció i el temps de resposta del [detector de moviment PIR de la dotació](#) està predeterminat i no es pot modular com en altres sensors PIR més potents.

El temps de resposta és d'uns 3 segons, el que vol dir que pot endarrerir l'execució de programes complexos.

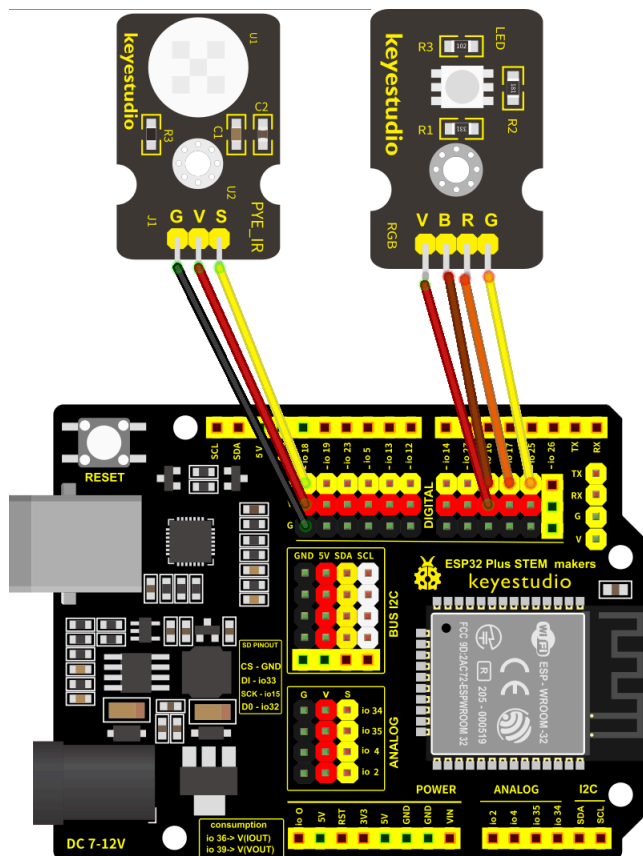


LLISTA D'ELEMENTS

- Placa ESP32
- Detector de moviments PIR 
- LED 
- 1 cables Dupont RBL NN 
- 4 cables Dupont senzills

MUNTATGE

Connectem el detector de moviment PIR al pin IO18 (D13) i el LED RGB l'hem de connectar de manera diferent, en cable que surt del positiu V al vermell del D5, i les senyals de cada color així: Blau al pin groc del D5, Red al pin groc del D4, Green al pin groc del D3

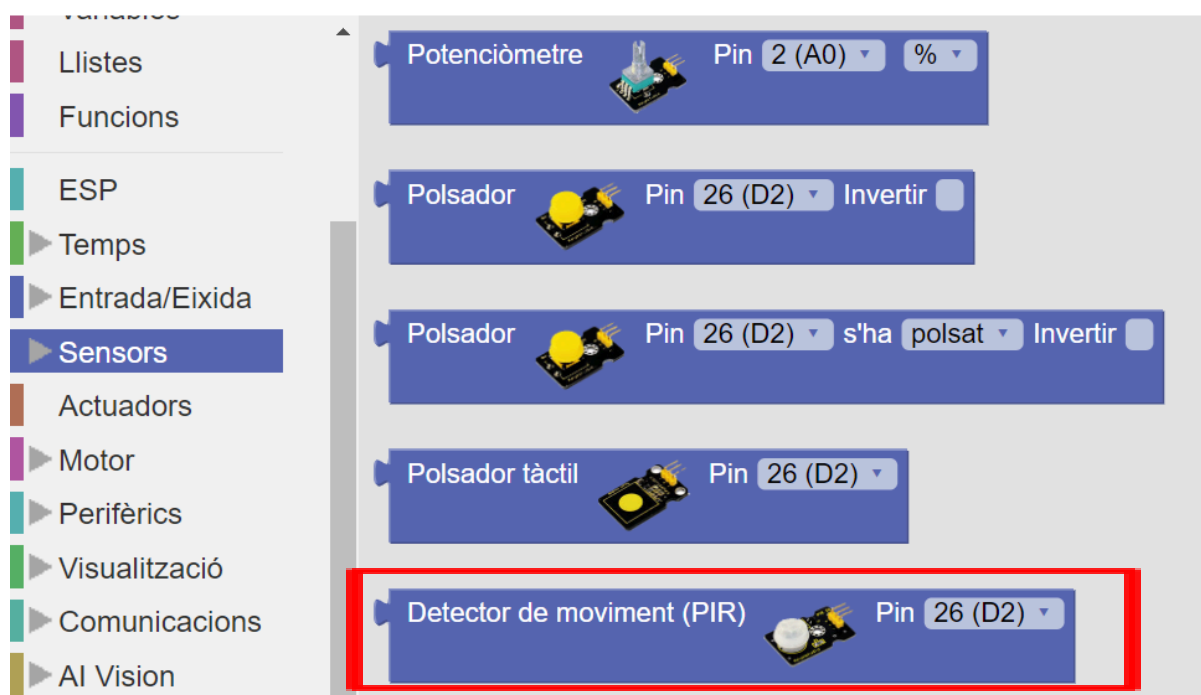


fritzing

PROGRAMACIÓ

Ara realitzarem un programa en el qual en passar per davant del detector de moviment s'encengui el LED RGB amb color vermell i quan deixi de detectar es quedi en verd.

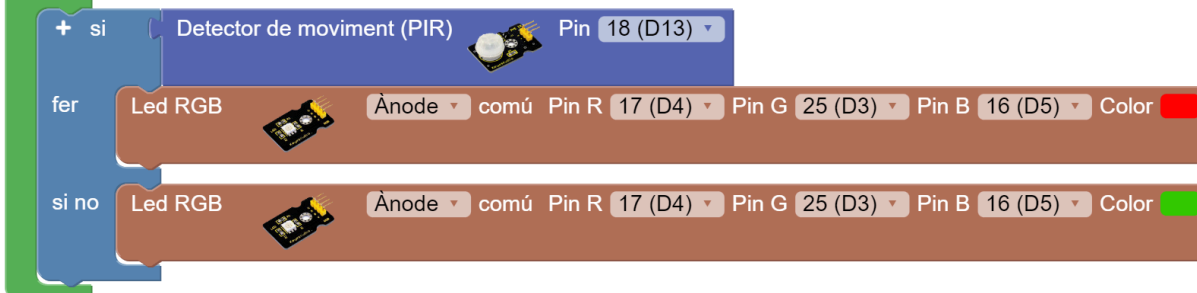
Al menú d'arduinoblocks trobem el bloc del detector de moviment PIR a l'apartat de sensors. Els blocs corresponents al LED RGB ja s'han explicat a pràctiques anteriors.



Simplement, fem un bloc condicional, en el que la condició és que el sensor detecta moviment, l'acció és posar en vermell el LED, i posar-lo en verd si no es compleix la condició.

Inicialitzar

Bucle



REPTES AMPLIACIÓ

- Estalvia energia al llum del passadís: podem fer servir aquest sensor per estalviar energia, si fem que comandi el llum d'un passadís. Normalment el llum estarà tancat i quan el sensor PIR detecti moviment que s'engegui el llum durant 20 segons (temps que una persona triga en travessar el passadís).

P19. Mòdul RFID



OBJECTIUS

- Obertura d'una porta controlada amb RFID: Al acostar el tag al mòdul RFID, el servomotor girarà simulant que s'obre una porta.

SABERS

El lector de targetes està compost per un mòdul de radiofreqüència i un camp magnètic d'alt nivell. El transponedor (el tag o la targeta) és un dispositiu de detecció, que no conté bateria. Només conté petits circuits integrats de xips i mitjans per emmagatzemar dades i antenes per rebre i transmetre senyals.

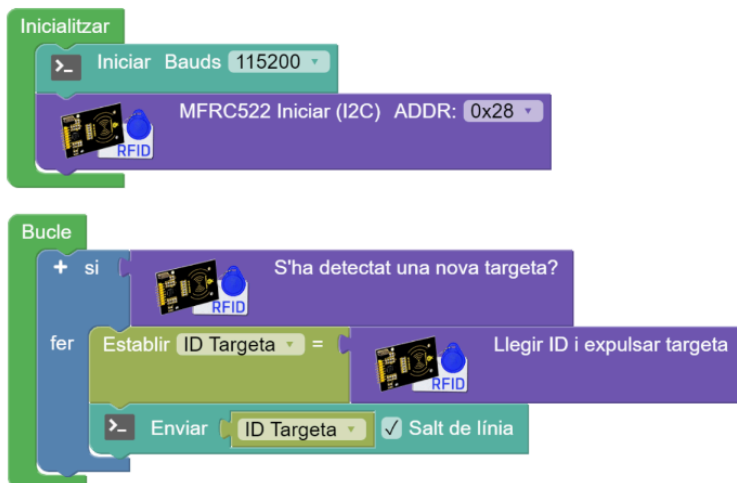
Per llegir les dades de la targeta, cal acostar-la al lector. Aquest generarà un camp magnètic, que produeix electricitat d'acord amb la llei de Lenz, activant així el dispositiu.






Per saber quin codi conté cada targeta o tag, llegirem el codi amb el sensor RFID connectat per I2C. El valor llegit, el guardem a un variable de text i l'enviem al port sèrie, o a la pantalla OLED per poder visualitzar-lo.

Els blocs per controlar-lo es troba al menú perifèrics → RFID (I2C)

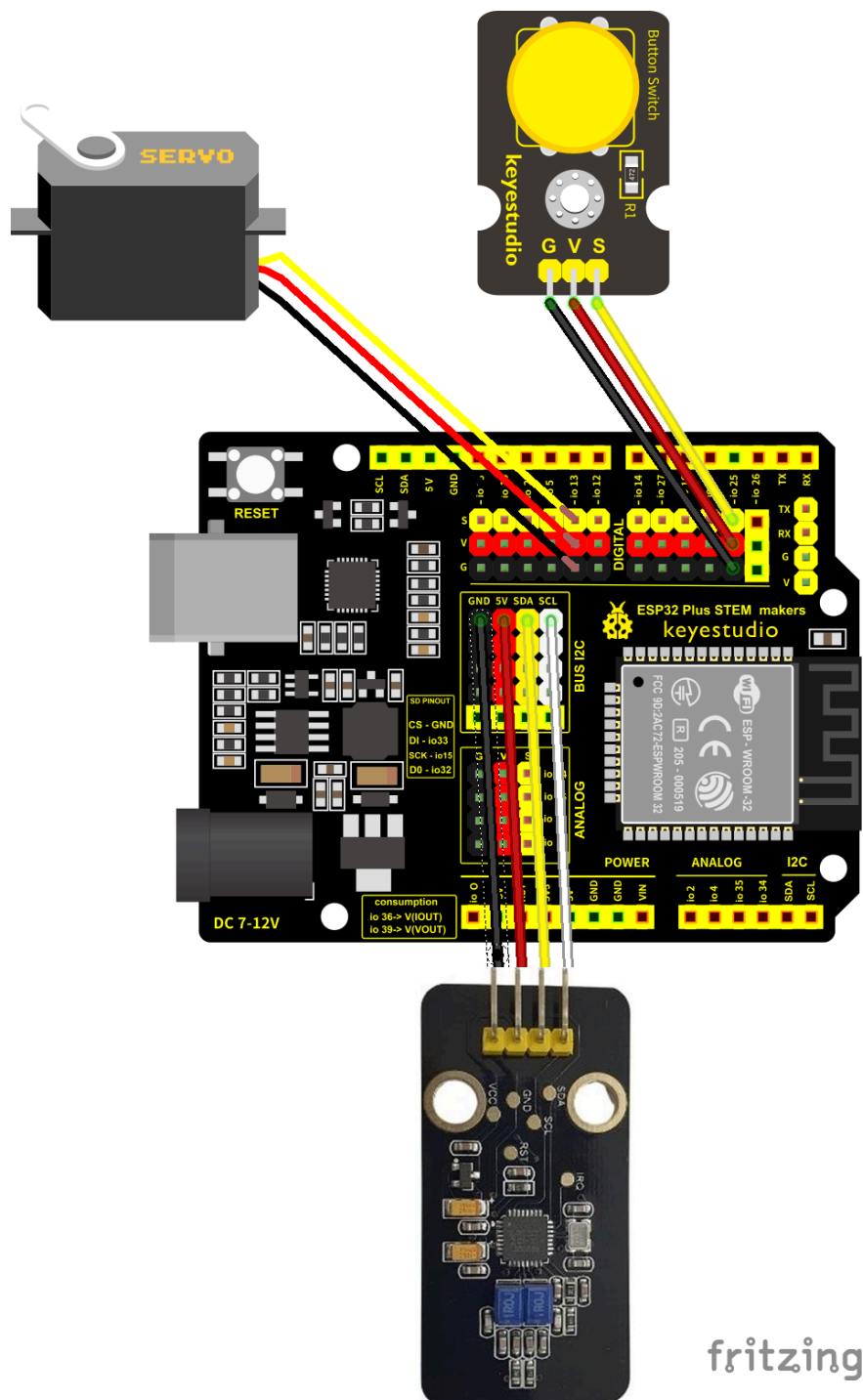
La programació següent ens permet inicialitzar el mòdul, llegir la targeta, guardar el codi en una variable i visualitzar-ho en la consola.



LLISTA D'ELEMENTS

- 1 Placa ESP32 STEAMakers
-  Mòdul RFID amb tag o targeta
-  Polsador
-  Servomotor

MUNTATGE



PROGRAMACIÓ





REPTES AMPLIACIÓ

- **Visualització:** La pantalla OLED ens indica l'estat de la porta.
- **Porta de garatge:** La porta s'obre amb la targeta i es tanca si, passat un cert temps, el sensor de col·lisió està activat. Aquest sensor es treballa igual que el polsador, amb un bloc d'entrada digital. Sense prémer la palanca dona ON i al prémer et dona OFF.

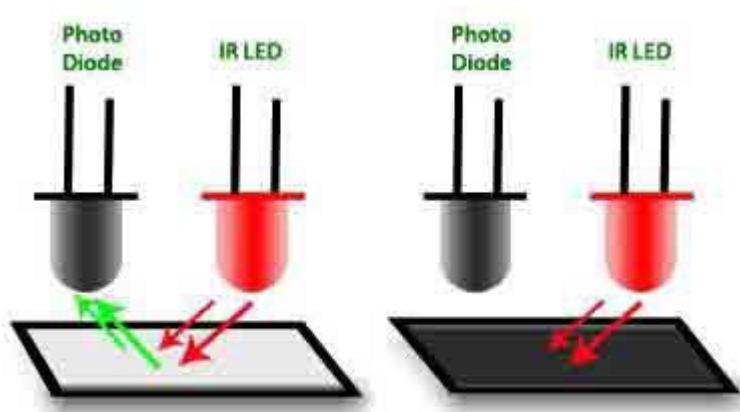
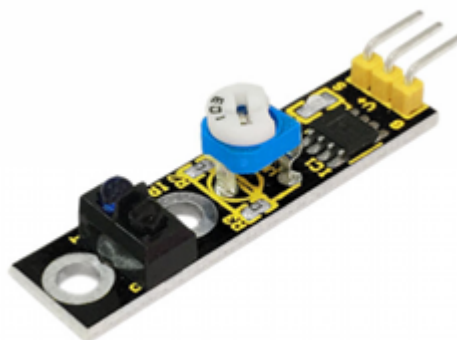
P20. Seguidor de línies.

OBJECTIUS

Fer servir el sensor de infrarojos tipus detector digital de línies negres i blanques com a un comptador de línies d'un paper imprès.



SABERS

El sensor de línies KS0050 està format per dos sensors infrarojos TCRT5000. El seu principi de funcionament és la d'emetre un senyal infraroig (emissor) i detectar la quantitat reflexada (receptor). En el cas de que el color sigui negre, aquest color absorbeix tota la llum però si és de color blanc la reflexa. Per tant, si el color és negre retornarà un valor Alt (1) i si és blanc retornarà un valor Baix (0). L'alçada de detecció està compresa entre 0,5 i 3 cm.



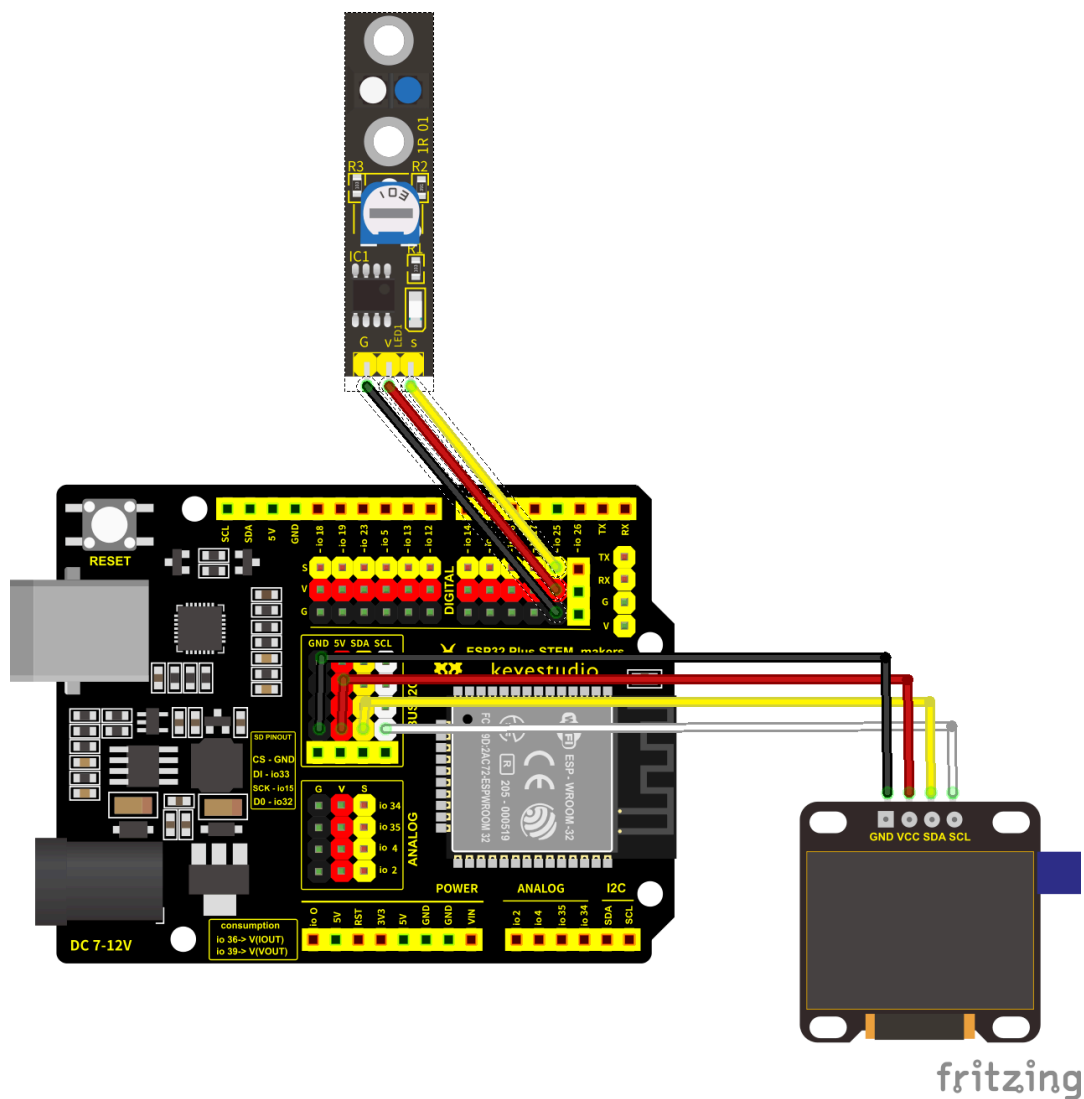
A la part superior hi ha un potenciòmetre per poder ajustar la sensibilitat. Al costat del potenciòmetre hi ha un petit led vermell que s'encén quan el sensor detecta el color blanc i s'apaga amb el color negre.

LLISTA D'ELEMENTS

- 1 Placa ESP32
- Sensor seguidor de línies 
- Pantalla OLED 
- Full imprès amb línies negres. [Descarrega-ho aquí.](#)

MUNTATGE

Connectem la pantalla OLED a un port I2C i el seguidor de línies al pin digital D3



PROGRAMACIÓ

Per realitzar aquest programa hem de tenir en compte que el sensor transmet senyal de manera continua sempre que està detectant un color o un altre, per tant si volem fer un comptador de línies el que hem de programar que es compti són els canvis d'estat, no la senyal. Per això necessitem crear dues variables booleanes "estat" i "estat anterior" amb les quals podem comprovar quan hi ha un canvi d'estat. A més a més necessitarem una tercera variable de tipus numèric "canvis" per fer el sumatori dels canvis d'estat, aquesta serà la que mostrem en pantalla.

A inicialitzar declarem les variables i iniciem la pantalla OLED, netegem la pantalla per si de cas queda alguna cosa anterior.

Al bucle associem el valor de la variable "estat" a la lectura del sensor, i després comprovem si aquest valor és diferent que l'anterior. Si és diferent vol dir que ha canviat de blanc a negre, i en aquest cas sumem 1 a la variable "canvis" cada vegada que es recorre el bucle es torna a donar a la variable "estat anterior" el valor de la variable "estat".

```

Inicialitzar
  # 1 Iniciar I2C 0x3C Mostra automàticament
  # 1 Netejar
  Establir estat = fals
  Establir estat_anterior = fals
  Establir canvis = 0

Bucle
  Establir estat = Segueix línies (IR) Pin 25 (D3)
  + si estat = cert i estat_anterior = fals
  fer
    Establir canvis = canvis + 1
    # 1 Netejar
    # 1 Text X 1 Y 1 + - crear text amb Formatar numèric canvis amb 0 decimals Led ON big
    # 1 Text X 45 Y 6 + - crear text amb "línies" Led ON medium
  Establir estat_anterior = estat
  
```

Un cop hem transferit el programa passem el full mirant cap a baix per sobre de lector. Hem d'intentar no fer-ho massa ràpid i que no passin les mans per sobre perquè poden donar errades.



REPTES AMPLIACIÓ

- **Ampliació 1: Escala musical.** Modificar el programa anterior de manera que cada vegada que passi una línia el brunzidor reproduïxi una nota musical diferent. Es pot fer una escala del DO al SI, i tornar a començar.
- **Ampliació 2: Comptador de voltes.** Dissenyar una roda amb un radi de color negre, imprimir-la en cartulina i acoblar-la al motor de CC.

P21. Sensors integrats i consum d'energia



OBJECTIUS

Conèixer i utilitzar els sensors integrats i el sistema de mesura de consum d'energia a la placa ESP32 STEAMakers.

SABERS

La placa ESP32 STEAMakers porta incorporat:

- Un sensor d'efecte hall
- Un sistema de mesura d'intensitat i voltatge

Amb el sensor d'efecte hall detectem variacions de camps magnètics a la proximitat de la placa.

Per explicar el funcionament del sistema mesurador de consum d'energia incorporat a la placa ESP32 STEAMakers, cal tenir en compte diversos sabers teòrics que cobreixen els aspectes fonamentals de l'electrònica i la física:

- Intensitat de Corrent (I): La quantitat de càrrega elèctrica que passa per un conductor per unitat de temps. Es mesura en ampers (A).
- Voltatge (V): La diferència de potencial elèctric entre dos punts. És la força que impulsa els electrons a través d'un circuit. Es mesura en volts (V).
- Resistència (R): La oposició que ofereix un material al pas de corrent elèctric. Es mesura en ohms (Ω).

La llei d'Ohm estableix que la intensitat del corrent que passa per un conductor entre dos punts és directament proporcional a la diferència de potencial entre aquests punts i inversament proporcional a la resistència del conductor:

$$V = I \cdot R$$

Potència Elèctrica (P): La quantitat d'energia consumida per unitat de temps. Es mesura en watts (W) i es calcula amb la fórmula:

$$P = V \cdot I$$

Energia (E): La capacitat per realitzar treball. En el context elèctric, es mesura en joules (J) o, més comunament, en kilowatts hora (kWh). La fórmula per calcular l'energia és:

$$E = P \cdot t$$




On t és el temps durant el qual la potència ha estat consumida

Si la tensió d'alimentació de la placa ESP32 STEAMakers baixa dels 4,8V, aquesta no funcionarà correctament (sobretot la transmissió wifi). És recomanable realitzar una comprovació de la tensió que està entregant el port USB de l'ordinador o la font externa d'energia que s'estigui utilitzant.

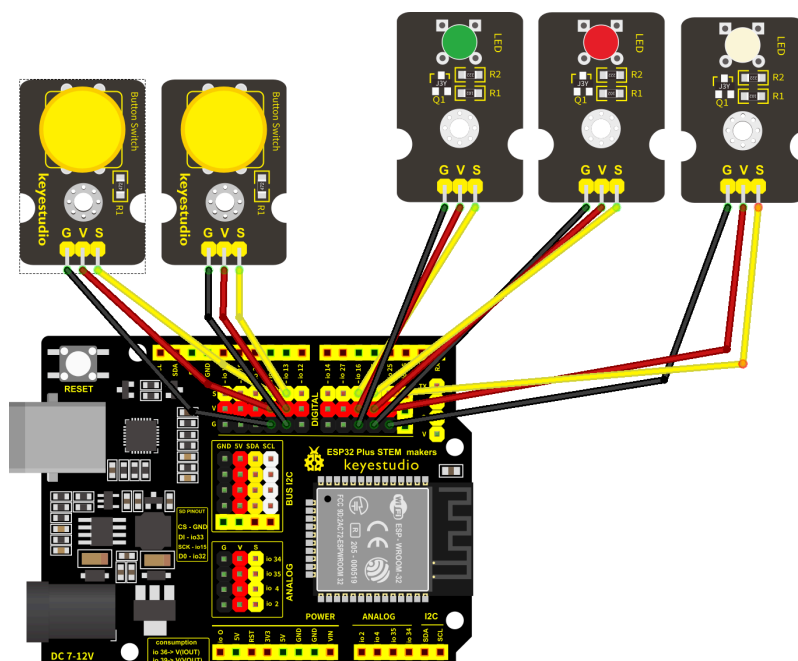
Dins d'arduinoblocks, els blocs per treballar amb aquests sensors els trobem a 'Sensors' → 'Integrats', i són els que veiem a la figura següent:



LLISTA D'ELEMENTS

- 1 Placa ESP32
-  1 polsador
-  1 sensor de col·lisió (Farà la funció de polsador)
-  3 leds

MUNTATGE



fritzing

PROGRAMACIÓ 21.1

Sensor de camp magnètic: Podem observar que en col·locar un objecte metàl·lic sobre la steamakers el valor de proximitat augmenta i en treure'l disminueix.

```

Inicialitzar
  >- Iniciar Bauds 115200

Bucle
  Establir proximitat = Sensor de camp magnètic
  >- Enviar " Proximitat: " Salt de línia
  >- Enviar proximitat Salt de línia
  Esperar 250 mil·lsegons
  
```

PROGRAMACIÓ 21.2

Llegir la tensió d'alimentació:

```

Inicialitzar
  >- Iniciar Bauds 115200
  Mesurador d'energia > Calibrar I=0
  
```

```

Bucle
  Establir Voltatge = Mesurador d'energia Volts (V)
  >- Enviar " Voltatge: " Salt de línia
  >- Enviar Voltatge Salt de línia
  >- Enviar " V " Salt de línia
  Esperar 1000 mil·lsegons
  
```

Consola sèrie

Baudrate: 115200 Connectar Desconnectar Netejar

Enviar

```

Voltatge: 5.08 V
Voltatge: 5.08 V
Voltatge: 5.09 V
Voltatge: 5.08 V
Voltatge: 5.09 V
Voltatge: 5.08 V
Voltatge: 5.09 V
Voltatge: 5.08 V
Voltatge: 5.09 V
Voltatge: 5.08 V
  
```

PROGRAMACIÓ 21.3

Controlar l'engegada i apagada de 3 leds mitjançant 2 polsadors (o un polsador i controlar l'estat amb una variable booleana). Comparar els 4 paràmetres de consum que podem calcular en els dos estats.

```

    Inicialitzar
    Iniciar Bauds 115200
    Mesurador d'energia > Calibrar I=0

    Bucle
    Establir intensitat = Mesurador d'energia Amps (A)
    Establir voltatge = Mesurador d'energia Volts (V)
    Establir potencia = Mesurador d'energia Power (W)
    Establir energia = Mesurador d'energia Energy (Wh)

    + si Polsador Pin 13 (D9) s'ha polsat Invertir
    fer engegar

    + si Polsador Pin 5 (D10) s'ha polsat Invertir
    fer apagar

    enviarPortSerie
    Esperar 250 mil·lisegons

    + a engegar
    Led Pin 25 (D3) Estat ON
    Led Pin 17 (D4) Estat ON
    Led Pin 16 (D5) Estat ON

    + a apagar
    Led Pin 25 (D3) Estat OFF
    Led Pin 17 (D4) Estat OFF
    Led Pin 16 (D5) Estat OFF

    + a enviarPortSerie
    Enviar "Intensitat: " Salt de línia
    Enviar intensitat Salt de línia
    Enviar " A " Salt de línia
    Enviar " Voltatge: " Salt de línia
    Enviar voltatge Salt de línia
    Enviar " V " Salt de línia
    Enviar " Potència: " Salt de línia
    Enviar potencia Salt de línia
    Enviar " W " Salt de línia
    Enviar " Energia: " Salt de línia
    Enviar energia Salt de línia
    Enviar " Wh " Salt de línia
    
```



Consola sèrie

Baudrate: 115200 ▾

Connectar

Desconnectar

Netejar

Enviar

Intensitat: 0.00A
 Voltatge: 5.07V
 Potència: 0.00W
 Energia: 0.00Wh
 Intensitat: 0.00A
 Voltatge: 5.07V
 Potència: 0.10W
 Energia: 0.00Wh
 Intensitat: 0.02A
 Voltatge: 5.07V
 Potència: 0.00W

REPTES AMPLIACIÓ

- **Desar les dades dels sensors interns:** graficar les 4 magnituds i recollir-les en un fitxer .csv
- **Consum per colors:** Comparar el consum d'energia si connectem la tira de leds neopixel i, quan engeguem els leds ho fem:
 - Tots en vermell
 - tots de color verd
 - tots de color blau
 - tots de color blanc

Glossari de termes de programació

Bucle: [P10](#)

Condicional: [P02](#)

Funcions [P11](#)

Llistes [P07](#)

Mapejat [P11](#)

Operadors aritmètics

Operadors lògics

Variables [P02](#)

Nombre aleatori [P04](#)

Reconeixements

Aquest manual està elaborat basant-se en documents anteriors, reconèixer i agrair el treball de:

- Physical Computing. ESP32 Plus STEAMakers, Imagina TDR STEAM, arduinoblocks. Versió 4.0. Fernando Hernández i Innova Didàctic.
- GUIA D'USUARI PLACA ESP32 STEAMakers. Innova Didactic.

Altres fonts de documentació utilitzades:

<https://www.luisllamas.es/>. Luis Llamas. Ingeniería, informática y diseño.

Imatges i material gràfic

- P17 [Imatge funcionament PIR](#). Lavina Saldhana, sota llicència internacional Creative Commons Attribution-Share Alike 4.0.
- [P17 Imatge sensor KS0349](#). Wiki Keystudio.
- [P19 Imatge sensor RFID amb i2C](#). Innovadidactic.