



ROBÒTICA · 3r ESO

UNITAT 6: COMUNICACIÓ SÈRIE + PROJECTE FINAL

Nom: _____ Curs/Grup: _____ Data: _____

Setmanes 10-11. Al final sabràs:

- Fer servir el monitor sèrie per comunicar PC ↔ placa.
- Enviar dades dels sensors al PC i llegir ordres del teclat.
- Comunicar dues plaques Arduino entre elles.

1. Comunicació entre dispositius electrònics

Els sistemes electrònics complexos solen estar formats per diversos elements que s'intercanvien dades: sensors, microcontroladors, actuadors, PC... La comunicació més senzilla entre Arduino i el PC (o entre dues plaques) és la COMUNICACIÓ SÈRIE ASÍNCRONA.

Tasa de transferencia (Baud Rate)

És la velocitat de transmissió en bits per segon (bps/baudis). El valor més comú és 9600. Cal que les dues parts estiguin a la MATEIXA velocitat.

Connexió i hardware

Quan connectes l'Arduino per USB al PC, el cable USB fa la conversió: els pins 0 (RX, rebre) i 1 (TX, transmetre) són els de la comunicació sèrie. ⚠ Si fas servir Serial al programa, NO connectis res als pins 0 i 1.

2. Usant el port sèrie a Arduino

Serial.begin(9600); — comença la comunicació a 9600 baudis (dins setup())

Serial.println('text'); — envia text al monitor sèrie i salta de línia

PROPOSTA 10 · Hola món sèrie

```
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  Serial.println('Hola món des d'Arduino!');
  delay(1000);
}
```

Puja'l i obre el Monitor Sèrie (icona de la lupa): veuràs el missatge cada segon.

PROPOSTA 11 · Llegir dades del PC

Ara al revés: el PC envia dades a l'Arduino.

```
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  if (Serial.available() > 0) // hi ha dades al buffer?
  {
    int dada = Serial.read(); // llegeix caràcter a caràcter
    Serial.write(dada); // els torna al monitor
  }
}
```

Envia 'ARDUINO' des del monitor: la funció write() envia els valors rebuts amb el seu caràcter corresponent de la taula ASCII. Si uses char en lloc de int, el mateix amb println().

PROPOSTA 12 · Potenciòmetre al monitor sèrie

Mostrem el valor d'un sensor analògic al monitor:

```
// Transmissió serie del valor d'un potenciòmetre
const int POT = A0;
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  int val = analogRead(POT);
  int porcentaje = map(val, 0, 1023, 0, 100);
```

```
Serial.print('Lectura analògica: ');
Serial.print(val);
Serial.print(' Percentatge: ');
Serial.print(Percentatge);
Serial.println('%');
delay(1000);
}
```

Observa la funció map() que ja coneixes: ajusta 0-1023 a 0-100 (percentatge).

PROPOSTA 13 · Encendre un LED des del PC

```
// Control d'un LED des del monitor serie
const int LED = 11;
char dato;
void setup()
{
  Serial.begin(9600);
  pinMode(LED, OUTPUT);
}
void loop()
{
  if (Serial.available() > 0)
  {
    dato = Serial.read();
    if (dato == 'R') // si rep R majúscula
    {
      digitalWrite(LED, HIGH);
      Serial.println('LED VERMELL ON');
    }
    else if (dato == 'r') // si rep r minúscula
    {
      digitalWrite(LED, LOW);
      Serial.println('LED VERMELL OFF');
    }
  }
}
```

Envia R → s'encén. Envía r → s'apaga. Aquest programa, modificat fàcilment, permet controlar qualsevol dispositiu des del teclat!

Activitat 10

Fes les 4 propostes. Registre:

Proposta

P10 Hola món

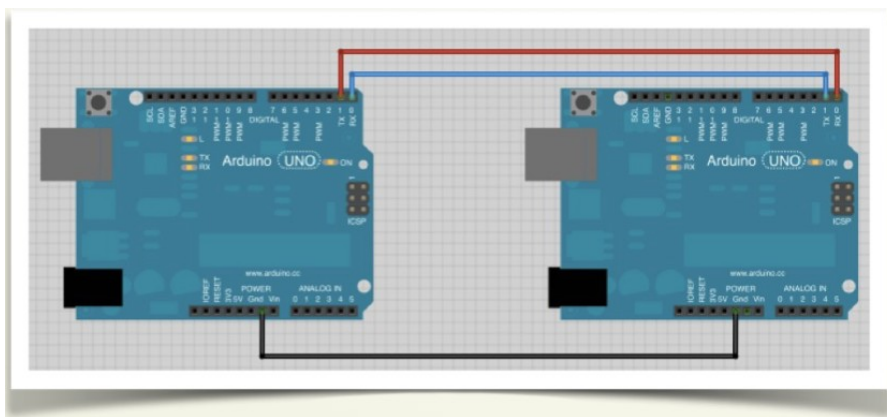
P11 Echo

P12 Pot al monitor

P13 LED des del PC

3. Comunicant dues Arduino

Es pot establir comunicació sèrie entre dues plaques: una EMISSORA i una RECEPTORA. Fem servir la llibreria SoftwareSerial (pins digitals qualssevol).



Cablejat creuat: TX₁→RX₂, RX₁→TX₂, GND comú. Falta connectar 5V comú.

- Uneix les dues terres (GND-GND) i alimenta la receptora des de l'emissora (5V-5V).
- Connecta el pin TX de l'emissora al RX de la receptora, i viceversa (creuat).

Programa de l'EMISSORA (rep ordres del monitor i les reenvia):

```
// Arduino UNO Emissora
#include <SoftwareSerial.h>
SoftwareSerial emisor(8, 9); // Rx, Tx
void setup()
{
  Serial.begin(9600);
  emisor.begin(9600);
}
void loop()
{
  if (Serial.available() > 0)
  {
    int dada = Serial.read();
    if (dada == 'R')
```

```

{
emisor.write('R');
Serial.println('LED VERMELL ON ENVIADA ORDRE');
}
else if (dato == 'r')
{
emisor.write('r');
Serial.println('LED VERMELL OFF ENVIADA ORDRE');
}
}
delay(1000);
}

```

Programa de la RECEPTORA (rep ordres i encén/apaga el LED):

```

// Arduino UNO Receptora
#include <SoftwareSerial.h>
int pinLed = 13;
int lectura = 0;
SoftwareSerial receptor(8, 9);
void setup()
{
pinMode(pinLed, OUTPUT);
receptor.begin(9600);
}
void loop()
{
if (receptor.available() > 0)
{
lectura = receptor.read();
if (lectura == 'R')
{
digitalWrite(pinLed, HIGH);
}
else if (lectura == 'r')
{
digitalWrite(pinLed, LOW);
}
}
delay(1000);
}

```

Puja cada programa a la seva placa (has de canviar de port quan canviïs de placa!). Des del monitor de l'emissora: envia R i el LED de la receptora s'encén. Has construït un sistema de telecomanda amb fil! També es pot fer per WiFi per Bluetooth... si hi arribem, ja ho farem!

Activitat 11

Munta la comunicació entre dues plaques. Prova:

- a) Afegeix més instruccions (G/g per un LED verd a la receptora).
- b) EXTRA: que la receptora respongui confirmació a l'emissora.

4. PROJECTE FINAL ARDUINO (opcional)

Disseny d'un Sistema de Control d'Accés amb Arduino

4.1. Descripció del Projecte

L'objectiu d'aquest projecte és construir un pany electrònic intel·ligent. L'alumnat aprendrà a integrar diferents components d'entrada i sortida controlats per una placa Arduino per crear un sistema de seguretat interactiu.

Quan un usuari introdueixi una contrasenya de 4 xifres mitjançant un teclat matricial, el sistema l'avaluarà:

- Si la contrasenya és **correcta**, la pantalla LCD mostrarà un missatge de benvinguda i s'obrirà el pany (simulat amb un missatge i un senyal sonor).
- Si la contrasenya és **errònia**, mostrarà un avís d'error a la pantalla i emetrà un so d'alerta.
- Durant la introducció de les dades, la pantalla mostrarà asteriscs * * * * per mantenir la privacitat del codi.

4.2. Objectius d'Aprenentatge

- **Electrònica:** Connectar correctament un teclat matricial (4x4), una pantalla LCD amb interfície I2C i un brunzidor a una placa Arduino.
- **Programació:**
 - Utilitzar llibreries externes específiques (Keypad i LiquidCrystal_I2C).
 - Gestionar variables de tipus text (String) i la comparació de cadenes.
 - Implementar estructures condicionals (if / else) i bucles per llegir entrades de l'usuari.
- **Disseny i UX:** Oferir retroalimentació visual i sonora adequada per a l'usuari final.

4.3. Materials Necessaris

Per dur a terme el muntatge pràctic, cada grup de treball disposarà de:

- 1 placa Arduino (o compatible) amb cable USB.
- 1 protoboard (placa de proves) i cables de connexió (mascle-mascle / mascle-femella).

- 1 teclat matricial de 4x4.
- 1 pantalla LCD 16x2 amb mòdul adaptador I2C (4 pins).
- 1 brunzidor (*brunzidor*) piezoelèctric.

Components necessaris:

- 1 placa Arduino (com l'Arduino Uno)
- 1 teclat matricial de 4x4 (*Keypad*)
- 1 pantalla LCD 16x2 amb mòdul I2C (per connectar-la fàcilment amb només 4 cables)
- 1 brunzidor (*brunzidor*) per donar feedback acústic (opcional però molt recomanable)
- Cables de connexió i protoboard

Instal·lació de llibreries necessàries

Abans de carregar el codi, assegura't de tenir instal·lades aquestes dues llibreries a l'Arduino IDE (les pots buscar al *Gestor de llibreries*):

1. **Keypad** (per gestionar el teclat matricial)
2. **LiquidCrystal_I2C** (per controlar la pantalla LCD)

4.4. Esquema de Connexions (Resum tècnic)

- **Pantalla LCD (I2C):**
 - GND $\rightarrow$ GND d'Arduino
 - VCC $\rightarrow$ 5V d'Arduino
 - SDA $\rightarrow$ Pin analògic A4 (en Arduino Uno)
 - SCL $\rightarrow$ Pin analògic A5 (en Arduino Uno)
- **Teclat Matricial (4x4):** Connexió de les 8 línies als pins digitals del 2 al 9 de l'Arduino.
- **Brunzidor:** Connectat al pin digital 10 i a GND.

4.5. El Codi del Projecte (Exemple Base)

```
#include <Wire.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
#include <Keypad.h>
```

```
// Inicialitzem la pantalla LCD (Adreça I2C 0x27 o 0x3F, 16 columnes i 2 files)
```

Jordi Tordera Soler · Robòtica 3r ESO · Curs 2026-2027

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
// Configuració del teclat matricial 4x4
```

```
const byte FILES = 4; // 4 files
```

```
const byte COLUMNES = 4; // 4 columnes
```

```
// Definim els caràcters de les tecles del teclat
```

```
char tecles[FILES][COLUMNES] = {
```

```
  {'1','2','3','A'},
```

```
  {'4','5','6','B'},
```

```
  {'7','8','9','C'},
```

```
  {'*','0','#','D'}
```

```
};
```

```
// Pins d'Arduino on es connecten les files i columnes del teclat
```

```
byte pinsFiles[FILES] = {9, 8, 7, 6}; // Connecta a les files 1 a 4
```

```
byte pinsColumnes[COLUMNES] = {5, 4, 3, 2}; // Connecta a les columnes 1 a 4
```

```
Keypad teclat = Keypad(makeKeymap(tecles), pinsFiles, pinsColumnes, FILES,  
COLUMNES);
```

```
// Variables del sistema de contrasenya
```

```
String contrasenyaSecret = "1234"; // Pots canviar-la per la que vulguis
```

```
String inputUsuari = ""; // Emmagatzema el que l'alumne va teclejant
```

```
const int pinBrunzidor = 10; // Pin opcional per a un brunzidor
```

```
void setup() {
```

```
  pinMode(pinBrunzidor, OUTPUT);
```

```
  // Inicialitzem la pantalla LCD
```

```
  lcd.init();
```

```
  lcd.backlight(); // Encenem la llum de fons de la LCD
```

```

    missatgeInici();
}

void loop() {
    char tecla, a;
    tecla = teclat.getKey(); // Llegim si s'ha premut alguna tecla

    if (tecla) {
        // Si prem '*', podem esborrar l'intent actual
        if (tecla == '*') {
            inputUsuari = "";
            missatgeInici();
            return;
        }

        // Si prem '#', comprovem directament el codi introduït
        if (tecla == '#') {
            verificarContrasenya();
            return;
        }

        // Si és un nombre (de 0 a 9) i portem menys de 4 xifres
        if (tecla >= '0' && tecla <= '9') {
            if (inputUsuari.length() < 4) {
                inputUsuari += tecla; // Afegim el dígit

                // Dibuixem els asteriscs corresponents al nombre de xifres entrades
                lcd.setCursor(0, 1);
                for (int i = 0; i < inputUsuari.length(); i++) {
                    lcd.print("*");
                }
            }
        }
    }
}

```

```

    // Si ja ha arribat a les 4 xifres, comprovem automàticament
    if (inputUsuari.length() == 4) {
        delay(300); // Petit respir per veure el darrer asterisc
        verificarContrasenya();
    }
}
}
}
}
}
}

```

```

void missatgeInici() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Introdueix codi:");
    lcd.setCursor(0, 1);
    lcd.print("  "); // Espai buit per començar a mostrar asteriscs
}

```

```

void verificarContrasenya() {
    lcd.clear();
    lcd.setCursor(0, 0);

    if (inputUsuari == contrasenyaSecret) {
        // PASSORD CORRECTE
        lcd.print("Porta");
        lcd.setCursor(0, 1);
        lcd.print("desbloquejada");

        // So agut d'èxit amb el brunzidor
        tone(pinBrunzidor, 1000, 200);
        delay(250);
    }
}

```

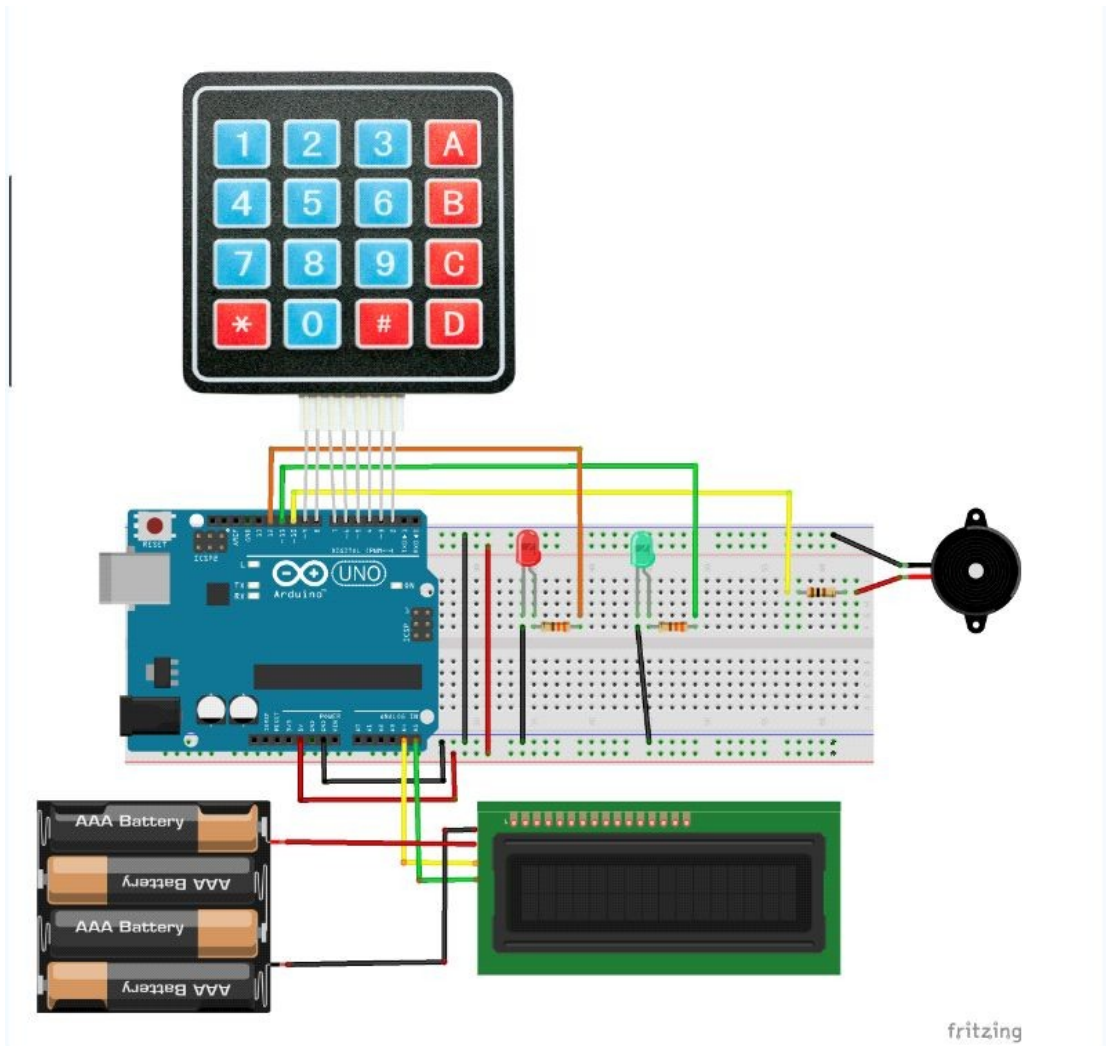
```
tone(pinBrunzidor, 1500, 300);

delay(3000); // Es manté el missatge 3 segons
} else {
    // CONTRASENYA ERRÒNIA
    lcd.print("Codi erroni");

    // So greu d'error amb el brunzidor
    tone(pinBrunzidor, 300, 500);

    delay(2000); // Es mostra l'error 2 segons
}

// Reiniciem la variable per a un nou intent
inputUsuari = "";
missatgeInici();
}
```



Cablejat keypad matricial, pantall i LCD. Brunzidor i LEDs indicators, opcionals.

4.6. Fases de Treball i Activitats Avaluables

El projecte es dividirà en les següents fases d'execució:

1. **Fase de Muntatge (Hardware):** Connectar tots els components a la protoboard de manera ordenada i segura sense connectar l'alimentació principal fins a la revisió del docent.
2. **Fase de Programació (Software):** Instal·lar les llibreries necessàries a l'IDE d'Arduino i carregar el codi base.

3. Fase de Proves i Modificació:

- Comprovar que els asteriscs apareixen correctament a mesura que es prem el teclat.
- Modificar el codi secret de l'Arduino per personalitzar la contrasenya del grup.
- Afegir millores opcionals (per exemple, un comptador de 3 intents màxims abans de bloquejar el sistema durant 10 segons).

4. **Fase de Memòria de Projecte:** Lliurament d'un informe breu explicant les dificultats trobades i com s'han resolt.

Repàs i autoavaluació

Rúbrica: 1 = No assolit · 2 = Satisfactori · 3 = Notable · 4 = Excel·lent

Objectiu
Fa servir Serial.begin i el monitor sèrie per comunicar amb el PC
Envio dades de sensors al PC (analogRead + map + println)
Rebo ordres del teclat (Serial.read) i contro actuadors
Comunico dues plaques Arduino amb SoftwareSerial
He acabat el projecte final i l'he defensat

Glossari

- Comunicació sèrie asíncrona: enviament bit a bit sense rellotge compartit.
- Baud rate: velocitat en bits/segon (9600 el més comú).
- RX / TX: pin de rebre / transmetre (0 i 1 a la UNO).
- Serial.begin(9600): inicia la comunicació.

- `Serial.println` / `print`: envia text amb / sense salt de línia.
- `Serial.available() > 0`: comprova si hi ha dades al buffer.
- `Serial.read` / `write`: llegir / enviar un caràcter (ASCII).
- `SoftwareSerial`: llibreria per fer sèrie amb pins digitals qualssevol.
- ASCII: taula que assigna un número a cada caràcter.