

Kommunicera med LCD monitorn

Ett flertal apparater i vår vardag innehåller en LCD monitor. Exempel på detta är radioapparater, larmsystem, hushållsmaskiner samt digital visning av tryck och temperatur.

Monitorn är ett bra sätt att med ett text meddelande kunna kommunicera med användaren och då gärna tillsammans med en knapptabla för inmatning.

Den LCD monitor vi använder oss av i övningen är en av de enklare typerna med blå bakgrundsbelysning och vit text.

Den hanterar två rader med 16 tecken i varje. Med denna kan man visa både små och stora bokstäver samt siffror och tecken. Den har en begränsning språkmässigt, då den ej gillar våra svenska å, ä och ö rakt upp och ner. Därför håller vi oss till engelska i övningarna.

Det första tecknet på första raden benämns (0,0) d.v.s. kolumn 0, rad 0 Sista tecknet på andra raden blir följaktligen (16,1).

I de kommande övningarna ska vi bygga ett system för poängräkning för ett enklare flipper spel. Då nickelkulan far ner i olika hålrum passerar den en mikrobrytare, som rapporterar till Arduino att ingången är hög. Med lite enkel matematik skickar vi upp nyss tagen poäng samt en summering av den totala poängställningen till LCD monitorn.

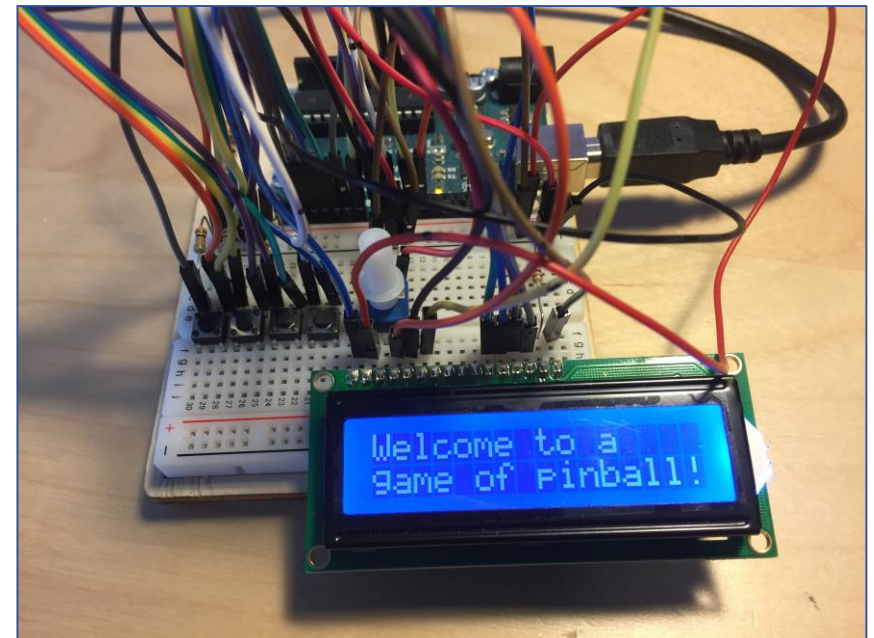
Vi förser spelet med totalt fyra strömställare.

Brytare A ger 10 poäng på port 6

Brytare B ger 15 poäng på port 7

Brytare C ger 20 poäng på port 8

Brytare D återställer poängräkningen till 0 d.v.s. nytt spel på port 9



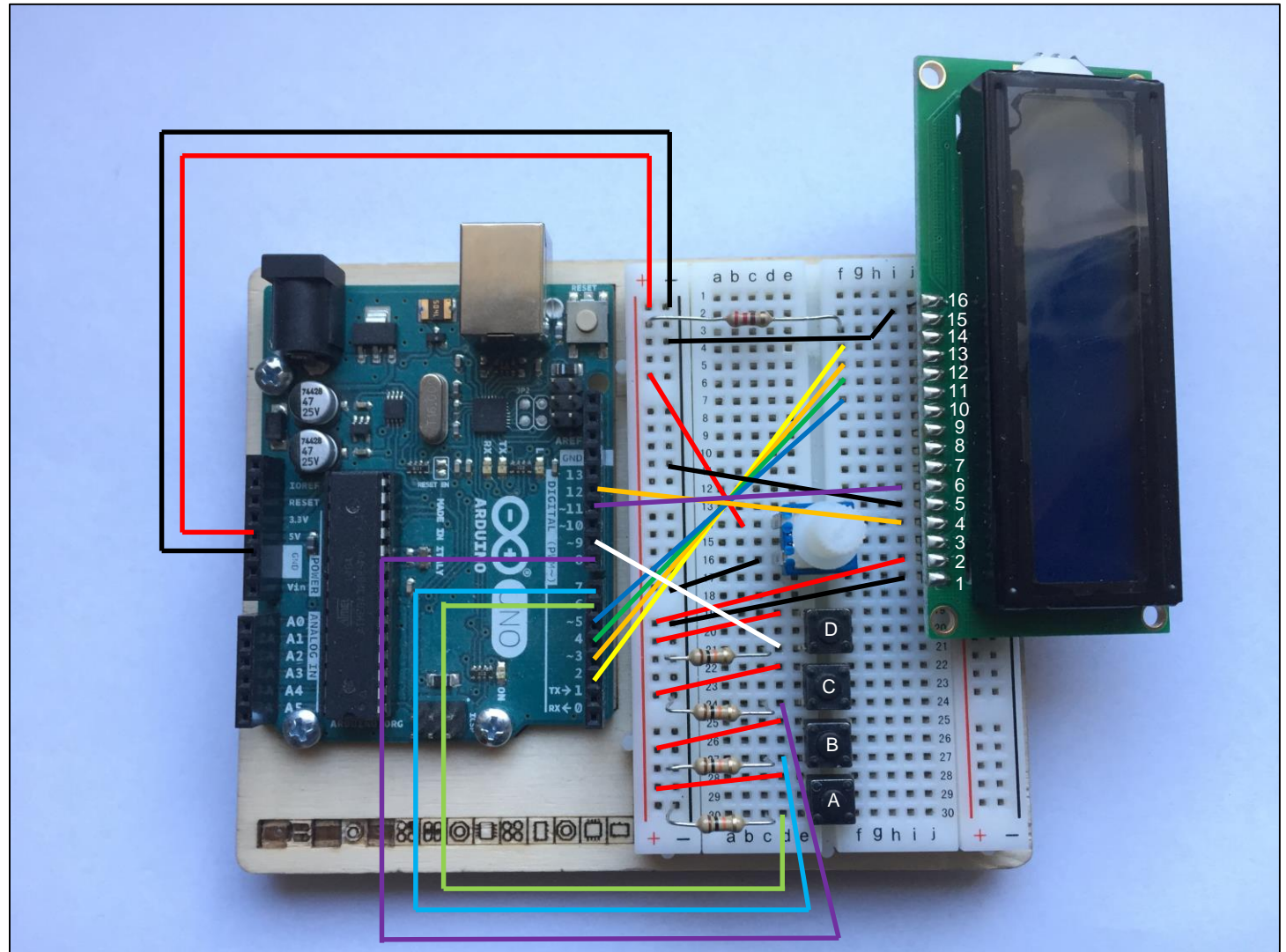
Övning 1a – Koppla upp hårdvaran

Vi använder oss av följande:

- 1 st. Arduino
- 1 st. kopplingsblock
- 1 st. LCD monitor
- 1 st. 220 Ω motstånd
- 4 st. 10 k Ω motstånd
- 1 st. potentiometer
- 4 st. strömbrytare

Potentiometern gör att vi kan ställa ljusstyrkan på monitorn.

Lägg märke till att potentiometerns mitre ben står i kontakt med LCD monitorns pin 3.



Övning 1b – Upprätta kommunikation med LCD monitorn

Precis som med servomotorn i föregående övning, hämtar vi in en färdig bakomliggande rutin för LCD-monitorn från vårt bibliotek.

```
#include <LiquidCrystal.h>
```

På Arduinon använder vi portarna 12, 11, 5, 4, 3 och 2 för kommunikation med monitorn. Vi går här inte in på hur detta går till.

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

I setup-delen talar vi om för vår bakomliggande rutin att vi använder oss av en monitor med 16 tecken och 2 rader

```
lcd.begin(16, 2);
```

Markören ställs på kolumn 0 och rad 0 med:

```
lcd.setCursor(0,0);
```

Ett meddelande inom citationstecken skickas till monitorn:

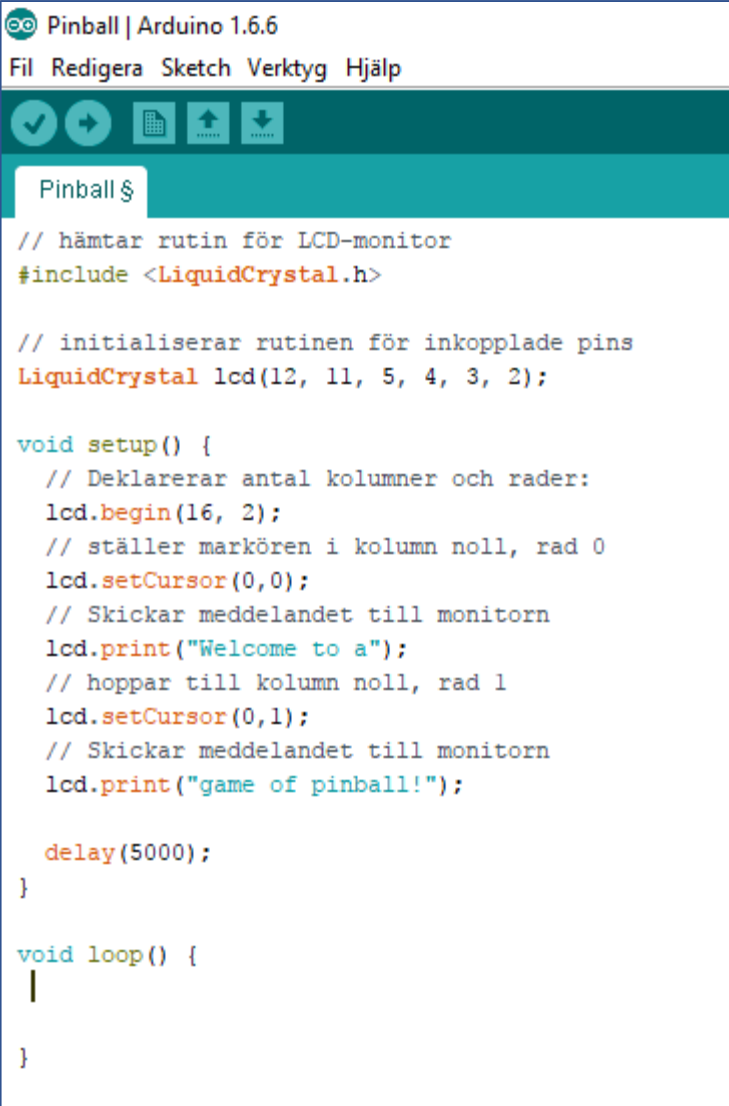
```
lcd.print("Welcome to a");
```

Raden skiftas och ett nytt meddelande skickas:

```
lcd.setCursor(0,1);  
lcd.print("game of pinball!");
```

Vi inför även en paus på 5 sekunder.

```
delay(5000);
```



```
Pinball | Arduino 1.6.6  
Fil Redigera Sketch Verktyg Hjälp  
✓ → 📄 ⬆️ ⬇️  
Pinball §  
// hämtar rutin för LCD-monitor  
#include <LiquidCrystal.h>  
  
// initialiserar rutinen för inkopplade pins  
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);  
  
void setup() {  
  // Deklarerar antal kolumner och rader:  
  lcd.begin(16, 2);  
  // ställer markören i kolumn noll, rad 0  
  lcd.setCursor(0,0);  
  // Skickar meddelandet till monitorn  
  lcd.print("Welcome to a");  
  // hoppar till kolumn noll, rad 1  
  lcd.setCursor(0,1);  
  // Skickar meddelandet till monitorn  
  lcd.print("game of pinball!");  
  
  delay(5000);  
}  
  
void loop() {  
  |  
}
```

Övning 1c – Deklarera parametrar och ingångar

Vi deklarerar parametrarna "points" och "totalpoints" som heltal med "int" och fäller in följande kod i skriptet.

Port 6, 7, 8 och 9 deklarerar som ingångar.

```
Pinball | Arduino 1.6.6
Fil Redigera Sketch Verktøy Hjälp
Ladda upp
Pinball §
// hämtar rutin för LCD-monitor
#include <LiquidCrystal.h>

// initialiserar rutinen för inkopplade pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

int points = 0;
int totalpoints= 0;

void setup() {
  // Deklarerar antal kolumner och rader:
  lcd.begin(16, 2);
  // ställer markören i kolumn noll, rad 1
  lcd.setCursor(0,0);
  // Skickar meddelandet till monitorn
  lcd.print("Welcome to a");
  // hoppar till kolumn noll, rad 1
  lcd.setCursor(0,1);
  // Skickar meddelandet till monitorn
  lcd.print("game of pinball!");

  pinMode(6, INPUT);
  pinMode(7, INPUT);
  pinMode(8, INPUT);
  pinMode(9, INPUT);

  delay(5000);
}

void loop() {
}
}
```

Övning 1d – Skapa rubriker

Skriptet i setup-delen fylls på med kod som gör att vi byter meddelande efter 5 sek.

Vårt tidigare meddelande i LCD monitorn tar vi bort genom att rensa hela skärmen med kommandot:

```
lcd.clear();
```

Vi ställer markören i olika lägen och skriver ut rubrikerna Points och Total på översta raden

```
// Skickar meddelandet till monitorn
lcd.print("game of pinball!");

pinMode(6, INPUT);
pinMode(7, INPUT);
pinMode(8, INPUT);
pinMode(9, INPUT);

delay(5000);
// Rensar skärmen
lcd.clear();
//Ställer markören på första kolumnen och i första raden
lcd.setCursor(0,0);
// Skriver ut rubriken Points
lcd.print("Points");
//Ställer markören på nionde kolumnen och i andra raden
lcd.setCursor(9,0);
// Skriver ut rubriken Total
lcd.print("Total");

}

void loop() {
|
}
```

Övning 1e – Poängräkning

Vi fyller på loop-delen med inledande kod som detekterar ingångarna samt skriver in poängställningen i variablerna "points" och "totalpoints".

Ett tryck på Reset-knappen medför att den totala poängen raderas både i minnet och på LCD-monitorn.

Totalpoängen räknas ut ifall ett nytt poäng inrapporterats. I koden beskrivs detta som att variabeln "points" är större än noll.

Data skickas till LCD monitorn.

Variabeln "points" nollställs inför nästa varv eller loop, medans värdet av variabeln "totalpoints" inte gör det så länge inte Reset-knappen aktiveras.

```
void loop() {  
  
  //Detekterar ingångar och fördelar poäng  
  if (digitalRead(6) == HIGH) {points=10;}  
  if (digitalRead(7) == HIGH) {points=15;}  
  if (digitalRead(8) == HIGH) {points=20;}  
  //Nollställer för nytt spel  
  if (digitalRead(9) == HIGH) {totalpoints=0;  
    //Ställer markören på nionde kolumnen och på andra raden  
    lcd.setCursor(9,2);  
    // Rensar total poäng  
    lcd.print("  ");  
  }  
  
  // Räknar ut totalpoängen om poängen är högre än noll  
  if (points>0){totalpoints = totalpoints + points;  
    //Ställer markören på första kolumnen och på andra raden  
    lcd.setCursor(0,1);  
    // Skriver ut nyss tagen poäng  
    lcd.print(points);  
    //Ställer markören på nionde kolumnen och på andra raden  
    lcd.setCursor(9,2);  
    // Skriver ut total poäng  
    lcd.print(totalpoints);  
  
    //Nollställer parametern points  
    points=0;  
    //Paus på 2 sek så att tagen poäng hinner uppfattas  
    delay(2000);  
    //Ställer markören på första kolumnen och på andra raden  
    lcd.setCursor(0,1);  
    //Rensar  
    lcd.print("  "); }  
}
```