

PROGRAMOWANIE UKŁADÓW ELEKTRONICZNYCH **ARDUINO**

Program definiujący pracę zegara cyfrowego na wyświetlaczu Multifunction SHIELD

```
// PINY DLA REJESTRÓW PRZESUWNYCH MULTI FUNCTION SHIELD
const int LATCH_PIN = 4;
const int CLK_PIN = 7;
const int DATA_PIN = 8;

// MAPOWANIE CYFR 0-9 DLA WYŚWIETLACZA SIEDMIOSEGMENTOWEGO
const byte SEGMENT_MAP[] = {
  0xC0, 0xF9, 0xA4, 0xB0, 0x99, 0x92, 0x82, 0xF8, 0x80, 0x90
};

// Pozycje cyfr na wyświetlaczu (od lewej do prawej)
const byte DIGIT_MAP[] = { 0xF1, 0xF2, 0xF4, 0xF8 };

// ZMIENNE CZASU POCZĄTKOWEGO
int godziny = 15;
int minuty = 34;

unsigned long poprzedniCzas = 0;
const unsigned long sekunda = 1000;
int licznikSekund = 0;

void setup() {
  pinMode(LATCH_PIN, OUTPUT);
  pinMode(CLK_PIN, OUTPUT);
  pinMode(DATA_PIN, OUTPUT);
}

void wyswietlCyfre(int pozycja, int wartosc, bool kropka = false) {
  byte wzor = SEGMENT_MAP[wartosc];
  if (kropka) wzor &= 0x7F; // Zapalenie kropki (bit 7)
  digitalWrite(LATCH_PIN, LOW);
  shiftOut(DATA_PIN, CLK_PIN, MSBFIRST, wzor);
  shiftOut(DATA_PIN, CLK_PIN, MSBFIRST, DIGIT_MAP[pozycja]);
  digitalWrite(LATCH_PIN, HIGH);
  delay(2); // Krótkie opóźnienie dla multipleksowania
}

void loop() {
```

```
// AKTUALIZACJA CZASU CO SEKUNDĘ
```

```
unsigned long obecnyCzas = millis();
```

```
if (obecnyCzas - poprzedniCzas >= sekunda) {
```

```
    poprzedniCzas = obecnyCzas;
```

```
    licznikSekund++;
```

```
    if (licznikSekund >= 60) {
```

```
        licznikSekund = 0;
```

```
        minuty++;
```

```
    }
```

```
    if (minuty >= 60) {
```

```
        minuty = 0;
```

```
        godziny++;
```

```
    }
```

```
    if (godziny >= 24) {
```

```
        godziny = 0;
```

```
    }
```

```
}
```

```
// WYŚWIETLANIE (MULTIPLEKSOWANIE)
```

```
// HH:MM
```

```
wyswietlCyfre(0, godziny / 10);
```

```
wyswietlCyfre(1, godziny % 10, (obecnyCzas / 500) % 2); // Kropka miga co pół sekundy
```

```
wyswietlCyfre(2, minuty / 10);
```

```
wyswietlCyfre(3, minuty % 10);
```

```
}
```