

E-Ink display 2,13"

1. POPIS

E-Ink displej se vyznačuje dobrou čitelností na přímém slunci, širokými pozorovacími úhly a nízkou energetickou náročností. Tento typ displejů najde své využití zejména v elektronických čtečkách knih, jelikož neunavuje oči jako klasické LCD, LED nebo OLED displeje. Tento konkrétní E-Ink komunikuje prostřednictvím komunikačního rozhraní SPI. Displej je kompatibilní s platformou Arduino, Raspberry a ST(ARM).



Základní charakteristika:

- rozlišení 250 x 122
- úhlopříčka 2,13"
- černobílý displej
- rozhraní SPI

2. SPECIFIKACE

Napájení	3,3 V	Počet barev	2
Logická napěťová úroveň	3,3 V	Pozorovací úhel	cca 170°
Pracovní proud	až 12 mA	Komunikační rozhraní	SPI
Spotřeba při vykreslení	až 40 mW	Operační teplota	0 až 40 °C
Rozlišení displeje	250 x 122 pixelů	Rozteč mont. otvorů (mm)	63 x 23
Úhlopříčka	2,13"	Rozměry displeje (mm)	59 x 29
Úplná odezva (při 25 °C)	cca 680 ms	Rozměry zařízení (mm)	68 x 30



3. ZAPOJENÍ

Pro správnou funkčnost je nutné snížit logickou napěťovou úroveň datových pinů Arduino, a to na úroveň 3,3 V. Pokud tak nebude učiněno, displej může být poškozen a nemusí pracovat správně. Doporučujeme využít level-shifter 1489315030 nebo napěťový dělič (vizte <https://navody.arduino-shop.cz/technikuv-blog/zmena-logickych-napetovych-urovni.html>).

Pin displeje	Pin Arduino Uno
BUSY	D7
D/C	D9
CS	D10
CLK	D13
SDI	D11
GND	GND
VCC	3,3 V



4. UKÁZKA PROGRAMU

Pro správnou funkci je nutné stáhnout adresář s knihovnou [fepd2in13.h](#) (společně s příkladem vykreslování bitmapy).

Příložený kód je příklad obsahu .ino souboru. K správné funkci je nutné, společně s .ino souborem, otevřít (ve stejném projektu) soubory imagedata.h a imagedata.cpp. Tyto soubory se nacházejí ve stejném adresáři na výše zmíněném odkazu. Po nahrání kódu do Arduino bude na displeji zobrazena bitmapa.

```
#include <SPI.h>
#include "fepd2in13.h"
#include "imagedata.h"

Epd epd;

void setup()
{
  // put your setup code here, to run once:
```

```
Serial.begin(9600);
if (epd.Init() != 0) {
    Serial.print("e-Paper init failed\n");
    return;
}
Serial.print("SetFrameMemory\n");
epd.SetFrameMemory(gImage_IMAGE);
Serial.print("DisplayFrame\n");
epd.DisplayFrame();
}

void loop()
{
}
```