ECE 2534

OLED interfacing (Digilent's PmodOLED)

Where to get information

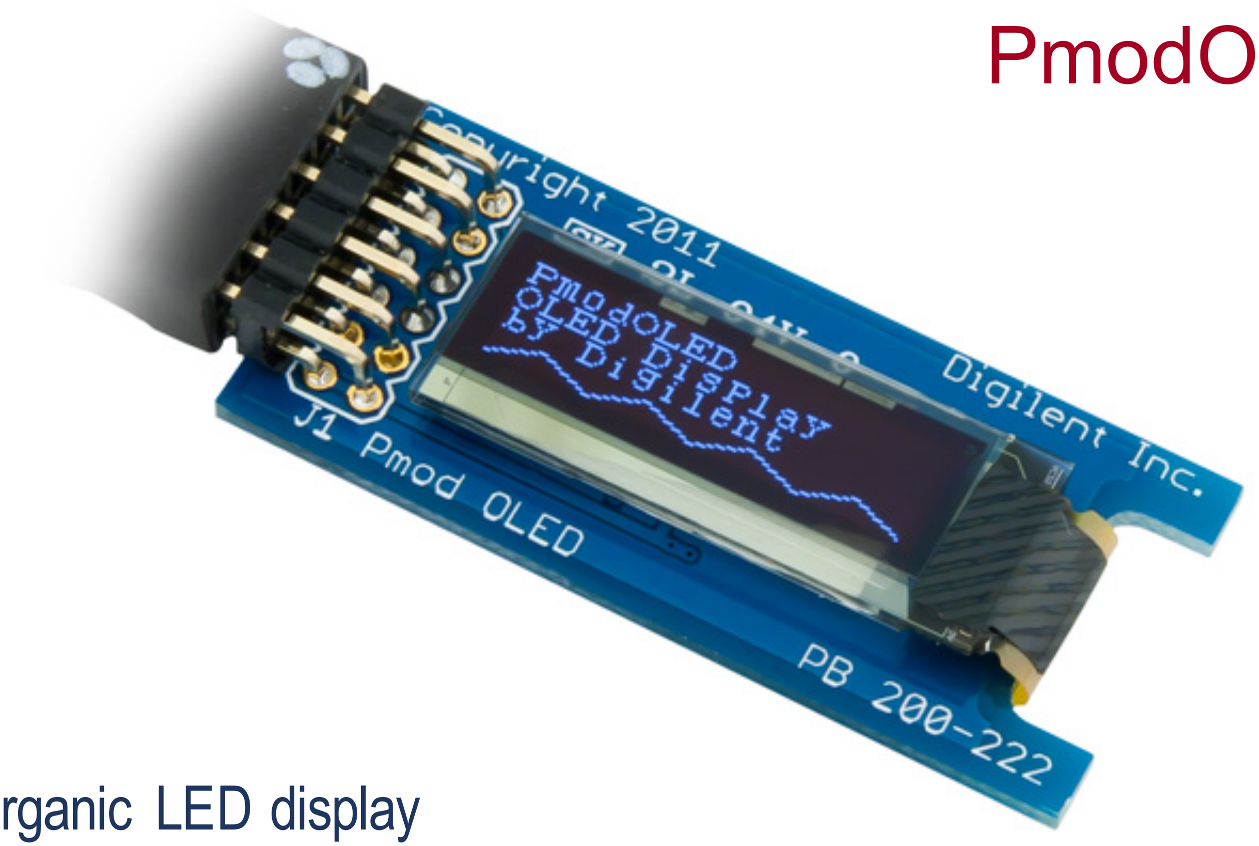
❑ **Reference manual for the Pmod**
  - PmodOLED_rm.pdf

❑ **Schematics** for the Pmod
  - PmodOLED_sch.pdf

❑ **Source code library** for the Pmod
  - PmodOLED_MPLAB_Library.zip
  - Contains PmodOLED.pdf (library reference manual)

❑ **These are available from Digilent's web site:**
  - www.digilentinc.com → Peripheral Modules → PmodOLED

PmodOLED



- 0.9” organic LED display
- 128 by 32 pixels
- Standard SPI interface
- Clock speeds of up to 10MHz
- Internal display buffer

It is a “write-only” module

- Displays the last image drawn on the screen until it is powered down or a new image is drawn to the display
- Refreshing and updating is handled internally

PmodOLED provides a “bitmap” display

- Rectangular grid of pixels (picture elements)
- Each pixel is either on (illuminated) or off (dark)

- ❑ The PmodOLED has a particular power-on/power-off sequence that must be followed to prolong the life of the display.
- ❑ Power-on sequence:

  1. Apply power to VDD.
  2. Send Display Off command.
  3. Initialize display to desired operating mode.
  4. Clear screen.
  5. Apply power to VBAT.
  6. Delay 100ms.
  7. Send Display On command.

Supporting files from from Microchip

- delay{.c, .h} ← uses Timer 1 for delays needed by PmodOLED
- PmodOLED{.c, .h} ← Init, Clear, Update the display
- OledChar{.c, .h} ← Set cursor, send a string, define new chars, …
- OledGraph{.c, .h}
- ChrFont0.c
- FillPat{.c, .h}

In PmodOLED{.c, .h}

```
void    OledInit();

void    OledClear();

void    OledClearBuffer();

void    OledUpdate();
```

In OledChar{.c, .h}

```
void    OledSetCursor(int xch, int ych);

void    OledGetCursor(int * pxcy, int * pych);

int     OledDefUserChar(char ch, BYTE * pbDef);

void    OledSetCharUpdate(int f);

int     OledGetCharUpdate();

void    OledPutChar(char ch);

void    OledPutString(char * sz);
```

In OledGrph{.c, .h}

```
void      OledSetDrawColor(BYTE clr);
void      OledSetDrawMode(int mod);
int       OledGetDrawMode();
BYTE *    OledGetStdPattern(int ipat);
void      OledSetFillPattern(BYTE * pbPat);

void      OledMoveTo(int xco, int yco);
void      OledGetPos(int * pxco, int * pyco);
void      OledDrawPixel();
BYTE      OledGetPixel();
void      OledLineTo(int xco, int yco);
void      OledDrawRect(int xco, int yco);
void      OledFillRect(int xco, int yco);
void      OledGetBmp(int dxco, int dyco, BYTE * pbBmp);
void      OledPutBmp(int dxco, int dyco, BYTE * pbBmp);
void      OledDrawChar(char ch);
void      OledDrawString(char * sz);
```

PmodOLED

- Small, versatile display
- It is a “bitmap” display
  - Supports graphics by allowing your code to turn each pixel on/off
  - An alphanumeric “character” interface is provided through software calls to lower-level graphics functions
- Need to look at the source files
- The documentation for PmodOLED code consists of
  - PmodOLED.pdf, and
  - comments in the source files