Keywords: How to use SSD1928, cmos camera interface to SSD1928, PIC32 interface SSD1928, cmos camera interface to MCU

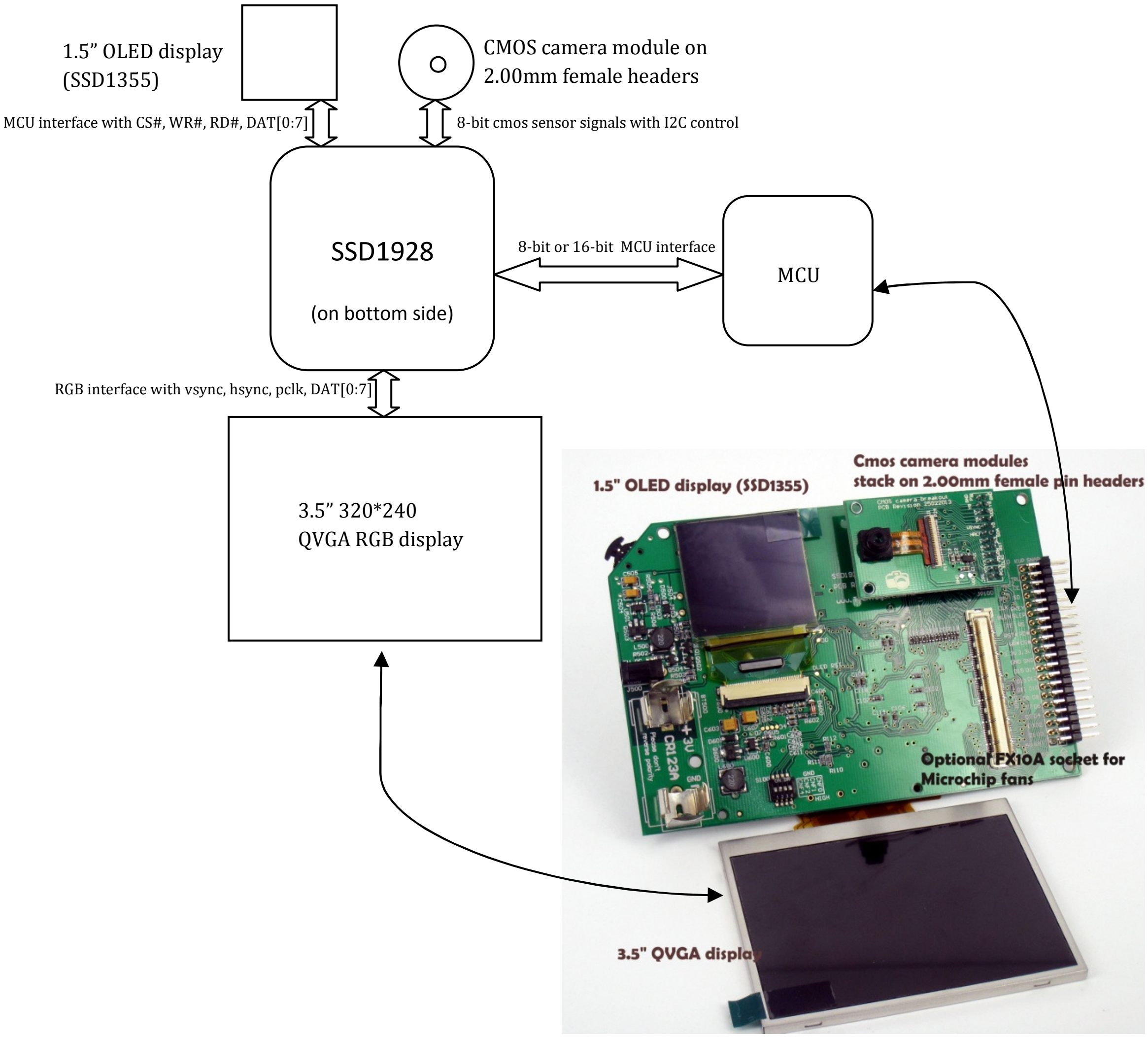
How to use SSD1928

This software demonstrates hardware features unique to the Solomon Systech SSD1928 Image Processor. These features include

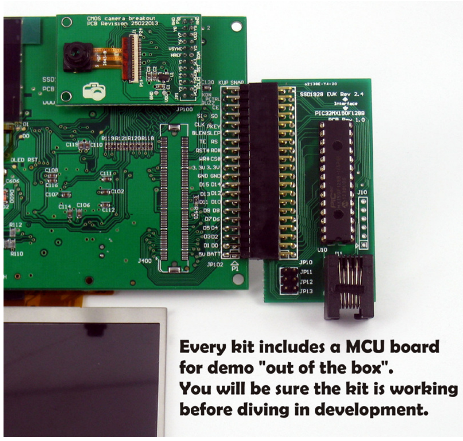
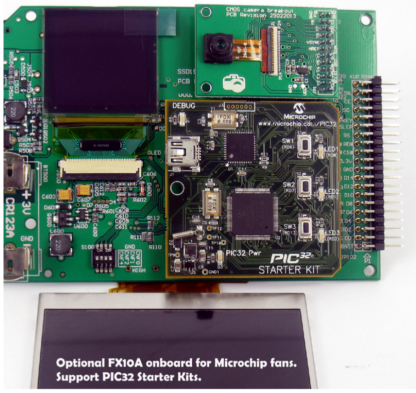
- cmos camera interface
- hardware JPEG decode
- hardware 4-bit SD card interface
- dual display connection in 8-bit MCU interface and 8-bit serial RGB interface

Hardware:

The hardware is illustrated with block diagram below.


1.5” OLED display
(SSD1355)
CMOS camera module on
2.00mm female headers
MCU interface with CS#, WR#, RD#, DAT[0:7]
8-bit cmos sensor signals with I2C control
SSD1928
(on bottom side)
8-bit or 16-bit MCU interface
MCU
RGB interface with vsync, hsync, pclk, DAT[0:7]
3.5” 320*240
QVGA RGB display
Cmos camera modules
stack on 2.00mm female pin headers
1.5" OLED display (SSD1355)
Optional FX10A socket for
Microchip fans
3.5" QVGA display

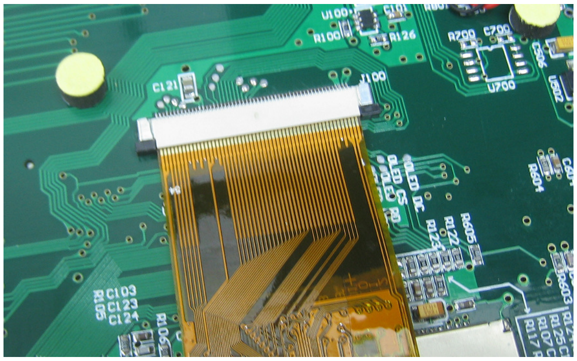
Four MCU models are supported with this demo. They are PIC32MX250F128B, PIC32MX360F512L, PIC32MX460F512L, and PIC32MX795F512L.

Every SSD1928 evaluation kit includes a small MCU board so that you know the kit is working before diving in application development. Firmware described later was preloaded in the PIC32MX250F128L MCU to drive the OLED, cmos camera, QVGA display, and get all .jpeg and .rgb files displayed from a SD card (not included). *Please make sure pin 1 of the female receptacle on PIC32MX250F128L PCB matches with pin1 of JP102 2.54mm pin header on SSD1928 Evaluation Kit. A reverse connection may damage both the MCU and SSD1928.*	 Every kit includes a MCU board for demo "out of the box". You will be sure the kit is working before diving in development.
The SSD1928 evaluation kit is sold with *optional* [i]Hirose FX10A socket for Microchip PIC32 Starter Kits. All current models of the starter kits are supported, including PIC32MX GP Starter Kit, PIC32MX USB Starter Kit (I/II), and PIC32MX Ethernet Starter Kit.	 Optional FX10A onboard for Microchip fans. Support PIC32 Starter Kits.

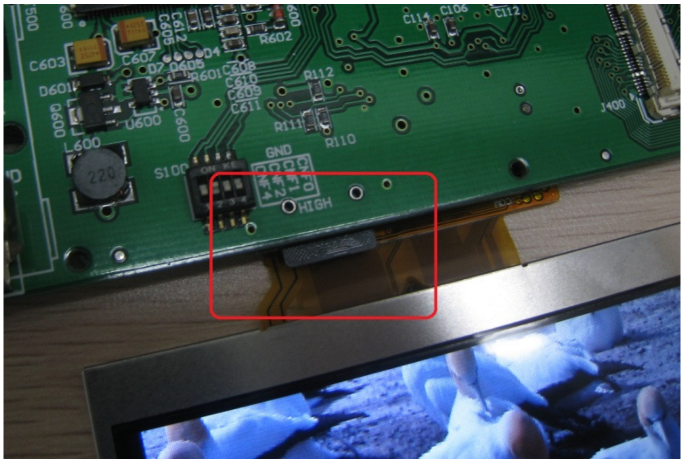
Two screens are included in every kit. They are 3.5” QVGA (320x240) display in 8-bit serial RGB interface and a 1.5” OLED (128x128) display in 8-bit MCU interface. The OLED display is used as an auxiliary display and the QVGA display as the main screen. This hierarchy is a direct consequence of SSD1928’s hardware feature that, no parallel display update to RGB and MCU interfaces is possible. A continuous pixel clock is required to maintain the display content on the 3.5” screen. On the other hand, the OLED display is a MCU-mode display that does not require an external pixel clock. Display data is written to internal frame buffer built-in the OLED display with CS line strobe and WR line strobe. Therefore, it is possible to maintain a static picture on OLED while showing a life video on 3.5” QVGA display.

An implication of this demo reveals that it is possible to use only the MCU-mode display when dual screen is not required. This leads to higher flexibility in hardware design because smaller displays of less than 3.2” usually come in MCU interface while a RGB-type display usually comes in a size larger than 3.5”. It is possible to connect a smaller smart-display to MCU port of SSD1926/8 and use it as the main panel simply by software configuration (please refer to data sheet of SSD1926/8).

Install the QVGA display to J100 with gold fingers up. Four sponges with double-tape are included in the package. These sponges will be useful if you want to stick the 3.5” display on PCB.



To run the demo, prepare a micro SD card (not included) by copying all files from the “Resources” folder to the root path of the SD card. Insert the card into the socket (J101) on SSD1928 EVK board.



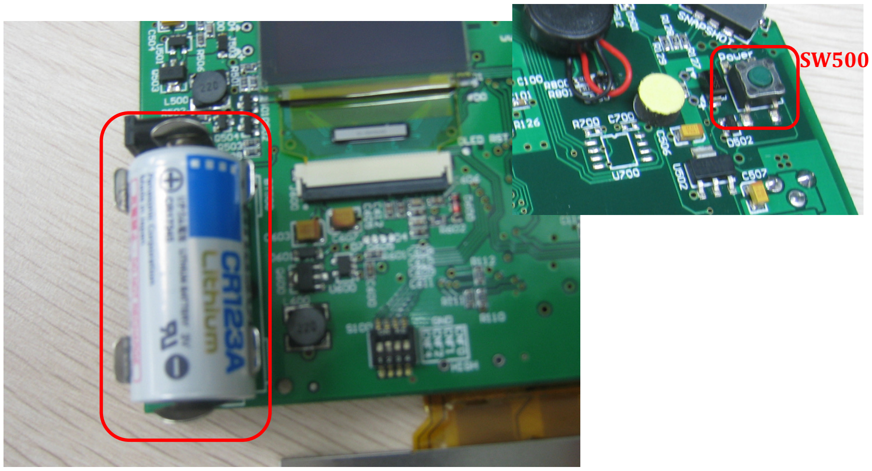
The card must be present prior to applying power.

Two power paths are possible. Readers are advised to look at the schematic for reference. They are 5V DC with pin positive for development or a 3V CR123A lithium battery (at BT500) for mobile applications. When 5V DC is applied at J500, the connection via R500 (1k) pulls up the gate of Q500 (IRLML6402TR) to switch it off. This leaves only the forward diode voltage at source pin of Q500 to source pin of Q504. This voltage at 3V (minus 0.2V) meets 3.3V from KIA1117-3.3V regulator output therefore reverse biasing the diode built-in Q500. As a result no battery drains when external power applied at J500.

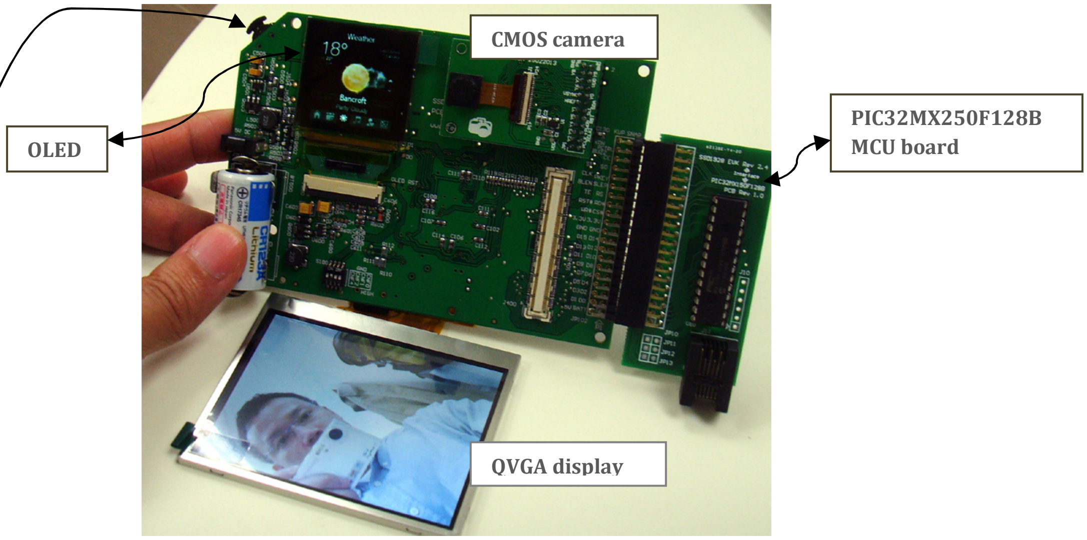
When a 3V CR123A lithium battery is used, MOSFET (Q504: IRLML6402TR) is used to cut off (or connect) the battery from the whole circuit. This ensures long battery life when the device is left idle with battery inserted.

The switching action is explained as follows.

- When switch SW500 is released, Q501 (MMBT3904) is switched off. Resistor R502 pulls gate and source of Q504 to the same voltage level. MOSFET Q504 is switched off therefore cutting off current from the battery.
- When SW500 is pressed and held, transistor Q501 (MMBT3904) conducts, pulling gate of Q504 low to turn it on. Battery power drains from the source pin of Q500 and conducts via Q504 with a resistance as low as 0.065Ω. Q500 is used as a reverse polarity protection device when the battery was inserted in wrong polarity out of mistake. If it was not a development kit, Q500 could be omitted.
- Now 3V is present at the inductor end of L500. DC-DC converter XC9104 switches 3V to 3.3V to supply the whole circuit including the MCU.
- MCU boots up from POR. Software in MCU sets PWR_CTRL pin an output high to turn on Q502, pulling gate of Q504 low to maintain its state. After this, the user may release SW500 because the state of Q504's gate is now maintained by software.
- Power cut-off is possible with PWR_CTRL switched low after a press-and-hold on SW500 like most mobile phones and digital cameras.



The software starts with displaying a static image on OLED and then capturing a life video from a cmos camera (optional). Picture below shows a self-portrait with 3V lithium battery installed. Three cmos sensor models are supported in current firmware version. They are OmniVision OV965x 1.3Mpixel cmos camera, OV2640 2Mpixel cmos camera, and OV2643 cmos camera. The software distinguishes which cmos sensor has been inserted by reading the PID register.



Different sensors return different PIDs. The software simply compares the return value with known values to tell which camera is used. If no camera is in place, the QVGA screen will display a blank screen.

Pulling the scroll button anti-clockwise will browse to the next file on SD card.

The demo shows JPEG files with *.jpg extension. The unique setup of this demo is that the JPEG data is not sent back to the MCU, rather, it is buffered and decoded by SSD1928 JPEG hardware decoder to the frame buffer on the SSD1928 RAM directly. The MCU only manages the SSD1928 hardware registers and the file system. When displaying a JPEG file, the SSD1928 frame buffer is set to use the YUV color mode. Resolution of the images must be less than 2560x1920 (maximum scale down factor supported by the SSD1928 is 8).

The demo will also play video files formatted as RGB565 for each frame. Use ffmpeg (http://ffmpeg.org) to do the video conversion, an example ffmpeg command line that outputs a video compatible with this demo is:

```
ffmpeg -i video.avi -s 320x240 -pix_fmt rgb565 video.rgb
```

The demo attempts to play video files with *.rgb extension. When playing an RGB file, the SSD1928 frame buffer is set to use the RGB color mode.

Software:

This software was modified from the project "SSD1926 JPEG and SDCard Demo PIC32" released by Microchip in her Graphics Library version 2012-10-15. The major difference between SSD1926 and SSD1928 is an extra cmos camera interface in SSD1928. The penalty is that, only 8-bit serial RGB screen allowed for SSD1928 whereas 24-bit RGB screen can be used with SSD1926. Driver from Microchip for SSD1926 is well written and easy to understand. Full source code of the Graphics Library is available for download at www.Microchip.com[ii] . Demo of for this note extracts only the relevant files. The environment to compile this demo is summarized below.

- Microchip MPLAB Integrated Development Environment (IDE) version 8.83.

  At time of writing, the latest software version is v8.86. MPLAB can be downloaded from this hyperlink

  http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002

  By following the installation procedure for default settings, MPLAB was installed under ***C:\Program Files\Microchip\MPLAB IDE***.

- MPLAB C Complier for PIC32 MCUs version 2.00.

  The MPLAB C Compiler for PIC32 (C32) is a full-featured ANSI compliant C compiler for Microchip's PIC32 family of 32-bit microcontrollers. A free evaluation is available by downloading the Evaluation Edition. It has no code size limit and provides complete functionality for 60 days. The compiler is completely usable after 60 days other than certain optimization levels are disabled.

  The hyperlink for downloading C32 is http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2615&dDocName=en532454.

  By following the installation procedure for default settings, C32 was installed under ***C:\Program Files\Microchip\mplabc32\v2.00***. All example programs from us are working on the Evaluation Edition before and after 60 days.

[i] It is necessary to purchase the option with Hirose FX10A socket to use Microchip Starter Kits.

[ii] http://www.microchip.com/pagehandler/en-us/technology/graphics/tools/software.html