STM32H750VBT6으로 소형 칼라 LCD 구동하기

3.5” 이하의 작은 칼라 LCD들은 크게 3가지의 인터페이스를 가지고 있습니다. SPI, CPU BUS, RGB 인터페이스가 그것입니다.

용어는 통일되지 않고 같은 것을 두고도 여러 가지로 쓰이고 있습니다. 물론 다른 인터페이스가 없는 것은 아닙니다.

핸드폰용 고해상도 LCD들은 이 범주를 벗어나며, 앞으로 일반화 되겠지만 현재 Cortex-Mx 정도의 마이크로 컨트롤러로 쉽게 구동하고,

쉽게 구할 수 있는 칼라 LCD들은 대부분 이들 3가지 중 하나의 인터페이스를 갖고 있습니다.

목적에 맞는 LCD를 선정하기 위하여는 인터페이스 방식 각각의 특징을 알아야 합니다.

간단히 표로 먼저 정리해 보겠습니다.

|  |  |  |  |
| --- | --- | --- | --- |
| **방식** | **장점** | **단점** | **특징** |
| **SPI** | * 적은 핀으로 가능 * 거의 모든 마이크로 컨트롤러에서 사용 가능. | * 속도가 느림 | * 2인치 이하의 소형 LCD에서 대부분 채택 |
| **CPU(MCU)** | * 외부 메모리 확장 기능이 있는 마이크로 컨트롤러에서 사용 가능. * 소형 LCD에 충분한 속도 | * 핀 컨트롤을 통해서도 제어 가능하지만 FMC가 필요 | * 2.4~4인치 급에서 많이 채택 |
| **RGB** | * 속도가 빠름. * 전용 컨트롤러의 다양한 기능을 사용할 수 있음. * CPU에 부담을 거의 주지 않음. | * 마이크로 컨트롤러 내부에 많은 메모리(Frame Buffer)가 필요. * 메모리 확장으로 많은 핀카운트 필요. * 작은 LCD의 경우 Configuration을 위해 SPI 통신을 필요로 하는 경우가 많음. | * 5인치 이상에서 대부분 채택 * STM32F429 이상에 LTDC 내장. * LCD 전용 인터페이스 |

많은 소형 LCD의 경우 여러 가지 내부 설정(Configuration)을 해 줘야 합니다.

8비트 또는 16비트를 사용할 지, 감마 설정, 파워 설정 등등이 그것입니다. 이를 위해서 RS 또는 CD 로 이름 붙여진 핀이 사용됩니다.

“LCD 내부 특정 기능 레지스터에 설정 값을 써 넣은다”는 개념을 쓰기도 하고, “특정 명령어를 먼저 보내고 설정 값을 이어서 보낸다”는

개념을 사용하기도 하는 데, 같은 것입니다. 전자가 RS, 후자가 CD 라는 핀 이름을 쓰는 데, 지금 보내는 데이터가 레지스터

주소인지(RS Low) 설정 값인지(RS High) 구분 하는 데 이 핀이 쓰이는 것입니다. 또는 지금 보내는 데이터가 명령어인지(CD Low)

설정 값인지(CD High) 로 생각할 수도 있습니다. SPI 방식이나 CPU 방식일 경우 이 핀으로 Configuration을 할 수 있지만

RGB(LTDC 사용)방식에서는 표준 설정 방법이 없습니다. 그래서 RGB 방식 LCD에서는 이미지 데이터는 RGB 라인으로 보내고 설정을 위해

SPI를 덤으로 가지고 있는 경우가 많습니다. 하지만 5인치 이상인 경우는 내부 설정없이 RGB 만으로 구성된 LCD들이 대부분입니다.

또 하나 중요한 차이는 그래픽 데이터(GRAM)를 어디에 가지고 있을 것인가 입니다.

SPI 및 CPU 버스방식에서는 그래픽 램을 LCD내부에 있는 것을 사용합니다. 좌표 데이터를 먼저 보내고 그 위치의 칼라 값을 이어서

보내는 방식으로 드라이브합니다. 하지만 RGB 방식에서는 마이크로 컨트롤러에 그래픽 램을 가지고 있어야 합니다. LTDC는 CLK,

HSYNC, VSYNC 및 데이터 버스를 통해서 마이크로 컨트롤러 내의 그래픽 데이터를 계속 LCD로 퍼 날라줍니다. 마이크로 컨트롤러

내부에 있는 그래픽 램을 보통 Frame Buffer 라고 부릅니다. STM32 계열에서는 2개의 Frame Buffer를 지원하고 Layer 라고 부릅니다.

VGA 해상도인 640 x 480 만 하더라도 한 픽셀 당 16bit 칼라(RGB565)를 사용한다고 하면 640 x 480 x 2 = 614,400Bytes 가 되어

내장 메모리로는 감당이 안됩니다. 32bit 칼라 모드를 쓴다면 더더욱 힘든 상황이 됩니다. 그래서 보통은 외장 메모리, 그것도 SDRAM을

확장해서 사용합니다. 보통 144 핀 이상인 마이크로 컨트롤러를 사용해야 합니다.

우리가 준비한 테스트 보드는 STM32H750VBT6으로 100핀이고 외부에 SDRAM을 확장할 수 없습니다. 내부 메모리는 1Mbytes 가

있지만 여러 영역으로 나뉘어 있고 연속적으로 사용할 수 있는 영역은 가장 큰 곳이 RAM\_D1으로 512Kbyte 입니다. 그래서 해상도가

320 x 240인 ILI9341 을 채택했습니다. 내장 메모리에 2개의 Layer를 설정해서 테스트해 볼 수 있습니다. 다른 하나는 ST7796을

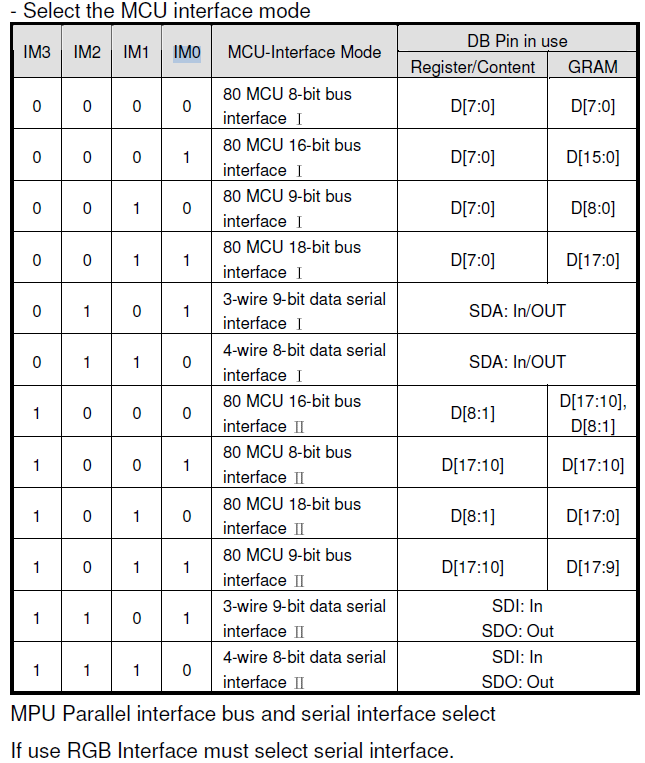
컨트롤러로 내장한 320 x 480 해상도의 LCD로 내장 메모리에 하나의 Frame Buffer를 설정해서 RGB Interface를 테스트해 볼 수 있습니다.

1. **통신 인터페이스**

SPI, CPU BUS, RGB 인터페이스의 특징부터 각각 알아보겠습니다.

1. **통신 인터페이스의 선택**

소형 LCD 에 쓰이는 ILI9341, ST7796 등의 컨트롤러는 여러 가지 통신 인터페이스를 동시에 가지고 있고 선택해서 쓰게 되어 있습니다.



위 표는 ILI9341의 인터페이스 선택 방법을 나타냅니다만 다른 LCD들도 비슷 합니다 . 보면 IM0, IM1, IM2, IM3의 4개 핀으로

선택을 하게 되어 있습니다.

보통 많이 쓰이는 방법이 MCU 8-bit bus 또는 16-bit bus, 4-wire 8-bit data serial interface 입니다. 맨 아랫줄에 쓰여 있듯이

RGB 인터페이스를 사용할 경우 시리얼 인터페이스를 선택해야 됩니다. 소형 LCD들은 핀 수를 줄이기 위해서 모든 I/O를 핀으로

뽑아 놓지 않습니다. IMx핀들을 고정시키고 외부에서 선택할 수 없게 만든 것도 있으며, 외부에서 선택하게 했지만 실제 통신에

필요한 핀을 외부로 뽑아 놓지 않아서 사용할 수 없기도 합니다. 예를 들어 시리얼 인터페이스를 선택할 수 있게 했지만

HSYNC, VSYNC, DE 핀들을 뽑아 놓지 않아서 RGB 인터페이스는 사용할 수 없는 경우가 많습니다. CPU BUS 방식을 사용하기 위해서는

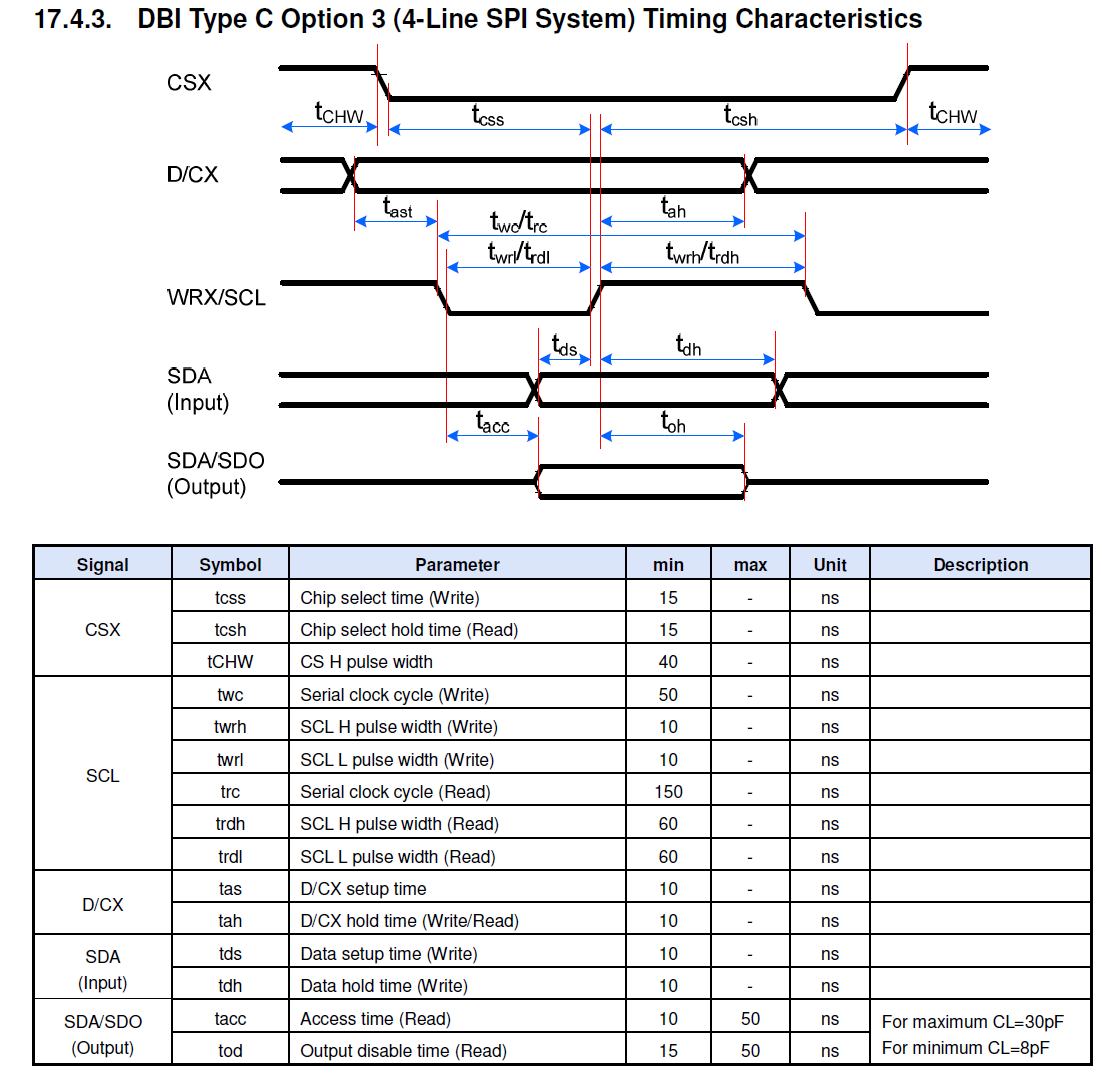
RD, WR 핀들이, 시리얼 인터페이스를 사용하기 위해서는 CLK, SDI 핀들이 있는 지 꼼꼼히 살펴보고 LCD를 선택해야 합니다.

1. **SPI 인터페이스**

SPI 인터페이스 방식은 3-Line 또는 4-Line 방식이 흔히 사용되는 데, 일반 마이크로 컨트롤러에 내장되어 있는 SPI를 그대로

사용할 수 있는 4-Line 방식이 많이 사용됩니다. 이름에서 알 수 있듯이 최소 4개의 I/O만 있으면 구동 가능하므로 핀 수가

빈약한 조건에서 많이 선택이 됩니다. 하지만 일반 로직 레벨의 시리얼 통신이기 때문에 속도가 셋 중에 가장 느립니다.



위의 그림은 ILI9488 데이터시트에서 발췌한 것으로 SPI 통신 방식의 타이밍도 입니다. 위에 그려지 있지 않은 RESET 신호를

포함해서 보통 6개의 I/O가 필요합니다. LCD 쪽에서 출력인 가장 아래 SDO 라인은 LCD내의 그래픽 램(GRAM)을 읽어올 필요가

없을 때는 사용하지 않아도 됩니다.

표에서 가장 주의 깊게 보아야 할 부분은 twc 의 min 값으로 클럭의 빠르기를 결정하는 것입니다. 클럭이 빠르면 그리기 속도가

빨라지는 것이므로 빠를수록 좋겠지만 마냥 빠르게 할 수는 없습니다. twc의 최소 요구 값이 50ns이므로 최대 클럭 주파수는

역수를 취해보면 20MHz 인 것을 알 수 있습니다. 그래서 CubeMX의 SPI 설정 중 Parameter Settings에서 Prescaler의 값을

잘 조정해서 Baud Rate를 20MBits/s가 넘지 않도록 설정해야 합니다.

표에서 trc는 읽어올 때의 클럭 속도를 표시하고 있는 데, 쓰기 때보다 보통 느립니다. LCD의 그래픽 램을 읽어올 때는 trc 값을

참고하여 더 느린 클럭을 사용해야 합니다.

SPI 인터페이스를 써야하는 조건인데 속도는 빠른 LCD를 써야 한다면, twc, trc 값을 보고 작은 것을 선택하는 것이 요령입니다.

타이밍도에서 CSX로 표시된 것은 칩 셀렉트로 STM32 시리즈에서 하드웨어 방식으로 설정해 놓으면 그냥 LOW 상태를

계속 유지하는 등, 생각하는 데로 동작하지 않는 경우가 있습니다. 그래서 보통은 수동으로 GPIO처럼 제어를 하는

경우가 많습니다. STM32H750에서는 “Hardware NSS Output Signal”로 설정해 놓으면 잘 동작합니다.

1. **CPU BUS 인터페이스**

MCU 방식, 패러랠 버스 방식 등으로 불리는 것입니다.

SPI 보다 빠르고 전용의 LCD 인터페이스를 갖추지 못한 마이크로 컨트롤러들을 위한 통신 방식입니다. LCD를

외장 메모리처럼 취급하여 그래픽 램에 읽고 쓰기를 하겠다는 것이지요. 전용 LCD 인터페이스를 갖지 못한 마이크로 컨트롤러도

외장 메모리를 붙일 수 있는 것은 많으니까요. CubeMX에서도 FMC에서 Memory type을 LCD Interface로 선택하면 이 방식을

지원하지만 약간의 편의성만 제공할 뿐 특별히 하는 일은 없고, 결국은 외장 메모리처럼 LCD를 붙여 쓰는 것입니다.

여기서 한 가지 짚고 넘어가야 할 것이 있습니다. STM32H750VBT6은 핀 수가 적기 때문에 외부 메모리에 많은 핀을

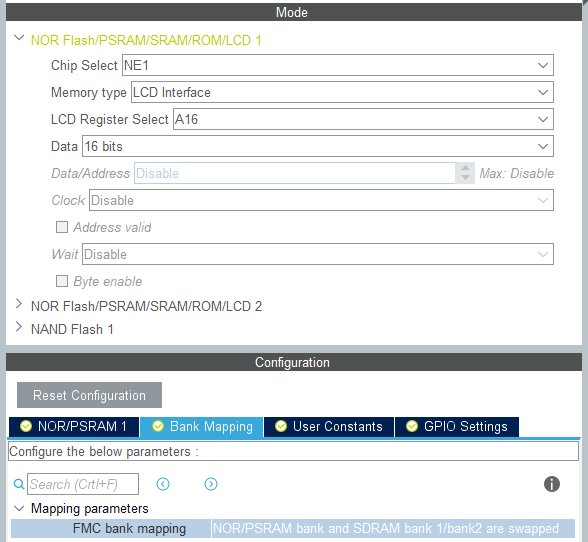
쓰기가 부담스럽습니다. 그래서 어드레스 버스와 데이터 버스를 공유하는 Muxed 방식을 쓰는데, LCD로서는 좀 헷갈릴 수

있게 되는 것입니다. 여러 가지 해결 방법이 있겠지만 간단한 방법 중 하나는 CubeMX 설정할 때

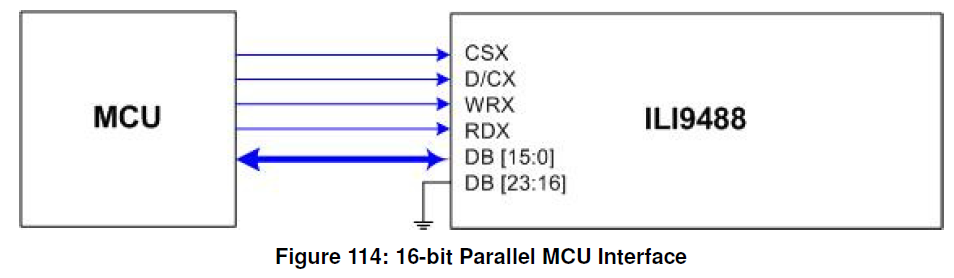
아래 그림처럼 FMC -> Bank Mapping 에서 "NOR/PSRAM bank and SDRAM bank 1/ bank2 are swapped" 를 선택하면

기본 매핑 주소인 0x60000000 대신에 0xC0000000에 매핑 되는데, 이 주소에서는 LCD가 이상없이 동작합니다.

아래 그림은 CubeMX에서 설정하는 예를 보여줍니다.



아래의 그림은 패러랠 인터페이스 방식(CPU BUS)의 도식입니다.



SPI 방식과 마찬가지로 데이터시트에서 통신 속도를 확인하시고 CubeMX에서 읽기, 쓰기 타이밍을 잘 설정해 주어야 합니다.

1. **RGB 인터페이스**

우리의 STM32H750VBT6은 LTDC를 가지고 있음에도 불구하고 핀 수가 적어서 외부에 SDRAM을 확장할 수 없습니다.

그래서 연속으로 사용할 수 있는 512Kbytes 에 프레임 버퍼를 잡을 수 있는 320 x 480 정도 해상도의 LCD가

RGB 인터페이스를 쓸 수 있는 사실상 최대치가 됩니다.

RGB 인터페이스를 쓰기 위해서는 크게 두 가지 설정이 필요합니다. 하나는 SPI를 통하여 LCD 내부 Configuration 을

하는 것입니다. 보통 void LCD\_Init(void); 라는 함수에서 초기화 데이터를 보내는 것으로 처리 합니다.

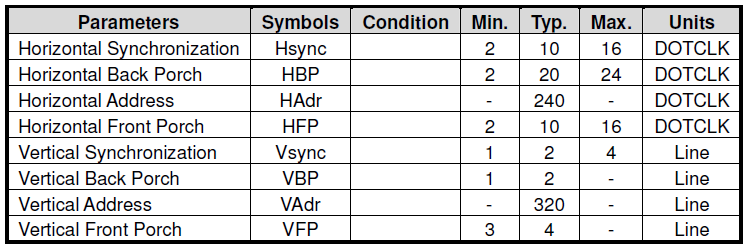
다른 하나는 STM32의 LTDC를 설정하는 것입니다. 물론 필요한 I/O 핀들은 제대로 연결되어 있어야 합니다.

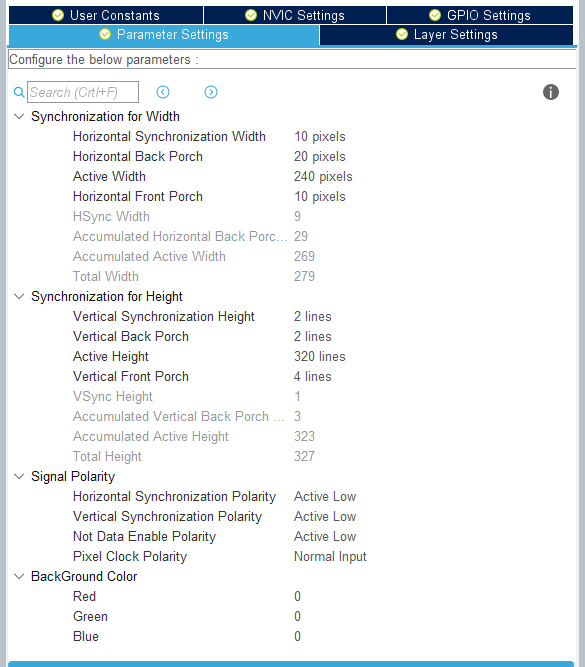
데이터 시트를 보고 최대 속도를 넘지 않도록 클럭을 설정해야 합니다. 예로 ILI9341의 DOTCLK cycle time 의 최소값이

100ns 이므로 CubeMX의 클럭 설정화면에서 To LTDC(MHz) 항목이 10을 넘기지 않게 설정합니다.

다음은 각 타이밍을 설정해야 하는데 아래는 ILI9341의 데이터시트 일부와 이를 참고로 CubeMX의 LTDC 파라메터를

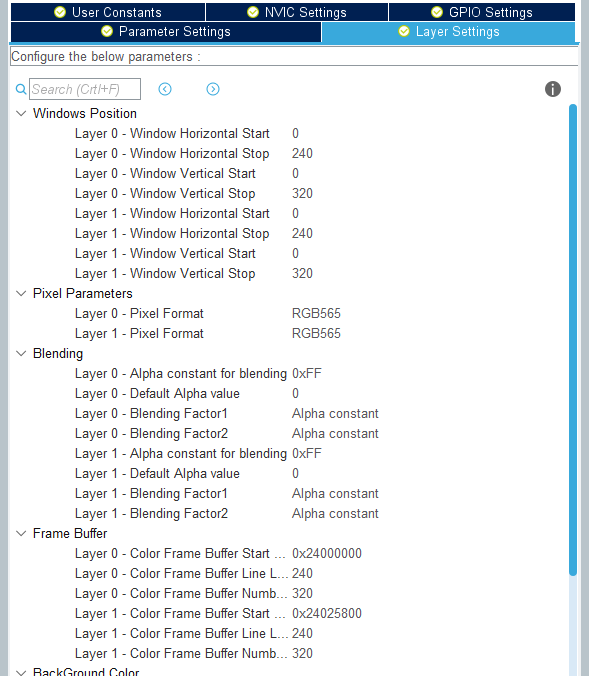
설정한 예입니다.





다음은 LTDC에서 LCD의 해상도와 프레임버퍼 주소에 맞게 Layer를 설정하면 됩니다. 아래는 ILI9341를 사용한

LCD를 내부 메모리에 2개의 Layer를 사용하도록 설정한 예입니다.



1. **개발 키트 선택**

글재주 없는 사람이 이 문서를 만든 것은 저희가 만든 LCD 개발을 돕기 위한 몇 가지의 개발 키트 중 선택에 도움을 드리기

위해서입니다. 저희 URAETECH에서는 흔히 구할 수 있는 LCD들을 이용해 앞에서 설명한 LCD 구동 방식을

저희의 STM32H750VBT6 개발 보드에 붙여서 테스트해 볼 수 있는 몇 가지의 제품을 준비했습니다.

저희 GitHub(<https://github.com/uraetech>)에 공개해 놓은 소스와 자료를 보시고 참고하시기 바랍니다.

LCD 컨트롤러는 ILI9341, ILI9488, ST7796 이렇게 3가지를 선택해서 6가지 보드를 만들었습니다.

이 보드는 저희 회사의 STM32H750 Demo board 에 딸보드로 꼽아서 사용할 수 있습니다.

각 특징은 다음과 같습니다. 참고로 CPU 버스 보드에서는 저희가 곧 출시할 OV2640 카메라 보드를 동시에 연결하여

카메라 영상을 LCD로 보면서 테스트할 수 있습니다.

1. **ILI9341 CPU**

* 해상도 320 x 240
* CPU BUS 통신 모드
* 터치스크린 없음.
* 카메라 보드 동시 연결 가능

1. **ILI9341 SPI/RGB**

* 해상도 320 x 240
* SPI 또는 RGB 인터페이스를 테스트할 수 있음.
* 터치스크린 없음.
* 카메라 보드 동시 연결 불가능

1. **ILI9488 CPU**

* 해상도 480 x 320
* CPU BUS 통신 모드
* 터치스크린 있음.
* 카메라 보드 동시 연결 가능

1. **ILI9488 SPI**

* 해상도 480 x 320
* SPI 통신 모드
* 터치스크린 있음.
* RGB 모드 불가능

1. **ST7796 CPU**

* 해상도 480 x 320
* CPU BUS 통신 모드
* 터치스크린 있음.
* 카메라 보드 동시 연결 가능

1. **ST7796 SPI/RGB**

* 해상도 480 x 320
* SPI 또는 RGB 인터페이스를 테스트할 수 있음.
* 터치스크린 있음.
* 카메라 보드 동시 연결 불가능