

Mango-AM335x 10.4 인치 감압식 LCD display 및 터치 보정하기

<http://www.mangoboard.com/>

<http://cafe.naver.com/embeddedcrazyboys>

Crazy Embedded Laboratory



Document History

Revision	Date	Change note
Init	2021-07-06	전종인

1. 소스 다운로드 및 컴파일.....	5
2. LCD display Timming 변경	5
3. 터치 드라이버 포팅	6
4. 커널 이미지 Write하기.....	8
5. QT 어플 example 테스트	9
6. 터치 및 display 테스트 방법.....	9

1. 소스 다운로드 및 컴파일

<http://crztech.iptime.org:8080/Release/mango-am335x-ST/Linux/>

에서 최종 릴리즈 소스를 다운로드 합니다.

```
$ wget http://crztech.iptime.org:8080/Release/mango-am335x-ST/Linux/kernel4.4/20200702/m3358_buildroot_k4419-20200702.tgz
```

이미지도 다운로드 한다.

```
$ wget http://crztech.iptime.org:8080/Release/mango-am335x-ST/Linux/kernel4.4/20200702/m3358_buildroot_k4419-20200702-image.tgz
```

압축 풀기

```
$ tar xf m3358_buildroot_k4419-20200702.tgz
```

```
$ tar xf m3358_buildroot_k4419-20200702-image.tgz
```

```
$ mv m3358_buildroot_k4419-20200702-image m3358_buildroot_k4419-20200702/image
```

```
$ ./build_kernel defconfig m3358_audio_defconfig
```

```
$ ./build_kernel
```

2. LCD display Timming 변경

" arch/arm/boot/dts/am335x-evmsk.dts " 수정

```
#if 1 //crazyboys 20210706 10.4inch
    display-timings {
        800x600 {
            hactive      = <800>;
            vactive      = <600>;
            hback-porch  = <46>;
            hfront-porch = <210>;
            hsync-len    = <1>;
            vback-porch  = <23>;
            vfront-porch = <12>;
            vsync-len    = <1>;
            clock-frequency = <40000000>;
            hsync-active = <0>;
            vsync-active = <0>;
        }
    }
#endif
```

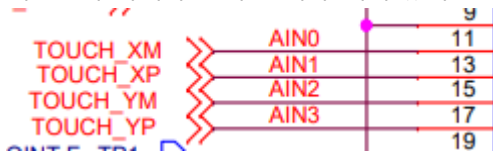
```

        };
};
#endif

```

3. 터치 드라이버 포팅

회로도예 아래와 같이 정의가 되어 있다.



코드가 아래와 같이 되어 있으므로

```

static int titsc_config_wires(struct titsc *ts_dev)
{
    u32 analog_line[4];
    u32 wire_order[4];
    int i, bit_cfg;

    for (i = 0; i < 4; i++) {
        /*
         * Get the order in which TSC wires are attached
         * w.r.t. each of the analog input lines on the EVM.
         */
        analog_line[i] = (ts_dev->config_inp[i] & 0xF0) >> 4;
        wire_order[i] = ts_dev->config_inp[i] & 0x0F;
        if (WARN_ON(analog_line[i] > 7))
            return -EINVAL;
        if (WARN_ON(wire_order[i] > ARRAY_SIZE(config_pins)))
            return -EINVAL;
    }

    for (i = 0; i < 4; i++) {
        int an_line;
        int wi_order;

```

```

        an_line = analog_line[i];
        wi_order = wire_order[i];
        bit_cfg = config_pins[wi_order];
        if (bit_cfg == 0)
            return -EINVAL;
        switch (wi_order) {
        case 0:

            ts_dev->bit_xp = bit_cfg;
            ts_dev->inp_xp = an_line;

            break;

        case 1:
            ts_dev->bit_xn = bit_cfg;
            ts_dev->inp_xn = an_line;

            break;

        case 2:
            ts_dev->bit_yp = bit_cfg;
            ts_dev->inp_yp = an_line;

            break;

        case 3:
            ts_dev->bit_yn = bit_cfg;
            ts_dev->inp_yn = an_line;

            break;

        }
    }
    return 0;
}

```

" arch/arm/boot/dts/am335x-evmsk.dts "

```

&tscadc {
    status = "okay";
}

```

```

tsc {
    ti,wires = <4>;
    ti,x-plate-resistance = <200>;
    ti,coordinate-readouts = <5>;
    ti,wire-config = <0x01 0x10 0x23 0x32>;
    ti,charge-delay = <0x400>;
};
};

```

CONFIG_TOUCHSCREEN_TI_AM335X_TSC=y

CONFIG_MFD_TI_AM335X_TSCADC=y

4. 커널 이미지 Write하기

커널 컴파일 시 image 디렉토리는 아래와 같이 존재를 해야 합니다.

```

$ ls
app  buildroot-2013.11  image  kernel  mango_build_clean.sh  uboot

$ ls image/
boot-images  boot_tar_make.sh  custom_datafs          custom_datafs_make.sh  rootfs.tar.gz
boot.tar.gz  create-sdcard.sh  custom_datafs.tar.gz  rootfs.tar

```

Kernel 이미지는 custom_datafs/boot 디렉토리에 복사가 된다.

```
$ ./custom_datafs_make.sh
```

실행 후 custom_datafs.tar.gz 파일이 생성이 된다.

PC에 Micro SD를 삽입 후 Write를 한다. 디바이스 노드를 확인 한다.

```
$ sudo ./create-sdcard-ubuntu16.06.sh /dev/sdc
```

전체 이미지가 Write된다.

커널 이미지만 Write하기

```

$ cd image
$ mkdir fs
$ sudo mount /dev/sdc2 fs
$ sudo cp custom_datafs/boot/* fs/boot/
$ sudo umount fs

```


5. QT 어플 example 테스트

```
[root@localhost ~]# cat /proc/bus/input/devices
I: Bus=0000 Vendor=0000 Product=0000 Version=0000
N: Name="ti-tsc"
P: Phys=
S: Sysfs=/devices/platform/ocp/44e0d000.tscadc/TI-am335x-tsc/input/input0
U: Uniq=
H: Handlers=mouse0 event0
B: PROP=0
B: EV=b
B: KEY=400 0 0 0 0 0 0 0 0 0
B: ABS=1000003
```

```
export QWS_MOUSE_PROTO="tslib:/dev/input/event0"
export TSLIB_TSEVENTTYPE=INPUT
export TSLIB_TSDEVICE=/dev/input/event0
export TSLIB_CALIBFILE=/etc/pointercal
export TSLIB_CONFFILE=/etc/ts.conf
export TSLIB_PLUGINDIR=/usr/lib/ts
export LD_LIBRARY_PATH=/usr/lib:/lib
export QWS_SIZE=800*600
export QWS_DISPLAY=Transformed:Rot270:VNC:LinuxFb
```

6. 터치 및 display 테스트 방법

<터치 테스트>

```
# ts_calibrate
xres = 800, yres = 600
[ 43.868857] random: nonblocking pool is initialized
Took 62 samples...
Top left : X = 3815 Y = 3686
Took 66 samples...
Top right : X = 357 Y = 3657
Took 46 samples...
Bot right : X = 341 Y = 437
Took 61 samples...
Bot left : X = 3766 Y = 419
```

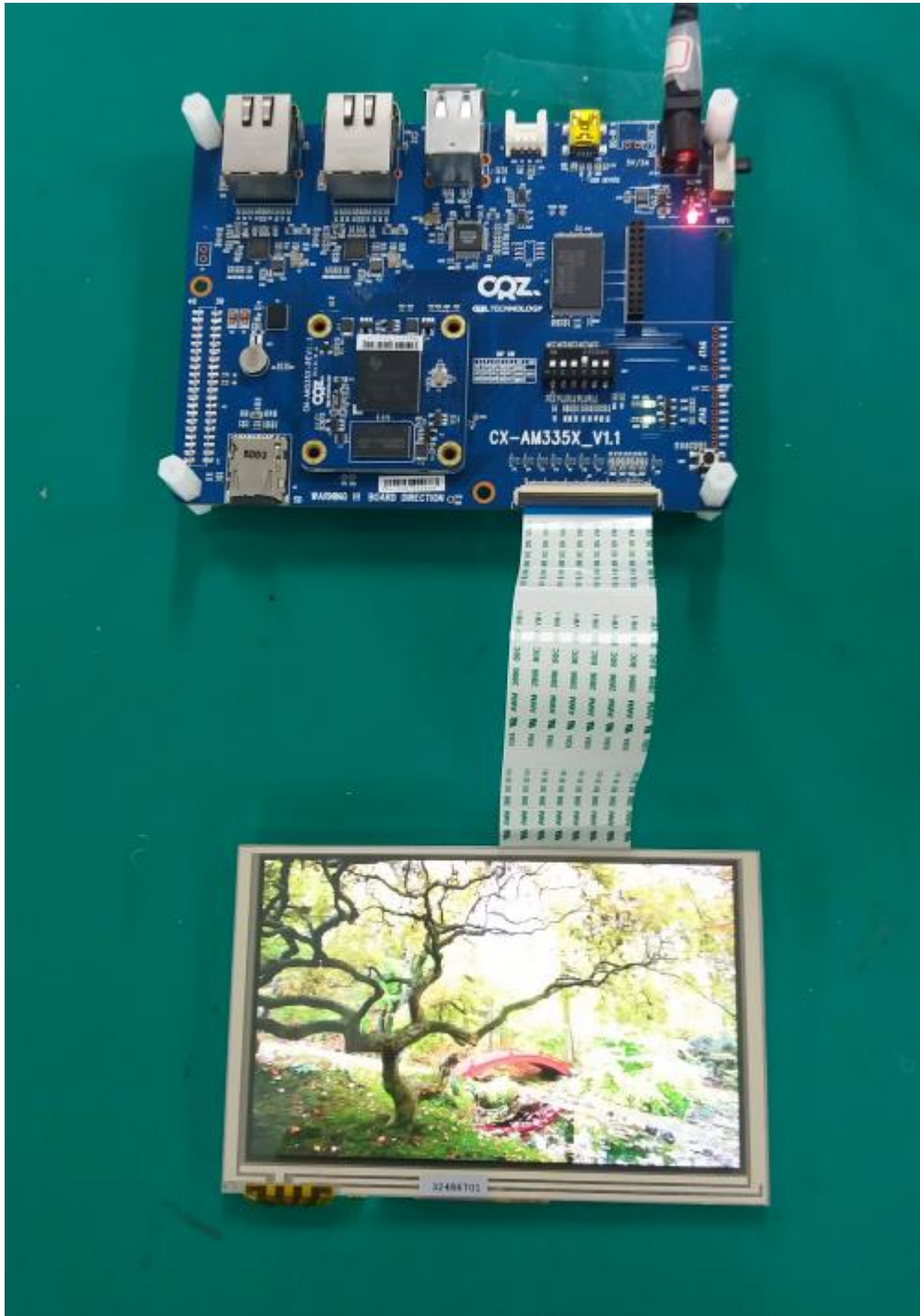
```
Took 60 samples...  
Center : X = 2059 Y = 2062  
816.349548 -0.203397 0.002043  
615.823853 0.000250 -0.154147  
Calibration constants: 53500284 -13329 133 40358632 16 -10102 65536
```

```
[root@localhost ~]# ts_test
```

```
[root@localhost ~]# fb-test  
fb-test 1.0.0 (tablet_rosa)  
fb res 800x600 virtual 800x600, line_len 3200  
dim -1mm x -1mm
```

이미지를 삽입해서 fbv명령을 이용하여 이미지를 display 할 수 있다.

```
#fbv 1.jpg
```



사용법은 아래와 같습니다.

```
[root@localhost ~]# fbv
```

Usage: fbv [options] image1 image2 image3 ...

Available options:

--help | -h : Show this help
--alpha | -a : Use the alpha channel (if applicable)
--dontclear | -c : Do not clear the screen before and after displaying the image
--donthide | -u : Do not hide the cursor before and after displaying the image
--noinfo | -i : Suppress image information
--stretch | -f : Stretch (using a simple resizing routine) the image to fit onto screen if necessary
--colorstretch | -k : Stretch (using a 'color average' resizing routine) the image to fit onto screen if necessary
--enlarge | -e : Enlarge the image to fit the whole screen if necessary
--ignore-aspect | -r : Ignore the image aspect while resizing
--delay <d> | -s <delay> : Slideshow, 'delay' is the slideshow delay in tenths of seconds.

Keys:

r : Redraw the image
a, d, w, x : Pan the image
f : Toggle resizing on/off
k : Toggle resizing quality
e : Toggle enlarging on/off
i : Toggle respecting the image aspect on/off
n : Rotate the image 90 degrees left
m : Rotate the image 90 degrees right
p : Disable all transformations

Copyright (C) 2000 - 2004 Mateusz Golicz, Tomasz Sterna.

