

Handle one USB port on CCP XA/XS.

Note that CCPilot XA/XS standard models have one USB port while Net+ and CrossCore XS models have two USB host port. (From CCP XA/XS TechManual!)

Use command “fdisk” to check if USB is inserted!

No USB inserted gives the following output:

```
#
# fdisk -l

Disk /dev/mmcblk0: 4012 MB, 4012900352 bytes
4 heads, 16 sectors/track, 122464 cylinders, total 7837696 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00069bae

    Device Boot      Start         End      Blocks   Id  System
/dev/mmcblk0p1        12288        77823       32768    83   Linux
/dev/mmcblk0p2         88064       600063      256000    83   Linux
/dev/mmcblk0p3       600064       7837695     3618816    83   Linux
#
```

When an USB is inserted in CCP XS, you get the following output:

```
# fdisk -l

Disk /dev/mmcblk0: 4012 MB, 4012900352 bytes
4 heads, 16 sectors/track, 122464 cylinders, total 7837696 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00069bae

    Device Boot      Start         End      Blocks   Id  System
/dev/mmcblk0p1        12288        77823       32768    83   Linux
/dev/mmcblk0p2         88064       600063      256000    83   Linux
/dev/mmcblk0p3       600064       7837695     3618816    83   Linux

Disk /dev/sda: 503 MB, 503316480 bytes
4 heads, 51 sectors/track, 4818 cylinders, total 983040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x4ca5e7e9

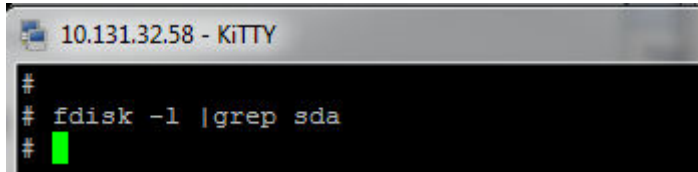
    Device Boot      Start         End      Blocks   Id  System
/dev/sda1             504         983039      491268    6   FAT16
#
```

Proposed solution:

When there is no USB inserted and you use the command:

```
# fdisk -l | grep /dev/sda
```

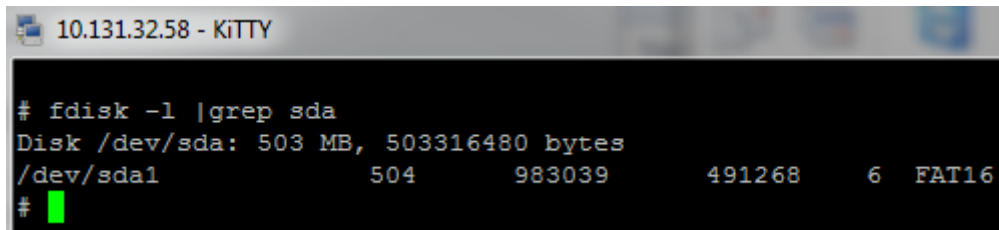
You get the following result:



```
10.131.32.58 - KITTY
#
# fdisk -l | grep sda
#
```

But if you insert an USB and then use the same command:

```
# fdisk -l | grep /dev/sda
```



```
10.131.32.58 - KITTY
# fdisk -l | grep sda
Disk /dev/sda: 503 MB, 503316480 bytes
/dev/sda1          504      983039      491268      6  FAT16
#
```

This fact could be used within your application code, as could be seen in the following example!

Code example:

```
#define + MAX_NUM_DEVICES 7
void check_usb()
{
    char i;
    char cmd[256];

    for (i = 0x61; i < (0x61 + MAX_NUM_DEVICES); i++)
    {
        // Check if device /dev/sdx1 exists
        sprintf(cmd, "fdisk -l | grep /dev/sd%c", i);
        if (system(cmd) == 0)          // NOTE! 0(zero) means NO error!
        {
            // Check if device is mounted anywhere
            sprintf(cmd, "mount | grep /dev/sd%c", i);
            if (system(cmd) == 0)          // NOTE! 0(zero) means NO error!
            {
                // Device is mounted
                printLog("Found device /dev/sd%c, device is already mounted.\n");
                // The USB is present! Use it as you wish!

                ...
            }
            else
            {
                ...
            }
        }
    }
}
```

Note!

You need to call this function "check_usb()" in a loop somewhere in your program...

When the program detects that a USB is mounted – do your thing!