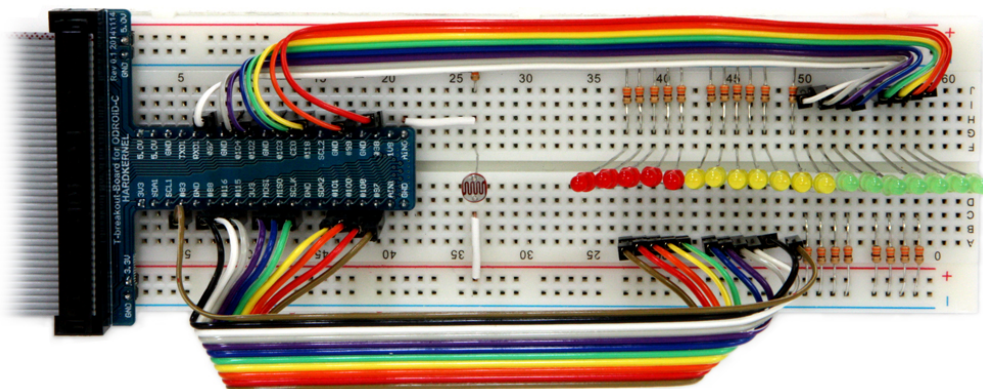


Introduction C Tinkering Kit on Ubuntu

- This works fine on **ODROID-XU4**, **ODROID-N2** as well.



You're probably itching to make some fun embedded computer projects with **ODROID-C**. What you need is an add on prototyping T-breakout board, which can break out all those tasty power, GPIO, I2C, ADC pins from the 40 pin header onto a solderless breadboard. This set will make “cobbling together” prototypes with the **ODROID-C** super easy.

This kit comes with below many items.

- Assembled T-breakout PCB - 40Pin GPIO Breakout board
- Breadboard - 630 Tie-points with dual power lanes
- 40pin Ribbon cable - IDC Flat cable 100mm
- 40pin Male-to-Male Dupont jumper Wire 170mm
- 7 x Green LED 3mm
- 7 x Yellow LED 3mm
- 7 x Red LED 3mm
- 2 x Photo Cell (*cds* Light sensor)
- 6 x Tact Switches
- 50 x 330 Ohm 1/6W resistor
- 50 x 10K Ohm 1/6W resistor

[Where to buy](#)

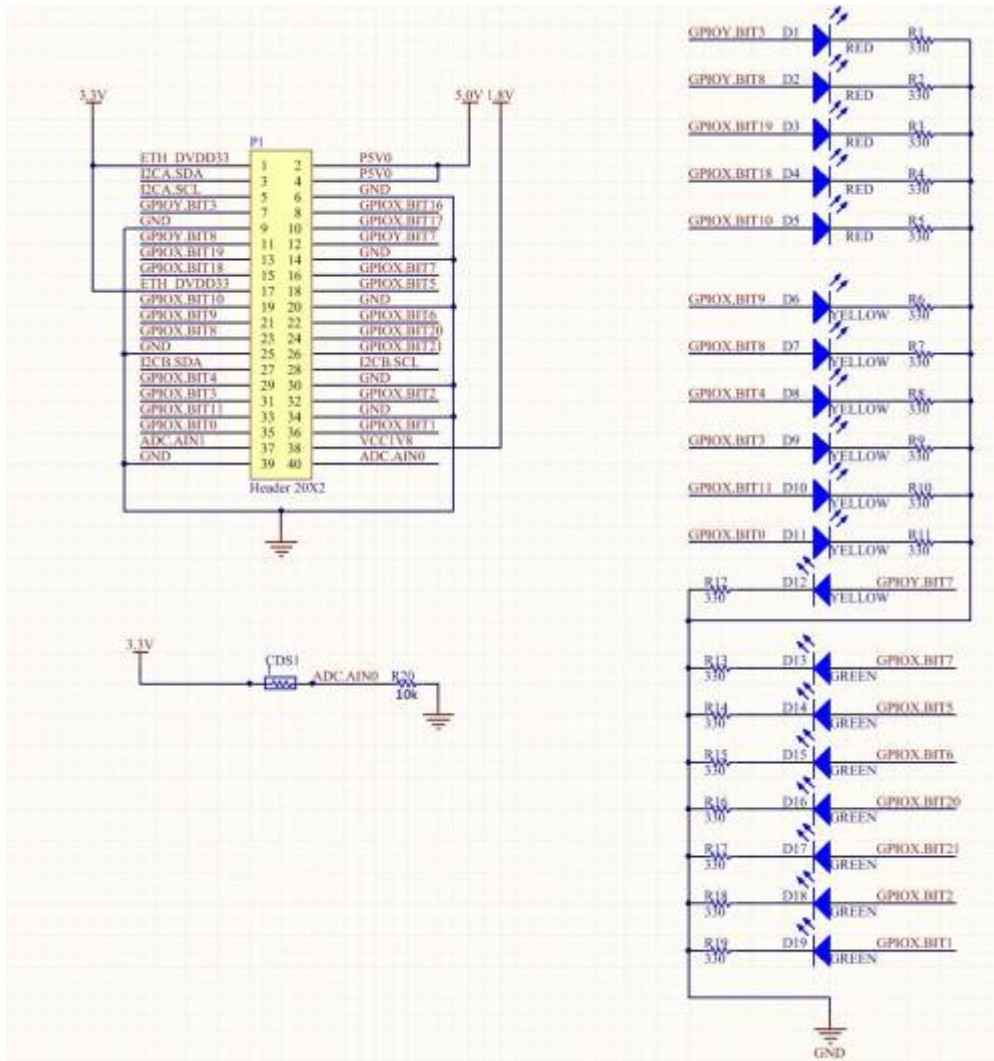
DIY light level meter project

Configuration tinkering kit such as below schematic.

Please refer to below links for ODROID-C2 user or details related to gpio mapping.

[C1 Expansion Connectors](#)

[C2 Expansion Connectors](#)



Linux

C example (With WiringPi)

1. Install our [WiringPi](#). Please refer to this document: [WiringPi and Python Wrapper](#)
2. Compile and run the example source code.

```
$ wget http://dn.odroid.com/source_peripherals/ctinkeringkit/example-led.c
$ gcc -o example-led example-led.c -lwiringPi -lwiringPiDev -lm -pthread -lrt -lcrypt
$ sudo ./example-led
```

example-led.c

```
//-----
//
// ODR0ID-C GPIO(LED) / ADC Test Application.
//
```

```
// Defined port number is wiringPi port number.
//
// Compile : gcc -o <create excute file name> <source file name> -lwiringPi
-lwiringPiDev -lpthread
// Run : sudo ./<created excute file name>
//
//-----
-----
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>

#include <unistd.h>
#include <string.h>
#include <time.h>

#include <wiringPi.h>
#include <wiringPiI2C.h>
#include <wiringSerial.h>
#include <lcd.h>

//-----
-----
//
// Global handle Define
//
//-----
-----

#define DATA_UPDATE_PERIOD 100 // 100ms

//-----
-----
//
// ADC:
//
//-----
-----

#define PORT_ADC1 0 // ADC.AIN0

static int adcValue = 0;

//-----
-----
//
// LED:
//
//-----
-----

static int ledPos = 0;
```

```
const int ledPorts[] = {
    24,
    23,
    22,
    21,
    14,
    13,
    12,
    3,
    2,
    0,
    7,

    1,
    4,
    5,
    6,
    10,
    11,
    26,
    27,
};

#define MAX_LED_CNT sizeof(ledPorts) / sizeof(ledPorts[0])

//-----
//-----
//-----
//
// system init
//
//-----
//-----

int system_init(void)
{
    int i;

    // GPIO Init(LED Port ALL Output)
    for(i = 0; i < MAX_LED_CNT; i++)    pinMode (ledPorts[i], OUTPUT);

    return 0;
}

//-----
//-----
//
// board data update
//
//-----
```

```
-----  
void boardDataUpdate(void)  
{  
    int i;  
  
    // adc value read  
    if((adcValue = analogRead (PORT_ADC1))    {  
        ledPos = (adcValue * MAX_LED_CNT * 1000) / 1024;  
        ledPos = (MAX_LED_CNT - (ledPos / 1000));  
    }  
    else  
        ledPos = 0;  
  
    // LED Control  
    for(i = 0; i < MAX_LED_CNT; i++)    digitalWrite (ledPorts[i], 0); //  
LED All Clear  
    for(i = 0; i < ledPos;    i++)    digitalWrite (ledPorts[i], 1); //  
LED On  
}  
  
//-----  
-----  
//  
// Start Program  
//  
//-----  
-----  
int main (int argc, char *argv[])  
{  
    static int timer = 0 ;  
  
    wiringPiSetup ();  
  
    if (system_init() < 0)  
    {  
        fprintf (stderr, "%s: System Init failed\n", __func__);    return  
-1;  
    }  
  
    for(;;)    {  
        usleep(100000);  
        if (millis () < timer)    continue ;  
  
        timer = millis () + DATA_UPDATE_PERIOD;  
  
        // All Data update  
        boardDataUpdate();  
    }  
  
    return 0 ;  
}
```

```
//-----  
-----  
//-----  
-----
```

Python example

1. Install our [WiringPi Python Wrapper](#). Please refer to this document: [WiringPi Python Wrapper](#)
2. Get/Run the example source code

```
wget http://dn.odroid.com/source_peripherals/ctinkeringkit/example-led.py  
sudo python example-led.py
```

[example-led.py](#)

```
#!/usr/bin/python  
import wiringpi2 as wpi  
import time  
  
leds = [24, 23, 22, 21, 14, 13, 12, 3, 2, 0, 7, 1, 4, 5, 6, 10, 11, 26, 27]  
  
wpi.wiringPiSetup()  
  
# GPIO pin setup  
for x in leds:  
    wpi.pinMode(x, 1)  
    wpi.digitalWrite(x, 0)  
  
adc_unit = 4095 / len(leds)  
while True:  
    time.sleep(0.05)  
  
    # Read the adc value  
    adcValue = wpi.analogRead(0)  
    print adcValue  
  
    # Set the LEDs  
    ledPos = (adcValue * len(leds) * 1000) / 1024  
    ledPos = len(leds) - (ledPos / 1000)  
  
    for x in leds:  
        wpi.digitalWrite(x, 0)  
  
    for x in xrange(ledPos):  
        wpi.digitalWrite(leds[x], 1)
```

Android

Having fun with Tinkering kit GPIO on Android!

From:

<http://wiki.odroid.com/> - **ODROID Wiki**

Permanent link:

http://wiki.odroid.com/accessory/development/c_tinkering/c_tinkering

Last update: **2019/02/13 05:17**

