Decade Counter through Odroid-C2 GPIO

G V V Sharma*

Contents

***Abstract*—This manual provides a simple introduction for using the GPIO pins of the ODROID-C2 using the WIRINGPI library. This is done by displaying numbers on a seven segment display.**

1 Components

Component	Value	Quantity
Breadboard		1
Resistor	≥ 220Ω	1
Seven Segment Display		1
Jumper Wires		20

TABLE 1

2 Pin Connections

The pin connections between the ODROID and the seven segment display are available in Table 2. Please refer to Fig. 3.3 for details.

3 GPIO Programming

Problem 3.1. Run the following program on the odroid. This will blink the DOT pin of the display.

```
#include <wiringPi.h>
int main (void)
{
  wiringPiSetup () ;
  pinMode (13, OUTPUT) ;
  for (;;)
```

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in.

Type	GPIO Pin No
5V	2
GND	6
a	11
b	12
c	13
d	15
e	16
f	18
g	19
DOT	21

TABLE 2

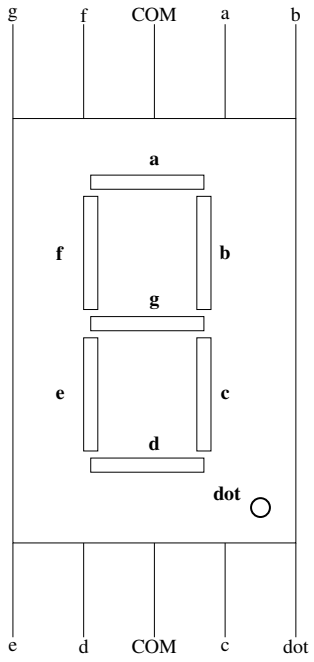

Fig. 2.0

```
  {
    digitalWrite (13, HIGH) ;
       delay (1000) ;
    digitalWrite (13,  LOW) ;
       delay (1000) ;

  }
  return 0 ;
}
//Run following statements on
   terminal
//gcc -Wall -o blink blink.c -
   lwiringPi
//followed by
// sudo ./blink
```

Problem 3.2. The following code generates the number 0 on the seven segment display. Use it to generate all the numbers from 1-9.

```
#include <wiringPi.h>
int main (void)
{
  wiringPiSetup () ;
  pinMode (0, OUTPUT) ;//a
  pinMode (1, OUTPUT) ;//b
  pinMode (2, OUTPUT) ;//c
  pinMode (3, OUTPUT) ;//d
  pinMode (4, OUTPUT) ;//e
  pinMode (5, OUTPUT) ;//f
  pinMode (12, OUTPUT) ;//g
  for (;;)
  {
    digitalWrite (0,  0) ;
        digitalWrite (1,  0) ;
    digitalWrite (2,  0) ;
        digitalWrite (3,  0) ;
    digitalWrite (4,  0) ;
        digitalWrite (5,  0) ;
        digitalWrite (12,  1) ;



  }
  return 0 ;
}
//Run following statements on
   terminal
//gcc -Wall -o 7seg 7seg.c -
   lwiringPi
//followed by
// sudo ./7seg
```

Problem 3.3. The following code implements the decade counter

```
#include <wiringPi.h>

int A=0,B=0,C=0,D=0,a,b,c,d,e,f,g,
   W,X,Y,Z,i,j;
int r0;
unsigned int initialtime, elapsed;
void showit(int x);

int main (void)
{
//Declarations
  wiringPiSetup () ;
  pinMode (0, OUTPUT) ;//a
  pinMode (1, OUTPUT) ;//b
  pinMode (2, OUTPUT) ;//c
  pinMode (3, OUTPUT) ;//d
  pinMode (4, OUTPUT) ;//e
  pinMode (5, OUTPUT) ;//f
  pinMode (12, OUTPUT) ;//g
  for (;;)
  {

//Decade Counting
for( r0=0;r0<=9;r0++)
{
initialtime=millis();
//Counting 1000 milliseconds
for(elapsed=0;elapsed<=1000;
   elapsed=millis()-initialtime)
{

//Write number to display
showit(r0);

}//end counting 10 sec
} //end counting 1 sec


    digitalWrite (0,  0) ;
        digitalWrite (1,  0) ;
    digitalWrite (2,  0) ;
        digitalWrite (3,  0) ;
    digitalWrite (4,  0) ;
        digitalWrite (5,  0) ;
        digitalWrite (12,  1) ;
```

```
  }
  return 0 ;
}
//Display logic
void showit(int x)
{
int D,C,B,A;

//Decimal to Binary conversion
A=x%2;
x=x/2;
B=x%2;
x=x/2;
C=x%2;
x=x/2;
D=x%2;

//BCD to seven segement decoder
a=(!D&&!C&&!B&&A)||(!D&&C&&!B&&!A)
    ;
b=(!D&&C&&!B&&A)||(!D&&C&&B&&!A);
c=(!D&&!C&&B&&!A);
d=(!D&&!C&&!B&&A)||(!D&&C&&!B&&!A)
    ||(!D&&C&&B&&A);
e=(!D&&!C&&!B&&A)||(!D&&!C&&B&&A)
    ||(!D&&C&&!B&&!A)||(!D&&C&&!B&&A
    )||(!D&&C&&B&&A)||(D&&!C&&!B&&A)
    ;
f=(!D&&!C&&!B&&A)||(!D&&!C&&B&&!A)
    ||(!D&&!C&&B&&A)||(!D&&C&&B&&A);
g=(!D&&!C&&!B&&!A)||(!D&&!C&&!B&&A
    )||(!D&&C&&B&&A);

//Writing to display
    digitalWrite (0, a) ;
        digitalWrite (1, b) ;
    digitalWrite (2, c) ;
        digitalWrite (3, d) ;
    digitalWrite (4, e) ;
        digitalWrite (5, f) ;
        digitalWrite (12, g) ;
}

//Run following statements on
   terminal
//gcc -Wall -o decade decade.c -
   lwiringPi
//followed by
// sudo ./decade
```

GPIO PIN-MAP

ODROID-C2 40pin Layout

Power Pin
Special Function
GPIO/Special Function

WiringPi GPIO#	Export GPIO#	ODROID-C2 PIN	Label	HEADER		Label	ODROID-C2 PIN	Export GPIO#	WiringPi GPIO#
			3V3	1	2	5V0			
	205	I2CA_SDA	SDA1	3	4	5V0			
	206	I2CA_SCL	SCL1	5	6	GND			
7	249	GPIOX.BIT21	#249	7	8	TXD1	TXD_B	113	
			GND	9	10	RXD1	RXD_B	114	
0	247	GPIOX.BIT19	#247	11	12	#238	GPIOY.BIT10	238	1
2	239	GPIOX.BIT11	#239	13	14	GND			
3	237	GPIOX.BIT9	#237	15	16	#236	GPIOX.BIT8	236	4
			3V3	17	18	#233	GPIOX.BIT5	233	5
12	235	GPIOX.BIT7	#235	19	20	GND			
13	232	GPIOX.BIT4	#232	21	22	#231	GPIOX.BIT3	231	6
14	230	GPIOX.BIT2	#230	23	24	#229	GPIOX.BIT1	229	10
			GND	25	26	#225	GPIOY.BIT14	225	11
	207	I2CB_SDA	SDA2	27	28	SCL2	I2CB_SCL	77	
21	228	GPIOX.BIT0	#228	29	30	GND			
22	219	GPIOY.BIT8	#219	31	32	#224	GPIOY.BIT13	224	26
23	234	GPIOX.BIT6	#234	33	34	GND			
24	214	GPIOY.BIT3	#214	35	36	#218	GPIOY.BIT7	218	27
		ADC.AIN1	AIN1	37	38	1V8	1V8		
			GND	39	40	AIN0	ADC.AIN0		

Fig. 3.3