parallax propeller p1(P8X32A-D40) make your own from scratch

Author:- Alok Kumar Mishra (akmishra_99@yahoo.com) Jan 21,2021

parallax’s propeller (p1) has eight core (called cogs) in it, and it is very easy to make your own board with few components and EEPROM,here is list of components

(1) propeller p1 (40 PIN DIP)

(2) propeller plug, (https://www.parallax.com/product/prop-plug-rev-d or Prop Plug (#32201))

(3) push button switch , and

(4) two 330 Ohm resistor ,

(5) one 10 K Ohms resistor

(6) one 5 Mhz crystal .

(7) 512Kbit (64KB) EEPROM or 32KB , 256Kbit EEPROM

(8) PIN headers

(9) three 0.1 micro farad capacitor

following page shows complete schematics this schematic is based on manual from parallax (https://www1.parallax.com/sites/default/files/downloads/P8X32A-Web-PropellerManual-v1.2.pdf page 17)

I made this schematic as minimal as possible, yes EEPROM and crystal are optional but I wanted to have persistence in program after boot up.

in figure 1 (minimal p1 schematic) there are two LEDs,one is power LED (LED1) which is always on when 3.3 V DC is applied to the board , other LED (LED2) is attached to port P0 of p1 chip. there are current limiting resistors (R1,R2 of 330 Ohms each).

There are four pin headers JP1 to JP4, pin header JP1 is used to apply power 3.3V and GND, DTR of pin header JP1 is connected to capacitor C1 (0.1 Micro Farad) and other end of capacitor C1 is connected to p1 (P8X32A-D40) RESET Pin 11 (RESET or RES), low on Pin 11 resets the p1,also push button SW1 is connected to Pin 11 of p1, pressing switch will reset p1.

other two pins of pin header JP1 is Tx and Rx (for serial communications from host to P1)

Pin header JP2 and JP3 brings all the port pins of p1 (P8X32A-D40)

crystal Q1 (5 Mhz) is attached to pin 30 (X1) and pin 31 (X0), this provides stable clock to p1, it is possible to use internal oscillator of chip p1, but from simpleIDE (GUI program to program this chip. I did not had success in loading program using internal oscillator) , this 5 Mhz is multiplied by internal PLL and 80 Mhz clock is generated in p1

C2 and C3 (0.1uF) are bypass capacitor for p1 and EEPROM.

C4 is 100uF capacitor to make power supply stable for p1 (to minimize ripple)

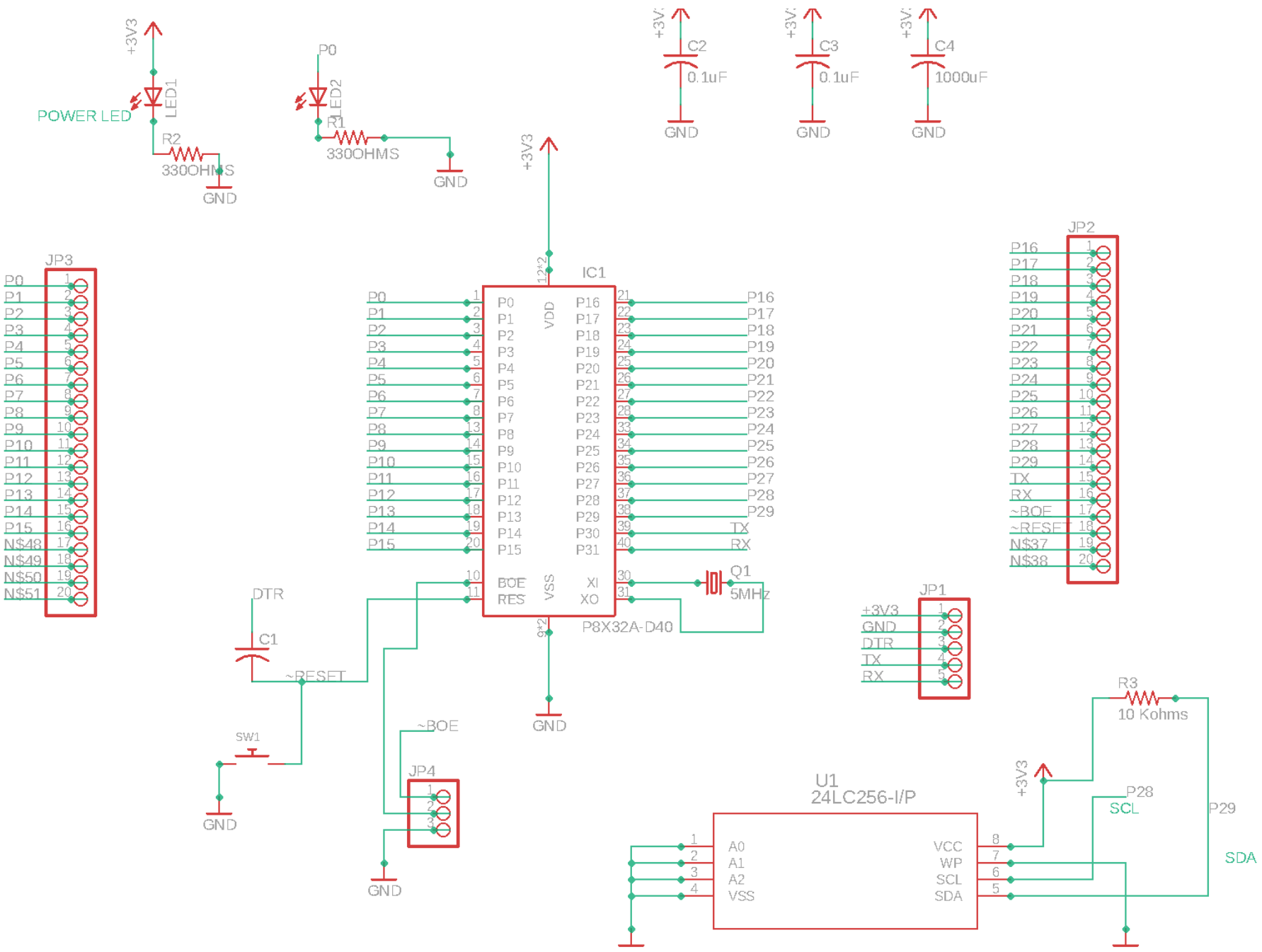

Figure 1 minimal p1 schematic

U1 (24LC256-I/P) is 32 Kilo byte (32 KB) EEPROM, it is possible to substitute (64 KB EEPROM) in this circuit, I replaced 32KB EEPROM with 64KB EEPROM and it worked.

JP4 is used to connect pin 10 (BOE) to ground or any other voltage to enable Burn out reset.

U1 is EEPROM (24LC256), i2c is used to communicate with it, resistor R3 (10 Kilo Ohms) is pull up resistor connected to SDA (pin 5 of EEPROM) which is connected to P29 of p1 and SCL (pin 6 of EEPROM) is connected to P28 of p1.

now let us make connections to propeller plug and p1 board which we made , connect VSS of propeller plug to GND of jumper JP1 , connect Rx of propeller plug (USB to serial TTL level adapter) to Tx of JP1, connect Rx of propeller plug to Tx of JP1, connect RES of propeller plug to DTR of JP1.

connect a 3.3v power supply (https://www.amazon.com/gp/product/B01LZVRP83/ref=ppx_yo_dt_b_asin_title_o05_s00?ie=UTF8&psc=1) to 3.3v pin of JP1 and connect ground of 3.3v power supply to ground of JP1

power LED should turn on indicating power to board/circuit is applied

parallax provides simpleIDE for programming their p1 chip in C language

here is code which we are going to use to blink LED2 attached to port P0

```
/*
  Blank Simple Project.c
  http://learn.parallax.com/propeller-c-tutorials
*/
#include "simpletools.h"                      // Include simple tools

int main()                                    // Main function
{
  // Add startup code here.
  while(1)
  {
    print("hello there\n");
    // Add main loop code here.
    high(0);                                  // Set P0 I/O pin high
```

```
    pause(100);                     // Wait 1/10 second


    low(0);                     // Set P0 I/O pin low
    pause(100);
  }
}
```

here is picture of simpleIDE with above code entered(led_blink_P0.c)

led_blink_P0.c

```
/*
  Blank Simple Project.c
  http://learn.parallax.com/propeller-c-tutorials
*/
#include "simpletools.h"                      // Include simple tools

int main()                                    // Main function
{
  // Add startup code here.


  while(1)
  {
    print("hello there\n");
    // Add main loop code here.
    high(0);                                 // Set P0 I/O pin high
    pause(100);                               // Wait 1/10 second

    low(0);                                  // Set P0 I/O pin low
    pause(100);

  }
}
```

Figure 2 simpleIDE with code to blink LED2 on port P0

in the above program we are printing “hello there” on serial port of p1 board and then we are turning LED2 (high(0)) on and then we are waiting for 1/10 th (pause(100)) of second then turning LED2 off(low(0)) , then waiting for 1/10th (pause(100)) of seconds and then repeating all over in infinite loop (while(1))

figure 3

click on right most icon in above picture (Run with terminal), simple IDE will compile and download program to RAM of p1 , if everything goes well LED2 on p1 board will start blinking and a terminal window will popup with “hello there” printed in it.

running led_blink_P0.c code from EEPROM

to run from EEPROM click on button (third from right) Load “EEPROM and Run”, when download is successful , remove power from p1 board and reconnect power and you”ll see LED2 blinking

to know about power up and boot loader invocation(boot up procedure) , please see https://www1.parallax.com/sites/default/files/downloads/P8X32A-Web-PropellerManual-v1.2.pdf page 18

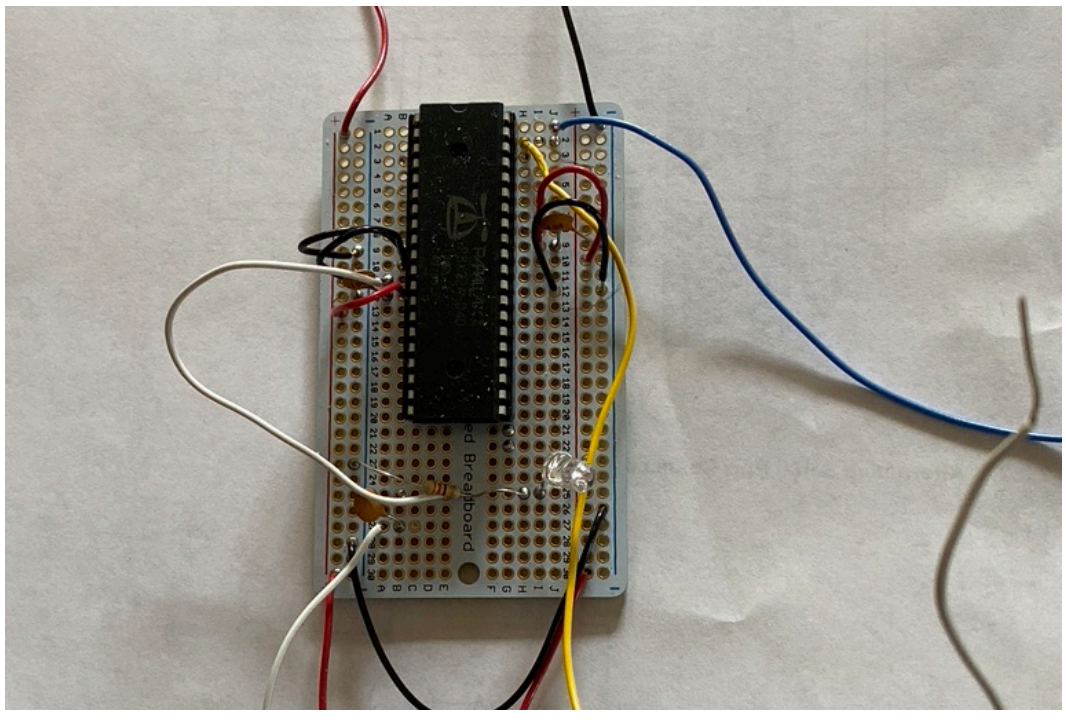

Figure 4 , p1 soldered on perf board

I soldered these components on half size perf board before making PCB (I made sure that my circuit and tools simpleIDE which I installed on my mac, works)

a picture of it is shown in Figure 4 left.

when my circuit worked , I designed PCB for it , here is picture of PCB with top and bottom

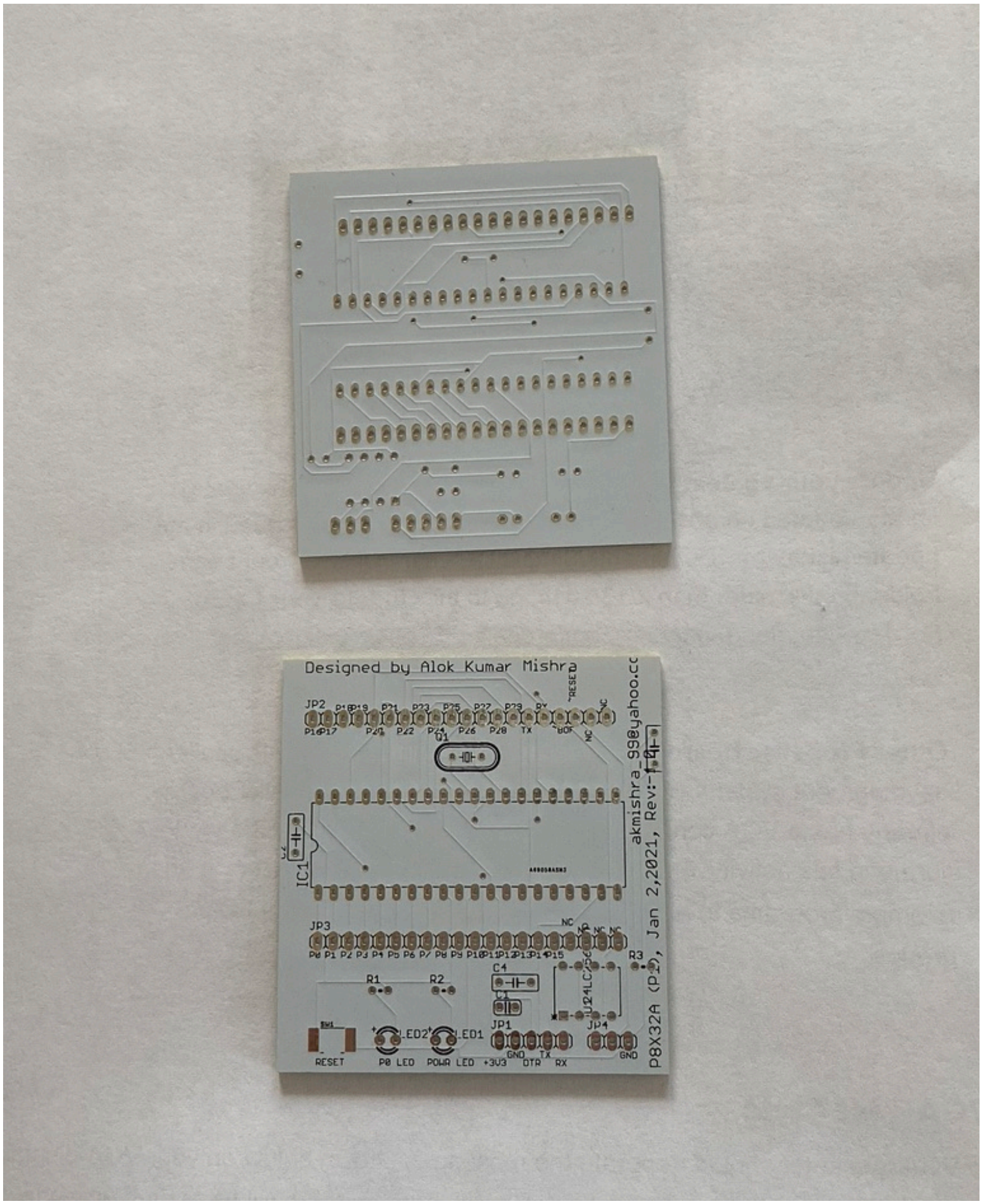

Figure 5 , PCB bottom in above and top side in below

here is picture with all components (except reset button SW1) soldered

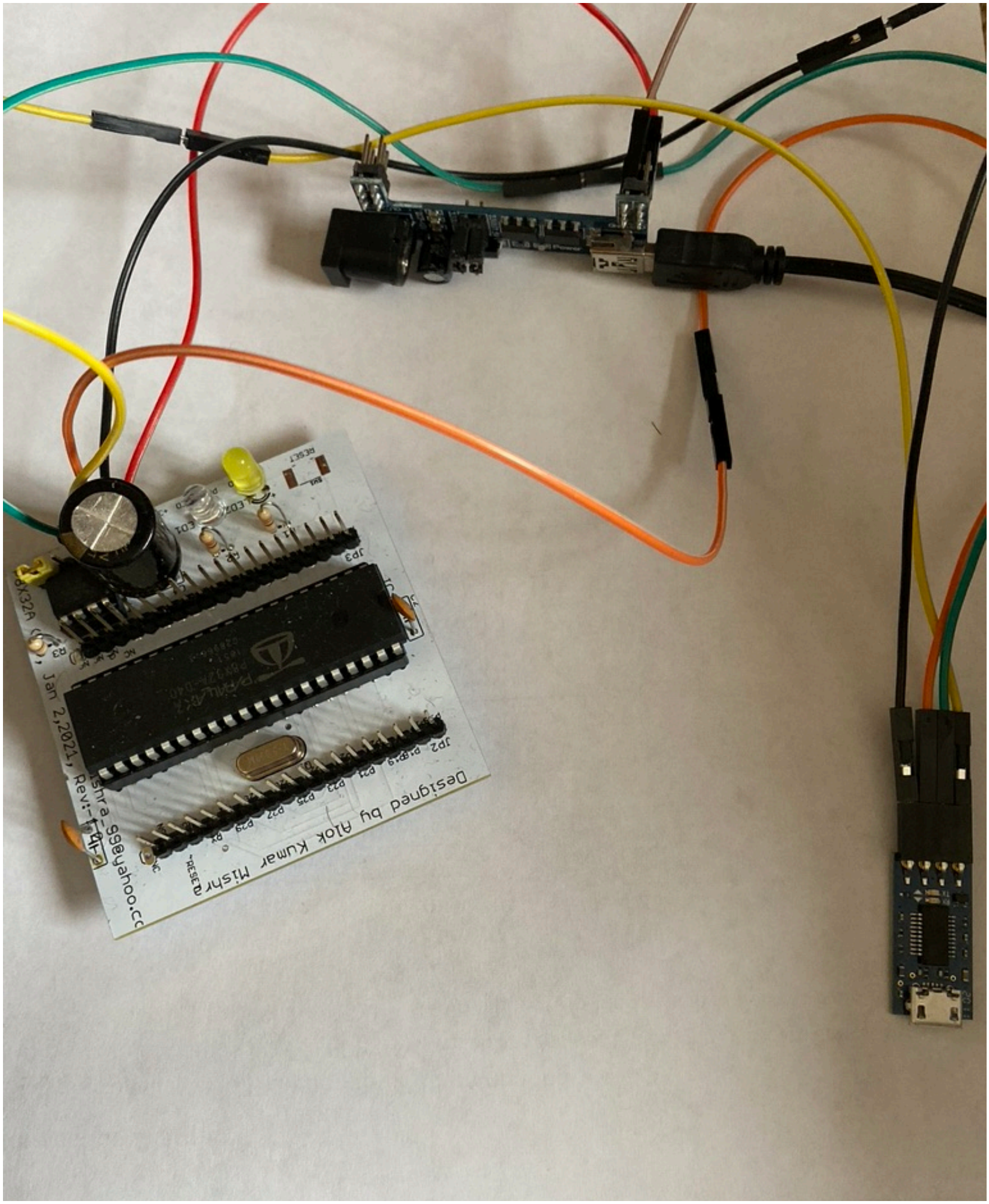

Figure 6

in figure (figure 6) on previous page on the left is p1 board with all components soldered (except reset button SW1) , on the bottom right is propeller plug and in center top is power supply