DEPARTMENT OF ELECTRONIC AND TELECOMMUNICATION
UNIVERSITY OF MORATUWA

EN 2160 : Electronic Design Realization

Reciprocal Frequency Counter - report

200687M : Vishwamith P.G.H

$22^{th} of July, 2023$

Contents

1 Product Description

The Reciprocal Frequency Counter is a state-of-the-art, battery-powered electronic device designed for precise frequency and duty cycle measurements of signals with amplitudes up to 100V. Whether in the laboratory, fieldwork, or industrial settings, this compact and portable instrument provides accurate frequency and duty cycle readings with ease and convenience. It caters to a wide range of applications, including electronics testing, signal analysis, and troubleshooting tasks.

1.1 Key Features

1. **Wide Frequency Range:** The frequency counter covers an extensive range, supporting frequencies from 1 Hz to 100 MHz, making it suitable for various signal types and applications.
2. **High Accuracy:** The reciprocal counting technique ensures high accuracy and stability in frequency measurements, making it ideal for demanding applications that require precise results.
3. **Duty Cycle Measurement:** Apart from frequency measurement, the device is equipped to measure the duty cycle of square wave signals, providing valuable insights into the signal's on-off ratio.
4. **Large Display:** The product features a clear and easy-to-read LCD display that shows frequency and duty cycle readings simultaneously. The backlight allows users to operate the device in low-light conditions.
5. **User-Friendly Interface:** With a simple and intuitive user interface, the frequency counter is easy to operate, even for non-technical users. Its one-button measurement functionality provides quick and hassle-free results.
6. **Battery-Powered:** The device is powered by a long-lasting, rechargeable battery, ensuring portability and uninterrupted usage in the field without the need for external power sources.
7. **Auto Power-Off:** To conserve battery life, the device incorporates an auto power-off feature that activates after a period of inactivity.
8. **Input Protection:** The device is designed with robust input protection circuitry, safeguarding it against accidental voltage spikes and ensuring user safety during measurements.
9. **Compact and Rugged Design:** The frequency counter's compact and sturdy construction makes it suitable for on-the-go usage, withstanding rugged environments and providing reliable performance in challenging conditions.
10. **Banana Input Connector:** The device includes a standard Banana input connector, allowing easy and secure connections to various signal sources.

1.2 Specifications

- Frequency Range: Frequency Range: 1 Hz to 100 MHz
- Frequency Resolution: 1 Hz
- Duty Cycle Range: 1% to 99%
- Duty Cycle Resolution: 0.1%
- Input Voltage Range: Up to 100V peak
- Input Impedance: 1 MΩ
- Accuracy: $\pm 10^{-6}$
- Timebase: Crystal Oscillator
- Display: LCD, 7 digits, with backlight
- Power Source: Rechargeable battery pack
- Battery Life: Approximately 300 hours (depending on usage)
- Auto Power-Off: 10 minutes
- Operating Temperature: -10°C to 50°C
- Dimensions: 120 mm x 150 mm x 40 mm (L x W x H)
- Weight: 256 gram

1.3 Product Validation

During the product validation process, the device undergoes rigorous testing and evaluation to verify its capabilities, functionality, and safety. The goal is to ensure that the frequency counter not only meets the specifications claimed but also performs optimally in real-world scenarios, satisfying the needs of its target users.

1.4 Validation Process

1. **Performance Testing:** The frequency counter is subjected to comprehensive performance testing across its entire frequency range. Test signals of known frequencies are applied, and the accuracy of the measurements is assessed. The counter's ability to handle various types of input waveforms is verified, including sine waves, square waves, and pulse waveforms.

2. **Duty Cycle Measurement Verification:** The duty cycle measurement feature is thoroughly validated by applying square wave signals with known duty cycles. The results are compared with the expected values to ensure accurate duty cycle readings.

3. **Frequency Resolution and Sensitivity:** The counter's frequency resolution and sensitivity are evaluated by testing its ability to measure low-frequency signals accurately and detect weak signals close to the noise floor.

4. **Input Protection Testing:** Robustness of the input protection circuitry is validated by applying voltage spikes and transient signals to the input. The device's response is analyzed to confirm its ability to withstand accidental voltage surges and protect both the instrument and the user.

5. **Temperature and Environmental Testing:** The frequency counter is tested under various environmental conditions, including temperature extremes, humidity, and vibration, to ensure it maintains stable and accurate measurements in challenging operating environments.

6. **Battery Performance:** The battery life and performance are assessed under different usage scenarios to determine the expected operational duration between charges accurately.

7. **User Interface and Usability Evaluation:** The device's user interface is evaluated for intuitiveness and ease of use. Human factors testing helps identify and address any potential usability issues to make the product more user-friendly.

8. **Safety Compliance:** The product is tested to meet safety standards and regulations to ensure it poses no hazards to users during normal operation.

9. **Endurance Testing:** To assess the device's long-term reliability, endurance testing is performed, subjecting it to continuous operation for an extended period to identify any potential wear and tear issues.

2 Product Goals

1. **Accuracy and Reliability:** The primary goal of the product is to provide accurate and reliable frequency and duty cycle measurements across its entire frequency range. Users should be able to trust the counter's readings for critical applications and signal analysis tasks.

2. **User-Friendly Operation:** The product aims to offer a user-friendly interface that is easy to navigate, making it accessible to both experienced professionals and non-technical users. Intuitive controls and clear display of measurements enhance usability.

3. **Portability and Convenience:** As a battery-powered device, the frequency counter aims to be highly portable, allowing users to take it to different locations for fieldwork or on-the-go tasks. It should have a long-lasting battery life to ensure uninterrupted usage.

4. **Wide Frequency Range:** The goal is to provide a frequency counter with a wide range of 1 Hz to 100 MHz, covering a broad spectrum of applications and accommodating various signal types.

5. **Duty Cycle Measurement:** By incorporating duty cycle measurement capabilities, the product seeks to offer valuable insights into square wave signals' on-off behavior, enhancing its versatility and usability.

6. **Input Protection and Safety:** The product aims to prioritize user safety by integrating robust input protection circuitry, safeguarding both the device and the user from accidental voltage spikes or transients.

2.1 Market Goals

1. **Target Audience:** The product seeks to cater to a diverse range of professionals and enthusiasts, including electronic engineers, technicians, hobbyists, and educators. It aims to appeal to those who require accurate frequency and duty cycle measurements for various applications.

2. **Competitive Pricing:** The market goal is to offer a competitive price point that aligns with the product's features and performance. This approach ensures accessibility to a broader customer base and positions the device favorably in the market.

3. **Differentiation:** The product aims to stand out from competitors by emphasizing its wide frequency range, accuracy, and duty cycle measurement capabilities. By highlighting unique selling points, it seeks to establish itself as a top choice among similar devices.

4. **Market Penetration:** The goal is to penetrate the market efficiently by reaching out to relevant industry publications, online platforms, and trade shows to create awareness about the product's features and benefits.

5. **Customer Support and Satisfaction:** The product aims to deliver exceptional customer support and after-sales service to ensure customer satisfaction. Prompt response to inquiries and addressing any issues promptly enhances customer loyalty and positive word-of-mouth.

6. **Expansion and Product Iteration:** In the long term, the goal is to continuously improve the product based on user feedback, technological advancements, and changing market demands. This approach allows the company to expand its product line and offer enhanced versions with additional features and capabilities.

By aligning the product and market goals, the Reciprocal Frequency Counter with 100 MHz Bandwidth aims to establish itself as a reliable, versatile, and user-friendly tool, garnering customer trust and loyalty in the competitive frequency measurement market.

Manufacturing cost : Rs.8500

Unit Price : Rs. 10000

3 Implementation

3.1 Component Selection

- **74AHC1G4208GWH :** divide by 256 to use as a prescalar to in crease band width up to 200MHz.
- **SR201C223KAR :** Ceramic capacitors use as decoupling capacitors for operate in better frequency.
- **1N4148 :** High speed diodes have used.
- **Connectors :** JST and FPC
  FEC ribbon connectors have used as connectors.
- **ATMEGA32A-PU :** Used for higher number of input pins.

4 Circuit Design And Initial Testing

The circuit design of the Reciprocal Frequency Counter begins by converting the incoming signal into a square pulse with the same frequency using Schmitt triggers. This step ensures that the signal is standardized for accurate frequency measurements. To create a known gate time pulse (e.g., 100 ms) for counting, a SR latch is utilized, switching on and off in a controlled manner. The D flip-flop is employed to correct the edges of the pulses, ensuring precise timing.

During the gate time, the number of reference clock pulses and signal pulses within the gate time is determined using three separate binary counters. These counters keep track of the number of pulses, which are later used for calculating the frequency and duty cycle of the input signal.

The output of the binary counters is fed into a microcontroller, which plays a central role in the circuit. The microcontroller processes the counts from the binary counters and performs the necessary calculations to derive the frequency and duty cycle of the input signal accurately. It also controls the LCD display, which shows the frequency and duty cycle results in a user-friendly format.

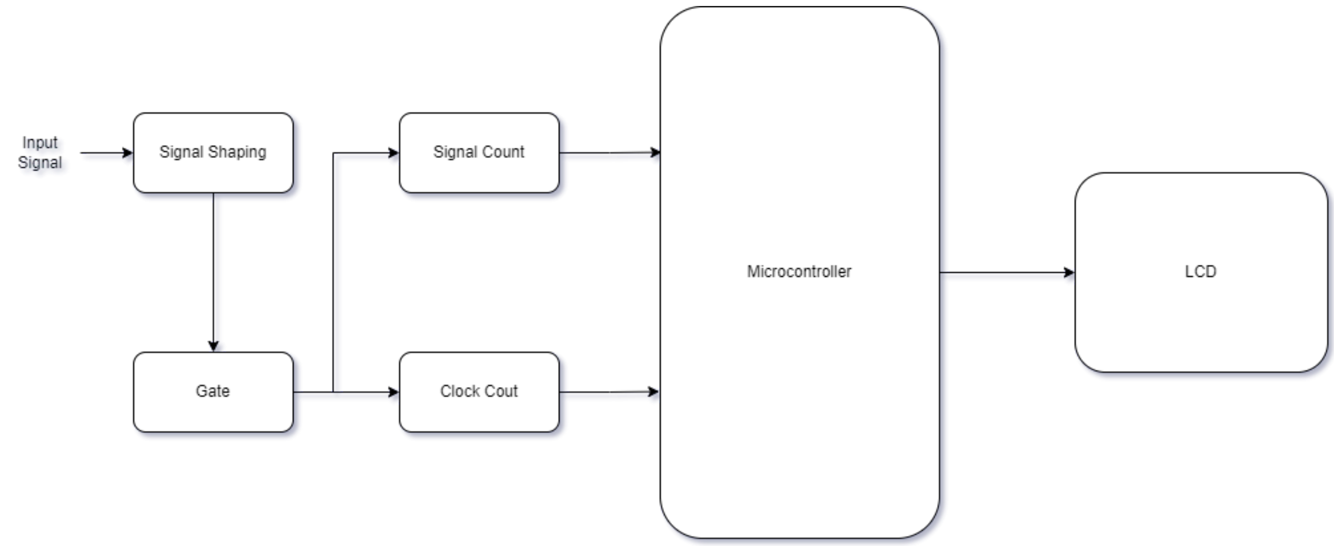


Figure 1: Functional Block Diagram

Before moving to the production phase, an extensive initial testing process is conducted to validate the circuit design and ensure its functionality and accuracy. Simulation software such as Multisim and Proteus is employed to model the circuit and verify its performance under various conditions.

The initial testing phase encompasses several crucial steps. Functionality tests are conducted to ensure that each circuit component, including the Schmitt triggers, SR latch, D flip-flop, and binary counters, operates as intended. The correct conversion of the signal into square pulses and the accurate generation of the gate time pulse are verified.

The frequency range of the counter is tested using known input frequencies, and the calculated frequency results are compared to expected values to ensure accuracy. Duty cycle measurements are also validated by applying square wave signals with known duty cycles.

Input protection measures are tested to verify the circuit's resilience to voltage spikes and transient signals, guaranteeing the safety of both the device and the user during measurements.

Additionally, noise and interference tests are performed to identify and address any potential issues that could affect measurement accuracy. The stability of the circuit over time is assessed to ensure consistent performance during extended use.

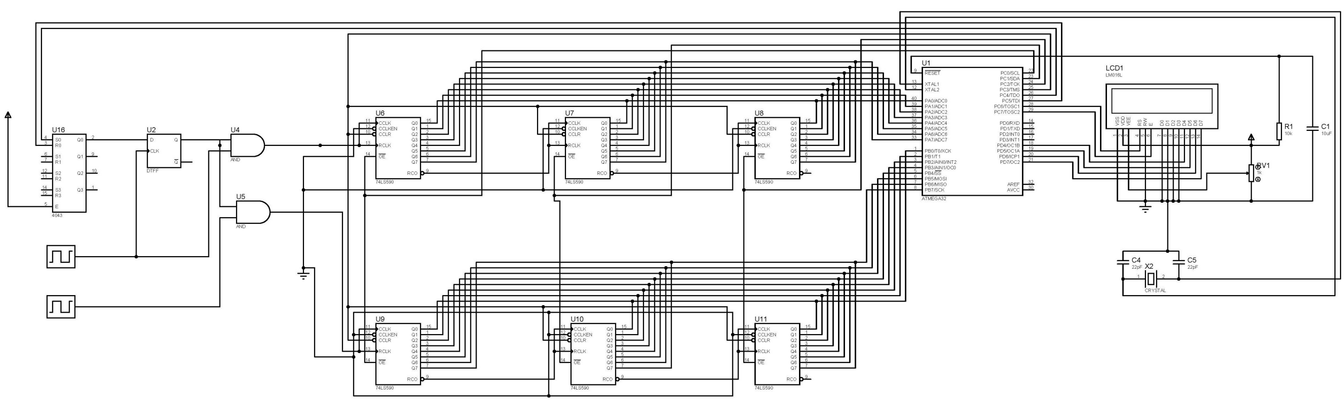


Figure 2: Simulation Circuit from Multisim

By meticulously conducting initial testing and simulation using Multisim and Proteus, any design flaws or performance discrepancies are identified and addressed before proceeding with mass production. The Reciprocal Frequency Counter is thus equipped to provide reliable and precise frequency and duty cycle measurements, making it a valuable tool for engineers, technicians, and hobbyists in various applications.

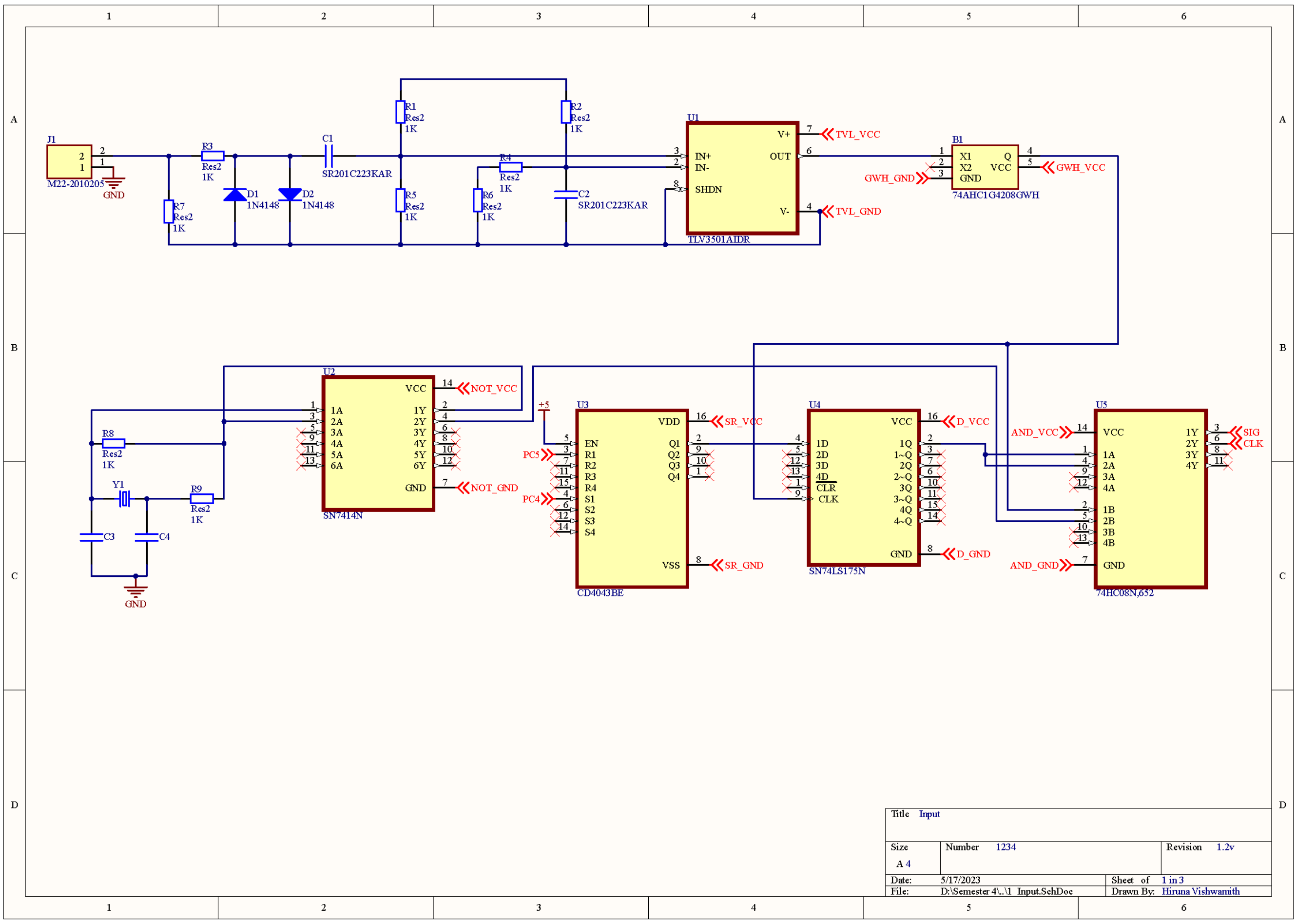


Figure 3: Input Circuit

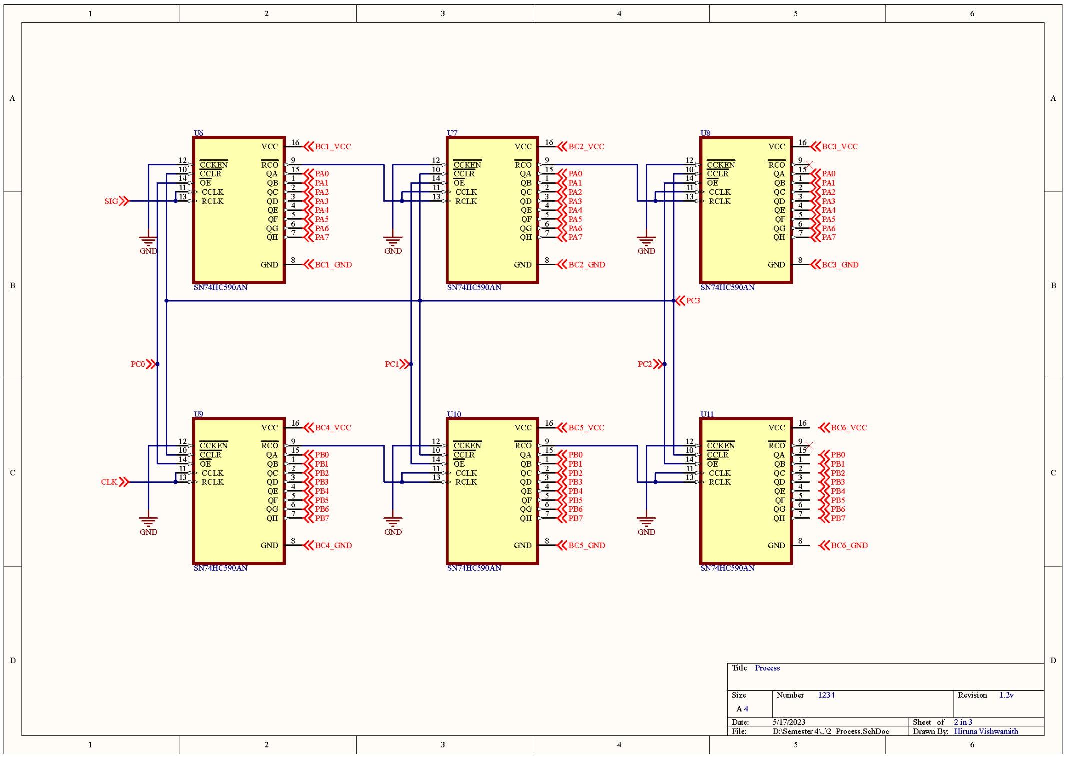


Figure 4: Processing Circuit

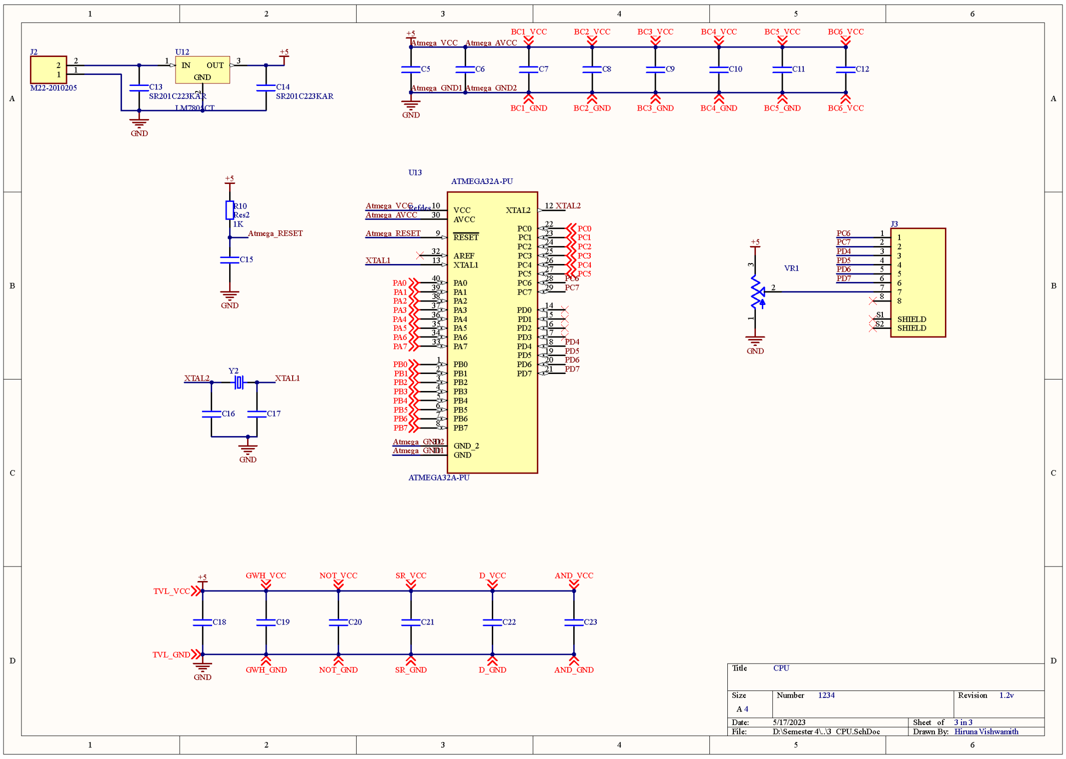
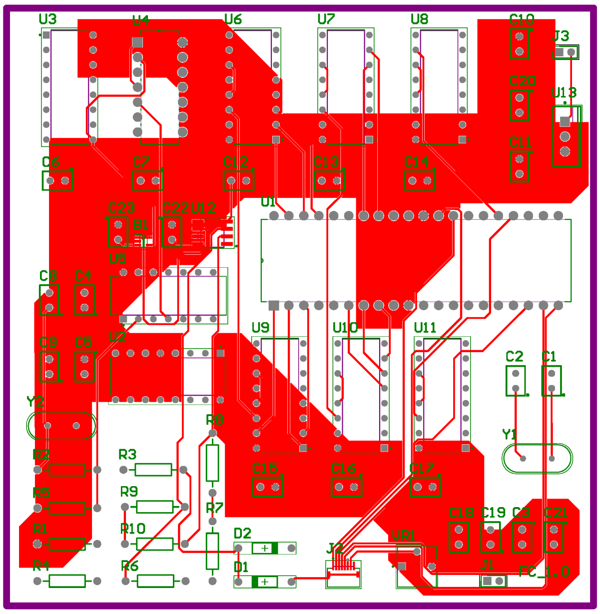
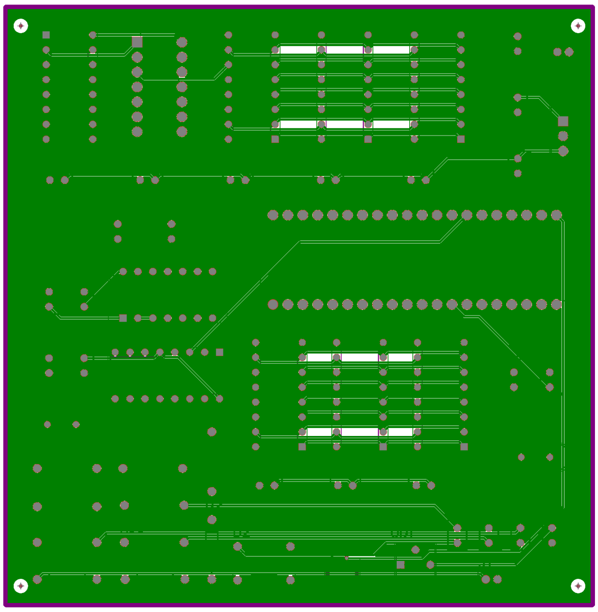


Figure 5: Output Circuit

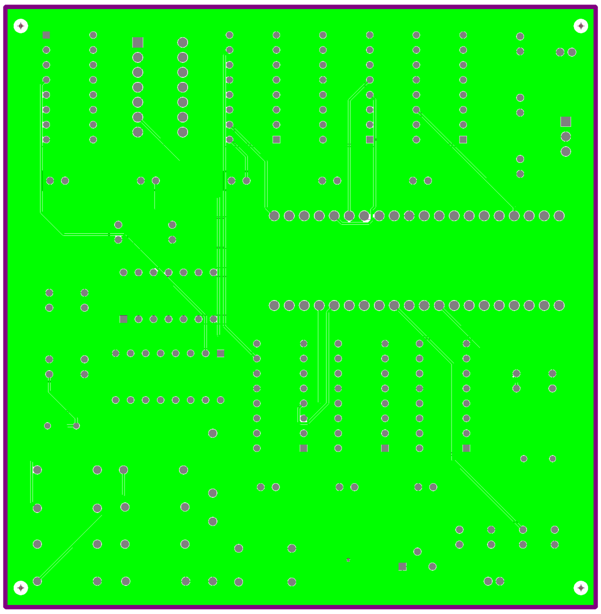
5 PCB



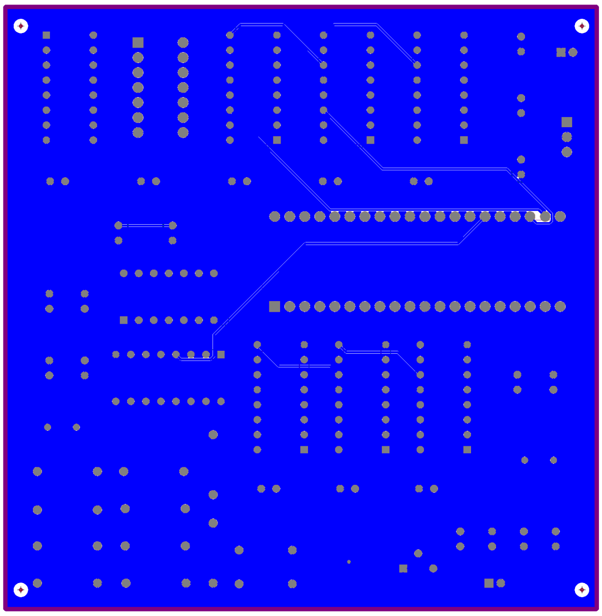
(a) Top layer



(b) Mid layer 1

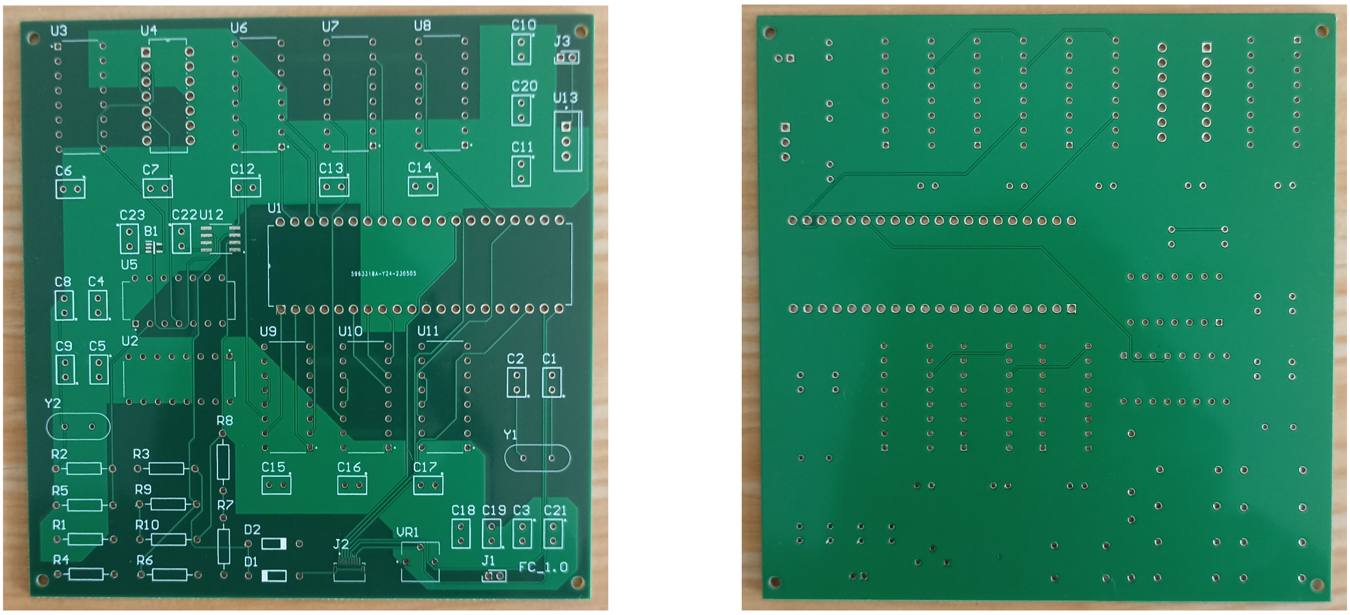


(a) Mid layer 2



(b) Bottom layer

Figure 7: PCB layers



(a) Top View (b) Bottom View

Figure 8: Manufactured PCB

Manufacturing was done at JLCPCB in China. Therefore, online available JLCPCB design rules were imported to Altium Designer. Refer Appendix A for schematic diagram and layout respectively.

6 Enclouser

Premade Encounter is brought in chinese manufacturer. Bulk of these enclosures can be bought in a chinese company like alibaba.

Figure 9: Front View

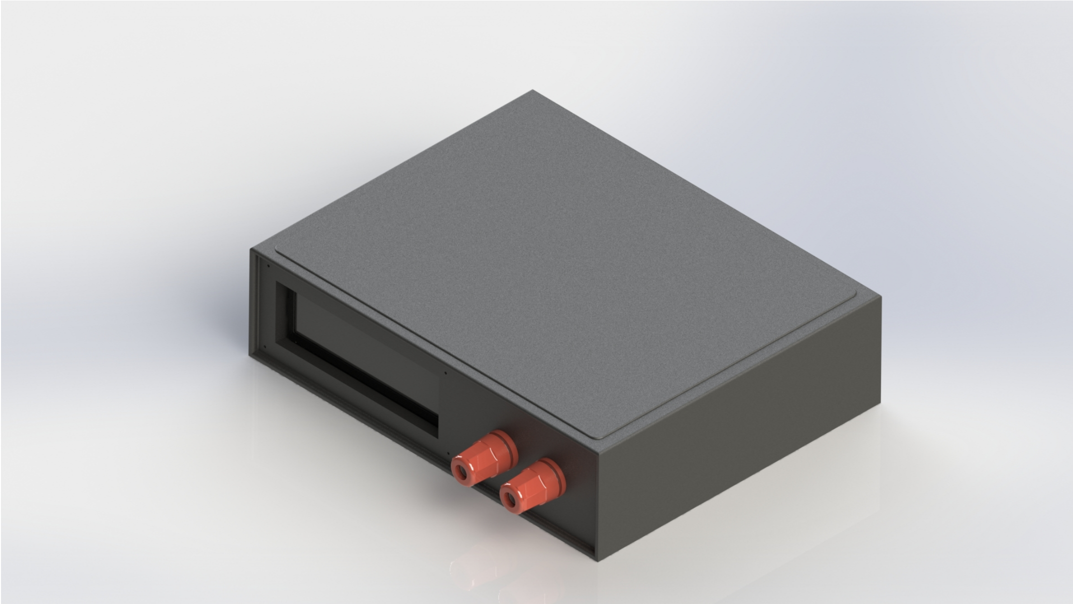


Figure 10: Isometric View

Figure 11: Final Product

Figure 12: Output of LCD

7 Instruction for Assembly

This guide will walk you through the assembly process to build your frequency counter. Before you begin, ensure you have all the required components and tools ready. Take your time and follow the instructions carefully for a successful assembly.

- **Step 1: Component Check** Inspect the contents of the kit and verify that all components are present and undamaged. Refer to the included component list to ensure nothing is missing.
- **Step 2: Soldering** Start by soldering the components on the provided printed circuit board (PCB) following these guidelines:
  - Begin with low-profile components, such as resistors and capacitors, and work your way up to taller components.
  - Pay attention to the correct orientation of polarized components, such as diodes and electrolytic capacitors.
  - Use an adjustable temperature soldering iron and lead-free solder for best results.
  - Ensure all solder joints are clean and free from shorts or cold solder joints.
- **Step 3: ICs and IC Sockets** Insert the ICs into their designated IC sockets carefully. Ensure that the orientation matches the PCB silkscreen markings. Take care not to bend any pins during insertion.
- **Step 4: LCD Display** Attach the LCD display to its designated location on the PCB. Align the pins with the corresponding pads and gently press the display into place.
- **Step 5: Input Connectors** Mount the input connectors (BNC or other specified connectors) onto the PCB. Secure them with nuts or screws provided in the kit.
- **Step 6: Power Supply** Connect the provided rechargeable battery pack to the designated connector on the PCB. Alternatively, you can connect an external power supply within the specified voltage range.
- **Step 7: Casing** If your kit includes an enclosure, carefully place the assembled PCB inside the casing. Secure it using screws or clips provided with the enclosure. Casing is fixed using the given 3mm screw nails.
- **Step 8: Final Check** Double-check all solder joints and connections to ensure they are secure and correct. Verify that no components are missing or incorrectly placed.
- **Step 9: Initial Testing** Before applying power, perform an initial test by examining the assembled components for any visible issues or soldering errors. Ensure that there are no solder bridges or cold joints that may affect the circuit's performance.

- **Step 10: Power On** With the initial testing complete, it's time to power on the Reciprocal Frequency Counter. Check that the LCD display lights up and shows the startup message.
- **Step 11: Calibration (Optional)** Depending on the kit, you may need to calibrate the frequency counter using a known reference signal. Follow the calibration instructions provided in the kit documentation.

8 Testing the Product Functionality

Ensuring the functionality and performance of the Reciprocal Frequency Counter is a critical phase in the product development process. Rigorous testing is conducted to verify that the device meets its specifications and provides accurate and reliable frequency and duty cycle measurements. Here's an overview of the testing process:

1. **Basic Functionality Test:** The first step in testing the product is to verify its basic functionality. This includes turning on the frequency counter, checking if the LCD display shows the startup message, and ensuring that all buttons and controls are responsive.
2. **Frequency Range Verification:** To validate the frequency measurement capabilities, the frequency counter is tested using known input frequencies across its specified range. The measured frequencies are compared against the known values to verify accuracy. The counter should display the correct frequency readings within an acceptable tolerance.
3. **Duty Cycle Measurement Test:** The duty cycle measurement feature is thoroughly tested by applying square wave signals with various duty cycles. The results are compared with the expected values to ensure accurate duty cycle readings. The device should accurately display the duty cycle of the input signal.
4. **Input Signal Testing:** The frequency counter is tested with different types of input signals, including sine waves, square waves, and pulse waveforms. It should accurately measure and display the frequencies and duty cycles of these signals.
5. **Stability and Long-Term Testing:** The device's stability and long-term performance are assessed by continuously measuring the same input signal over an extended period. The frequency counter should maintain accuracy and stability over time without significant drift.
6. **Input Protection Testing:** The input protection circuitry is thoroughly tested to ensure it effectively guards the device against voltage spikes and transient signals. The frequency counter should withstand external disturbances without affecting its functionality.
7. **Battery Performance Test:** For battery-powered models, the battery life and performance are evaluated under different usage scenarios. The device's operation time on a full charge is measured to determine its practical usage duration in the field.
8. **User Interface Evaluation:** The user interface and controls are assessed for ease of use and intuitiveness. The display should be clear and easily readable, and buttons should be responsive to user input.
9. **Calibration Verification (If Applicable):** If the device requires calibration, the process is carried out using known reference signals. The calibration results are verified to ensure the accuracy of subsequent frequency and duty cycle measurements.
10. **Field Testing (Optional):** In certain cases, the frequency counter is subjected to real-world field testing, where it is used in various environments and conditions. The device's performance is evaluated in practical applications to validate its suitability for diverse use cases.

By conducting comprehensive testing, we ensure that the Reciprocal Frequency Counter with 100 MHz Bandwidth meets the highest standards of accuracy, reliability, and user-friendliness. The successful completion of these tests confirms that the product is ready to deliver precise frequency and duty cycle measurements, empowering professionals and enthusiasts in their signal analysis and electronics testing endeavors.

9 BOM

Table 1: Bill of Materials

Item	Quantity	Cost in $	Cost in Rs.
74HC590	6	5.64	1850.24
Atmega32A-PU	1	5.18	1699.34
CD4043BE	1	0.92	301.81
SN74LS175N	1	0.94	308.37
SN74HC08AN	1	0.6	196.83
74AHC1G4208GWH	1	0.59	193.55
TLV3501AIDR	1	3.15	1033.38
ECS-160-20-4X-EM	2	0.72	236.20
SN74HC04N	1	0.6	196.83
Total		18.34	6016.57

10 References

https://www.arduino.cc/reference/en/libraries/liquidcrystal/liquidcrystal/
https://linuxhint.com/lcd-16x2-pin-configuration/
https://electronics.stackexchange.com/questions/513098/configuring-atmega32-timer-using-the-oscillator-without-prescaling-gives-wrong-d
https://www.mouser.com/ProductDetail/595-SN74HC04N
https://www.mouser.com/ProductDetail/595-CD40106BE
https://electronics.stackexchange.com/questions/8112/how-do-i-convert-9-v-dc-to-5-v
https://e2e.ti.com/blogs$_/b/analogwire/posts/how-fast-can-it-run-determine-the-speed-of-your-logic-device$
https://www.crystek.com/documents/appnotes/pierce-gateintroduction.pdf
https://electronics.stackexchange.com/questions/497272/calculate-c-and-r-in-a-cmos-crystal-oscillator
https://electrosome.com/interfacing-lcd-atmega32-microcontroller-atmel-studio/
https://electrosome.com/interfacing-lcd-atmega32-microcontroller-atmel-studio/
https://www.openhacks.com/uploadsproductos/eone-1602a1.pdf
https://datasheet.lcsc.com/lcsc/1804240041$_Texas-Instruments-74HC590ADWR_C8101.pdf$
https://www.radiokot.ru/konkursCatDay2014/30/02.pdf
https://rubiola.org/pdf-slides/2012T-IFCS-Counters.pdf
https://www.electronicsforu.com/electronics-projects/line-frequency-meter-reciprocal-counting
https://www.febo.com/pipermail/time-nuts/attachments/20071201/e7833af5/attachment.pdf
https://www.eevblog.com/forum/projects/8-11-digits-reciprocal-frequency-counter-0-1-hz-150-mhz/
https://www.best-microcontroller-projects.com/fast-reciprocal-frequency-counter.html

11 Appendix

11.1 Software Implementation

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(3,4,5,6,7,8);
#define pulse_ip 9
int ontime,offtime,duty;
float freq,period;

void setup()
{
  pinMode(pulse_ip,INPUT);
  lcd.begin(16, 2);
  lcd.clear();
  lcd.print("Freq:");
```

```
  lcd.setCursor(0,1);
  lcd.print("Duty:");
}
void loop()
{
   ontime = pulseIn(pulse_ip,HIGH);
   offtime = pulseIn(pulse_ip,LOW);
   period = ontime+offtime;
   freq = 1000000.0/period;
   duty = (ontime/period)*100;
   if(period==0){
    freq=0;

    }
   lcd.setCursor(5,0);
   lcd.print("         ");
   lcd.setCursor(5,0);
   if(freq<1000){
   lcd.print(freq);
   lcd.print("Hz");
   }else if(freq>999|freq<1000000){
   lcd.print(freq/1000);
   lcd.print("KHz");
   }else if(freq>999999|freq<1000000000){
   lcd.print(freq/1000000);
   lcd.print("MHz");
   }
   lcd.setCursor(6,1);
   lcd.print("         ");
   lcd.setCursor(6,1);
   lcd.print(duty);
   lcd.print('%');
   delay(1000);
}
```

11.2 Atmega32A-PU Configuration

```

#include <mega328p.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
unsigned int m,n;

unsigned char s[16];

// Alphanumeric LCD functions
#include <alcd.h>

// Declare your global variables here

// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
// Place your code here
n++;

}
```

```

// Timer 0 output compare A interrupt service routine
interrupt [TIM0_COMPA] void timer0_compa_isr(void)
{
// Place your code here

}

// Timer2 overflow interrupt service routine
interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
// Place your code here

}

// Timer2 output compare interrupt service routine
interrupt [TIM2_COMPA] void timer2_compa_isr(void)
{
// Place your code here
m=n*256+TCNT0;
n=0;
TCNT0=0;
lcd_clear();
lcd_gotoxy(0,0);
ltoa(m,s);
strcat(s,"Hz");
lcd_puts(s);

}

void main(void)
{
// Declare your local variables here

// Crystal Oscillator division factor: 1
#pragma optsize-
CLKPR=(1<<CLKPCE);
CLKPR=(0<<CLKPCE) | (0<<CLKPS3) | (0<<CLKPS2) | (0<<CLKPS1) | (0<<CLKPS0);
#ifdef _OPTIMIZE_SIZE_
#pragma optsize+
#endif

// Input/Output Ports initialization
// Port B initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRB=(0<<DDB7) | (0<<DDB6) | (0<<DDB5) | (0<<DDB4) | (0<<DDB3) | (0<<DDB2) | (0<<DDB1) |
    (0<<DDB0);
// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) |
    (0<<PORTB1) | (0<<PORTB0);

// Port C initialization
// Function: Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRC=(0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<DDC1) | (0<<DDC0);
// State: Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTC=(0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) | (0<<PORTC1) |
    (0<<PORTC0);

// Port D initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
```

```
DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<DDD1) |
    (0<<DDD0);
// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (0<<PORTD3) | (0<<PORTD2) |
    (0<<PORTD1) | (0<<PORTD0);

// Timer/Counter 0 initialization
// Clock source: T0 pin Rising Edge
// Mode: Fast PWM top=0xFF
// OC0A output: Disconnected
// OC0B output: Disconnected
TCCR0A=(0<<COM0A1) | (0<<COM0A0) | (0<<COM0B1) | (0<<COM0B0) | (1<<WGM01) | (1<<WGM00);
TCCR0B=(0<<WGM02) | (1<<CS02) | (1<<CS01) | (1<<CS00);
TCNT0=0x00;
OCR0A=0x80;
OCR0B=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Disconnected
// OC1B output: Disconnected
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) | (0<<WGM10);
TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) | (0<<CS10);
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: Crystal on TOSC1 pin
// Clock value: PCK2/128
// Mode: Fast PWM top=0xFF
// OC2A output: Disconnected
// OC2B output: Disconnected
ASSR=(0<<EXCLK) | (1<<AS2);
TCCR2A=(0<<COM2A1) | (0<<COM2A0) | (0<<COM2B1) | (0<<COM2B0) | (1<<WGM21) | (1<<WGM20);
TCCR2B=(0<<WGM22) | (1<<CS22) | (0<<CS21) | (1<<CS20);
TCNT2=0x00;
OCR2A=0x50;
OCR2B=0x00;

// Timer/Counter 0 Interrupt(s) initialization
TIMSK0=(0<<OCIE0B) | (1<<OCIE0A) | (1<<TOIE0);

// Timer/Counter 1 Interrupt(s) initialization
TIMSK1=(0<<ICIE1) | (0<<OCIE1B) | (0<<OCIE1A) | (0<<TOIE1);

// Timer/Counter 2 Interrupt(s) initialization
```

```
TIMSK2=(0<<OCIE2B) | (1<<OCIE2A) | (1<<TOIE2);

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// Interrupt on any change on pins PCINT0-7: Off
// Interrupt on any change on pins PCINT8-14: Off
// Interrupt on any change on pins PCINT16-23: Off
EICRA=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00);
EIMSK=(0<<INT1) | (0<<INT0);
PCICR=(0<<PCIE2) | (0<<PCIE1) | (0<<PCIE0);

// USART initialization
// USART disabled
UCSR0B=(0<<RXCIE0) | (0<<TXCIE0) | (0<<UDRIE0) | (0<<RXEN0) | (0<<TXEN0) | (0<<UCSZ02) |
    (0<<RXB80) | (0<<TXB80);

// Analog Comparator initialization
// Analog Comparator: Off
// The Analog Comparator's positive input is
// connected to the AIN0 pin
// The Analog Comparator's negative input is
// connected to the AIN1 pin
ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) |
    (0<<ACIS0);
ADCSRB=(0<<ACME);
// Digital input buffer on AIN0: On
// Digital input buffer on AIN1: On
DIDR1=(0<<AIN0D) | (0<<AIN1D);

// ADC initialization
// ADC disabled
ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) |
    (0<<ADPS0);

// SPI initialization
// SPI disabled
SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) |
    (0<<SPR0);

// TWI initialization
// TWI disabled
TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTB Bit 0
// RD - PORTB Bit 1
// EN - PORTB Bit 2
// D4 - PORTC Bit 2
// D5 - PORTC Bit 3
// D6 - PORTC Bit 4
// D7 - PORTC Bit 5
// Characters/line: 16
lcd_init(16);

// Globally enable interrupts
#asm("sei")

```

```
while (1)
      {
      // Place your code here

      }
}
```

11.3 3D model of PCB

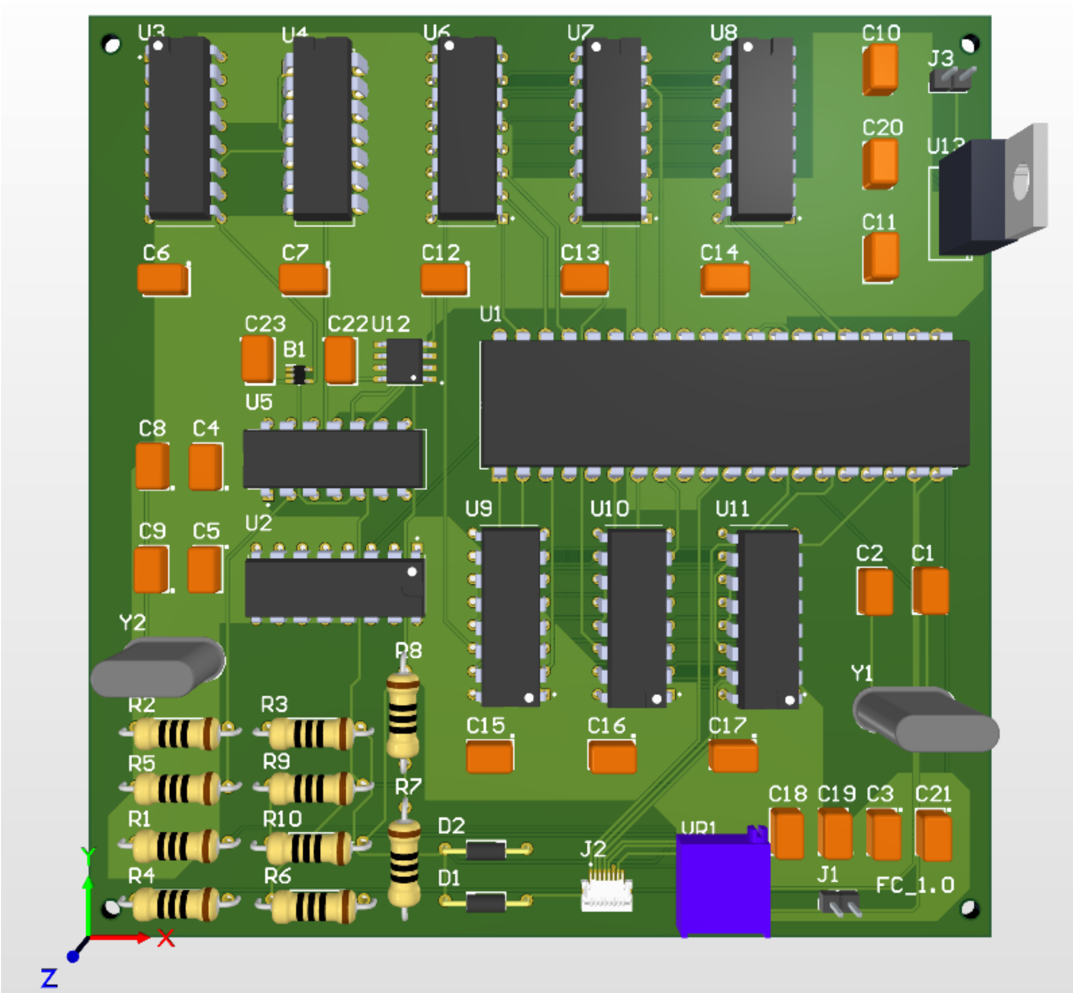
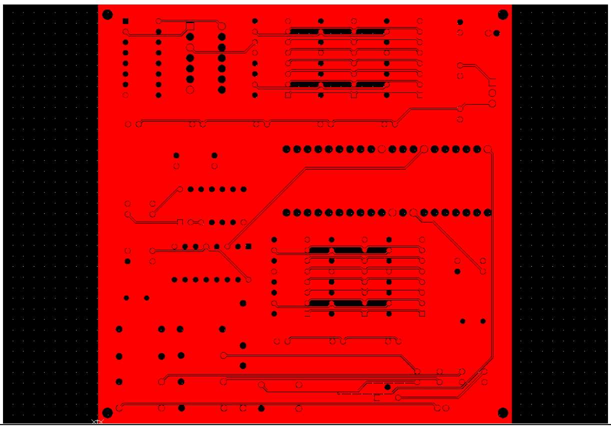
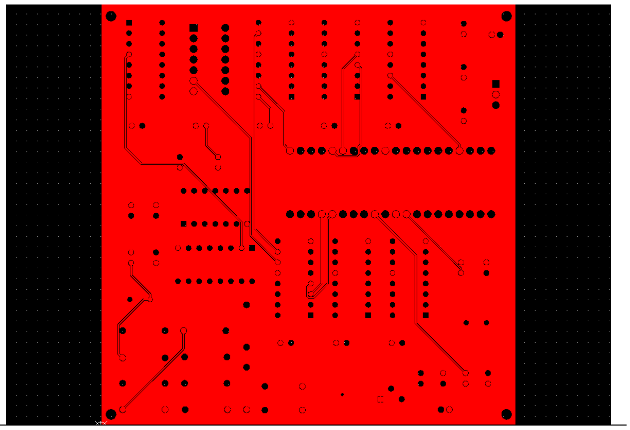
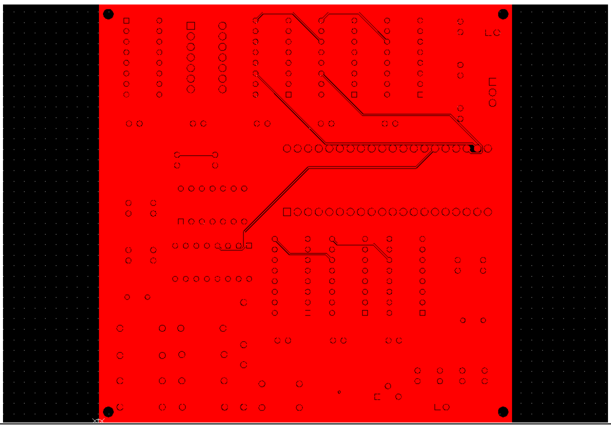
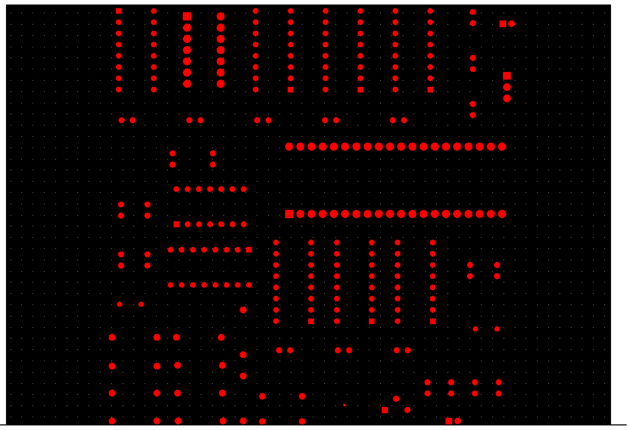
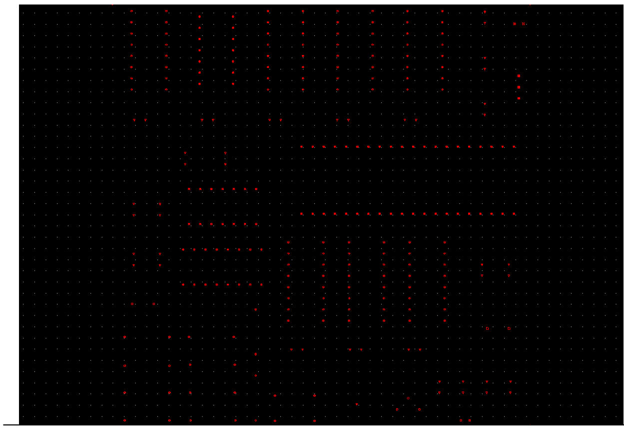
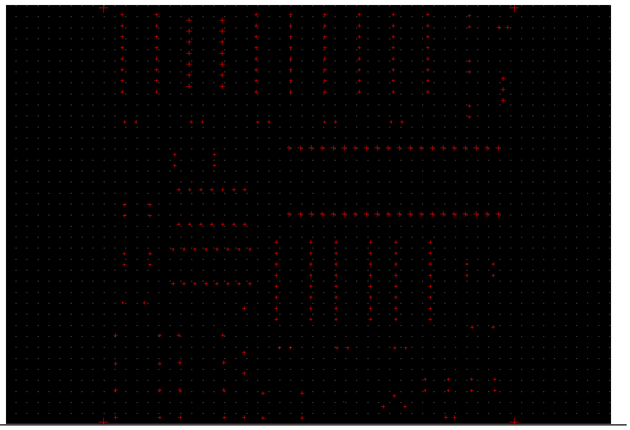
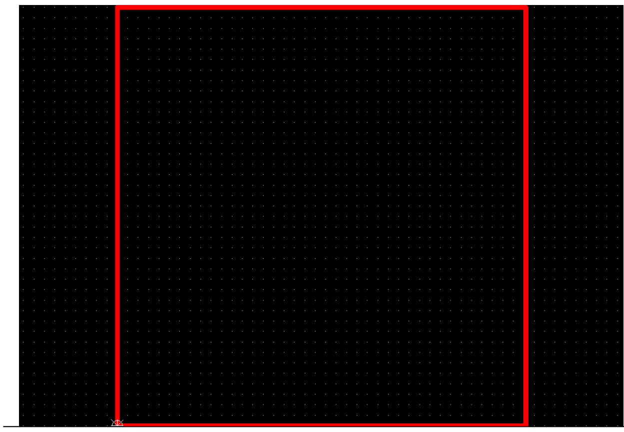
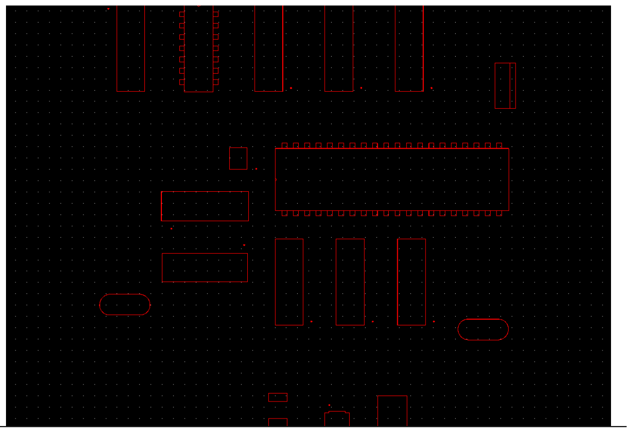
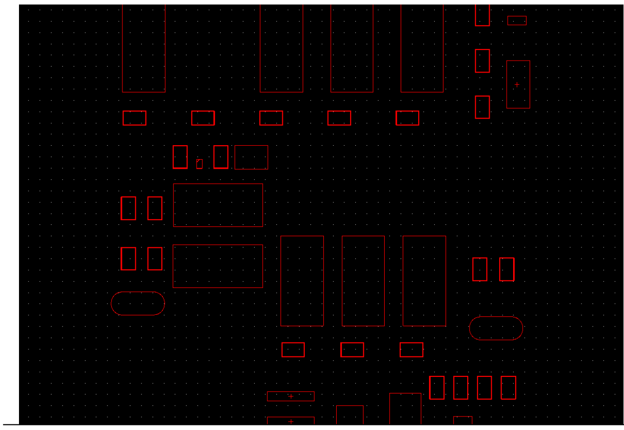
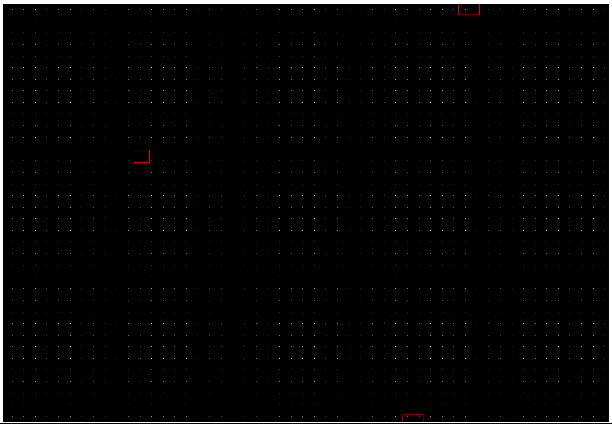


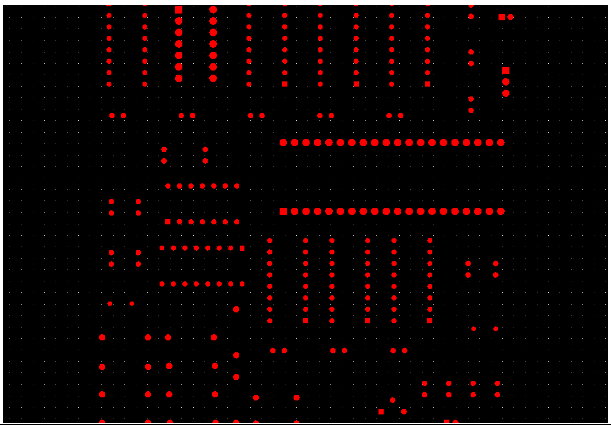
Figure 13: PCB 3D Model

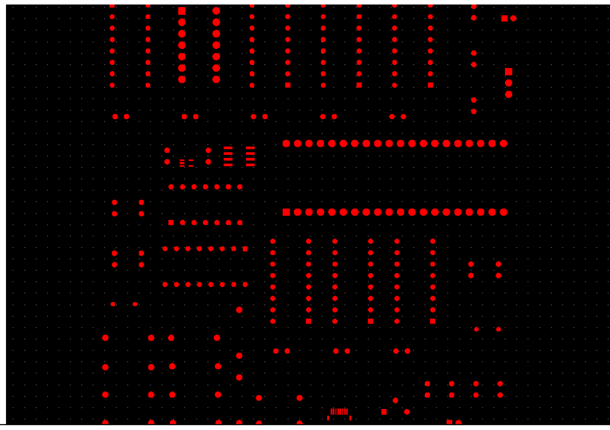
11.4 Manufacturing files - Gerber

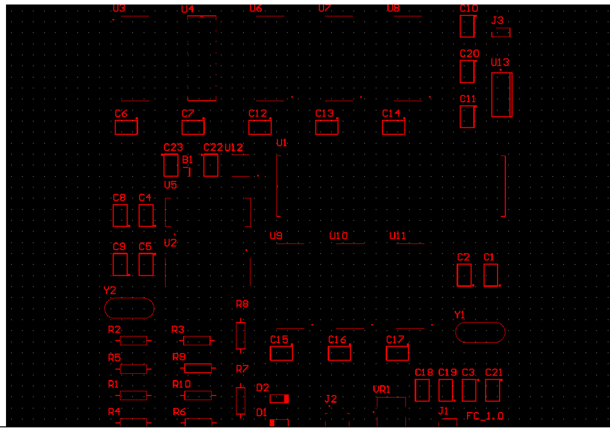


















U3
U4
U6
U7
U8
C10
J3
C20
U13
C11
C6
C7
C12
C13
C14
C23
C22
U12
B1
U1
U5
C8
C4
C9
C5
U2
U9
U10
U11
C2
C1
Y2
R8
Y1
R2
R3
C15
C16
C17
R5
R9
R7
C18
C19
C3
C21
R1
R10
D2
VR1
J2
R4
R6
D1
J1
FC_1.0