Control Circuit Design of 7-segment Common Cathode LED Display GPS Receiver Clock and Arduino Program

Seiichi Nakamori

Department of Technology, Faculty of Education, Kagoshima University, Kagoshima, Japan

k6165161@kadai.jp

Abstract

This paper newly designs the control circuit of the 7-segment common cathode LED display global positioning system (GPS) receiver clock and presents an Arduino sample program. The computer numerical control (CNC) router engraves the copper foil substrate to manufacture the printed wiring boards (PWBs) of the 7-segment common cathode LED display and its control circuits according to each G-code produced by the printed circuit board (PCB) design software Target3001! For each segment of the 7-segment common cathode LED display circuit, the two LEDs are connected in series. From an experiment, the two N-Channel metal-oxide-semiconductor field-effect transistors (MOSFETs) EKI04047, one at the high side and the other at the low side around the two LEDs, do not light on the two LEDs connected in series at all, unlike the case of the two LEDs connected in parallel. For this reason, upon lighting on/off the LEDs, this paper, newly, proposes the switching on/off control circuit with the optical relays at the high and low sides around the two LEDs connected in series.

Keywords: 7-Segment LED Display, GPS Receiver Clock, Arduino, Photo Relay, CNC Router

1. Introduction

The Global Positioning System (GPS) [5] provides us positioning, navigation, and timing (PNT) services. The GPS receiver clock with Arduino is found in [1], [12], [2], etc. The 7-segment light emission diode (LED) display [1], [12] and the liquid crystal display (LCD) [2], [3], [14] are used for the display of time. In [1], [12], the commercially available 7-segment LED display operates at DC 5 [V]. In [12], [3], the circuit and Arduino program are shown. In [7], by using the positioning data of the GPS, the circuit and the Arduino program of the latitude and longitude meter are presented. The large 7-segment LED display clock in [8] consists of the LED strip, the I2C real-time clock module and the large digit driver. The Arduino program of the large 7-segment LED display clock is also presented. In [8], direct current (DC) 12 [V] is supplied to the LED strip.

The lighting on/off operations for the three LEDs, connected in parallel, by one N-Channel metal-oxide-semiconductor field-effect transistor (MOSFET) are explained in [11]. Tanji [15] shows a circuit of the big 7-segment LED display clock with six LEDs, connected in parallel, per segment. In this paper each segment of the 7-segment common cathode LED display circuit has two LEDs connected in series. The two N-Channel MOSFETs EKI04047, one at the high side and the other at the low side, cannot light on the two LEDs connected in series (see Fig.1 (a) and Table 1, section 2).

In the parallel connection of the two LEDs, the two N-Channel MOSFETs work normally (see Fig.1 (b) and Table 2). For this reason, this paper newly designs the control circuit of the 7-segment common cathode LED display GPS receiver clock and presents an Arduino sample program. Namely, to light on/off the two LEDs, connected in series, for each segment, we introduce the optical relays [13] at the high and low sides around the LEDs.

The computer numerical control (CNC) router [12] engraves the copper foil substrate to produce the printed wired boards (PWBs) for both the control and 7-segment common cathode LED display circuits according to the G-codes produced by the printed circuit board (PCB) design software Target3001! [16].

2. Experiments to light on/off two LEDs in series or parallel connections

Let us consider the circuit diagrams with two LEDs as shown in Fig.1. In Fig.1 (a), the two LEDs are connected in
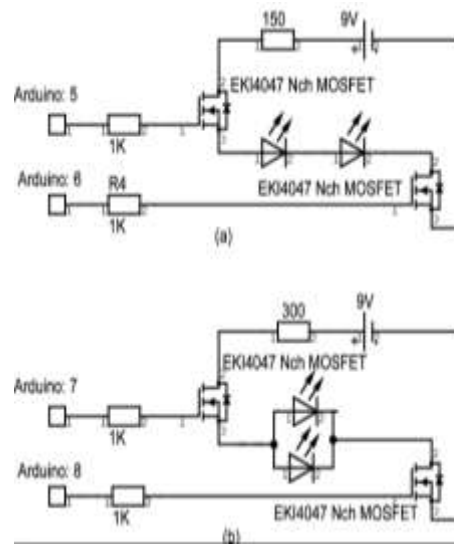
Fig.1 Experimental circuit diagrams to light on/off two LEDs connected in series or parallel.

Table 1 Experiment results for Fig.1 (a).

Arduino digital port number	Output voltage from digital pint	Two LEDs connected in series
5	5[V]	Not lighting
6	5[V]	
5	5[V]	Not lighting
6	0 [V]	
5	0 [V]	Not lighting
6	5[V]	
5	0 [V]	Not lighting
6	0 [V]	

Table 2 Experiment results for Fig.1 (b).

Arduino digital port number	Output voltage from digital pin	Two LEDs connected in parallel
7	5[V]	Lighting
8	5[V]	
7	5 [V]	Not lighting
8	0 [V]	
7	0 [V]	Not lighting
8	5[V]	
7	0 [V]	Not lighting
8	0 [V]	

series. From the Arduino digital pins 5 and 6, the voltages 0 [V] or 5 [V] are output. For the combinations of the output voltages, the experimental results are summarized in Table 1. In Fig.1 (b), from the digital pins 7 and 8, the voltages 0 [V] or 5 [V] are output similarly. The experimental results are summarized in Table 2. The actual voltage for 5 [V], output from each Arduino digital pin, is slightly smaller than 5 [V]. From Table 1, we see, in case of series connection of the two LEDs, that the N-Channel MOSFETs EKI04047 cannot light on the two LEDs in any cases. The problem lies in the high side N-Channel MOSFET connected to the digital pin 5. The low side N-Channel MOSFET works normally according to the voltage output from the digital pin 6. The turn-on and turn-off times of the N-Channel MOSFET EKI04047, used in the experiment, are very short, 9.7 [ns] and 31.6 [ns], respectively. From Table 2, it is seen that the two N-Channel MOSFETs can light on/off the two LEDs, connected in parallel, properly. The two LEDs light on when both output voltages from the digital pins 7 and 8 are 5 [V]. At this time, the two N-Channel MOSFETs turn on. The two N-Channel MOSFETs in Fig.1 (b) can be replaced with the two NPN bipolar junction transistors. In [2], the NPN bipolar junction transistors are used to turn current on or off for the 7-segment LED display of a 4-digit clock. LED strip [8] is also used to make a big 7-segment LED display clock. Section 3 proposes a circuit to light on/off the two LEDs, connected in series, with the photo relays for each segment and digit so that it can also be used for the 7-segment common cathode LED display GPS receiver clock with the LED strip.

3. Control circuit of 7-segment common cathode LED display GPS receiver clock with two LEDs, connected in series, for each segment

Fig.2 illustrates the control circuit diagram of the 7-segment LED common cathode display circuit in Fig.6 with the two LEDs, connected in series, for each segment. The terminals "Digit 1" and "Digit 2" correspond to hours, and "Digits 3" and "Digit 4" minutes. In this circuit, the photo relay TLP222AF is used to light on/off the two
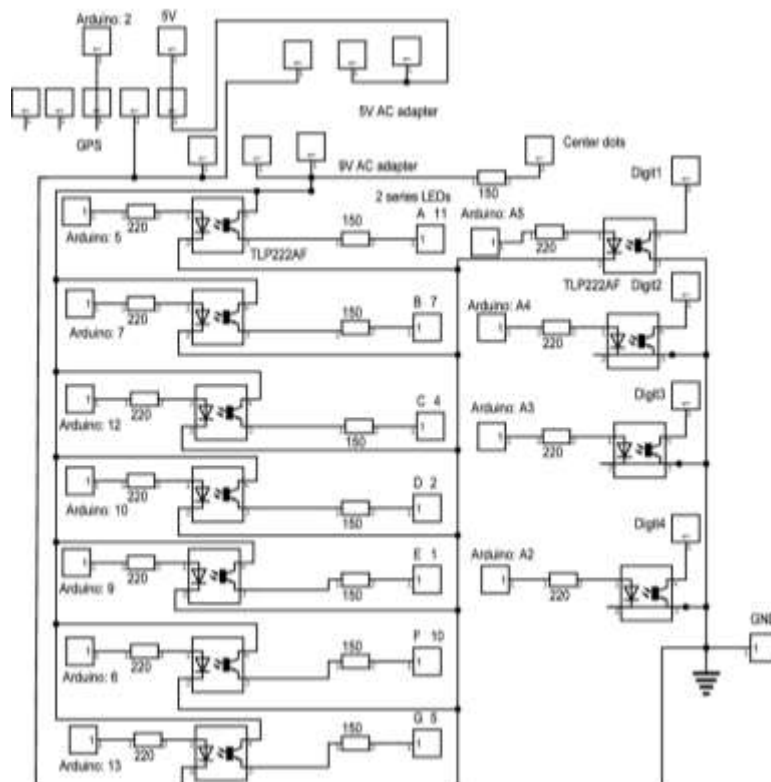

Fig.2 Control circuit diagram in 7-segment common cathode LED display GPS receiver clock drawn by Target3001!

LEDs connected series for each segment. In Fig.2, the terminal "Arduino: 5" is connected to the digital pin 5 of

Arduino. Similarly, the terminals "Arduino: 7", "Arduino: 12", "Arduino: 10", "Arduino: 9", "Arduino: 6" and "Arduino: 3" are connected to the digital pins 7, 12, 10, 9, 6 and 3 of Arduino, respectively. The terminals "A", "B", "C", "D", "E", "F" and "G" in Fig.2 are connected to the terminals "A", "B", "C", "D", "E", "F" and "G" of the 7-segment common cathode LED display in Fig.6, respectively. The terminals "Arduino A5", "Arduino A4", "Arduino A3" and "Arduino A2" in Fig.2 are connected to the analog pins of Arduino, A5, A4, A3 and A2, respectively. The terminals "Digit1", "Digit2", "Digit3" and "Digit4" are connected to the terminals "Digit1", "Digit2", "Digit3" and "Digit4" in Fig.6, respectively. The "GND" terminal in Fig.2 is connected to the GND pin of Arduino. The terminal "Arduino: 02" at the GPS receiver in Fig.2 is connected to the digital pin 2 of Arduino. The terminal "Center dots" in Fig.2 is connected to the terminal in the upper center of the 7-segment common cathode LED display of Fig. 6. In Fig.2, assuming that the electric current 20 [mA] flows through the two LEDs connected in series, we calculate the resistance 150 [Ω] for each segment by

$$\frac{9[V] - 6[V]}{20[mA]} = 150[\Omega]$$

from Ohm's law. In Fig.2, "A 11", "B 7", "C 4", "D 2", "E 1", "F 10" and "G 5" represent the pin numbers 11, 7, 4, 2, 1, 10 and 5 of the commercially available 4-digit cathode or anode common 7-segment LED displays corresponding to the segments A, B, C, D, E, F and G respectively. In this paper, the pin numbers of the 7-segment LED display need not be taken into account because the 4-digit 7-segment cathode common LED display is made by using the CNC router. The maximum DC current per I/O pin output from Arduino is 50 [mA] and the maximum value of the sum of the DC current output from Arduino is 100 [mA] [3]. The maximum trigger electric current of the photo relay TLP222AF, with the non-contact switching property, is 3 [mA] and electric on-current is 500 [mA]. The standard turn-on and turn-off times of the photo relay TLP222AF are 0.8 [ms] and 0.1 [ms], respectively. Hence, the photo relay is suitable for lighting on/off the two LEDs, connected in series, by Arduino. Since the maximum electric current supplied to the GPS receiver is 40 [mA], DC 5 [V] is supplied to the GPS receiver from an AC adapter. DC 9 [V] is supplied to the control circuit of the 7-segment common cathode LED display circuit and Arduino individually from two AC adapters. The proposed control circuit might also be applicable to the large 7-segment common cathode LED display with the LED strip [9], [6]. Fig.3 illustrates the PCB display of the control circuit diagram in Fig.2 by Target3001! Fig.4 shows the 3D view of the PCB display in Fig.3 by Target3001! Fig.5 shows the PWB engraving trajectory preview of the control circuit in the 7-segment common cathode LED display GPS receiver clock based on the G-code file with the extension "cnc" in the web site https://ncviewer.com.
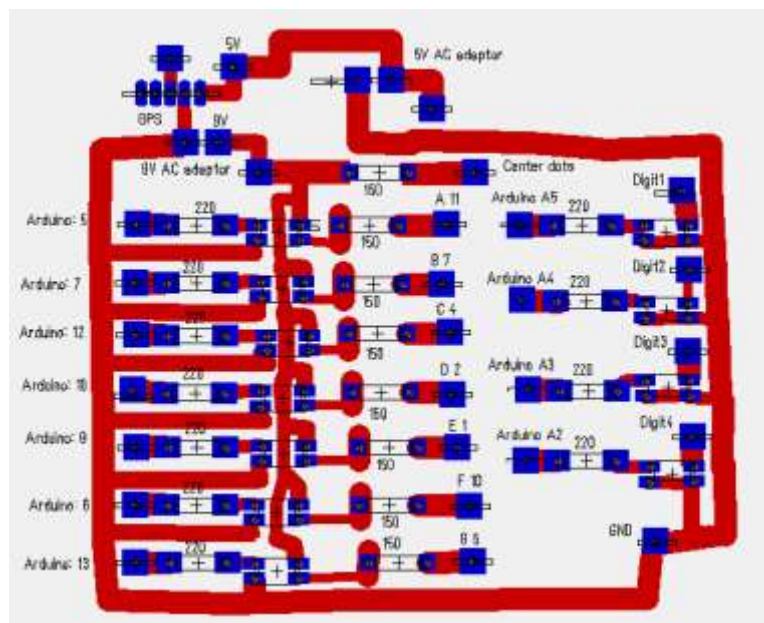

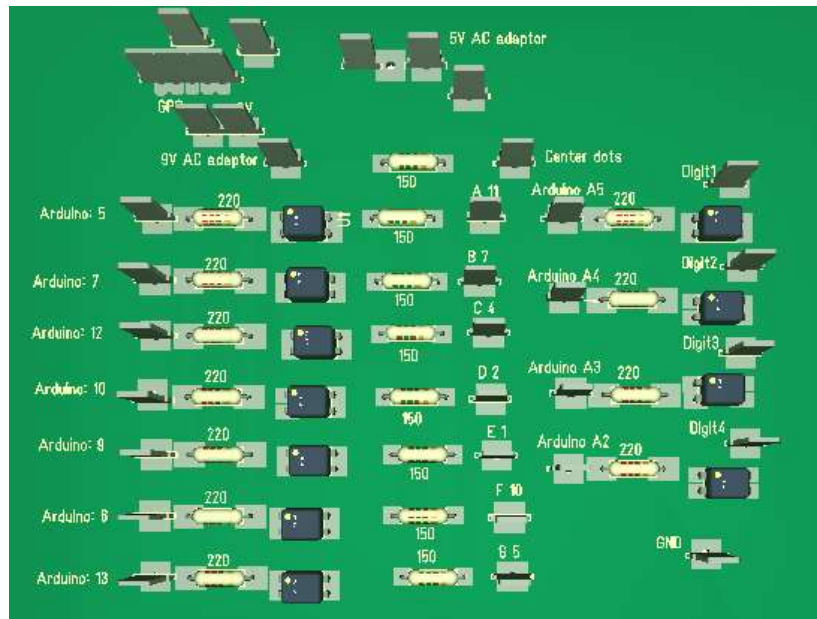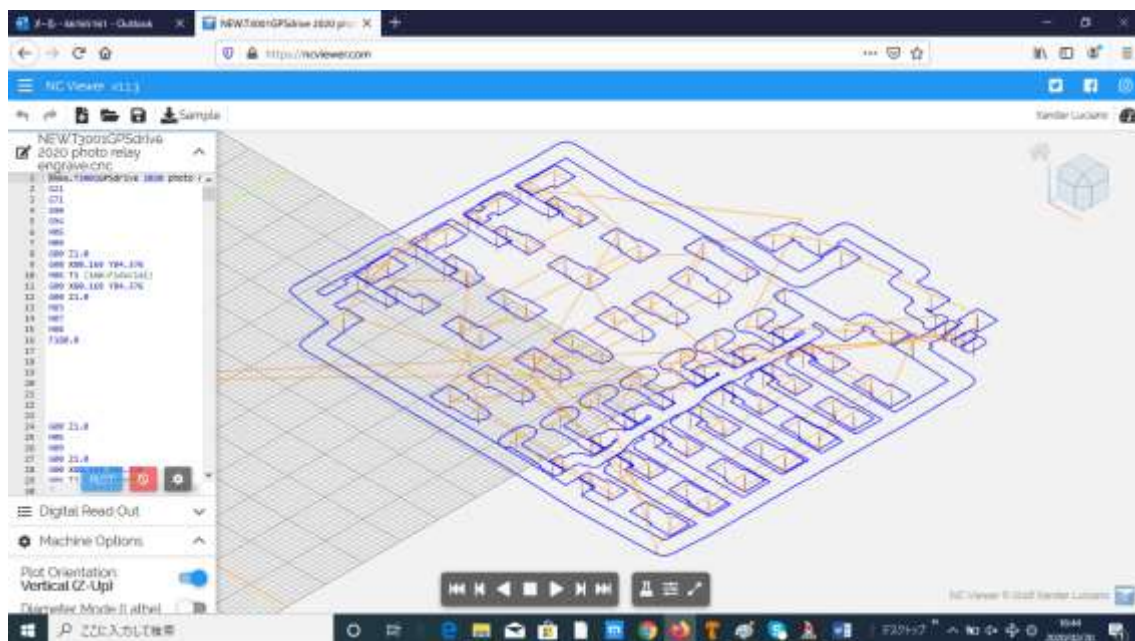Fig.3 PCB display of control circuit in 7-segment common cathode LED display GPS clock by Target3001!

Fig.4 3D view of control circuit by Target3001!

Fig.5 PWB engraving trajectory preview of control circuit in 7-segment common cathode LED display GPS receiver clock based on G-code with file extension "cnc". (Website: https://ncviewer.com)

4. Manufacturing 4-digit 7-segment common cathode LED display

Fig.6 illustrates the circuit diagram of the 4-digit 7-segment common cathode LED display by Target3001! Terminals "A" in Fig.6 are connected with each other. The same connections are applied to the segment terminals "B", "C", "D", "E", "F" and "G", respectively with each other. The "GND" terminal in Fig.6 is connected to the GND pin of Arduino. Fig.7 shows the PCB display of the 7-segment common cathode LED display by Target3001! Fig.8 shows the 3D view of the 7-segment common cathode LED display by Target3001! Fig.9 shows the engraving trajectory preview of the 7-segment common cathode LED display based on the G-code.
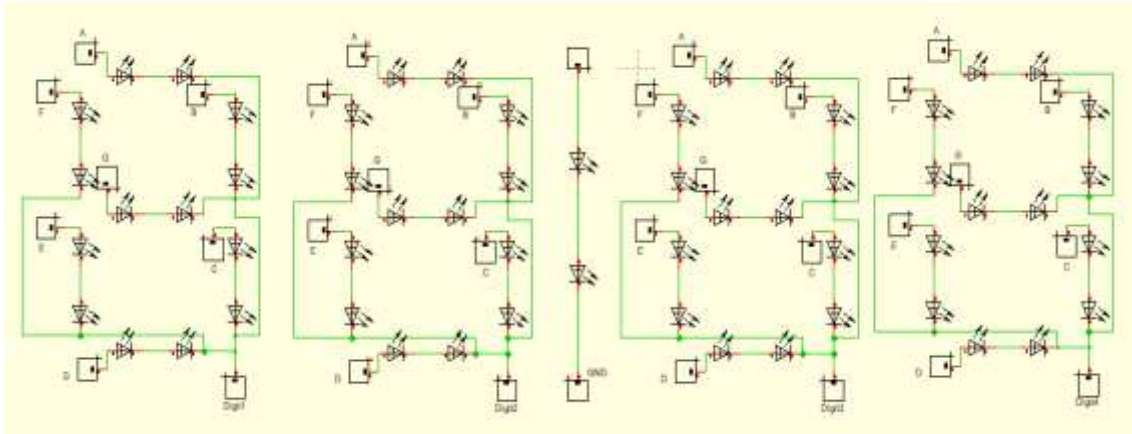
Fig.6 Circuit diagram of 4-digit 7-segment common cathode LED display drawn by Target3001!
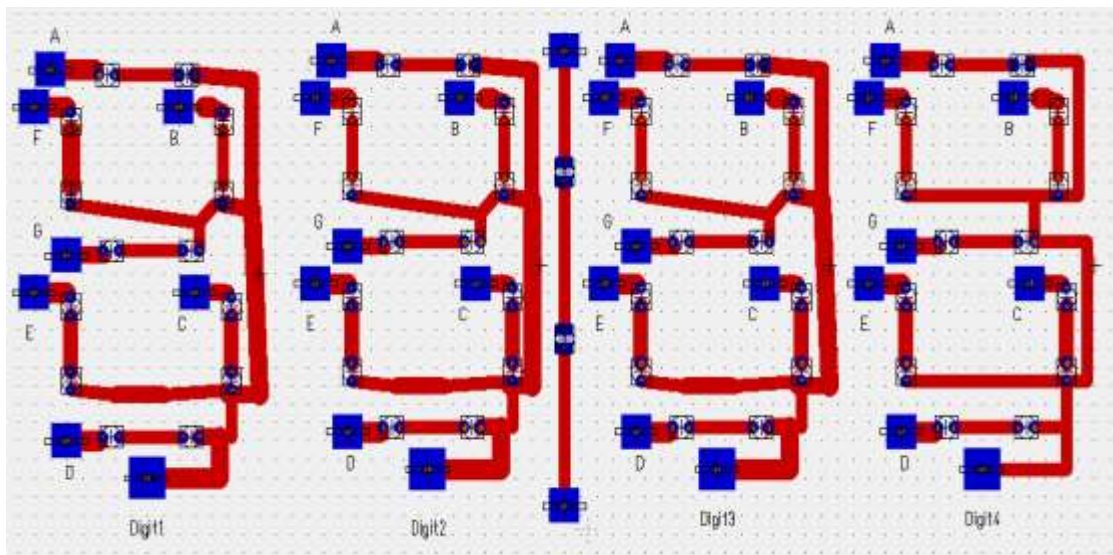

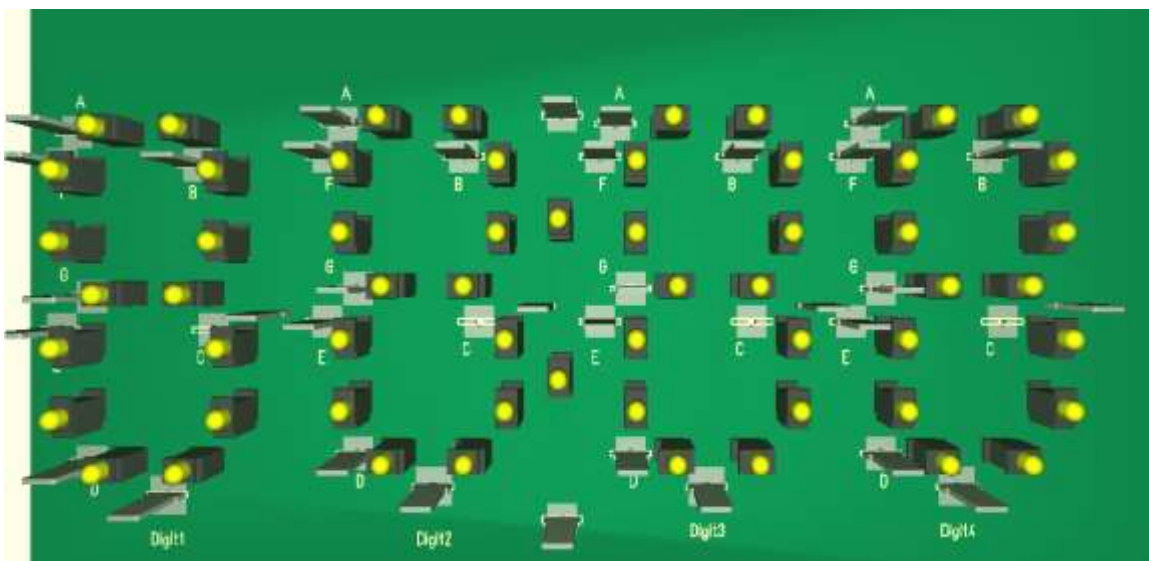Fig.7 PCB display of 4-digit 7-segment common cathode LED display by Target3001!

Fig.8 3D view of 4-digit 7-segment common cathode LED display by Target3001!
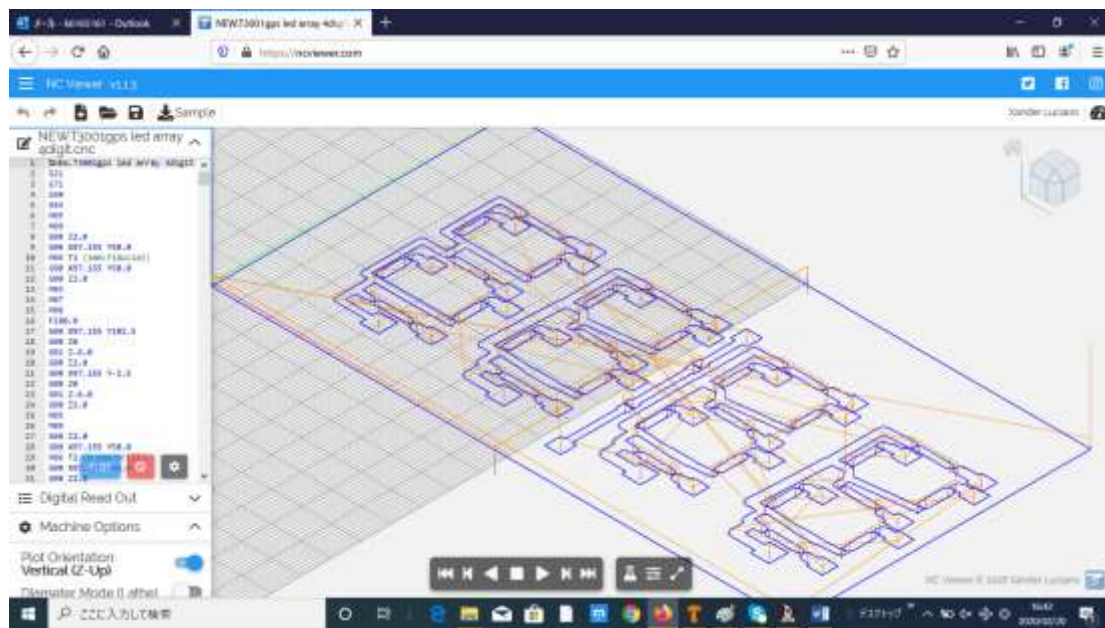
Fig.9 Engraving trajectory preview of 4-digit 7-segment common cathode LED display based on G-code generated by Target 3001! (Website: https://ncviewer.com)

Target3001! produces the G-code file with the extension "cnc" by the following operations. In the menu "Planning / general considerations": (1) click "File" of the menu selections; (2) select "Input / Output Formats"; (3) select "Production"; (4) select "Engraving insulation channels…"; (5) select "copper bottom only" in "Production type"; (6) select "CNC ISO 6983 / DIN 66025 (G-code Mach3)" in "output format". In "Drill", "Immersion depth of drills" is set to 3.00 (mm). In "Copper top", "Immersion depth of the engraving bit" is set to 0.37 (mm). If necessary, check the box "Erase rest copper top (Rub Out)". If checked, the rest copper top is engraved. In "Start output", click "Start", then the CNC G-code file is output to the assigned "Output file." Here, the copper foil thickness, used for the engraving by the CNC router, is 1.6 [mm].

5. Arduino sample program of 7-segment common cathode LED display GPS receiver clock

Fig.10 shows the sample program of the 7-segment common cathode LED display GPS receiver clock with the Arduino/Genuino Uno board. In the Arduino program, the libraries "SoftwareSerial.h" and "TinyGPS.h" are included. The 7-segment common cathode LED display shows the numbers 0-9 for each digit. From the left digit 1 to the right digit 4, each 7-segment LED display iterates lighting on/off quickly. If the 7-segment LED display for one digit lights and displays a certain number of hour or minute at the current time for a moment, the 7-segment LED displays for the other three digits are lighting off. It looks as if the 7-segment LED display indicates the current time in four digits at the same time by optical illusion. Also, as commented in the program, Japanese Standard Time (JST) is obtained by adding 9 hours to Coordinated Universal Time (CUT). Due to the value "10" in the statement "if (ii % 10 == 0)" in the Arduino program, almost every four seconds, the 7-segment LED display in every digit lights off instantaneously. Since the GPS signal is sent to Arduino every second from the GPS receiver, the maximum delay of time, displayed, is almost four seconds. By adjusting the power-on timing of the clock so that the end of the 4-second interval matches the minute change timing, the time will be almost accurate.

// 7 segment LED GPS receiver clock

// Sample program by Seiichi Nakamori, February 28, 2020

#include <SoftwareSerial.h>

#include <TinyGPS.h>

// NMEA parser TinyGPS

```
TinyGPS gps;
// Softwareserial(RX=2, TX=3) Connect RXD pin of GPS receiver to Arduino digital pin 2 (no connection of TXD
pin of GPS receiver)
SoftwareSerial nss(2, 3);
int h1;
int h2;
int hourj;
int m1;
int m2;
int hour;
int minute;
const int anode_a = 5; // Arduino pin number related to LED segment A (see Fig.7)
const int anode_b = 7; // Arduino pin number related to LED segment B (see Fig.7)
const int anode_c = 12;
const int anode_d = 10;
const int anode_e = 9;
const int anode_f = 6;
const int anode_g = 13;
const int cathode_pin1 = A5; // Arduino pin number related to digit 1
const int cathode_pin2 = A4; // Arduino pin number related to digit 2
const int cathode_pin3 = A3; // Arduino pin number related to digit 3
const int cathode_pin4 = A2; // Arduino pin number related to digit 4
// relationship between segment data and numbers from 0 to 9
const int digits[] = {
0b00111111, // 0
0b00000110, // 1
0b01011011, // 2
0b01001111, // 3
0b01100110, // 4
0b01101101, // 5
0b01111101, // 6
0b00100111, // 7
0b01111111, // 8
0b01101111, // 9
};
// display of numbers for each digit
```

```
void display_number (int n) {
// digits[n] lighting or turning lights off of each segment in accordance with each bit
digitalWrite(anode_a, digits[n] & 0b00000001 ? HIGH : LOW);
digitalWrite(anode_b, digits[n] & 0b00000010 ? HIGH : LOW);
digitalWrite(anode_c, digits[n] & 0b00000100 ? HIGH : LOW);
digitalWrite(anode_d, digits[n] & 0b00001000 ? HIGH : LOW);
digitalWrite(anode_e, digits[n] & 0b00010000 ? HIGH : LOW);
digitalWrite(anode_f, digits[n] & 0b00100000 ? HIGH : LOW);
digitalWrite(anode_g, digits[n] & 0b01000000 ? HIGH : LOW);
}
void setup () {
pinMode(anode_a, OUTPUT);
pinMode(anode_b, OUTPUT);
pinMode(anode_c, OUTPUT);
pinMode(anode_d, OUTPUT);
pinMode(anode_e, OUTPUT);
pinMode(anode_f, OUTPUT);
pinMode(anode_g, OUTPUT);
pinMode(cathode_pin1, OUTPUT);
pinMode(cathode_pin2, OUTPUT);
pinMode(cathode_pin3, OUTPUT);
pinMode(cathode_pin4, OUTPUT);
Serial.begin(115200);
nss.begin(9600);
}
void loop()
{
int year;
byte month, day, hour, minute, second, hundredths;
bool newdata = false;
// while there exist GPS data received
while(nss.available()){
char c = nss.read();
// send each character received to serial monitor
// Serial.print(c);
// analyze each character by encode()
```

```
if(gps.encode(c)){

newdata = true;

}

}

if(newdata){

// clear 7 segment display

// get date and time

for (int ii=0; ii <= 100; ii++){

if (ii % 10 == 0) {

gps.crack_datetime(&year, &month, &day, &hour, &minute, &second, &hundredths);

} else {

digitalWrite(cathode_pin1, LOW);

digitalWrite(cathode_pin2, LOW);

digitalWrite(cathode_pin3, LOW);

digitalWrite(cathode_pin4, LOW);

digitalWrite(cathode_pin1, LOW);

digitalWrite(cathode_pin2, LOW);

digitalWrite(cathode_pin3, LOW);

digitalWrite(cathode_pin4, LOW);

// }

delay(5);

digitalWrite(cathode_pin1, LOW);

digitalWrite(cathode_pin2, LOW);

digitalWrite(cathode_pin3, LOW);

digitalWrite(cathode_pin4, LOW);

delay(5);

hourj=(hour+9) % 24; // Japanese Standard Time (JST)=Coordinated Universal Time (CUT)+9

// residual residue

h1=hourj / 10; // the first digit of hour

h2= hourj % 10; // the second digit of hour

m1=minute / 10; // the first digit of minute

m2=minute % 10; // the second digit of minute

int okure=5;

int ite=5;

int okure1=1;

long int i;
```

```
for(i=0;i<ite;i++);{

digitalWrite(cathode_pin1,  LOW);

delay(okure);

digitalWrite(cathode_pin1,  HIGH);

display_number(h1);

delay(okure);

digitalWrite(cathode_pin1,  LOW);

}

for(i=0;i<ite;i++);{

digitalWrite(cathode_pin2,  HIGH);

display_number(h2);

delay(okure);

digitalWrite(cathode_pin2,  LOW);

}

for(i=0;i<ite;i++);{

digitalWrite(cathode_pin3,  HIGH);

display_number(m1);

delay(okure);

digitalWrite(cathode_pin3,  LOW);

}

for(i=0;i<ite;i++);{

digitalWrite(cathode_pin4,  HIGH);

display_number(m2);

delay(okure);

digitalWrite(cathode_pin4,  LOW);

}

}

}

}

}
```

Fig.10 Arduino sample program of 7-segment common cathode LED display GPS receiver clock.

Fig.11 shows the overall views of the 7-segment cathode common LED display GPS receiver clock.   The arrangement of the components in Fig.11 might be compared with those in the 3D views of Fig.4 and Fig.8. In Fig.4 and Fig.8, the package of the blade terminal is used as the terminals in Fig.2 and Fig.6.
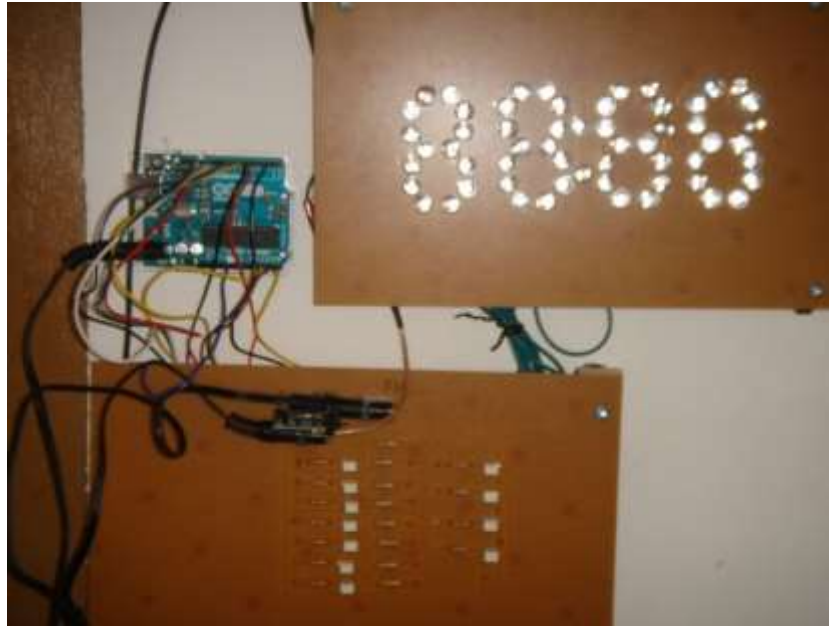
Fig.11 Overall views of 7-segment common cathode LED display GPS receiver clock.

6. Conclusions

This paper has newly proposed the 7-segment common cathode LED display GPS receiver clock. The 7-segment common cathode LED display GPS receiver clock consists of the control circuit with the photo relay and the resistance, the 7-segment common cathode LED display, Arduino and the GPS receiver. In section 2, the N-Channel MOSFET at the high side does not work properly to light on the two LEDs connected in series. Instead of the N-Channel MOSFETs, this paper, by using 11 photo relays, has proposed the control circuit diagram of the 7-segment common cathode LED display GPS receiver clock to light on/off the two LEDs, connected in series, per each segment. From the standard turn-on and turn-off times of the photo relay TLP222AF, the photo relay is suitable for lighting on/off the LEDs in high speed by Arduino. The electric current of the trigger LED in TLP222AF is 3 [mA], the maximum value of the electric current, output from respective Arduino pin, is 50 [mA] and the maximum value of the sum of the electric currents output from Arduino is 100 [mA]. From these facts, TLP222AF is suitable for being used in the control circuit of the 7-segment common cathode LED display GPS receiver clock.

Finally, for the purpose of manufacturing large 7-segment common cathode LED display GPS receiver clock, the proposed 7-segment LED display control circuit diagram in Fig.2 might be combined with that of the 7-segment common cathode LED display with the LED strip. Depending on the LED strip used, we select DC 12 [V] or 24[V] AC adapters.

References

1. "7-Segment GPS Receiver Clock," https://easyeda.com/amohan/7_SEGMENT_GPS_RECEIVER_CLOCK-61ed8e2345e7414b924c234028a30acf

2. "Arduino GPS Clock," https://circuitdigest.com/microcontroller-projects/arduino-gps-clock

3. "Arduino GPS Real Time Clock with NEO-6M Module," https://simple-circuit.com/arduino-gps-real-time-clock-neo-6m/

4. "Digital Temperature Sensor with 4 Digit 7-Segment Display," https://haroonwardak1995.wordpress.com/2016/03/09/digital-temperature-sensor-with-4-digit-7-segment-display/

5. "GPS.GOV," https://www.gps.gov/

6. "High CRI LED Strip Lights 12V & 24V DC Flexible Tape Featuring 95 CRI & 90 R9," https://www.waveformlighting.com/high-cri-led-strip-lights

7. "How to Display Latitude and Longitude on an LCD using Tracking Kit (GPRS + GPS)," https://www.cooking-hacks.com/documentation/tutorials/how-to-display-latitude-and-longitude-on-lcd-gprs-gps/

8. "Large 7 Segment LED Pace Clock," https://www.instructables.com/id/Large-7-Segment-LED-Pace-Clock/

9. **"LED Strip Light Internal Schematic and Voltage Information," https://www.waveformlighting.com/pcb-designs/led-strip-light-schematic-and-voltage-information**

10. "Mini PCB Milling Machine, CNC Router Kit CNC 3018, Rattmmotor," https://www.amazon.co.jp/CNC%E3%83%AB%E3%83%BC%E3%82%BF%E3%82%AD%E3%83%83%E3%83%88-CNC3018-ER11-%E3%83%9F%E3%83%8B%E3%83%95%E3%83%A9%E3%82%A4%E3%82%B9%E7%9B%A4-USB%E3%83%87%E3%82%B9%E3%82%AF%E3%83%88%E3%83%83%E3%83%97%E5%BD%AB%E5%88%BB%E6%A9%9F%E3%80%81%E6%9C%A8%E6%9D%90%E3%80%81%E6%9C%A8%E5%B7%A5%E7%94%A8/dp/B06Y2HWMTH

11. "N-Channel E-Type MOSFET Switching LED's,"http://iamtechnical.com/n-channel-e-type-mosfet-ONing-leds

12. O. Iijima, GPS radio clock, *Arduino Sample Programs*, pp.118-140, Sanbariki inc., 2018   (in Japanese).

13. "Photocoupler Photorelay: TLP222AF," http://akizukidenshi.com/catalog/g/gI-07330/

14. "Play with Arduino-High-Precision Clock using GPS," http://arms22.blog91.fc2.com/blog-entry-313.html

15. S. Tanji, *Big clock by using 172 LEDs, Latest Use of LED Work Guide*, pp. 128-145, Dempa Publications, Inc., 2008   (in   Japanese). https://www.amazon.co.jp/%E6%9C%80%E6%96%B0LED%E6%B4%BB%E7%94%A8%E5%B7%A5%E4%BD%9C%E3%82%AC%E3%82%A4%E3%83%89-%E9%81%A0%E8%97%A4-%E6%95%8F%E5%A4%AB/dp/4885549515

16. "Target 3001!," https://ibfriedrich.com/en/index.html