

DAFTAR PUSTAKA

- Aditya, D. (2013). *Metodologi Penelitian*. Surakarta: Poltekkes Kemenkes.
- Anonymous. (n.d.). *Arduino*. Retrieved 3 2018, 2, from Wikipedia Ensiklopedia Bebas: <https://id.wikipedia.org/wiki/Arduino>
- Darmawan, S. D. (2008). Rancang Bangun Jam Digital Waktu Shalat Berbasis Mikrokontroler AT89S52. *UNDIP Tembalang*.
- Depag. (2008). *Al-Qur'an dan Terjemahannya*. Jakarta: Departemen Agama RI (Al Waah).
- Departemen Pendidikan Nasional. (2008). *Kamus Besar Bahasa Indonesia*. Jakarta: PT Gramedia Pustaka.
- Efendi, I. (2014). *Pengertian dan Kelebihan Arduino*. Retrieved 4 2, 2018, from IT-Jurnal.com: <https://www.it-jurnal.com/pengertian-dan-kelebihan-arduino/>
- Efendi, I. (n.d.). *Pengertian dan Kelebihan Arduino*. Retrieved 3 28, 2018, from IT JURNAL: <https://www.it-jurnal.com/pengertian-dan-kelebihan-arduino/>
- Fathulloh, A. G. (2015). *Irsyadul Murid*. Sampang: Lafal.
- Isa, A., Zakaria, Z., Johar, F., & Othman, A. (2007). Mobile Prayer Times and Qiblat Direction Using GPS. *Apace*, 1-5.
- Mujiono. (2015, 10). *Pengertian dan Fungsi Bluetooth*. Retrieved 3 28, 2018, from Teori Komputer: <http://www.teorikomputer.com/2015/10/pengertian-dan-fungsi-bluetooth.html>
- Nufail, M. (2012). *Kasyful Qomaroin*. Pasuruan: Al-Falah Print.
- Peter A Crossley, H. G. (2016). Time Synchronization for Transmission Substations Using GPS and IEEE 1588. *CSEE JOURNAL OF POWER AND ENERGY SYSTEMS*, Vol.2, 91.
- Ratnasari, Y. (2015). *Arduino and Raspberry Pi*. Retrieved 4 2018, 1, from Academia: http://www.academia.edu/23310766/Arduino_and_Raspberry_Pi
- Safryanti, E., Feranita, & Ardiles, R. (2008). Rancang Bangun Indikator Jam Sholat Abadi Menggunakan Atmel 89S52. *Seminar Nasional Teknik Kimia Oleo & Petrokimia Indonesia*, 1.
- Schille, J., & Voisar, A. (2004). *Location-Based Services*. San Francisco: Morgan Kaufmann.

- Siregar, R. M., Jabar, A., & Usman, A. (2016). Perancangan Perangkat Keras Sistem Pengingat Waktu Sholat Elektronik. *Sekolah Tinggi Teknik Harapan Medan*, 1.
- Sutinah, E. (2016). Flowchart Sistem Informasi Penjadwalan Waktu Sholat Berbasis Mikrokontroler Atmega16. *INFORMATICS FOR EDUCATORS AND PROFESSIONALS VOL. 13*, 37-49.
- Yu-Ling Hsueh, H.-C. C. (2017). Map Matching for Low-Sampling-Rate GPS Trajectories by Exploring Real-time Moving Directions. *Journal of Information Sciences*, 1-27.
- Yusuf, A. M. (2014). *Metode Penelitian: Kuantitatif, Kualitatif dan Penelitian Gabungan*. Jakarta: Kencana.
- Yusuf, R. M. (2017, November 10). *Realtime Clock*. Retrieved March 29, 2018, from Scribd, Read Anywhere Anytime: <https://www.scribd.com/doc/22362718/Real-Time-Clock>
- Zakiyudin, A. (2012). *Sistem Informasi Manajemen*. Bogor: Mitrawacanamedia.

Lampiran 1 : Tabel *Pin Mapping on Board Arduino Mega*

No. Pin	Nama Pin	Nama Pin di Map
1	PG5 (OC0B)	Digital pin 4 (PWM)
2	PE0 (RXD0/PCINT8)	Digital pin 0 (RX0)
3	PE1 (TXD0)	Digital pin 1 (TX0)
4	PE2 (XCK0/AIN0)	
5	PE3 (OC3A/AIN1)	Digital pin 5 (PWM)
6	PE4 (OC3B/INT4)	Digital pin 2 (PWM)
7	PE5 (OC3C/INT5)	Digital pin 3 (PWM)
8	PE6 (T3/INT6)	
9	PE7 (CLK0/ICP3/INT7)	
10	VCC	VCC
11	GND	GND
12	PH0 (RXD2)	Digital pin 17 (RX2)
13	PH1 (TXD2)	Digital pin 16 (TX2)
14	PH2 (XCK2)	
15	PH3 (OC4A)	Digital pin 6 (PWM)
16	PH4 (OC4B)	Digital pin 7 (PWM)
17	PH5 (OC4C)	Digital pin 8 (PWM)
18	PH6 (OC2B)	Digital pin 9 (PWM)
19	PB0 (SS/PCINT0)	Digital pin 53 (SS)
20	PB1 (SCK/PCINT1)	Digital pin 52 (SCK)
21	PB2 (MOSI/PCINT2)	Digital pin 51 (MOSI)
22	PB3 (MISO/PCINT3)	Digital pin 50 (MISO)
23	PB4 (OC2A/PCINT4)	Digital pin 10 (PWM)
24	PB5 (OC1A/PCINT5)	Digital pin 11 (PWM)
25	PB6 (OC1B/PCINT6)	Digital pin 12 (PWM)
26	PB7 (OC0A/OC1C/PCINT7)	Digital pin 13 (PWM)
27	PH7 (T4)	
28	PG3 (TOSC2)	
29	PG4 (TOSC1)	
30	RESET	RESET
31	VCC	VCC
32	GND	GND
33	XTAL2	XTAL2
34	XTAL1	XTAL1
35	PL0 (ICP4)	Digital pin 49
36	PL1 (ICP5)	Digital pin 48
37	PL2 (T5)	Digital pin 47
38	PL3 (OC5A)	Digital pin 46 (PWM)
39	PL4 (OC5B)	Digital pin 45 (PWM)
40	PL5 (OC5C)	Digital pin 44 (PWM)

41	PL6	Digital pin 43
42	PL7	Digital pin 42
43	PD0 (SCL/INT0)	Digital pin 21 (SCL)
44	PD1 (SDA/INT1)	Digital pin 20 (SDA)
45	PD2 (RXDI/INT2)	Digital pin 19 (RX1)
46	PD3 (TXD1/INT3)	Digital pin 18 (TX1)
47	PD4 (ICP1)	
48	PD5 (XCK1)	
49	PD6 (T1)	
50	PD7 (T0)	Digital pin 38
51	PG0 (WR)	Digital pin 41
52	PG1 (RD)	Digital pin 40
53	PC0 (A8)	Digital pin 37
54	PC1 (A9)	Digital pin 36
55	PC2 (A10)	Digital pin 35
56	PC3 (A11)	Digital pin 34
57	PC4 (A12)	Digital pin 33
58	PC5 (A13)	Digital pin 32
59	PC6 (A14)	Digital pin 31
60	PC7 (A15)	Digital pin 30
61	VCC	VCC
62	GND	GND
63	PJ0 (RXD3/PCINT9)	Digital pin 15 (RX3)
64	PJ1 (TXD3/PCINT10)	Digital pin 14 (TX3)
65	PJ2 (XCK3/PCINT11)	
66	PJ3 (PCINT12)	
67	PJ4 (PCINT13)	
68	PJ5 (PCINT14)	
69	PJ6 (PCINT 15)	
70	PG2 (ALE)	Digital pin 39
71	PA7 (AD7)	Digital pin 29
72	PA6 (AD6)	Digital pin 28
73	PA5 (AD5)	Digital pin 27
74	PA4 (AD4)	Digital pin 26
75	PA3 (AD3)	Digital pin 25
76	PA2 (AD2)	Digital pin 24
77	PA1 (AD1)	Digital pin 23
78	PA0 (AD0)	Digital pin 22
79	PJ7	
80	VCC	VCC
81	GND	GND
82	PK7 (ADC15/PCINT23)	Analog pin 15
83	PK6 (ADC14/PCINT22)	Analog pin 14

84	PK5 (ADC13/PCINT21)	Analog pin 13
85	PK4 (ADC12/PCINT20)	Analog pin 12
86	PK3 (ADC11/PCINT19)	Analog pin 11
87	PK2 (ADC10/PCINT18)	Analog pin 10
88	PK1 (ADC9/PCINT17)	Analog pin 9
89	PK0 (ADC8/PCINT16)	Analog pin 8
90	PF7 (ADC7)	Analog pin 7
91	PF6 (ADC6)	Analog pin 6
92	PF5 (ADC5/TMS)	Analog pin 5
93	PF4 (ADC4/TMK)	Analog pin 4
94	PF3 (ADC3)	Analog pin 3
95	PF2 (ADC2)	Analog pin 2
96	PF1 (ADC1)	Analog pin 1
97	PF0 (ADC0)	Analog pin 0
98	AREF	Analog Reference
99	GND	GND
100	AVCC	VCC

Sumber : Datasheet Atmel-ATmega2560

Lampiran 2 : *Listing Code Jam Digital Jadwal Waktu Sholat*

```

#include <DFPlayer_Mini_Mp3.h>
#include <EEPROM.h>
#include <TimerOne.h>
#include <DMD.h>
#include <Wire.h>
#include <RTClib.h>
#include <SystemFont5x7.h>
#include <Small4x7.h>
#include <angka6x13.h>
#include <Bernard16.h>
#include <SPI.h>
#include <TinyGPS.h>
#define PIN_BUZZER 2

//AltSoftSerial altSerial;
//Public Declarations
uint8_t   reset_x;
byte      hour,minute,second, day, month, Tgl_H, Bln_H,WSNow;
int       Thn_H, year, countdown;
uint8_t   harpas[2],Run,cpt;
float     Waktu_Sholat[8],saikif,JedaIqomah,LamaSholat;
char *    DispWaktu;//,* infone;
char      infone[150];

//Structure of Variable
struct struct_param{ // loaded to EEPROM
  uint8_t state; // 1 byte add 0
  float Lt; // 4 byte add 1
  float Bj; // 4 byte add 5
  float TT; // 4 byte add 9
  uint8_t TZ; // 1 byte add 13
  uint8_t ihtiyath[5]; // 5 byte add 14
  float inkhifad[3]; // 12 byte add 19
  uint8_t Iqm[10]; // 10 byte add 31
  uint8_t Mrt[10]; // 10 byte add 41
  uint8_t Cerah; // 1 byte add 51
};
struct_param Prm;
DMD Disp(3,1);
TinyGPS gps;
RTC_DS3231 rtc;
void setup() {
  cpt=50;
  getPrm();
  Disp_init();
  Serial.begin(115200);
  Serial2.begin(4800);
  Serial3.begin(9600);
  mp3_set_serial (Serial3);
  rtc.begin();
  Wire.begin();
  delay(200);
  mp3_set_volume (30);
  Run=1;
  Hitung();
  Serial.print(Prm.Cerah);
  pinMode(PIN_BUZZER, OUTPUT);
  digitalWrite(PIN_BUZZER, HIGH);
}

void loop() {

```

```

SetPrm();
updateWaktu();
Disp.clear();
PlayTartil();
// if(!((harpas[0]==3 || harpas[0]==5) && hour >=22 || (harpas[0]==4 ||
harpas[0]==6) && hour <2)){
    if(Run==1){
        char out[500];
        sprintf(out,"%s | %s",DispWaktu, infone);
        dwMrq(out,cpt);
    }else if(Run==2){
        tampilInfo2();
    }else if(Run==3){
        ClrScr();
    }
// }else{
//     ClrScr();
// }
}
void ClrScr(){
    Disp.clear();
    Disp.swapBuffers();
}
void PlayTartil(){
    if(saikif<0.0006f) Hitung();
    uint8_t WSNext = (WSNow+1)%8;
    if(WSNext != 4 && !(WSNext > 5)){
        uint8_t idx_Mrt = WSNext==5?4:WSNext;
        if(saikif >= (Waktu_Sholat[WSNext]-Prm.Mrt[idx_Mrt]/60.0f) && saikif <
(Waktu_Sholat[WSNext]-Prm.Mrt[idx_Mrt]/60.0f+0.0003f))
mp3_play(Prm.Mrt[idx_Mrt+5]);
    }
    if(saikif >= Waktu_Sholat[WSNext] && saikif < (Waktu_Sholat[WSNext]+0.0006f)){
        if(WSNext==2) Hitung();
        WSNow = WSNext;
        if(WSNow != 4 && !(WSNow > 5)) {
            uint8_t idxIqm = WSNow<=3?0:1;
            JedaIqomah=(float)Prm.Iqm[WSNow+idxIqm]/60.0f;
            LamaSholat=(float)Prm.Iqm[WSNow+idxIqm+4]/60.0f;
        }
        Run=2;
    }
}

void UpdateGPS(){
    bool newData = false;
    // For one second we parse GPS data and report some key values
    for (unsigned long start = millis(); millis() - start < 800;){
        while (Serial2.available()){
            char c = Serial2.read();
            // Serial.write(c); // uncomment this line if you want to see the GPS data
            flowing
            if (gps.encode(c)) // Did a new valid sentence come in?
                newData = true;
        }
    }
    if (newData){
        unsigned long gps_age;
        int gps_year;
        byte gps_month, gps_day, gps_hour, gps_minute, gps_second, gps_hundredths;
        gps.crack_datetime(&gps_year, &gps_month, &gps_day,
            &gps_hour, &gps_minute, &gps_second, &gps_hundredths, &gps_age);
        Prm.TT = gps.f_altitude();
    }
}

```

```

    gps.f_get_position(&Prm.Lt, &Prm.Bj, &gps_age);
    EEPROM.put(1,Prm.Lt);
    EEPROM.put(5,Prm.Bj);
    EEPROM.put(9,Prm.TT);
    if(Prm.TZ != (uint8_t)Prm.Bj/15){
        Prm.TZ = (byte)(Prm.Bj/15);
        EEPROM.put(13,Prm.TZ);
    }
    DateTime utc = DateTime(gps_year, gps_month, gps_day, gps_hour, gps_minute,
gps_second);
    DateTime now(utc.unixtime() + Prm.TZ * 3600);
    rtc.adjust(now.unixtime());
}
}
void updateWaktu(){
    if(millis() % 60000 > 59980 || millis()<4000) UpdateGPS();
//Update GPS interval 10 menit
    DateTime now = rtc.now();
    hour = now.hour();
    minute = now.minute();
    second = now.second();
    day = now.day();
    month = now.month();
    year = now.year();
    saikif = hour+minute/60.0f+second/3600.0f;
}

void scan() { Disp.refresh();}
void Disp_init() {
    Disp.setDoubleBuffer(true);
    Timer1.initialize(2000);
    Disp.setBrightness(Prm.Cerah);
    Timer1.attachInterrupt(scan);
    Disp.clear();
    Disp.swapBuffers();
}

//Fungsi Trigonometri=====
float mtSin(float nilai){
    return (float)sin(nilai/180*3.141592654f);
}
float mtCos(float nilai){
    return (float)cos(nilai/180*3.141592654f);
}
float mtTan(float nilai){
    return (float)tan(nilai/180*3.141592654f);
}
float mtASin(float nilai){
    return (float)asin(nilai)/3.141592654f*180;
}
float mtACos(float nilai){
    return (float)acos(nilai)/3.141592654f*180;
}
float mtATan(float nilai){
    return (float)atan(nilai)/3.141592654f*180;
}
//=====

char* Konversi(byte number){ // Konversi angka agar ada nol di depannya jika di
bawah 10
    static char out[2];
    sprintf(out,"%02d",number);
    return out;
}

```



```

}

void floatToHourMin(float val, byte &hourS, byte &minuteS){
    hourS = floor(val);
    byte menitS = floor((val-hourS)*60);
    minuteS = ((val-hourS)*60-menitS)>=0.5?menitS+1:menitS;
}

byte ProsesManual(byte bulan, int tahun){
    byte HariAwal;
    if(bulan-1 == 0){
        bulan=12;tahun-=1;
    }else{
        bulan-=1;
    }
    float Dip = 1.76 / 60.0f * sqrt(Prm.TT);
    float HY = tahun + (bulan * 29.53) / 354.3671f;
    float K = floor(0.5+(HY-1410)*12);
    float JA = K / 1200.0f;
    float Tambahan_JD = 2447740;
    float JD = 0.652f + 29.53058868f * K + 0.0001178f * JA*JA;
    float M = fmod(207.9587074 + 29.10535608 * K - 0.00003333 * JA*JA, 360);
    float Mi = fmod(111.1791307f + 385.81691806f * K + 0.0107306f * JA*JA, 360);
    float F = fmod(164.2162296 + 390.67050646 * K - 0.0016528 * JA*JA, 360);
    float t[14];
    t[1] = (0.1734 - 0.000395f * JA) * mtSin(M);
    t[2] = 0.0021 * mtSin(2 * M);
    t[3] = -0.4068 * mtSin(Mi);
    t[4] = 0.0161 * mtSin(2 * Mi);
    t[5] = -0.0004 * mtSin(3 * Mi);
    t[6] = 0.0104 * mtSin(2 * F);
    t[7] = -0.0051 * mtSin(M + Mi);
    t[8] = -0.0074 * mtSin(M - Mi);
    t[9] = 0.0004 * mtSin(2 * F + M);
    t[10] = -0.0004 * mtSin(2 * F - M);
    t[11] = -0.0006 * mtSin(2 * F + Mi);
    t[12] = 0.001 * mtSin(2 * F - Mi);
    t[13] = 0.0005 * mtSin(M + 2 * Mi);
    t[0]=0;
    for(byte i=1;i<14;i++) t[0]+=t[i];
    float JD_Ijt = JD + 0.5 + t[0] + Prm.TZ / 24.0f;
    float WI_WD = (JD_Ijt - floor(JD_Ijt))*24.0f;
    float MailS, EoT, SDM, KWD;

    float JDne = floor(floor(JD_Ijt)-0.5)/36525.0f;
    float JDne2= (17-(float)Prm.TZ+12)/24.0f/36525.0f;
    float JDne3= 3805/36525.0f;
    float T = JDne + JDne2 - JDne3;

    Data_Matahari(T, MailS, EoT, SDM, KWD);
    float Maghrib = 12 + mtACos(-mtTan(Prm.Lt) * mtTan(MailS) + mtSin(-
(SDM+0.575f+0.29333f*sqrt(Prm.TT))) / mtCos(Prm.Lt) / mtCos(MailS))/15 - KWD;

    //Cari Hari Rukyah
    byte KorH=0;
    if(WI_WD > Maghrib && WI_WD < 24) KorH=1;
    HariAwal = fmod(floor(JD_Ijt) + Tambahan_JD + KorH+2,7);

    JDne = floor(floor(JD_Ijt+KorH)-0.5)/36525.0f;
    JDne2= (Maghrib-(float)Prm.TZ+12)/24.0f/36525.0f;
    JDne3= 3805/36525.0f;
    T = JDne + JDne2 - JDne3;

```

```
//DATA MATAHARI SAAT GHURUB HARI IJTIMA'
float S = fmod((280.46645 + 36000.76983f * T), 360);
float m = fmod((357.5291 + 35999.0503 * T), 360);
float N = fmod((125.04 - 1934.136f * T), 360);
float Ki = (17.264 / 3600.0) * mtSin(N) + (0.206 / 3600.0) * mtSin(2 * N);
float Kii = (-1.264 / 3600.0) * mtSin(2.0 * S);
float Ri = (9.23 / 3600.0) * mtCos(N) - (0.09 / 3600.0) * mtCos(2 * N);
float Rii = (0.548 / 3600.0) * mtCos(2.0 * S);
float Qi = 23.43929111 + Ri + Rii - (46.815 / 3600.0) * T;
float E = (6898.06 / 3600.0) * mtSin(m) + (72.095 / 3600.0) * mtSin(2.0 * m) +
(0.966 / 3600.0) * mtSin(3.0 * m);
float Si = S + E + Ki + Kii - 20.47 / 3600.0;
MailS = mtASin(mtSin(Si) * mtSin(Qi));
float PT = mtATan(mtTan(Si) * mtCos(Qi));
PT += Si<90?0:Si<270?180:360;
EoT = (S - PT) / 15;
SDM = 0.267 / (1 - 0.017 * mtCos(m));
float H = -(SDM + (34.5 / 60.0f) + Dip);
float nisfu_qousinnahar = mtACos(-mtTan(Prm.Lt) * mtTan(MailS) + mtSin(H) /
mtCos(Prm.Lt) / mtCos(MailS));
```

```
//DATA BULAN SAAT GHURUB HARI IJTIMA
float eM = fmod( 218.31617 + 481267.88088 * T,360);
float Aa = fmod(134.96292 + 477198.86753 * T, 360);
float eF = fmod(93.27283 + 483202.01873 * T, 360);
float De = fmod(297.85027 + 445267.11135 * T, 360);
t[0] = 0;
t[1] = (22640 / 3600.0f) * mtSin(Aa);
t[2] = (-4586 / 3600.0f) * mtSin(Aa - 2 * De);
t[3] = (2370 / 3600.0f) * mtSin(2 * De);
t[4] = (769 / 3600.0f) * mtSin(2 * Aa);
t[5] = (-668 / 3600.0f) * mtSin(m);
t[6] = (-412 / 3600.0f) * mtSin(2 * eF);
t[7] = (-212 / 3600.0f) * mtSin(2 * Aa - 2 * De);
t[8] = (-206 / 3600.0f) * mtSin(Aa + m - 2 * De);
t[9] = (192 / 3600.0f) * mtSin(Aa + 2 * De);
t[10] = (-165 / 3600.0f) * mtSin(m - 2 * De);
t[11] = (148 / 3600.0f) * mtSin(Aa - m);
t[12] = (-125 / 3600.0f) * mtSin(De);
t[13] = (-110 / 3600.0f) * mtSin(Aa + m);
t[14] = (-55 / 3600.0f) * mtSin(2 * eF - 2 * De);
for(byte i=1;i<15;i++) t[0]+=t[i];

float Mo = (eM + t[0] + Ki + Kii - (20.47 / 3600.0f));
float Ai = Aa + t[2] + t[3] + t[5];
float Li = (18461 / 3600.0f) * mtSin(eF) + (1010 / 3600.0f) * mtSin(Aa + eF) +
(1000 / 3600.0f) * mtSin(Aa - eF) - (624 / 3600.0f) * mtSin(eF - 2 * De) - (199
/ 3600.0f) * mtSin(Aa - eF - 2 * De) - (167 / 3600.0f) * mtSin(Aa + eF - 2 *
De);
float X = mtATan(mtSin(Mo) * mtTan(Qi));
float Y = (Li + X);
float nC = mtASin(mtSin(Mo) * mtSin(Qi) * mtSin(Y) / mtSin(X));
float PTc = mtACos(mtCos(Mo) * mtCos(Li) / mtCos(nC));
PTc = Mo<0?0:Mo>=180?360-PTc:PTc;
float tc = fmod((PT - PTc) + nisfu_qousinnahar,360);
float hc = mtASin(mtSin(Prm.Lt) * mtSin(nC) + mtCos(Prm.Lt) * mtCos(nC) *
mtCos(tc)); //Irtifa' Haqiqi
float P = (384401 * (1 - 0.00301401f)) / (1 + 0.0549f * mtCos(Ai + t[1]));
float HP = 0.9507f / (P/384401.0f);
float sdc = (0.5181f / (P/384401.0f)) / 2.0f;
float Pe = HP * mtCos(hc);
float Ref = 0.0167f / mtTan(hc + 7.31f / (hc + 4.4f));
float hci = hc - Pe;
```

```

    hci += (hc - Pe)>0?Ref + Dip - sdc:0;//IRTIFAK HILAL MAR'I dikurangi sdc kalo
    mau lower limb
    float Az = 270 + mtATan(-mtSin(Prm.Lt) / mtTan(nisfu_gousinnahar) +
    mtCos(Prm.Lt) * mtTan(Mails) / mtSin(nisfu_gousinnahar)); //AZIMUT MATAHARI
    HariAwal += hci>2?1:2;
    return HariAwal;
}

void Data_Matahari(float T, float &MailS, float &EoT, float &SDM, float &KWD){
    //float T = (JD - 2451545) / 36525;
    float S = fmod((280.46645f + 36000.76983f * T), 360);
    float m = fmod((357.5291f + 35999.0503f * T), 360);
    //DATA MATAHARI
    MailS = mtASin(mtSin(S) * 0.3976817514f);
    EoT = (-1.915f * mtSin(m) + -0.02f * mtSin(2.0f * m) + 2.466f * mtSin(2.0f *
    S) - 0.053f * mtSin(4.0f * S)) / 15.0f;
    SDM = 0.267f / (1 - 0.017f * mtCos(m));
    KWD = EoT + (float)(Prm.Bj - Prm.TZ * 15.0f)/15.0f; //Selisih Wis-WIB
}

void Hitung(){
    updateWaktu();
    byte bulan_ydh;int tahun_ydh;
    delay(20);
    if(month<3){
        bulan_ydh=month+12;
        tahun_ydh=year-1;
    }else{
        bulan_ydh=month;
        tahun_ydh=year;
    }
    int B = 2-(byte)(tahun_ydh/100)+(byte)((byte)(tahun_ydh/100)/4);
    float JD = (float)(1720994.5f + (long)(365.25f * tahun_ydh) +
    (int)(30.6001f*(bulan_ydh+1)) + B + day+0.5f);
    float T = (float)(JD - 2451545) / 36525.0f;

    harpas[0] = (byte)fmod(JD+2,7);
    harpas[1] = (byte)fmod(JD+1,5);
    float MailS, EoT, SDM, KWD; //Deklarasi variabel deklinasi,
    Perata Waktu (equation of time), Semi Diameter Matahari dan koreksi Waktu Daerah
    Data_Matahari(T, MailS, EoT, SDM, KWD); //Mengambil Data Matahari di Tab
    Hisab
    //Irtifa' Matahari masing-masing waktu sholat
    //Fungsi Trigonometri mtSin,mtCos,tan,asin dsb ada di tab Hisab
    float h[5];
    h[0] = (float)mtATan(1/(mtTan(abs(Prm.Lt-Mails))+1)); //Ashar
    h[1] = (float)(SDM+0.575f+(1.76/60.0f*sqrt(Prm.TT))+0.0024); //Maghrib &
    Thulu'
    h[2] = Prm.inkhifad[0]; //Isya'
    h[3] = Prm.inkhifad[1]; //Shubuh
    h[4] = Prm.inkhifad[2]; //Dluha

    //Waktu Sholat WD
    Waktu_Sholat[0] = 12+3/60.0f - KWD + Prm.ihtiyath[0]/60.0f;
    //Dhuhur
    Waktu_Sholat[1] = 12 + mtACos(-mtTan(Prm.Lt) * mtTan(Mails) + mtSin(h[0]) /
    mtCos(Prm.Lt) / mtCos(Mails))/15 - KWD + Prm.ihtiyath[1]/60.0f; //Ashar
    Waktu_Sholat[2] = 12 + mtACos(-mtTan(Prm.Lt) * mtTan(Mails) + mtSin(-h[1]) /
    mtCos(Prm.Lt) / mtCos(Mails))/15 - KWD + Prm.ihtiyath[2]/60.0f; //Maghrib
    Waktu_Sholat[3] = 12 + mtACos(-mtTan(Prm.Lt) * mtTan(Mails) + mtSin(-h[2]) /
    mtCos(Prm.Lt) / mtCos(Mails))/15 - KWD + Prm.ihtiyath[3]/60.0f; //Isya'
    Waktu_Sholat[5] = 12 - mtACos(-mtTan(Prm.Lt) * mtTan(Mails) + mtSin(-h[3]) /
    mtCos(Prm.Lt) / mtCos(Mails))/15 - KWD + Prm.ihtiyath[4]/60.0f; //Shubuh

```

```

Waktu_Sholat[4] = Waktu_Sholat[5] - 10/60.0f;
Waktu_Sholat[6] = 12 - mtACos(-mtTan(Prm.Lt) * mtTan(Mails) + mtSin(-h[1]) /
mtCos(Prm.Lt) / mtCos(Mails))/15 - KWD - Prm.ihtiyath[2]/60.0f; //Thulu'
Waktu_Sholat[7] = 12 - mtACos(-mtTan(Prm.Lt) * mtTan(Mails) + mtSin(h[4]) /
mtCos(Prm.Lt) / mtCos(Mails))/15 - KWD; //Dluha
WSNow=3;
for(uint8_t i=0;i<8;i++){
    if(saikif >= Waktu_Sholat[i] && saikif < Waktu_Sholat[(i+1)%8]) WSNow=i;
}
//Konversi Masehi-Hijri sederhana
double AH = JD - 1721423 - 227016;
int ToUrfi,SelUrfi,UpreadeUrfi,HariUrfi, HariTahqiqi = 0,HariTahqiqiNext = 0;
HariUrfi = 0;
if(saikif >= Waktu_Sholat[2]){
    HariUrfi++;AH++;
}
HariUrfi += fmod(harpas[0],7);
Thn_H = floor(AH / 354.3671f) + 1;
Bln_H = floor(fmod(AH, 354.3671f) / 29.53f)+1;
Tgl_H = floor(fmod(fmod(AH, 354.3671f), 29.53f)+1);

HariTahqiqi=ProsesManual(Bln_H, Thn_H); //Proses hitung hari awal
bulan berdasarkan ijtima' & hilal di file Hisab.ino
HariTahqiqiNext=ProsesManual(Bln_H+1, Thn_H); //Proses hitung hari awal
bulan selanjutnya berdasarkan ijtima' & hilal di file Hisab.ino
//Koreksi Hasil Konversi Urfi (sederhana)
ToUrfi = (int) fmod((HariTahqiqi + Tgl_H - 1), 7); //To Urfi = hari yang
sebenarnya berdasarkan ijtima' (ketika tanggal hasil konversi)
if (ToUrfi == HariUrfi) {
    //OK
} else {
    if ((ToUrfi > 5) & (HariUrfi < 3)) HariUrfi = (int) (HariUrfi + 7);
    if ((ToUrfi < 3) & (HariUrfi > 5)) ToUrfi = (int) (ToUrfi + 7);

    SelUrfi = (int) (HariUrfi - ToUrfi);
    SelUrfi -= SelUrfi>4?7:0;
    SelUrfi += SelUrfi<-4?7:0;

    UpreadeUrfi = (int) fmod(HariUrfi + SelUrfi, 7);
    UpreadeUrfi += UpreadeUrfi==0?7:0;

    if (ToUrfi == UpreadeUrfi) SelUrfi = (int) SelUrfi;

    // Koreksi hari=====
    Tgl_H += SelUrfi;
    if (Tgl_H < 1) {
        Tgl_H = 30;
        Bln_H -= 1;
        if (Bln_H < 1) {
            Bln_H += 12;
            Thn_H -= 1;
        }
    }
}
if (Tgl_H == 31 || Tgl_H == 30 && fmod(HariTahqiqi,7) == fmod(HariTahqiqiNext
- 1,7)) { //usia 29
    Tgl_H = 1;
    Bln_H += 1;
    if (Bln_H > 12) {
        Bln_H -= 12;
        Thn_H += 1;
    }
}
}

```

```

    DispWaktu = gabung(harpas[0], harpas[1], day, month, year, Tgl_H, Bln_H,
    Thn_H);
}

void dwMrq(const char* msg, int Speed){
    static uint16_t x;
    if (reset_x !=0) { x=0;reset_x = 0;}
    static uint16_t lsRn;
    int fullScroll = Disp.textWidth(msg) + 96;
    uint16_t Tmr = millis();
    if((Tmr-lsRn)> Speed){
        lsRn=Tmr;
        if (x < fullScroll+15) ++x;
        else {x = 0;return;}
        Disp.setFont(SystemFont5x7);
        Disp.drawText(96 - x, 9, msg);
        Disp.drawFilledRect(0,7,29,15,0);
        tampilJam();
        tampilJadwal();
        Disp.swapBuffers();
    }
}

void tampilJam(){
    updateWaktu();
    Disp.setFont(angka6x13);
    Disp.drawText(0, -1, Konversi(hour));
    Disp.drawText(16, -1, Konversi(minute));
    Disp.setFont(small4x7);
    Disp.drawText(30,0, Konversi(second));
    Disp.drawFilledRect(0, 14, 28, 15, 1);
    boolean is500 = millis()%1000<500?true:false;
    Disp.drawLine(14, 3, 14, 4, is500);
    Disp.drawLine(14, 9, 14, 10, is500);
}

void tampilInfo2(){
    updateWaktu();
    if(saikif >= Waktu_Sholat[WSNow] && saikif < Waktu_Sholat[WSNow]+0.033f){
        dwSaatSholat();
    }else{
        if(WSNow != 4 && !(WSNow > 5)) {
            if(saikif >= Waktu_Sholat[WSNow]+0.033f && saikif <=
Waktu_Sholat[WSNow]+JedaIqomah){
                countdown = (Waktu_Sholat[WSNow]+JedaIqomah-saikif)*3600;
                static char out[11];
                sprintf(out, "IQOMAH -%d", countdown);
                Disp.setFont(Bernard16);
                Disp.drawText((Disp.width()-Disp.textWidth(out))/2, 0, out);
                Disp.swapBuffers();
                if(countdown<=10){
                    digitalWrite(PIN_BUZZER, LOW);
                    delay(500);
                    digitalWrite(PIN_BUZZER, HIGH);
                    delay(500);
                }
            }else if(saikif >= Waktu_Sholat[WSNow]+JedaIqomah && saikif <=
Waktu_Sholat[WSNow]+JedaIqomah+LamaSholat){
                digitalWrite(PIN_BUZZER, LOW);
                delay(5000);
                ClrScr();
                digitalWrite(PIN_BUZZER, HIGH);
            }else if(saikif > Waktu_Sholat[WSNow]+JedaIqomah+LamaSholat){

```



```

    }
  }
  for(uint8_t i=0;i<3;i++)
EEPROM.update(i*4+1,BT_Param.substring(Koma[i]+1,Koma[i+1]).toFloat());
}else if(kode.equals("IHT")){
  uint8_t Koma[9], temp=0;
  for(uint8_t i=3;i<panjang;i++){
    if(BT_Param.substring(i,i+1)==":"){
      Koma[temp]=i;temp++;
    }
  }
  for(uint8_t i=0;i<5;i++) EEPROM.update(14+i,
BT_Param.substring(Koma[i]+1,Koma[i+1]).toInt());
  for(uint8_t i=5;i<8;i++) EEPROM.update((i-5)*4+19,
BT_Param.substring(Koma[i]+1,Koma[i+1]).toInt());
}else if(kode.equals("CRH")){
  Prm.Cerah = BT_Param.substring(3,panjang).toInt();
  EEPROM.update(51, Prm.Cerah);
  Disp.setBrightness(Prm.Cerah);
}else if(kode.equals("TXT")){
  String text;
  text = String(BT_Param.substring(3,panjang));
  text.toCharArray(infone,panjang>150?150:panjang+(150-panjang-3));
  EEPROM.put(100,infone);
}else if(kode.equals("PWR")){
  Run = Run==3?1:3;
}else if(kode.equals("GPS")){
  UpdateGPS();
}else{
  //nothing else
}
if(!kode.equals("TXT")) EEPROM.get(0, Prm);
Hitung();
}
}

const char static hari[] PROGMEM = {"SABTU\0\0"
  "AHAD\0\0\0"
  "SENIN\0\0"
  "SELASA\0"
  "RABU\0\0\0"
  "KAMIS\0\0"
  "JUM'AT\0"
  "SABTU\0\0"};
const char static pasaran[] PROGMEM = {"KLIWON\0"
  "LEGI\0\0\0"
  "PAHING\0"
  "PON\0\0\0\0"
  "WAGE\0\0\0"
  "KLIWON\0"};
const char static BulanH[] PROGMEM ={"DZULHIJJA\0"
  "MUHARRAM\0\0\0"
  "SHAFAR\0\0\0\0\0"
  "RAB.AWAL\0\0\0"
  "RAB.AKHIR\0\0"
  "JUM.AWAL\0\0\0"
  "JUM.AKHIR\0\0"
  "RAJAB\0\0\0\0\0\0"
  "SYA'BAN\0\0\0\0"
  "RAMADLAN\0\0\0"
  "SYAWWAL\0\0\0\0"
  "DZULQA'DAH\0"
  "DZULHIJJA\0"};

```

```

const char static BulanM[] PROGMEM ={ "DESEMBER\0\0"
                                        "JANUARI\0\0\0"
                                        "FEBRUARI\0\0"
                                        "MARET\0\0\0\0"
                                        "APRIL\0\0\0\0"
                                        "MEI\0\0\0\0\0"
                                        "JUNI\0\0\0\0\0"
                                        "JULI\0\0\0\0\0"
                                        "AGUSTUS\0\0\0"
                                        "SEPTEMBER\0"
                                        "OKTOBER\0\0\0"
                                        "NOVEMBER\0\0"
                                        "DESEMBER\0\0"};
const char static NamaSholat[] PROGMEM = { "Dhuhr\0"
                                             "Ashr\0\0"
                                             "Magrb\0"
                                             "Isya'\0"
                                             "Imsak\0"
                                             "Subuh\0"
                                             "Tulu'\0"
                                             "Dluha\0"
                                             "Jumat\0"};
const char static SaatSholat[] PROGMEM = { "DHUHUR\0\0"
                                             "ASHAR\0\0\0"
                                             "MAGHRIB\0"
                                             "ISYA'\0\0\0"
                                             "IMSAK\0\0\0"
                                             "SHUBUH\0\0"
                                             "THULU'\0\0"
                                             "DLUHA\0\0\0"
                                             "JUM'AT\0\0"};

char * cHari(byte number){ // get hari
    static char locBuff[7];
    int locLen = (number)*7;
    memccpy_P(locBuff,hari+locLen,0,7);
    return locBuff;
}
char * cPasaran(byte number){ // get hari
    static char locBuff[7];
    int locLen = (number)*7;
    memccpy_P(locBuff,pasaran+locLen,0,7);
    return locBuff;
}
char * cBulanM(byte number){ // get hari
    static char locBuff[10];
    int locLen = (number)*10;
    memccpy_P(locBuff,BulanM+locLen,0,10);
    return locBuff;
}
char * cBulanH(byte number){ // get hari
    static char locBuff[11];
    int locLen = (number)*11;
    memccpy_P(locBuff,BulanH+locLen,0,11);
    return locBuff;
}

char * gabung(byte hr, byte ps, byte tglM, byte blM, int thM, byte tglH, byte
blH, int thH){ // get hari

    static char out[70];
    sprintf(out,"%s %s, %02d %s %d M./%02d %s %d H.",cHari(hr), cPasaran(ps),
    tglM, cBulanM(blM), thM, tglH, cBulanH(blH), thH);
    return out;
}

```



```

}
void dwSaatSholat(){
    static char locBuff[8];
    static char out[11];
    uint8_t ganti = WSNOW;
    if(ganti == 0 && harpas[0]==6) {ganti = 8;}
    int locLen = (ganti)*8;
    memccpy_P(locBuff,SaatSholat+locLen,0,8);
    sprintf(out,"%s...",locBuff);
    Disp.setFont(Bernard16);
    Disp.drawText((96-Disp.textWidth(out))/2, 0, out);
    Disp.swapBuffers();
    digitalWrite(PIN_BUZZER, LOW);
    delay(1000);
    digitalWrite(PIN_BUZZER, HIGH);
    ClrScr();
    delay(500);
}

void tampilJadwal(){
    byte ganti = (millis()%24000)/3000;
    if(ganti == 0 && harpas[0]==6) {ganti = 8;}
    static char out[6];
    int locLen = (ganti)*6;
    memccpy_P(out>NamaSholat+locLen,0,6);
    Disp.setFont(SystemFont5x7);
    Disp.drawText(44, 0, out);
    Disp.setFont(small4x7);
    byte jmWS[2];
    floatToHourMin(Waktu_Sholat[ganti==8?0:ganti], jmWS[0], jmWS[1]);
    Disp.drawText(75, 0, Konversi(jmWS[0]));
    Disp.drawText(87, 0, Konversi(jmWS[1]));
    Disp.drawLine(85, 4, 85, 4, 1);
    Disp.drawLine(85, 2, 85, 2, 1);
}

```

Lampiran 3 : *Curriculum Vitae***CURRICULUM VITAE**

- Nama : Muhammad Nufail
- NIM : 2014.69.04.0096
- Tempat dan Tanggal Lahir : Pasuruan, 15 September 1989
- Pendidikan : - 1996 – 2002 : MI. Miftahul Khoir I (Karangrejo, Purwosari, Pasuruan).
- 2005 – 2008 : MTs. Darullughah Wadda'wah (Raci, Bangil, Pasuruan).
- 2008 – 2011 : MA. Darullughah Wadda'wah Program Bahasa (Raci, Bangil, Pasuruan).
- Pekerjaan : - Typist
- Publikasi Karya : - Kupas Tuntas Hukum Darah Wanita (2012)
- Kasyful Qomaroin Fii Ma'rifati Hisabis Sinin Wal Hilal (2008).
- Rancang Bangun Suspenduk (Sistem Kependudukan) Desa di CV. Fix Digital (2018).