DAFTAR PUSTAKA

[1] I. Kholiq, “Pemanfaatan Energi Alternatif Sebagai Energi Terbarukan untuk Mendukung Subtitusi BBM,” *Jurnal IPTEK*, vol. vol.19, no. 2, 2015.

[2] A. I. Ramadhan, E. Diniardi, and S. H. Mukti, “Analisis Desain Sistem Pembangkit Listrik Tenaga Surya Kapasitas 50 WP,” *TEKNIK*, 2016.

[3] A. Budiman, K. Das, A. Mohammad, K. T. Tan, and O. Tonby, “Sepuluh gagasan untuk menguatkan kembali sektor energi Indonesia,” McKinsey&COmpany, Tech. Rep., 2014.

[4] Direktorat Jenderal Energi Baru, Terbarukan dan Konversi Energi, “Rencana Strategis Kementerian Energi dan Sumber Daya Mineral 2015-2019,” Kementerian Energi dan Sumber Daya Mineral, Tech. Rep., 2015.

[5] A. Rahayuningtyas, S. I. Kuala, and I. F. Apriyanto, “Studi Perencanaan Sistem Pembangkit Listrik Tenaga Surya (PLTS) Skala Rumah Sederhana di Daerah Pedesaan Sebagai Pembangkit Listrik Alternatif untuk Mendukung Program Ramah Lingkungan dan Energi Terbarukan,” *Prosiding SNaPP2014 Sains, Teknologi, dan Kesehatan*, 2014.

[6] G.-R. Yu and J.-F. Chen, “Fuzzy Regulation of DC-Bus Voltage for Solar Energy System,” *IEEE International Conference on System Science and Engineering*, 2013.

[7] R. Arulmurugan and N. S. Vanitha, “Intelligent fuzzy MPPT controller using analysis of DC to DC novel buck converter for photovoltaic energy system applications,” *International Conference on Pattern Recognition, Informatics and Mobile Engineering*, 2013.

[8] H. A. Sher and K. E. Addoweesh, "A New Sensorless Hybrid MPPT Algorithm Based on Fractional Short-Circuit Current Measurement and P&O MPPT," *IEEE Transaction on Sustainable Energy*, vol. vol.6, no. 4, pp. 1426 – 1434, 2015.

[9] C. Sharma and A. Jain, "MAXIMUM POWER POINT TRACKING TECHNIQUES: A REVIEW," *International Journal of Recent Research in Electrical and Electronics Engineering (IJRREEE)*, vol. 1, pp. 25–33, 2014.

[10] M. R. Patel, *Wind and Solar Power Systems, Design, Analysis, and Operation*. U.S. Merchant Marine Academy Kings Point, New York, U.S.A., 2006.

[11] greentechlead, "Tsec, dupont to showcase v-series solar panels at tokyo expo." [Online]. Available: http://www.greentechlead.com/solar/tsec-dupont-to-showcase-v-series-solar-panels-at-tokyo-expo-21927

[12] G. M. Masters, *Renewable and Efficient Electric Power Systems*, New Jersey, 2004, ch. 8, pp. 446–502.

[13] D. C. Huynh, T. M. Nguyen, M. W. Dunnigan, and M. A. Mueller, "Global MPPT of Solar PV Modules using a Dynamic PSO Algorithm under Partial Shading Conditions," *IEEE Conference on Clean Energy and Technology (CEAT)*, 2013.

[14] L. Castaner and S. Silvestre, *Modeling Photovoltaic Using PSpice*. John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England, 2002.

Jäger, O. Isabella, A. H. M. Smets, R. A. C. M. M. van Swaaij, and Zeman, *Solar Energy Fundamentals, Technology, and Systems*. ft University of Technology, 2014.

[16] B. Hossam and K. Itako, “Real Time Hotspot Detection Using Scan-method Adopted with P&O MPPT for PV Generation System,” *IEEE 2nd Annual Southern Power Electronics Conference (SPEC)*, pp. 1–5, 2016.

[17] F. A. Samman, A. A. Rahmansyah, and Syafaruddin, “Iterative decremented step-size scanning-based MPPT algorithms for photovoltaic systems,” *Information Technology and Electrical Engineering (ICITEE), 2017 9th International Conference*, 2018.

[18] M. Effendy, N. A. Mardiyah, and K. Hidayat, “Implementasi Maximum Power Point Tracking pada Photovoltaic Berbasis P&O-Fuzzy,” *JNTETI*, vol. vol.6, no. 115-120, 2017.

[19] Syafaruddin, “ANN-Polar Coordinated Fuzzy Controller Based Real Time Maximum Power Point Tracking Control Of Photovoltaic System,” PhD dissertation, School of Science and Technology - Kumamoto University, Japan, 2009.

[20] Y.-H. Chang and C.-Y. Chang, “A Maximum Power Point Tracking of PV System by Scaling Fuzzy Control,” *Proceeding of The International Multi Conference of Engineers and Computer Scientists*, vol. II, 2010.

[21] C. Lohmeier, J. Zeng, W. Qiao, and L. Q. J. Hudgins, “A Current-Sensorless MPPT Quasi-Double-Boost Converter for PV Systems,” *Energy Conversion Congress and Exposition (ECCE)*, pp. 1069–1075, 2011.

[22] Y.-H. Wang, W.-C. Liu, and T.-H. Kuo, “A 200W MPPT Boost Converter for BIPV Applications with Integrated Controller,” *International Sympo- m on Computer, Consumer and Control*, pp. 1–8, 2014.

[23] K. Chen, S. Tian, Y. Cheng, and L. Bai, “An Improved MPPT Controller for Photovoltaic System Under Partial Shading Condition,” *IEEE Transaction on Suistainable Energy*, vol. 5, pp. 978 – 985, 2014.

[24] S. Mohanty, B. Subudhi, and P. K. Ray, “A New MPPT Design Using Grey Wolf Optimization Technique for Photovoltaic System Under Partial Shading Conditions,” *IEEE Transactions on Sustainable Energy*, vol. 7, pp. 181 – 188, 2015.

[25] A. A. S. Mohamed, A. Berzoy, and O. Mohammed, “Design and Hardware Implementation of FLMPPT Control of PV Systems Based on GA and Small-Signal Analysis,” *IEEE Transactions on Sustainable Energy*, vol. 8, pp. 279 – 290, 2017.

[26] P. S. Narsingrao and D. R. C. Prasad, “Design and Implementation of MPPT Algorithm for Solar Energy System,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, pp. 756–761, 2013.

[27] S. R. M. B.Sc., *DC/DC BOOK OF KNOWLEDGE*. RECOM Engineering GmbH & Co KG, Austria (hereafter RECOM), 2014.

[28] D. Zhang, “AN-1484 Designing A SEPIC Converter,” Texas Instrumens, Tech. Rep., 2013.

[29] J. Falin, “Designing DC/DC converters based on SEPIC topology,” Texas Instrumens, Tech. Rep., 2008.

[30] *8-bit Atmel with 8 KBytes InSystem Programmable Flash Datasheet*. [Online]. Available: http://www.atmel.com/Images/
nel-2486-8-bit-AVR-microcontroller-ATmega8_L_summary.pdf

Suryawinata, D. Purwanti, and S. Sunardiyo, “Sistem Monitoring pa-
Panel Surya Menggunakan Data logger Berbasis ATmega 328 dan

Real Time Clock DS1307 ," *Jurnal Teknik Elektro*, vol. 9, no. 1, pp. 30–36, 2017.

[32] O. Semiconductor, "MUR1560G Switch-mode Power Rectifiers," On Semiconductor, Tech. Rep., 2014.

[33] Cadence Design Systems, Inc, "About PSpice, PSpice History." [Online]. Available: http://www.pspice.com/about

[34] E. P. Dr. Agfianto, "Tentang PSpice!" [Online]. Available: http://agfi.staff.ugm.ac.id/blog/index.php/2008/11/tentang-pspice/

[35] V. Siliconix, "Power MOSFET IRFP460, SiHFP460," Vishay Siliconix, Tech. Rep., 2008.

[36] M. H. Rashid and H. M. Rashid, *SPICE for Power Electronics and Electric Power*. Taylor & Francis Group, 2006.

[37] T. Instrumenst, "LM2576xx Series SIMPLE SWITCHER® 3-A Step-Down Voltage Regulator," Texas Instrumenst, Tech. Rep., 2016.

LAMPIRAN

Source code simulasi pSpice pemodelan SEPIC dengan variasi f_{SW}

```
MPPT-SEPIC-VAR-FREKUENSI
.param Vpv = 22.41V
.param Freq = 1Khz
.param T = {1/Freq}
.param D = 0.5
.param _HIGH = {Vpv}
.param _LOW = 0V
.param Td = 0us
.param Tr = 1us
.param Tf = 1us
VIN 1 0 DC {Vpv}
VPULSE 7 0 PULSE ({_LOW} {Vpv} {Td} {Tr} {Tf} {T*D} {T})
R1 7 8 100
RL 6 0 100
C1 3 4 100NF IC=0V
C2 6 0 1000UF IC=0V
L1 2 3 1.5MH
L2 4 0 1.5MH
D1 4 5 MUR1560
.MODEL MUR1560 D(IS=2.2E-16 BV=600V IBV=10U CJO=2P TT=60N)
M1 3 8 0 0 IRFP460
.MODEL IRFP460 NMOS (VTO=2.83 KP=31.2153U L=1U
+W=3M CGDO=1N CGSO=3.2N)
Vy 1 2 0
Vx 5 6 0
*.STEP PARAM D List 0.1 0.5 0.9
.STEP PARAM Freq List 50 100 500 750 1000 2500 5000 7500 10000
.TRAN 100mS 300mS 0mS UIC
.PROBE
.END
```

Source code simulasi pSpice pemodelan SEPIC dengan variasi D

```
MPPT-SEPIC-VAR-FREKUENSI
       Vpv = 22.41V
       Freq = 500hz
       T = {1/Freq}
       D = 0.5
       _HIGH = {Vpv}
       _LOW = 0V
```

```
.param Td = 0us
.param Tr = 1us
.param Tf = 1us
VIN 1 0 DC {Vpv}
VPULSE 7 0 PULSE ({_LOW} {Vpv} {Td} {Tr} {Tf} {T*D} {T})
R1 7 8 100
RL 6 0 100
C1 3 4 100NF IC=0V
C2 6 0 1000UF IC=0V
L1 2 3 1.5MH
L2 4 0 1.5MH
D1 4 5 MUR1560
.MODEL MUR1560 D(IS=2.2E-16 BV=600V IBV=10U CJO=2P TT=60N)
M1 3 8 0 0 IRFP460
.MODEL IRFP460 NMOS (VTO=2.83 KP=31.2153U L=1U
+W=3M CGDO=1N CGSO=3.2N)
Vy 1 2 0
Vx 5 6 0
.STEP PARAM d List 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
.TRAN 0.1PS 300mS 0MS UIC
.PROBE
.END
```

Source code simulasi pSpice pemodelan SEPIC dengan variasi V_{IN}

```
MPPT-SEPIC-VAR-VIN
.param Vpv = 22.41V
.param Freq = 500hz
.param T = {1/Freq}
.param D = 0.7
.param _HIGH = {Vpv}
.param _LOW = 0V
.param Td = 0us
.param Tr = 1us
.param Tf = 1us
*VIN 1 0 DC {Vpv}
VIN 1 0 PWL (0 0 3.6S 60V)
VPULSE 7 0 PULSE ({_LOW} {Vpv} {Td} {Tr} {Tf} {T*D} {T})
R1 7 8 100
RL 6 0 100
C1 3 4 100NF IC=0V
C2 6 0 1000UF IC=0V
L1 2 3 1.5MH
L2 4 0 1.5MH
D1 4 5 MUR1560
       MUR1560 D(IS=2.2E-16 BV=600V IBV=10U CJO=2P TT=60N)
       0 0 IRFP460
       IRFP460 NMOS (VTO=2.83 KP=31.2153U L=1U
       GDO=1N CGSO=3.2N)
       0
       0
```

```
*.STEP PARAM Vpv List 0 1 5 10 15 20 25 30 35 40 45 50 55 60
.TRAN 0.1PS 3.6S 0MS UIC
.PROBE
.END
```

Source code simulasi pSpice pemodelan SEPIC dengan variasi R_L

```
MPPT-SEPIC-VAR-RL
.param Vpv = 22.41V
.param Freq = 500hz
.param T = {1/Freq}
.param D = 0.7
.param _HIGH = {Vpv}
.param _LOW = 0V
.param Td = 0us
.param Tr = 1us
.param Tf = 1us
.param RLoad =1k
VIN 1 0 DC {Vpv}
*VIN 1 0 PWL (0 0 3.6S 60V)
VPULSE 7 0 PULSE ({_LOW} {Vpv} {Td} {Tr} {Tf} {T*D} {T})
R1 7 8 100
RL 6 0 {RLoad}
C1 3 4 100NF IC=0V
C2 6 0 1000UF IC=0V
L1 2 3 1.5MH
L2 4 0 1.5MH
D1 4 5 MUR1560
.MODEL MUR1560 D(IS=2.2E-16 BV=600V IBV=10U CJO=2P TT=60N)
M1 3 8 0 0 IRFP460
.MODEL IRFP460 NMOS (VTO=2.83 KP=31.2153U L=1U
+W=3M CGDO=1N CGSO=3.2N)
Vy 1 2 0
Vx 5 6 0
.STEP PARAM RLoad List 1 10 50 100 250 500 750 800 900 1k
*.DC RLoad 0V 50V 1V
TRAN 0.1PS 300mS 0MS UIC
.PROBE
.END
```

Source code Kendali MPPT

```
       e <LiquidCrystal.h>
       e <TimerOne.h>
        Do 261
        Re 294
        Mi 329
        Fa 349
```

```
#define So 392
#define La 442
#define Si 493
#define DO 523
#define S1 10
#define S2 11
#define S3 12
//081543202548
//#define duty OCR1AL
int nada[3][2]={Do,80,Re,80,Mi,150};
int piezoPin = 13;
const int rs = 2, en = 3, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void duty(unsigned char Duty)
{unsigned char DUTY;
        DUTY=Duty;
        OCR1AL=DUTY;
}
#define volIn A4
#define volOut A5
#define curIn A3
#define curOut A0
#define curSw A2
#define curCtrl A1
const double calVolIn=0.05517;//0.0679245283018868;
//0.040983606557377;
const double calVolOut=0.1338;//0.1819226769230769;
const double calCurIn=384;
const double calCurOut=1580;
const double calCurSw=1580;
const double calCurCtrl=1580;
int mVperAmp = 185;
double Voltage = 0;
double VRMS = 0;
double AmpsRMS = 0;
const int adc_zero = 0;
const unsigned int numSamples = 100;
long currentAcc; int c,start_time;
unsigned char DC,mod;
char buff1[33];char buff2[33];
double vIn,vOut,iIn,iOut,iCtrl,iSw,pIn,pOut;
double pInPrev,pOutPrev;
float getVPP(unsigned sensorIn)
{
        float result;
        int readValue;
        start_time = 0;c=0;currentAcc=0;
        while(c < numSamples)
        {
                readValue = analogRead(sensorIn) - adc_zero;
                currentAcc += (unsigned long)(readValue );
                c++;
        }
        result = (float)currentAcc/(float)
        numSamples;// * (50 / 1024.0);

        //  result = analogRead(sensorIn);
        switch(sensorIn)
        {
                case volIn:result=(result*calVolIn)+1.9998;
                break;
                case volOut:if(result>0)
                {result=(result*calVolOut)+3.465;}
                else{result=(result*calVolOut);};break;
                case curIn:result=result/calCurIn;break;
                case curOut:result=result/calCurOut;break;
                case curSw:result=result/calCurSw;break;
                case curCtrl:result=result/calCurCtrl;break;
        }
        return result;
}
void bacaSensor()
{
        vIn=getVPP(volIn);
        vOut=getVPP(volOut);
        iIn=getVPP(curIn);
        iOut=getVPP(curOut);
        iSw=getVPP(curSw);
        iCtrl=getVPP(curCtrl);
        pIn=vIn*iIn;
        pOut=vOut*iOut;
}
void play()
{
        for(int i=0;i<3;i++){tone(piezoPin, nada[i][0]*5);
        delay(nada[i][1]);}noTone(piezoPin);
}
unsigned int scanTime;unsigned char pOutMax,cntBitMax;
boolean scanMode=false;

void sendSerial()
{
        //clearTime++;if(clearTime>clearInterval)
        {lcd.clear();clearTime=0;}
        Serial.print("ABCDE");Serial.print(vIn*100,0);
        Serial.print('F');Serial.print(vOut*100,0);
        Serial.print('G');Serial.print(iIn*100,0);
        Serial.print('H');Serial.print(iOut*100,0);
        Serial.print('I');Serial.print(iSw*100,0);
        Serial.print('J');Serial.print(iCtrl*100,0);
        Serial.print('K');Serial.print(DC);
        //Serial.print(DC/2.55*100,0);
        Serial.println('X');

       nt endBitRef=250;
```

```
unsigned char scanWidthRef=5;
unsigned char startBit=0,endBit=endBitRef,scanWidth=scanWidthRef;
unsigned char totalBit,balanceBit,cntBit;

void scanMPP()
{
        totalBit=endBit-startBit;
        balanceBit=(unsigned int)totalBit/scanWidth;
        //balanceBit=scanWidth;
        cntBit=startBit;pOutMax=0;
        while(scanMode==true)
        { duty(cntBit);
                bacaSensor();

                if(pOut>pOutMax){pOutMax=pOut;cntBitMax=cntBit;}
                DC=cntBit;
                sendSerial();

                cntBit+=balanceBit;
                if(cntBit>=endBit)
                {
                        startBit=cntBitMax-balanceBit;
                        endBit=cntBitMax+balanceBit;
                        totalBit=endBit-startBit;
                        if(totalBit<scanWidth)
                        {DC=cntBitMax;scanMode=false;break;}
                        else
                        {
                                balanceBit=(unsigned int)
                                totalBit/scanWidth;
                                cntBit=startBit;
                        }
                }//delay(10);
        }//endx:
}
boolean keyS1,keyS2,keyS3;
void setup() {
        char xx=0;
        unsigned char valScanWidth[5]={5,10,15,20,25};
        lcd.begin(16, 2);
        pinMode(S1, INPUT);pinMode(S2, INPUT);
        pinMode(S3, INPUT);pinMode(9, OUTPUT);
        Serial.begin(115200);
        TCCR1A=0x81;TCCR1B=0x0A;TCNT1H=0x00;
        TCNT1L=0x00;OCR1AH=0x00;OCR1AL=0;
        lcd.setCursor(0, 0);
        lcd.print("Bismillah");play();
        scanMode=true;
        //Timer1.initialize(1000);
        //Timer1.attachInterrupt(saklar);


        DC=0;delay(1000);
        //scanMPP();
        while(digitalRead(S3)==1)
        {
                if(digitalRead(S2)==0&&keyS2==true)
                {keyS2=false;xx++;if(xx>4){xx=4;}}
                else if(digitalRead(S1)==0&&keyS1==true)
                {keyS1=false;xx--;if(xx<0){xx=0;}}
                if(digitalRead(S1)==HIGH){keyS1=true;}
                if(digitalRead(S2)==HIGH){keyS2=true;}
                scanWidth=valScanWidth[xx];
                sprintf(buff1,"scanWidth= % 2u  ",scanWidth);
                lcd.setCursor(0, 0);lcd.print(buff1);
                lcd.setCursor(0, 1);lcd.print("tekan s3");

                bacaSensor();sendSerial();
        }delay(100);lcd.clear();
        OCR0A = 0xAF;
        TIMSK0 |= _BV(OCIE0A);
}
void loop() {
        //duty(DC*2.55);
        //sprintf(buff1, "Duty Cycle: %3u", DC/2.55);
        strcpy(buff1, "D-Cycle: ");
        dtostrf (DC/2.55,3,1, &buff1[strlen(buff1)]);
        bacaSensor();
        scanTime++;
        if(scanTime>25)
        {
                if(pOut==0)
                {
                        scanMode=true;startBit=0;endBit=endBitRef;
                        scanMPP();pOutPrev=pOut;pInPrev=pIn;
                }
                else
                {
                        if((pOut<pOutPrev-5.0||pOut>pOutPrev+5.0)
                        ||(pIn<pInPrev-5.0||pIn>pInPrev+5.0))
                        {
                                scanMode=true;startBit=0;
                                endBit=endBitRef;scanMPP();
                                pOutPrev=pOut;pInPrev=pIn;
                        }
                        scanTime=0;
                }
        }
        //if(scanTime>25)
        // {
                //      if((pOut<pOutPrev-5.0||
                pOut>pOutPrev+5.0)||(pIn<pInPrev-2.0
                ||pIn>pInPrev+2.0))
                //      {
                        //        scanMode=true;startBit=0;
                        endBit=endBitRef;scanMPP();
                        pOutPrev=pOut;pInPrev=pIn;
                        //     }
                //     scanTime=0;
                // }
        tampil();
        sendSerial();

}
void saklar()
{
        if(digitalRead(S2)==0&&keyS2==true)
        {keyS2=false;DC++;if(DC>100){DC=100;}}
        else if(digitalRead(S1)==0&&keyS1==true)
        {keyS1=false;DC--;if(DC==255){DC=0;}}
        //if(digitalRead(S3)==0&&keyS3==true)
        {keyS3=false;mod++;if(mod>8){mod=0;}}

        if(digitalRead(S1)==HIGH){keyS1=true;}
        if(digitalRead(S2)==HIGH){keyS2=true;}
        //if(digitalRead(S3)==HIGH){keyS3=true;}
}
SIGNAL(TIMER0_COMPA_vect)
{
        if(digitalRead(S2)==0&&keyS2==true)
        {keyS2=false;DC++;if(DC>255){DC=255;}}
        else if(digitalRead(S1)==0&&keyS1==true)
        {keyS1=false;DC--;if(DC==255){DC=0;}}
        //if(digitalRead(S3)==0&&keyS3==true)
        {keyS3=false;mod++;if(mod>8){mod=0;}}
        //if(digitalRead(S3)==0&&keyS3==true)
        {scanMode=true;startBit=0;endBit=endBitRef
        scanMPP();pOutPrev=pOut;pInPrev=pIn;}
        if(digitalRead(S1)==HIGH){keyS1=true;}
        if(digitalRead(S2)==HIGH){keyS2=true;}
        if(digitalRead(S3)==HIGH){keyS3=true;}
}
void tampil()
{
        switch(mod)
        {
                case 0:strcpy(buff2, "Vin: ");
                dtostrf (vIn,2,1, &buff2[strlen(buff2)]);break;
                case 1:strcpy(buff2, "Vout: ");
                dtostrf (vOut,2,1, &buff2[strlen(buff2)]);break;
                case 2:strcpy(buff2, "Iin: ");
                dtostrf (iIn,2,1, &buff2[strlen(buff2)]);break;
                case 3:strcpy(buff2, "Iout: ");
                dtostrf (iOut,2,1, &buff2[strlen(buff2)]);break;
                case 4:strcpy(buff2, "Isw: ");
                dtostrf (iSw,2,1, &buff2[strlen(buff2)]);break;
                case 5:strcpy(buff2, "Ictr: ");
                dtostrf (iCtrl,2,1, &buff2[strlen(buff2)]);break;
                case 6:strcpy(buff2, "Pin: ");
                dtostrf (pIn,2,1, &buff2[strlen(buff2)]);break;
                case 7:strcpy(buff2, "Pout: ");
                dtostrf (pOut,2,1, &buff2[strlen(buff2)]);break;
                case 8:strcpy(buff2, "Po/Pi: ");
                dtostrf ((pOut)/(pIn)*100,2,1,
                &buff2[strlen(buff2)]);
                break;
        }
        lcd.clear();
        lcd.setCursor(0, 0);lcd.print(buff1);lcd.print(" %");
        lcd.setCursor(0, 1);lcd.print(buff2);
        if(mod==0||mod==1){lcd.print(" V");}
        else if(mod==6||mod==7){lcd.print(" W");}
        else if(mod==8){lcd.print(" %");}
                else{lcd.print(" A");}
        }
```