

## REFERENCES

- Annapureddy, V. *et al.* (2017) 'Magnetic energy harvesting with magnetoelectrics: An emerging technology for self-powered autonomous systems', *Sustainable Energy and Fuels*, 1(10), pp. 2039–2052. doi: 10.1039/c7se00403f.
- Arduino Mega 2560 Rev 3* (no date). Available at: <https://store.arduino.cc/products/arduino-mega-2560-rev3>.
- Arduino Uno* (no date). Available at: <https://store.arduino.cc/products/arduino-uno-rev3>.
- Atmoko, R. A., Riantini, R. and Hasin, M. K. (2017) 'IoT real time data acquisition using MQTT protocol', *Journal of Physics: Conference Series*, 853(1). doi: 10.1088/1742-6596/853/1/012003.
- Cetinkaya, O., Member, S. and Akan, O. B. (no date) 'Electric-field Energy Harvesting Wireless Networks'.
- Datasheet, X. (2003) 'Datasheet Datasheet', *Connect*, 46(303), pp. 1–4.
- ESP (2021) 'ESP32 Series Datasheet', *Espressif Systems*, pp. 1–65. Available at: [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf).
- Ferencz, K. and József, D. (2020) 'Using Node-RED platform in an industrial environment.', (February), p. 13.
- Fraunhofer IZM* (no date). Available at: <https://www.astrose.de/en/astrose-system/astrose-so-funktioniert-das-system.html>.
- Hartono, A. T., Rusli, L. P. and Rudijanto, R. (2020) 'Development of Automated Pillbox With IoT Features for PT. Pharos indonesia', (15). Available at: [http://repository.sgu.ac.id/1951/%0Ahttp://repository.sgu.ac.id/1951/1/Alvin Tri Hartono 11601043-TOC.pdf](http://repository.sgu.ac.id/1951/%0Ahttp://repository.sgu.ac.id/1951/1/Alvin%20Tri%20Hartono%2011601043-TOC.pdf).
- 'MQTT' (no date). Available at: [https://www.hivemq.com/?utm\\_source=adwords&utm\\_campaign=&utm\\_term=hivemqtt&utm\\_medium=ppc&hsa\\_tgt=kwd-435614090536&hsa\\_cam=14220703563&hsa\\_src=g&hsa\\_net=adwords&hsa\\_kw=hivemqtt&hsa\\_ad=560533276684&hsa\\_grp=123021240462&hsa\\_ver=3&hsa\\_acc=3585854406&hsa\\_mt](https://www.hivemq.com/?utm_source=adwords&utm_campaign=&utm_term=hivemqtt&utm_medium=ppc&hsa_tgt=kwd-435614090536&hsa_cam=14220703563&hsa_src=g&hsa_net=adwords&hsa_kw=hivemqtt&hsa_ad=560533276684&hsa_grp=123021240462&hsa_ver=3&hsa_acc=3585854406&hsa_mt).
- OMICRON electronics GmbH* (no date). Available at: <https://www.omicronenergy.com/en/>.
- Sim Com (2009) 'SIM900 \_ AT Command', pp. 1–229. Available at: [https://www.espruino.com/datasheets/SIM900\\_AT.pdf](https://www.espruino.com/datasheets/SIM900_AT.pdf).
- Yilmaz, M. *et al.* (2014) 'High-efficiency passive full wave rectification for electromagnetic

harvesters’, *Journal of Applied Physics*, 116(13), pp. 0–8. doi: 10.1063/1.4896668.

Yuan, S. *et al.* (2017) ‘A high-efficiency helical core for magnetic field energy harvesting’, *IEEE Transactions on Power Electronics*, 32(7), pp. 5365–5376. doi: 10.1109/TPEL.2016.2610323.



## APPENDIX

### Current Measurement Program

```
#include <Adafruit_Sensor.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SH1106.h>

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
#define OLED_RESET -1
Adafruit_SH1106 display(OLED_RESET);

#include <DHT.h>;
#include <Wire.h>

//Constants
#define DHTPIN A3 // what pin we're connected to
#define DHTTYPE DHT11 // DHT 22 (AM2302)
DHT dht(DHTPIN, DHTTYPE); /// Initialize DHT sensor for normal 16mhz Arduino

float sampleArray[500] = {}; //array to store 20 samples of temperature signals
int sampleCounter = 0; //array pointer
float cube1 = 0.0;
float cube2 = 0.0 ;
float value;
float value1;
float mTesla;
float uTesla;
int hall_digital = 2;
int chk;
float hum; //Stores humidity value
float temp; //Stores temperature value

void setup()
{
  Serial.begin(9600); //Bd rate
  display.begin(SH1106_SWITCHCAPVCC, 0x3C);
  delay(2000);
  display.clearDisplay();
  display.setTextColor(WHITE);
}

void loop()
{
  //----check if 2-milisec has gone-----
```

```

unsigned long presentMillis = millis();
while (millis() - presentMillis < 2) // ms
{
  ; //wait here
}
presentMillis = millis(); //2-milli second has gone

//----take a reading from hall sensor
float Halldata = (analogRead(A1));
sampleArray[sampleCounter] = Halldata;
sampleCounter++;
if (sampleCounter == 500) //500 samples are collected
{

  for (int i = 0; i < 500; i++)
  {
    cube1 = cube1 + sampleArray[i];
  }
  cube1 = (cube1 / 500);
  value = cube1*5;
  value1 = value/1023;
  mTesla = abs(((value1 - 2.5)/0.015));
  uTesla = (mTesla*1000);
  hum = dht.readHumidity();
  temp= dht.readTemperature();

  Serial.print("The Hall Data is ");
  Serial.print(cube1, 2); //show data with 2-digit after decimal point
  Serial.print("Temperature: ");
  Serial.println(temp);
  display.clearDisplay();
  display.setTextSize(1);
  display.setCursor(0,0);
  display.print("Milli Tesla: ");
  display.setTextSize(2);
  display.setCursor(0,10);
  display.print(Halldata);
  display.print(" ");
  display.setTextSize(1);
  display.setTextSize(2);
  display.print("mT");
  display.setTextSize(1);
  display.setCursor(0,35);
  display.print(analogRead(A1));
  display.setTextSize(2);
  display.setCursor(0,45);
  display.print(cube1);

```

```
//----reset array and sampleCounter---  
// sampleArray[20] = {};  
sampleCounter = 0;  
// display.display();  
//-----  
}  
}
```

### ESP32 program

```
#include <WiFi.h>  
#include <PubSubClient.h>  
  
//char sampleArray[100] = {};  
//int sampleCounter = 0;  
const char* ssid = "Wifi Name";  
const char* password = "Wifi Password";  
const char* mqttServer = "test.mosquitto.org";  
const int mqttPort = 1883;  
const int potPin = 34;  
float sensor;  
  
WiFiClient espClient;  
PubSubClient client(espClient);  
  
void setup() {  
  Serial.begin(115200);  
  WiFi.begin(ssid, password);  
  
  while (WiFi.status() != WL_CONNECTED) {  
    delay(500);  
    Serial.println("Connecting to WiFi..");  
  }  
  
  Serial.println("Connected to the WiFi network");  
  
  client.setServer(mqttServer, mqttPort);  
  
  while (!client.connected()) {  
    Serial.println("Connecting to MQTT...");  
  
    if (client.connect("ESP32Client")) {  
      Serial.println("connected");  
    } else {
```

```
Serial.print("failed with state ");
Serial.print(client.state());
delay(2000);

}
}

client.publish("Energyiscool", "Hello from ESP32");

}

void loop() {

// sensor = analogRead(potPin);
// char sensor1 [8];
// dtostrf(sensor, 1, 2, sensor1);
  client.publish("Energyiscool", "Harvesting");
  client.loop();
  delay(500);
}
```

### Node-Red Program

```
var d = new Date();
var t = d.getTime();
payload={"date":d, "time": t, "payload":msg.payload, "topic":msg.topic};
//msg.payload=JSON.stringify(payload);
msg.payload=payload;
//msg.filename="Testing123.txt";
return msg;

msg.topic = "JSON File";
msg.payload = "Hey this is a test ";
msg.attachments = [
  { filename: "test4.json", path: "D:/SGU/Semester 8/Thesis/test4.json" },
  { filename: "test4.json", content: "HelloErik" },
];
return msg;
```

### Nyquist Theorem Calculation

Frequency:            50 Hz

Period: 0,02 Second

Ideally

Sampling rate:  $2 \times 50 \text{ Hz} = 100$

Sampling Interval =  $1/100 = 0,01 \text{ s} = 10 \text{ ms}$

Safety Factor

Sampling rate:  $5 \times 50 \text{ Hz} = 250$

Sampling Interval =  $1/250 = 0,004 \text{ s} = 4 \text{ ms}$

**XLSEMI**

Datasheet

180KHz 60V 5A Switching Current Boost DC/DC Converter	XL6019
---	--------

**XL6019 Electrical Characteristics**

$T_a = 25^\circ\text{C}$ ; unless otherwise specified.

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
<b>System parameters test circuit figure4</b>						
VFB	Feedback Voltage	$V_{in} = 12\text{V to } 20\text{V}, V_{out}=24\text{V}$ $I_{load}=0.1\text{A to } 1\text{A}$	1.213	1.25	1.287	V
$\eta$	Efficiency	$V_{in}=12\text{V}, V_{out}=24\text{V}$ $I_{out}=1\text{A}$	-	93	-	%

**Electrical Characteristics (DC Parameters)**

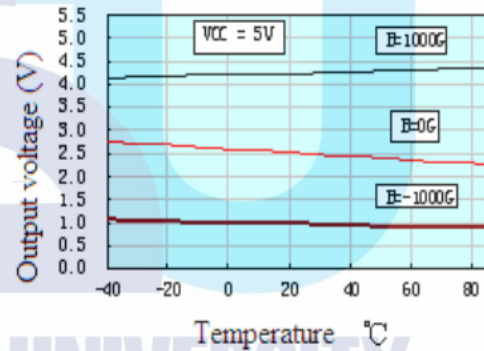
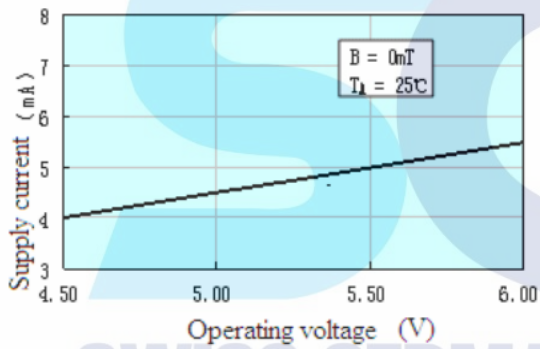
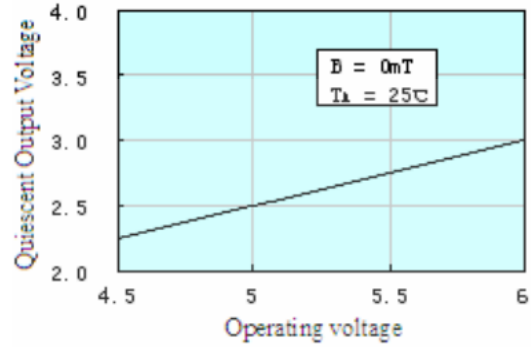
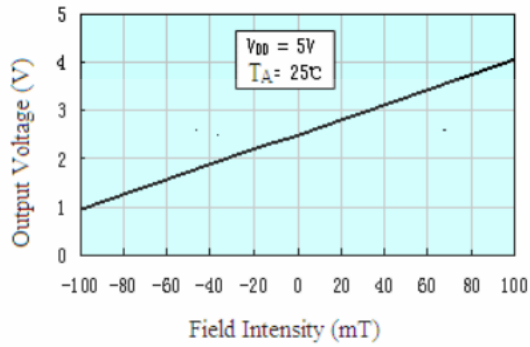
$V_{in} = 12\text{V}, GND=0\text{V}, V_{in}$  &  $GND$  parallel connect a  $100\mu\text{f}/50\text{V}$  capacitor;  $I_{out}=0.5\text{A}, T_a = 25^\circ\text{C}$ ; the others floating unless otherwise specified.

Parameters	Symbol	Test Condition	Min.	Typ.	Max.	Unit
Input operation voltage	$V_{in}$		5		40	V
Shutdown Supply Current	$I_s$	$V_{EN}=0\text{V}$		70	100	$\mu\text{A}$
Quiescent Supply Current	$I_q$	$V_{EN}=2\text{V},$ $V_{FB}=V_{in}$		2.5	5	mA
Oscillator Frequency	$F_{osc}$		144	180	253	KHz
SW OVP	$V_{sw}$	$V_{FB}=0\text{V}$		60		V
Switch Current Limit	$I_L$	$V_{FB}=0\text{V}$		5		A



**SS49E**  
*Linear Hall Effect Sensor*

**Performance Characteristics**



SWISS GERMAN UNIVERSITY





**LIR18650 Datasheet**  
Li-ion Battery  
Edition: NOV. 2010

**5. BASIC CHARACTERISTICS**

5.1 Capacity (25±5°C)	Nominal Capacity: 2600mAh (0.52A Discharge, 2.75V) Typical Capacity: 2550mAh (0.52A Discharge, 2.75V) Minimum Capacity: 2500mAh (0.52A Discharge, 2.75V)
5.2 Nominal Voltage	3.7V
5.3 Internal Impedance	≤ 70mΩ
5.4 Discharge Cut-off Voltage	3.0V
5.5 Max Charge Voltage	4.20±0.05V
5.6 Standard Charge Current	0.52A
5.7 Rapid Charge Current	1.3A
5.8 Standard Discharge Current	0.52A
5.9 Rapid Discharge Current	1.3A
5.10 Max Pulse Discharge Current	2.6A
5.11 Weight	46.5±1g
5.12 Max. Dimension	Diameter(Ø): 18.4mm Height (H): 65.2mm
5.13 Operating Temperature	Charge: 0 ~ 45°C Discharge: -20 ~ 60°C
5.14 Storage Temperature	During 1 month: -5 ~ 35°C During 6 months: 0 ~ 35°C



SWISS GERMAN UNIVERSITY

## CURRICULUM VITAE

### Adinata Danapati

#### Personal Data

First name und Last name : Adinata Danapati  
Place of Birth : Jakarta, Indonesia  
Date of Birth : 18 November 1999  
Address : Jl. Kelapa Lilin IV Blok DB 8 No 2  
Sektor 7A, Gading Serpong  
Tangerang 15810  
Banten, Indonesia  
Nationality : Indonesian  
Mobile Number : +62811-1683-518  
E-Mail : adi.danapati18@gmail.com  
Skype : Adinata Danapati



#### Education

Since 2018 : Mechatronics Engineering at Swiss German University

#### School Education

2015 – 2018 : Santa Laurensia Highschool, Tangerang, Indonesia  
2012 – 2015 : Santa Laurensia Junior Highschool Tangerang, Indonesia  
2006 – 2012 : Santa Laurensia Elementary School Tangerang, Indonesia

#### German Course

11/2019 – 05/2020 : German course at Optimal Studia in Indonesia

#### Training

01/07/2019 – 23/08/2019 : Training at Academy Technical Machine Industry (ATMI)  
Cikarang, Indonesia

Field of Work : Mechanical Engineering

Job Description : - Assembly & Reverse Engineering  
- Benchwork Mechanic  
- CNC - Simulation  
- Conventional Milling and Turning  
- PCB Making (Using Protheus ISIS & ARES)  
- Safety Technic  
- Welding

## **Internship**

13/04/2022 – 13/06/2022 Bachelor Thesis Student at Fachhochschule Südwestfalen

Working Area : Student Research Assistant

Job Description : - IoT Technology for 110-kV-Cable  
- Energy Harvesting  
- Node-Red Development for server

01/03/2021 – 31/08/2021 Internship at Fachhochschule Südwestfalen

Working Area : Student Research Assistant

Job Description : - Research on Energy Harvesting Technology  
- IoT for 110-kV-Cable

29/08/2019 – 25/10/2019 Internship at PT. Gajah Tunggal Tbk. (Tire Manufacturer)

Work Area : Maintenance Engineer

Job Description : - Maintenance of machines in production and workshop area  
- Designing electrical circuit

## **Organizational Experience**

2020 Vice Chairman of The Student's House of Representatives  
(The Students House of Representatives is a student organization that represents the students to the campus)

2020 Member of Marketing division for Mechatronics Day 2020  
(Mechatronics Day is a robotic competition)

2019 Chief of Assessor and Developer division of The Students House of Representatives

2019 Member of Marketing division for Mechatronics Day 2019

2018 Member of Logistics division for Mechatronics Day 4

2018 Liaison Officer for ICONIET 2018 (Seminar about industry 4.0)

### **Special Knowledge**

Language Skills : Indonesian (Mother Language)  
English (Intermediate)  
German (B.1.1)

Computer Skills : Microsoft Word, Excel and PowerPoint  
Solidworks  
Arduino  
LTSpice  
Node-Red  
Mosquitto (MQTT)  
Protheus ISIS & ARES  
Adobe Photoshop, InDesign, & Flash  
Fluid sim P  
Minitab 16  
PSIM

