

Perancangan Sistem Kendali Kendaraan Bermotor Jarak Jauh Menggunakan NodeMCU ESP8266

Choresy Michael G. Butar-Butar¹, Yusran Timur Samuel²

^{1,2} Fakultas Teknologi Informasi Universitas Advent Indonesia

e-mail: ¹ choressy911@gmail.com, ² y.tarihoran@unai.edu

Abstrak

Menghidupkan kendaraan bermotor dari jarak jauh adalah merupakan satu kendala bagi pemilik kendaraan bermotor, hal ini dapat terjadi apabila dia bepergian jauh dan tidak ada orang yang dapat menghidupkan kendaraannya. Untuk itu penulis membuat suatu penelitian untuk memanfaatkan teknologi IoT (*internet of things*) yang dapat mematikan dan menghidupkan kendaraan bermotor dari jarak jauh. Dengan menggunakan NodeMCU ESP8266 yang dihubungkan dengan 2 *relay* dan 2 sensor yaitu sensor getar SW420 dan sensor kemiringan (*tilt*) SW460D dapat menghasilkan konfigurasi *hardware* untuk mengendalikan kendaraan bermotor jarak jauh. Sebagai antar muka pengendali, telah dibuat satu aplikasi yang diprogram menggunakan aplikasi Blynk yang berbasis android. Rancangan yang penulis teliti ini, telah di uji cobakan kepada sepeda bermotor dari tanggal 1 Maret hingga 1 April, di mana hasilnya selalu berhasil mengendalikan kendaraan bermotor dari jarak jauh jika sistem ini mendapatkan sinyal *wifi* dengan koneksi internet.

Kata-kata Kunci: IoT, NodeMCU ESP8266, Sensor getar SW420, sensor kemiringan SW460D.

A Design of Distance Control System for Vehicle Using NodeMCU ESP8266

Abstract

Turning on a vehicle from a distance is an obstacle for the owners. This happens if he travels far and no one can turn on the vehicle. For this reason, the authors conduct a study to utilize IoT technology (*internet of things*) that can turn off and turn on a vehicle remotely. By using ESP8266 NodeMCU which is connected with 2 relays and 2 sensor that is a SW420 vibration sensor and a tilt sensor the SW460D can produce a hardware configuration to control the vehicles from a distance. As an interface controller, one application has been programmed using the Android-based Blynk application. This design has been tested on a motorcycle for a month, from March 1 to April 1. the result shows that it works, and it is always succeeded to control the motorcycle from a far if the system gets a wifi signal with an internet connection.

Keywords: IoT, NodeMCU, SW420 vibration sensor, SW460D tilt sensor.

1. Pendahuluan

Pemanfaatan teknologi internet sudah berkembang dengan sangat pesat, ini didorong oleh tersedianya perangkat keras internet dengan harga relatif murah dan tersedianya berbagai perangkat lunak yang dapat diprogram atau dikonfigurasi dengan cara yang relatif mudah. Berdasarkan fenomena tersebut maka dewasa ini berkembang konsep IoT (*Internet of Things*) yaitu memanfaatkan teknologi internet untuk mengendalikan peralatan-peralatan elektronik yang terdapat dalam kehidupan sehari-hari seperti mematikan dan menghidupkan lampu, mengendalikan kulkas, Tv dan lain-lain menggunakan perangkat komputer atau *smartphone*. Dengan adanya fenomena tersebut, maka berbagai peralatan dapat

dikendalikan dengan hanya menggunakan sebuah *smartphone* dan pengendalian tersebut dapat dilakukan dari jarak jauh.

Untuk lebih memperluas pemanfaatan internet dalam kehidupan sehari-hari maka penulis tertarik untuk mengadakan penelitian tentang pemanfaatan internet untuk pengendalian kendaraan bermotor jarak jauh. Hal ini ditujukan kepada mereka yang memiliki kendaraan bermotor, namun sering bepergian jarak jauh meninggalkan kendaraannya beberapa hari atau bahkan beberapa minggu. Sebagaimana kita ketahui bahwa kendaraan bermotor membutuhkan pemeliharaan. Salah satu hal yang perlu diperhatikan oleh pemilik kendaraan adalah, tidak baik jika kendaraan bermotor tidak dihidupkan dalam jangka waktu yang lama. Beberapa efek kendaraan bermotor yang tidak dihidupkan dalam jangka lama seperti, habis baterai [1], pelumas yang mengendap di dasar mesin dan efek lain yang tidak baik bagi kendaraan bermotor tersebut. Untuk itu diperlukan suatu sistem yang dapat mengendalikan kendaraan bermotor dari jarak jauh.

Sudah ada beberapa penelitian yang membahas tentang bagaimana menghidupkan kendaraan bermotor dari jarak jauh khususnya sepeda motor[2] yang menggunakan telepon selular dengan sistem pengenalan sinyal DTMF. Kemudian penelitian tentang bagaimana menyalakan (*start engine*) dan menyalakan alarm pada sepeda motor menggunakan *arduino uno* berbasis android [3]. Penelitian ini menggunakan *bluetooth* HC-05 untuk menerima perintah yang dikirim dari perangkat android. Namun sejauh yang peneliti ketahui belum ada penelitian yang menggunakan *WiFi* dan memiliki umpan balik untuk menentukan apakah motor sudah hidup atau belum dan menentukan apakah posisi motor miring atau tegak menggunakan kaki dua, sebab untuk motor khususnya jenis *matic*, tidak dapat dihidupkan apabila standar (kaki) samping tidak ditarik atau dilipat.

Oleh karena itu penulis tertarik untuk mengadakan penelitian tentang bagaimana merancang suatu sistem untuk menghidupkan kendaraan bermotor dari jarak jauh dengan memanfaatkan teknologi IoT menggunakan *WiFi* berbasis android. Kemudian penulis juga meneliti tentang bagaimana menggunakan sensor getar untuk menentukan apakah motor sudah menyala atau belum dan mengetahui apakah kondisi motor berada pada posisi yang siap untuk di *starter* dengan sensor kemiringan, sebab motor yang menggunakan kaki samping, akan mengakibatkan posisi motor yang miring sehingga tidak bisa dihidupkan. Dengan demikian diharapkan kendaraan bermotor yang ditinggal dengan jarak yang jauh dapat tetap dihidupkan sehingga meminimalisasi efek negatif dari kendaraan bermotor yang tidak dihidupkan dengan waktu yang cukup lama.

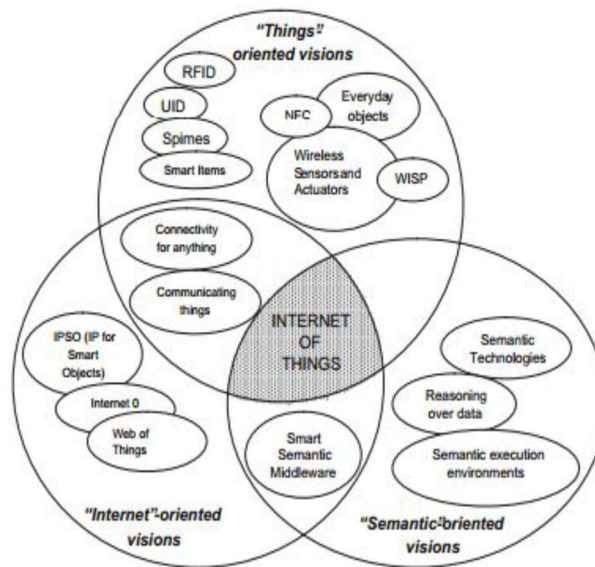
Dalam penelitian ini, kendaraan bermotor yang dimaksud hanya meneliti sepeda motor roda dua saja, di mana secara umum, cara untuk menghidupkan sepeda motor, mobil, dan kendaraan bermesin lainnya, menggunakan prinsip yang sama. Sedangkan pengendalian, dibatasi hanya pada menyalakan dan mematikan mesin motor.

2. Landasan Teori

Definisi *Internet of Things*

Internet of Things (IoT) adalah sebuah teknologi yang mampu menyelesaikan permasalahan sehari-hari dengan memanfaatkan interkoneksi baik secara fisik maupun virtual menggunakan layanan internet. *Internet of Things* terdiri dari dua buah kata yaitu *Internet* dan *Things*. *Internet* adalah jaringan komputer yang digunakan untuk berkomunikasi dan berbagi informasi di ruang lingkup tertentu menggunakan protokol internet (TCP/IP), sementara *things* dapat diartikan sebagai peralatan-peralatan yang dapat dihubungkan dengan sensor dan tersambung dengan internet untuk mengirimkan keadaan peralatan tersebut. Namun informasi yang didapatkan melalui sensor tersebut perlu diolah agar dapat dimengerti oleh pemakai, untuk itu diperlukan teknologi *semantic*. [4]

Dengan demikian komponen *Internet of Thing* terdiri dari 3 komponen yaitu *Internet*, *Things* dan *Semantic*. Gambar 1. berikut ini menggambarkan konsep utama, teknologi dan standarisasi dari paradigma *Internet of Things* [5].

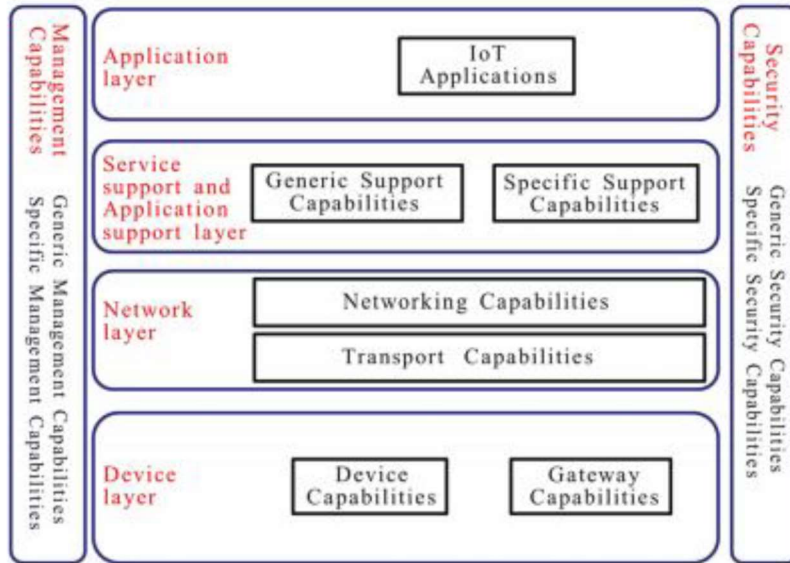


Gambar 1 Komponen Pendukung *Internet of Things* [4]

Model Referensi IoT

Menurut Sitrusta [4], Model referensi IoT (Gambar 2) terdiri dari 4 layer dan 2 kemampuan pendukung. 4 layer tersebut adalah :

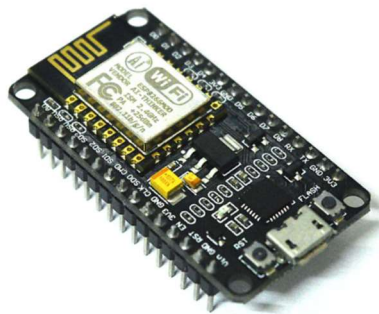
1. Layer Aplikasi : Pada layer inilah seluruh aplikasi dari IoT berada.
2. Layer Layanan dan Pendukung Aplikasi. Pada layer ini terdapat 2 kelompok yaitu :
 - o Layanan Dukungan *Generic* seperti pengolahan data atau penyimpanan data. Kemampuan ini dapat juga dipanggil oleh kemampuan dukungan khusus.
 - o Kemampuan Dukungan Khusus : Kemampuan untuk memberi fungsi pendukung yang berbeda beda untuk aplikasi IOT.
3. Layer Jaringan. Terbagi menjadi dua bagian yaitu :
 - o Kemampuan Membangun Jaringan yaitu kemampuan untuk menyediakan fungsi kontrol terhadap konektivitas jaringan.
 - o Kemampuan Transportasi : Menyediakan konektivitas untuk pengangkutan layanan IOT dan aplikasi informasi data khusus, serta transportasi kontrol IOT.
4. Layer Perangkat. Dapat dibagi ke dalam dua kelompok :
 - o Kemampuan Perangkat : Kemampuan interaksi langsung dengan jaringan komunikasi secara langsung maupun tidak langsung. Kemampuan interaksi dengan jaringan *Ad-Hoc*.
 - o Kemampuan *Gateway* : Kemampuan untuk mendukung berbagai jenis teknologi kabel atau nirkabel, seperti *Controller Area Network (CAN)*, *ZigBee*, *Bluetooth* atau *WiFi*. Pada layer jaringan, *gateway* memiliki kemampuan untuk dapat berkomunikasi dengan berbagai teknologi, seperti *PSTN (Public Switched Telephone Network)*, generasi kedua atau generasi ketiga (2G atau 3G), jaringan *LTE (Long-Term Evolution)*, *Ethernet* atau *DSL (Digital Subscriber Lines)*. [4]



Gambar 2 Model Referensi *Internet of Things* [4]

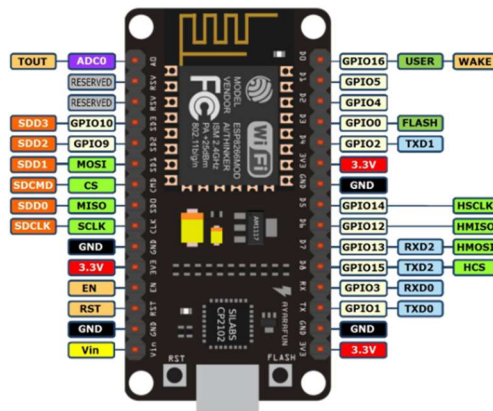
NodeMCU ESP8266 ESP12

Pada penelitian ini, penulis menggunakan *microcontroller* yang sudah dilengkapi dengan modul *WiFi* dengan *chipset* ESP8266. Gambar 3 berikut ini adalah gambar NodeMCU ESP8266 ESP12.



Gambar 3 NodeMCU ESP8266

NodeMCU adalah platform IoT yang *open source*, alat ini memiliki *firmware* yang jalan di atas ESP8266 *WiFi* SoC yang dibuat oleh *Espressif Systems*. Perangkat kerasnya berbasis modul ESP-12. NodeMCU ESP8266 menggunakan standar *wireless* IEEE 802.11 b/g/n dengan *range* frekuensi 2.412 - 2.484 GHz. Memiliki jenis *security* WEP/WPA-PSK/WPA2-PSK, dan jenis enkripsi WEP64/WEP128/TKIP/AES. Pin diagram dari NodeMCU ESP8266 dapat dilihat pada Gambar 4 berikut ini.



Gambar 4 Pin Diagram NodeMCU ESP8266 [6]

Relay

Relay adalah sakelar yang dioperasikan secara elektrik, kebanyakan relay menggunakan *electromagnetic* untuk mengadakan gerakan buka tutup atau *on/off* arus listrik. Biasanya relay ini menggunakan arus yang lemah (*volt* dan *ampere* kecil) untuk mematikan atau menghidupkan arus yang lebih tinggi baik *volt* dan *ampere*-nya. Pada gambar 5 berikut ini adalah relay yang digunakan untuk Arduino di mana arus pengendalinya 5V dan dapat digunakan sebagai sakelar arus untuk kapasitas AC hingga 250V 10A, dan DC hingga 30V 10A.



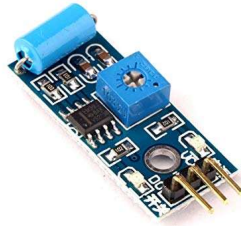
Gambar 5 Relay 1 Channel untuk Arduino

Relay biasanya digunakan sebagai sakelar untuk peralatan seperti: sistem kontrol lampu, peralatan telekomunikasi, pengontrol proses industri, kontrol lalu lintas, kontrol *drive* motor, sistem perlindungan tenaga listrik, komputer, peralatan otomotif serta peralatan rumah tangga.

Sensor Getaran SW420 (Vibration Switch Sensor)

Kegunaan sensor ini adalah untuk mendeteksi apakah ada getaran atau tidak, Bila tidak ada getaran, sensor getaran terhubung dan nilai *output* rendah. Lampu indikator menyala. Bila terdeteksi getaran, sensor segera terputus dan nilai *output* tinggi. Lampu indikator tidak menyala. *Output*-nya dapat langsung dihubungkan ke *microcontroller* untuk mendeteksi nilai rendah dan tinggi sehingga dapat diketahui apakah sedang terjadi getaran atau tidak.

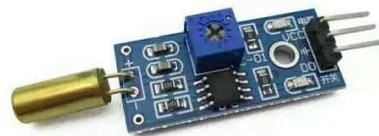
Sensor getar biasanya digunakan pada alarm motor/mobil/jendela/pintu rumah. Tegangan kerja sensor ini adalah 3.3V-5V dengan menggunakan *comparator* LM393. Dalam penelitian ini, sensor digunakan untuk mendeteksi apakah motor sudah menyala atau tidak dengan mendeteksi getaran. Karena motor yang menyala pasti akan menghasilkan getaran. Gambar 6 berikut adalah sensor getar yang digunakan pada penelitian ini.



Gambar 6 Sensor Getar SW420

Sensor Kemiringan SW460D

Sensor ini sering juga disebut dengan *tilt sensor*, yaitu sensor yang dapat mendeteksi perubahan pada sudut kemiringan. Ketelitian hingga 15-45 derajat. Sensor ini sering digunakan untuk sistem *monitoring* tekanan ban, lampu sepeda, rotasi bingkai foto digital, rotasi layer kamera video, aplikasi anti-maling, dll. Sensor ini menggunakan *chip comparator* LM393 yang stabil dan tingkat sensitifnya dapat diatur dengan potensiometer warna biru yang sudah tersedia. Sensor ini memiliki 3 kaki, yaitu VCC yang dapat dihubungkan dengan arus 3.3v s/d 5v, GND sebagai negatif dan DO untuk digital *output*.



Gambar 7 Sensor Kemiringan SW460D

Penelitian Sebelumnya

Sudah ada penelitian tentang bagaimana mengendalikan motor dari jarak jauh yaitu menghidupkan dan mematikan sepeda motor menggunakan telepon seluler. Metode yang digunakan adalah dengan menggunakan *microcontroller* AT Mega 16. Untuk menghidupkan sepeda motor menggunakan tombol 1 dan mematikan menggunakan tombol 2 dari telepon selularnya. Untuk mengenali sinyal ini digunakan IC DTMF MT8870, yang selanjutnya diproses oleh *microcontroller* untuk menghidupkan atau mematikan mesin motor[2].

Pengendalian jarak jauh lainnya adalah untuk menghidupkan dan mematikan lampu listrik di rumah. Penelitiannya menggunakan Arduino dan Modul *Wifi Esp8266*. *Interface* dibuat berbasis web dengan HTML5. Arduino dihubungkan dengan *relay* dan menerima instruksi untuk menghidupkan dan mematikan lampu. [7].

Kemudian penelitian lain tentang menghidupkan mesin (*start engine*) dan menyalakan alarm pada sepeda motor menggunakan arduino uno berbasis android [3]. Penelitian ini menggunakan *bluetooth* HC-05 untuk menerima perintah yang dikirim dari perangkat telepon berbasis android.

3. Metode Penelitian

Analisa Kebutuhan *Hardware* dan *Software*

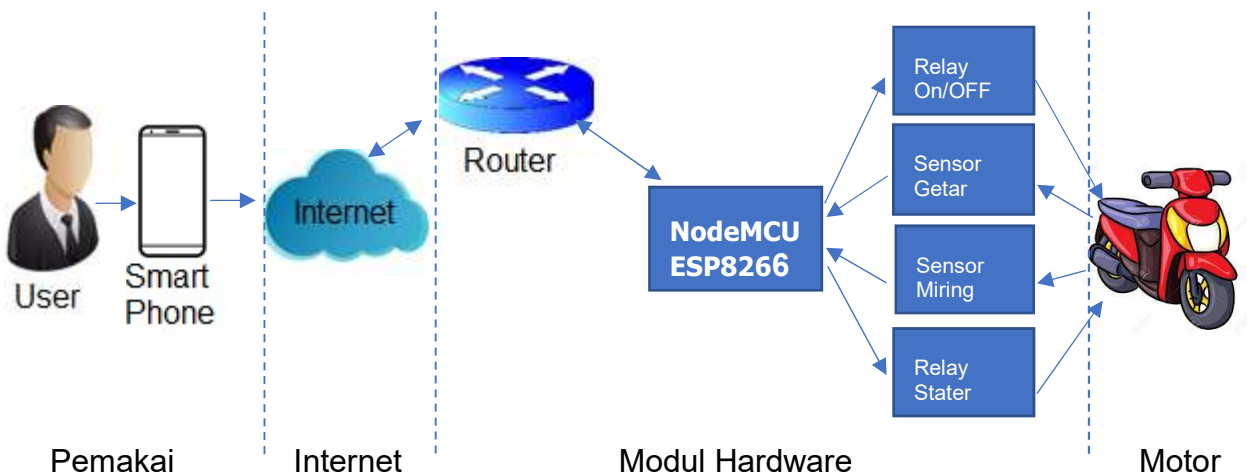
Pada Penelitian ini, penulis menggunakan beberapa peralatan (*hardware*) untuk dapat mengendalikan motor jarak jauh via internet di antaranya NodeMCU ESP8266, modul ini adalah *microcontroller* yang sudah

memiliki kemampuan untuk terhubung dengan *WiFi* dan dapat diprogram untuk menerima dan mengeluarkan sinyal melalui kaki-kakinya. NodeMCU ESP8266 dihubungkan dengan *relay* sehingga dapat digunakan sebagai pemutus atau penyambung arus. Arus yang perlu di kendalikan adalah arus posisi kunci kontak *on* atau *off* dan arus tombol *starter* (menekan atau melepas *starter*), sehingga dalam perancangan sistem ini membutuhkan dua buah *relay*. Untuk mengetahui apakah motor sudah hidup atau belum, bisa dipantau dari getaran dari motor. Sehingga dibutuhkan sensor getar. Sensor getar yang digunakan adalah SW420. Dan sebagaimana lazimnya motor berjenis *matic*, jika standar samping digunakan untuk menyanggah, maka motor tidak dapat di *starter*. Untuk mengetahui apakah standar samping digunakan atau tidak, bisa dilihat dari kemiringan. Untuk mengetahui kondisi ini, digunakan sensor kemiringan SW460D.

Software yang digunakan pada penelitian ini adalah Arduino (IDE), merupakan aplikasi yang sudah terintegrasi untuk melakukan pengembangan pada *microcontroller*. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Arduino IDE dibuat dengan menggunakan bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan *library C/C++* yang biasa disebut *Wiring* yang membuat operasi *input* dan *output* menjadi lebih mudah. *Software* lain yang digunakan adalah Blynk. Blynk adalah platform untuk aplikasi OS Mobile (iOS dan Android) yang bertujuan untuk kendali modul *Arduino*, *Raspberry Pi*, *ESP8266*, *WEMOS D1*, dan modul sejenisnya melalui Internet. Jadi pada program inilah diatur tampilan kendali motor jarak jauh yang dapat dijalankan melalui aplikasi berbasis android.

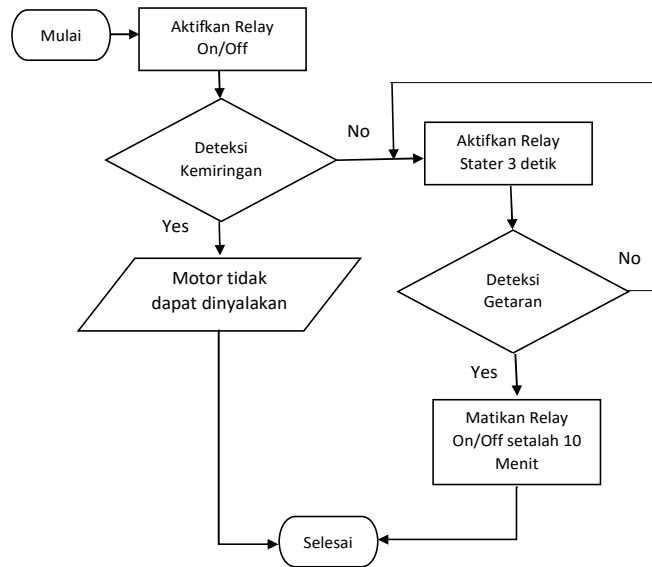
Perancangan Sistem

Arsitektur sistem pengendalian motor jarak jauh dapat dilihat pada Gambar 8 berikut ini.



Gambar 8 Arsitektur Pengendalian Motor Jarak Jauh

Sistem ini terbagi dalam 4 bagian, yaitu: pemakai, internet, modul *hardware* dan motor yang dikendalikan. Pemakai adalah orang yang akan menghidupkan motornya dari jarak jauh menggunakan aplikasi berbasis android. Aplikasi yang dirancang berbasis android akan melakukan komunikasi ke perangkat *microcontroller* via internet. *Microcontroller* dapat mengaktifkan *relay starter* atau *relay on/off* sesuai kebutuhan setelah mendapatkan informasi dari sensor kemiringan dan sensor getar. Apabila sensor getar menerima sinyal positif yang berarti motor hidup atau bergetar maka proses *starter* tidak dapat dilakukan sampai kondisi motor mati. Sedangkan sensor kemiringan berfungsi untuk melihat apakah motor menggunakan standar kaki dua atau satu (standar samping). Jika motor sedang menggunakan standar samping maka fungsi *starter* tidak dapat dijalankan atau tombol *starter* tidak aktif. Gambar 9 berikut ini menunjukkan *flowchart* dari sistem yang dibangun.

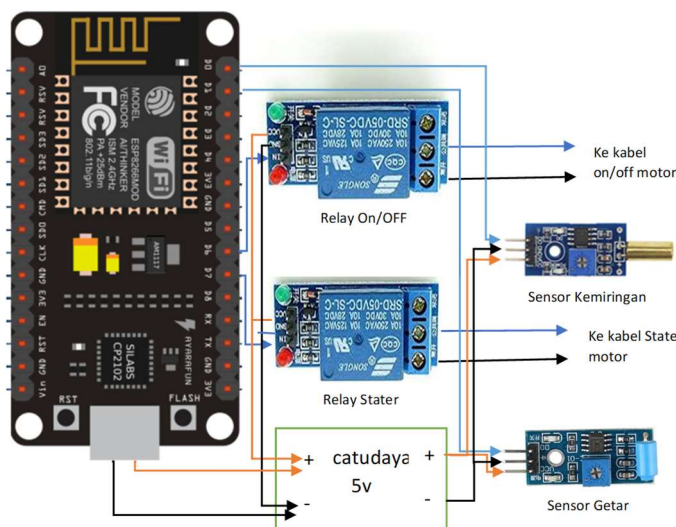


Gambar 9 Flowchart Sistem Pengendalian Motor Jarak Jauh

Dari Gambar 9 terlihat bahwa motor tidak dapat dinyalakan apabila sensor kemiringan mendeteksi motor miring, atau tidak menggunakan standar kaki dua. Hal tersebut sesuai dengan ketentuan pabrik bahwa motor tidak dapat dihidupkan jika menggunakan standar samping. Bila sensor kemiringan mendeteksi motor menggunakan standar kaki dua, maka motor di *starter* selama 3 detik dan apabila sudah hidup yang ditandai dengan sensor getar memberi nilai *HIGH*, maka motor akan menyala selama 10 menit dan akan membuat *relay on/off* berada pada posisi *off*.

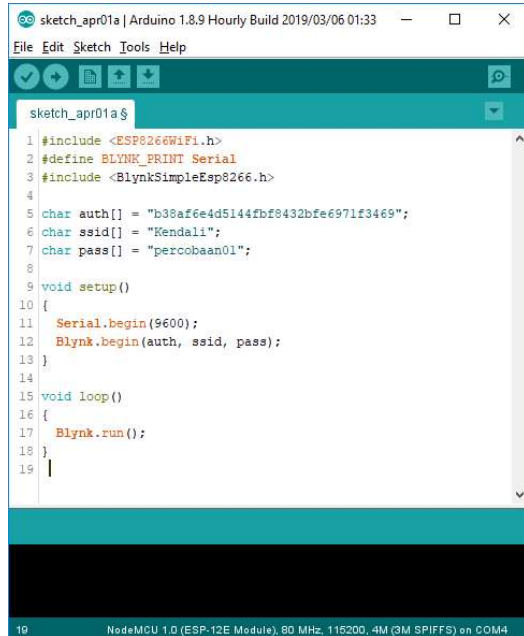
4. Hasil dan Implementasi

Konfigurasi *hardware* untuk pengendali motor jarak jauh dengan memanfaatkan sensor kemiringan dan sensor getar dapat dilihat pada Gambar 10 berikut.



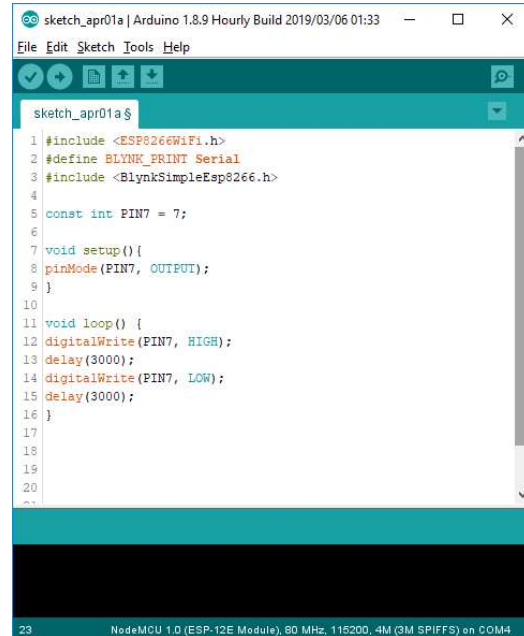
Gambar 10 Konfigurasi Hardware Sistem Pengendalian Motor Jarak Jauh

Pemrograman sistem ini menggunakan aplikasi Arduino IDE yaitu terdiri dari beberapa proses yaitu (1) proses sinkronisasi *WiFi* dan aplikasi Blynk (Gambar 11), (2) proses pengaturan *relay* (Gambar 12), (3) proses pengaturan sensor kemiringan (Gambar 13) dan (4) proses pengaturan sensor getar (Gambar 14).



```
sketch_apr01a | Arduino 1.8.9 Hourly Build 2019/03/06 01:33
File Edit Sketch Tools Help
sketch_apr01a $
1 #include <ESP8266WiFi.h>
2 #define BLYNK_PRINT Serial
3 #include <BlynkSimpleEsp8266.h>
4
5 char auth[] = "b38af6e4d51447bf8432bfe6971f3469";
6 char ssid[] = "Kendali1";
7 char pass[] = "percobaan01";
8
9 void setup()
10 {
11   Serial.begin(9600);
12   Blynk.begin(auth, ssid, pass);
13 }
14
15 void loop()
16 {
17   Blynk.run();
18 }
19 |
NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on COM4
```

Gambar 11 Sinkronisasi Rangkaian



```
sketch_apr01a | Arduino 1.8.9 Hourly Build 2019/03/06 01:33
File Edit Sketch Tools Help
sketch_apr01a $
1 #include <ESP8266WiFi.h>
2 #define BLYNK_PRINT Serial
3 #include <BlynkSimpleEsp8266.h>
4
5 const int PIN7 = 7;
6
7 void setup() {
8   pinMode(PIN7, OUTPUT);
9 }
10
11 void loop() {
12   digitalWrite(PIN7, HIGH);
13   delay(3000);
14   digitalWrite(PIN7, LOW);
15   delay(3000);
16 }
17
18
19
20
NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on COM4
```

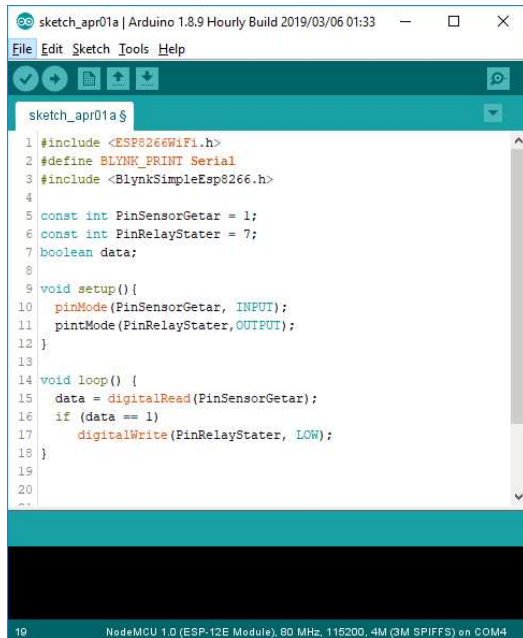
Gambar 12 Pengaturan Relay

Pada proses sinkronisasi *WiFi* dan aplikasi Blynk (Gambar 11), penulis menetapkan *SSID* yang nantinya akan digunakan pada *router* atau *hotspot* yang digunakan, penetapan ini harus dilakukan dari *coding*. Demikian juga dengan *password* untuk terhubung dengan *hotspot*, ditetapkan dari proses ini. Selain *SSID* dan *password*, pada proses ini juga harus di ikut sertakan kode autentifikasi yang diperoleh dari aplikasi Blynk. Setelah itu baru proses sinkronisasi dapat dilakukan dan *interface* Blynk dapat berkomunikasi dengan rangkaian pada Gambar 10.

Pada proses pengaturan *relay* (Gambar 12), *relay* yang digunakan ada 2 jenis yaitu untuk *relay* kunci kontak *on/off* dan *relay starter*. *Relay starter* dihubungkan dengan kaki D7 dan di set sebagai *OUTPUT*. *Starter* dilakukan dengan memberikan nilai *HIGH* pada kaki D7 tersebut. Demikian juga dengan *relay* kunci kontak, *relay* ini menggunakan kaki D6 yang diatur sebagai *OUTPUT*. Proses *starter* dilakukan dengan memberikan nilai *HIGH* pada kaki D6.

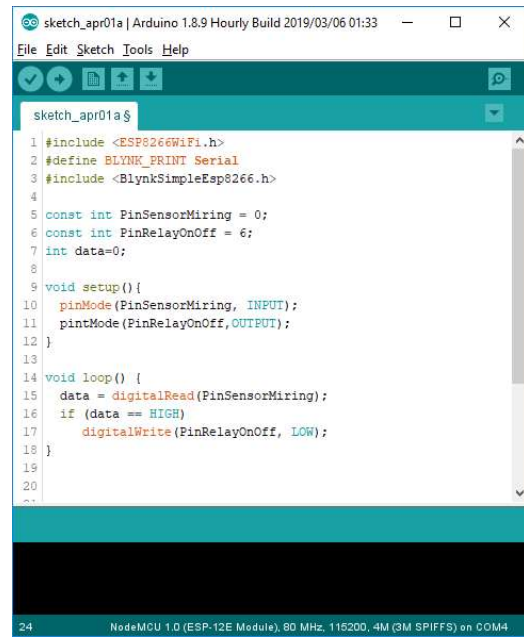
Untuk mengendalikan rangkaian dari jarak jauh, digunakan aplikasi android Blynk yang sudah dikonfigurasi agar memfungsikan *relay* sesuai dengan aturan *flowchart* pada Gambar 9 di atas. Adapun tampilan *interface* pengendaliannya dapat dilihat pada Gambar 15.

Sistem ini sudah di uji cobakan pada sebuah motor selama 1 bulan yaitu mulai dari tanggal 1 Maret hingga 1 April. Motor yang digunakan adalah Yamaha Mio M3 Tahun 2015 dan aplikasi android pengendali dipasang pada *Smartphone* Xiaomi Redmi Note 5 Pro, RAM 4 *Gigabyte* dan ROM 64 *Gigabyte*. Didapati bahwa sistem ini selalu dapat berjalan dengan baik yaitu dapat menghidupkan dan mematikan motor dari jarak jauh dan mesin dapat mati secara otomatis setelah menyala selama 10 menit atau dimatikan secara manual.



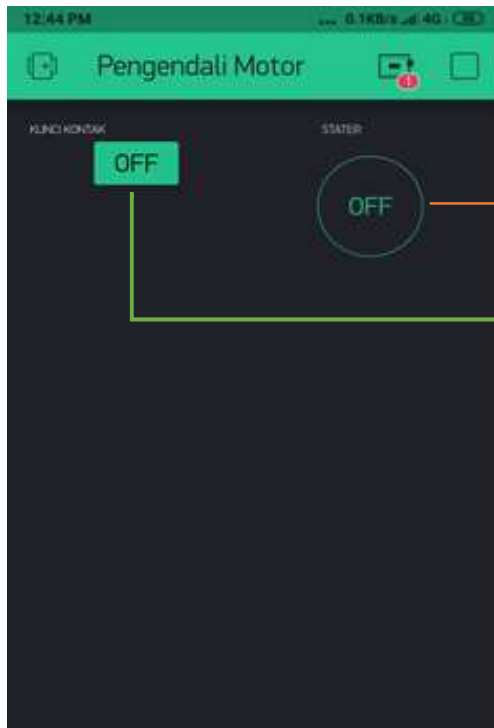
```
1 #include <ESP8266WiFi.h>
2 #define BLYNK_PRINT Serial
3 #include <BlynkSimpleEsp8266.h>
4
5 const int PinSensorGetar = 1;
6 const int PinRelayStater = 7;
7 boolean data;
8
9 void setup() {
10   pinMode(PinSensorGetar, INPUT);
11   pinMode(PinRelayStater, OUTPUT);
12 }
13
14 void loop() {
15   data = digitalRead(PinSensorGetar);
16   if (data == 1)
17     digitalWrite(PinRelayStater, LOW);
18 }
19
20
21
```

Gambar 13. Pengaturan Sensor Getar



```
1 #include <ESP8266WiFi.h>
2 #define BLYNK_PRINT Serial
3 #include <BlynkSimpleEsp8266.h>
4
5 const int PinSensorMiring = 0;
6 const int PinRelayOnOff = 6;
7 int data=0;
8
9 void setup() {
10   pinMode(PinSensorMiring, INPUT);
11   pinMode(PinRelayOnOff, OUTPUT);
12 }
13
14 void loop() {
15   data = digitalRead(PinSensorMiring);
16   if (data == HIGH)
17     digitalWrite(PinRelayOnOff, LOW);
18 }
19
20
21
```

Gambar 14. Pengaturan Sensor Kemiringan



Mengendalikan Tombol Starter

Mengendalikan kunci kontak (ON atau OFF)

Gambar 15 Tampilan *Interface* Pengendali Motor Jarak Jauh

5. Kesimpulan

Berdasarkan penelitian yang sudah dilakukan dalam membangun sebuah sistem pengendali motor jarak jauh dapat disimpulkan :

1. Sistem pengendali motor jarak jauh dapat dibuat dengan menggunakan *microcontroller* NodeMCU ESP8266 yang memiliki modul *WiFi*. Sistem ini menggunakan 2 buah *relay* untuk mengendalikan kunci kontak (*on/off*) dan tombol *starter* di motor. Sistem ini dilengkapi dengan 2 buah sensor yaitu sensor getar dan sensor kemiringan.
2. *Interface* yang dirancang menggunakan aplikasi Blynk berbasis android dapat mengendalikan peralatan *hardware* yang telah dikonfigurasi sesuai dengan tujuannya. Aplikasi ini dijalankan pada *smartphone* android yang terhubung dengan internet, baik menggunakan *WiFi* maupun menggunakan kartu jaringan 3G maupun 4G.
3. Hasil uji coba menunjukkan bahwa aplikasi ini dapat digunakan selama NodeMCU ESP8266 mendapatkan sinyal *WiFi* yang memiliki koneksi internet.

6. Referensi

- [1] R. Saputra and D. Tami, "Trik Sempel Hidupkan Motor Lama Tak Dipakai, Langsung Tokcer," 29 Jun 2018. [Online]. Available: <https://www.msn.com/id-id/otomotif/berita/trik-sempel-hidupkan-motor-lama-tak-dipakai-langsung-tokcer/ar-AAzkyn4>.
- [2] J. Arifin, A. H. Saptadi and A. Silalahi, " Aplikasi Telepon Seluler Sebagai Sistem Kendali Jarak Jauh untuk Menyalakan dan Mematikan Mesin Sepeda Motor," in *PROSIDING SEMINAR NASIONAL MULTI DISIPLIN ILMU UNISBANK*, Semarang, 2017.
- [3] M. I. Andi, Rancangan Sism Start Engine dan Alarm pada Sepeda Motor Menggunakan Arduino Uno Berbasis Android, Surakarta, 2017.
- [4] S. P. Sitrusta Sukaridhoto, Bermain dengan Internet of Things dan Big Data, Surabaya, 2016.
- [5] A. Luigi, I. Antonio and M. Giacomo, "The Internet of Things : A Survey," *Computer Networks*, 2010.
- [6] P. Singh, "NodeMCU ESP12 Dev Kit V1.0 Pin Definition," *IoT Bytes*, [Online]. Available: <https://iotbytes.wordpress.com/nodemcu-pinout/>. [Accessed April 2019].
- [7] S. Samsugi, Ardiansyah2 and D. Kastutara, "INTERNET OF THINGS (IOT): Sistem Kendali Jarak Jauh," in *Prosiding Seminar Nasional XII "Rekayasa Teknologi Industri dan Informasi 2017*, Yogyakarta, 2017.