**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

Tembalang merupakan salah satu daerah yang ada di Semarang. Di Tembalang terdapat Perguruan Tinggi Negeri (PTN) maupun Perguruan Tinggi Swasta (PTS) misalnya Politeknik Negeri Semarang, Universitas Diponegoro, Universitas Pandanaran, dan Politeknik Kesehatan Semarang. Keberadaan 4 perguruan tinggi tersebut mengakibatkan jumlah penduduk di Tembalang semakin meningkat. Hal ini menyebabkan usaha - usaha yang ada di Tembalang semakin meningkat.

Salah satu usaha yang meningkat adalah usaha jasa laundry. Jumlah usaha laundry yang ada di Tembalang lebih dari 50 outlet (Yuniarti Listia, 2017). Jasa laundry merupakan usaha yang menawarkan jasa cuci pakaian mulai dari cuci kering hingga cuci setrika. Dengan adanya jasa laundry semakin memudahkan dalam menyelesaikan pekerjaan yang lain, hanya dengan menyerahkan pakaian yang kotor dan menunggu selesai dilaundry. Biasanya hal tersebut sering dilakukan mahasiswa, karyawan, bahkan ibu rumah tangga yang tidak memiliki waktu untuk mencuci pakaian. Energi mereka sudah digunakan untuk aktivitas mereka yang lebih padat, sehingga lebih memilih menyerahkannya ke jasa laundry.

Sementara itu, jasa laundry sangat bergantung pada kondisi cuaca. Kondisi cuaca setiap harinya tidak bisa diprediksi terkadang hujan terkadang cerah. Apabila terjadi hujan jasa laundry memiliki kendala yaitu ketika proses penjemuran terjadi hujan maka pakaian harus segera dimasukkan ke dalam rumah agar tidak basah. Ketika pakaian yang terjemur basah lagi maka akan memakan waktu kembali untuk mengeringkannya dan tentunya sangat merugikan jasa laundry. Kondisi seperti itu sering terjadi di Berkah Laundry, salah satu jasa laundry di daerah Bulusan, Tembalang. Selain itu, Berkah Laundry masih belum memiliki karyawan, oleh karena itu membutuhkan waktu yang lebih lama untuk memasukkan pakaian ke dalam rumah.

Berdasarkan permasalahan tersebut, diperlukan sistem kendali dan otomasi dengan judul “**Sistem Kendali Jemuran Antisipasi Hujan Berbasis NodeMCU dan Android**” yang dapat mengendalikan jemuran apabila terjadi hujan sehingga pemilik jasa laundry bisa meninggalkan jemuran ketika berada di luar rumah.

1. **Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan, maka rumusan masalah pada Tugas Akhir ini adalah bagaimana merancang dan membangun Sistem Kendali Jemuran Antisipasi Hujan.

1. **Manfaat**

Hasil dari Tugas Akhir ini diharapkan dapat memberikan manfaat, diantaranya adalah sebagai berikut :

1. Bagi Penulis
2. Menerapkan semua ilmu yang didapatkan di perkuliahan.
3. Memahami masalah-masalah yang ada pada usaha *laundry*.
4. Bagi Pembaca
5. Dapat dijadikan informasi bagi khalayak umum serta dapat menambah pengetahuan dan wawasan mengenai implementasi sistem kendali jemuran antisipasi hujan pada usaha *laundry*.
6. Sebagai bahan referensi bagi pembaca yang akan menyusun tugas akhir mengenai implementasi sistem sendali jemuran antisipasi hujan pada usaha *laundry*.
7. Bagi Pemilik *Laundry*
8. Membantu pemilik *laundry* dalam mengontrol kondisi jemuran.
9. Membantu pemilik *laundry* dalam mempercepat proses pengeringan pakaian.
10. Membantu pemilik *laundry* dalam melindungi jemuran pakaian apabila terjadi hujan.
11. **Tujuan**

Tujuan dari pembuatan Tugas Akhir ini adalah merancang dan membangun Sistem Kendali Jemuran Antisipasi Hujan yang terintegrasi dengan *smartphone* berbasis android yang digunakan sebagai antisipasi hujan apabila terjadi hujan sehingga pemilik *laundry* dapat mengontrol keadaan jemuran.

1. **Batasan Masalah**

Dalam pembuatan Tugas Akhir ini, masalah yang akan dibahas terbatas pada :

1. Sistem yang dibangun menggunakan bahasa *C* untuk hardwarenya dan bahasa *Java* untuk Androidnya.
2. Sistem jemuran dibuat hanya untuk *laundry*.
3. Menggunakan *NodeMCU ESP8266* sebagai alat kontrol.
4. Menggunakan sensor hujan, sensor *LDR*, dan sensor *DHT11.*
5. Menggunakan motor servo sebagai penggerak jemuran.

1. **Sistematika Penulisan**

Sistematika penulisan dari laporan tugas akhir ini adalah :

**BAB I : Pendahuluan**

Berisi tentang latar belakang, perumusan masalah, batasan masalah, manfaat pembuatan sistem, tujuan pembuatan sistem, dan sistematika penulisan laporan.

**BAB II : Tinjauan Pustaka**

Membahas materi – materi yang berkaitan dengan permasalahan tugas akhir.

**BAB III : Kegiatan Pelaksanaan**

Membahas mengenai bahan dan perancangan cara kerja sistem dan metodologi.

**BAB IV : Analisis dan Pembahasan**

Berisi tentang pembuatan sistem, pengoperasian sistem, pengujian sistem dan hasil pengujian sistem.

**BAB V : Kesimpulan**

Berisi tentang kesimpulan dari keseluruhan pembahasan laporan dan saran-saran untuk pengembangan lebih lanjut.

**BAB II**

**TINJAUAN PUSTAKA**

1. **Studi Sebelumnya**

Terdapat beberapa penelitian yang memiliki topik yang hampir serupa dengan penelitian ini, diantaranya:

Tabel 2.1 Penelitian Sebelumnya

|  |  |
| --- | --- |
| Judul | Rancang Bangun Jemuran Otomatis Berbasis Web dengan Kendali Raspberry Pi |
| Nama, Tahun | Faisal, Aliyadi dan Angga, 2018 |
| Gambaran umum penelitian | Komponen yang digunakan adalah Raspberry Pi yang berfungsi untuk mengendalikan jemuran secara otomatis dengan bantuan motor DC. Motor DC bisa dikendalikan melalui web browser. |
| Perbedaan dengan tugas akhir ini | Komponen pengendali yang digunakan adalah NodeMCU. Kemudian, kontrol sistem daan pembacaan data sensor menggunakan aplikasi android. |

Tabel 2.2 Penelitian Sebelumnya

|  |  |
| --- | --- |
| Judul | Rancang Bangun Prototype Alat Penjemur Pakaian Otomatis Berbasis Arduino Uno |
| Nama, Tahun | Darusman, Dahlan dan Hilyana, 2018 |
| Gambaran umum penelitian | Alat jemuran pakaian secara otomatis bergerak menuju ke dalam ruangan apabila terjadi hujan dengan bantuan motor DC. Pengendali yang digunakan adalah Arduino Uno. Sensor yang digunakan hanya sensor hujan. |
| Perbedaan dengan tugas akhir ini | Komponen pengendali yang digunakan adalah NodeMCU. Kemudian sensor yang digunakan adalah sensor hujan, sensor cahaya, dan sensor kelembaban. |

Tabel 2.3 Penelitian Sebelumnya

|  |  |
| --- | --- |
| Judul | Rancang Bangun Sistem Jemuran Otomatis Berbasis Arduino Uno |
| Nama, Tahun | Yuwono dan Alam, 2018 |
| Gambaran umum penelitian | Alat jemuran otomatis menggunakan arduino UNO sebagai kendalinya. Komponen yang digunakan adalah sensor LDR, sensor hujan FC-37, sensor kelembaban digital, dan motor DC. |
| Perbedaan dengan tugas akhir ini | Kendali yang digunakan adalah NodeMCU. Kemudian komponen yang ditambahkan adalah blower fan yang berfungsi sebagai pengering apabila atap tertutup dan terjadi hujan. |

1. **NodeMCU ESP8266**

NodeMcu merupakan sebuah opensource platform IoT dan pengembangan *Kit* yang menggunakan bahasa pemrograman Lua untuk membantu programmer dalam membuat prototype produk IoT atau bisa dengan memakai sketch dengan Arduino IDE. Pengembangan *Kit* ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC, 1-Wire dan ADC (Analog to Digital Converter) semua dalam satu board.



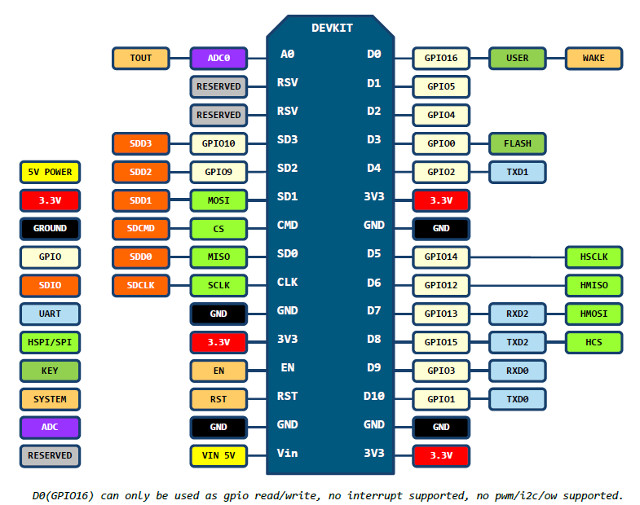
Gambar 2.1 NodeMCU ESP8266

Gambar 2.1 merupakan bentuk fisik dari NodeMCU. Board pada NodeMCU berukuran sangat kecil dengan panjang 4.83 cm dan lebar 2.54 cm, serta dengan berat 7 gram. Walaupun ukurannya yang kecil, board ini sudah dilengkapi dengan fitur WiFi dan *firmwarenya* yang bersifat *opensource.* Penggunaan NodeMcu lebih menguntungkan dari segi biaya maupun efisiensi tempat, karena NodeMcu yang ukurannya kecil, lebih

praktis dan harganya jauh lebih murah dibandingkan dengan Arduino Uno. Arduino Uno sendiri merupakan salah satu jenis mikrokontroler yang banyak diminati dan memiliki bahasa pemrograman C++ sama seperti NodeMcu, namun Arduino Uno belum memiliki modul WiFi dan belum berbasis IoT.

Spesifikasi yang dimliki oleh NodeMCU sebagai berikut :

1. Board ini berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan onboard USB to TTL. Wireless yang digunakan adalah IEE 802.11b/g/n.
2. tantalum capasitor 100 micro farad dan 10 micro farad.
3. 3.3v LDO regulator.
4. Blue led sebagai indikator.
5. Cp2102 usb to UART bridge.
6. Tombol reset, port usb, dan tombol flash.
7. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX
8. 3 pin ground.
9. S3 dan S2 sebagai pin GPIO.
10. S1 MOSI (Master Output Slave Input) yaitu jalur data dari master dan masuk ke dalam slave, sc cmd/sc.
11. S0 MISO (Master Input Slave Input) yaitu jalur data keluar dari slave dan masuk ke dalam master.
12. SK yang merupakan SCLK dari master ke slave yang berfungsi sebagai clock.
13. Pin Vin sebagai masukan tegangan.
14. Built in 32-bit MCU.



Gambar 2.2 GPIO NodeMCU ESP8266 v3

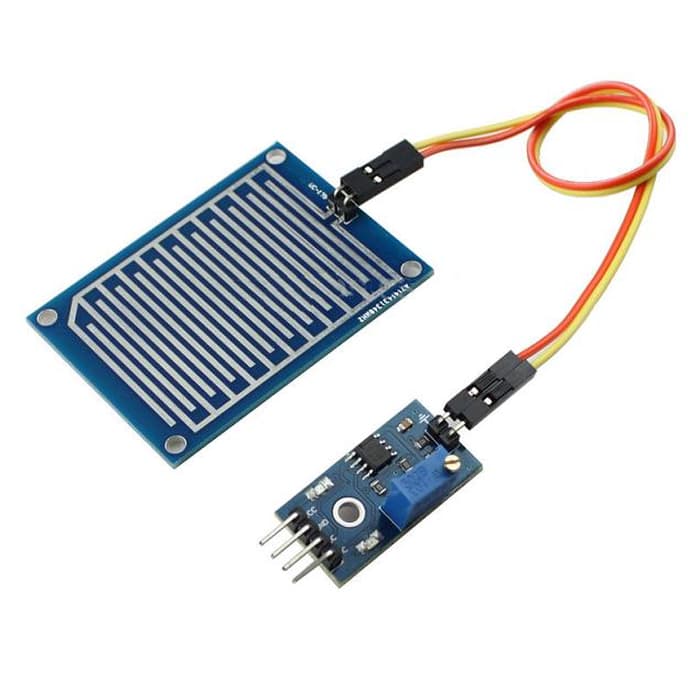
1. RST : berfungsi mereset modul
2. ADC: Analog Digital Converter. Rentang tegangan masukan 0-1v, dengan skup nilai digital 0-1024.
3. EN: Chip Enable, Active High
4. IO16 :GPIO16, dapat digunakan untuk membangunkan chipset dari mode deep sleep.
5. IO14 : GPIO14; HSPI\_CLK
6. IO12 : GPIO12: HSPI\_MISO
7. IO13: GPIO13; HSPI\_MOSI; UART0\_CTS
8. VCC: Catu daya 3.3V (VDD)
9. CS0 :Chip selection
10. MISO : Slave output, Main input
11. IO9 : GPIO9
12. IO10 GBIO10
13. MOSI: Main output slave input
14. SCLK: Clock
15. GND: Ground
16. IO15: GPIO15; MTDO; HSPICS; UART0\_RTS
17. IO2 : GPIO2;UART1\_TXD
18. IO0 : GPIO0
19. IO4 : GPIO4
20. IO5 : GPIO5
21. RXD : UART0\_RXD; GPIO3
22. TXD : UART0\_TXD; GPIO1
23. **Sensor**

Menurut fraden (2003:64), Sensor berasal dari kata *Sense* (merasakan atau mengindera), adalah mengidefinisikan sensor sebagai Piranti yang menerima sebuah stimulus dan meresponnya dengan sebuah sinyal listrik. Lebih jauh *fraden* mendefinisikan stimulus, atau rangsangan, sebagai kuantitas, sifat atau kondisi tertentu yang dapat dirasakan dan diubah menjadi sinyal listrik. Tujuan dari sebuah sensor adalah merespon sejenis masukan dan mengubah masukan tersebut menjadi sinyal listrik. Keluaran output dari sensor dapat berupa arus atau beda potensial. Setiap sensor pada prinsipnya adalah mengubah energi (energy converter). Sensor adalah jenis tranduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar dan kimia menjadi tegangan dan arus listrik. Sensor sering digunakan untuk pendeteksian pada saat melakukan pengukuran dan pengendalian.

Sensor adalah alat yang digunakan untuk mendeteksi dan mengetahui magnitude tertentu. Sensor merupakan jenis transduser yang digunakan untuk mengubah variasi mekanis, magnetis,panas,sinar dan kimia menjadi tegangan dan arus listrik. Sensor memegang peranan penting dalam mengendalikan proses pabrikasi modern (Petruzella, 2001 : 157).

1. **Sensor Hujan**

Sensor hujan adalah sebuah alat yang dapat mendeteksi hujan atau adanya cuaca hujan yang berada di sekitarnya. Sensor hujan dapat digunakan sebagai switch, saat adanya tetesan air hujan yang jatuh melewati raining board yang terdapat pada sensor. Selain itu. sensor hujan juga dapat digunakan untuk mengukur intensitas curah hujan (Sulastri, 2016; Katyal et al., 2016; Ünsal et al., 2016). Output analog sensor hujan digunakan untuk melakukan pendeteksian hujan, dengan kondisi nilai output sensor tinggi pada saat tidak mendeteksi hujan, sedangkan pada saat sensor mendeteksi hujan, nilai output sensor rendah. Gambar 2.3, memperlihatkan modul raindrop sensor (Katyal et al., 2016).



Gambar 2.3 Sensor Hujan

1. **Sensor Cahaya (LDR)**

Sensor cahaya adalah alat yang digunakan dalam bidang elektronika yang berfungsi untuk mengubah besaran cahaya menjadi besaran listrik. Sensor cahaya LDR (Light Dependent Resistor) merupakan suatu jenis resistor yang peka terhadap cahaya. Nilai resistansi LDR akan berubah-ubah sesuai dengan intensitas cahaya yang diterima. Jika LDR tidak terkena cahaya maka nilai tahanan akan menjadi besar (sekitar 10MΩ) dan jika terkena cahaya nilai tahanan akan menjadi kecil (sekitar 1kΩ). (Novianty,Lubis,& Tony, 2012 : 1).Cara kerja dari sensor ini adalah mengubah energi dari foton menjadi elektron, umumnya satu foton dapat membangkitkan satu elektron. Sensor ini mempunyai kegunaan yang sangat luas salah satunya yaitu sebagai pendeteksi cahaya pada jemuran otomatis. Beberapa komponen yang biasanya digunakan dalam rangkaian sensor cahaya adalah LDR (Light Dependent Resistor), Photodiode, dan Photo Transistor.



Gambar 2.4 Sensor Cahaya (LDR)

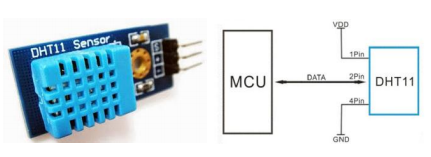
Pada dasarnya sensor cahaya ini merupakan suatu resistor yang memiliki nilai hambatan bergantung pada jumlah cahaya yang jatuh pada permukaan sensor tersebut. LDR dapat dibuat dari semikonduktor beresistensi tinggi yang tidak dilindungi dari cahaya. Jika cahaya yang mengenainya memiliki frekuensi yang cukup tinggi, foton yang diserap oleh semikonduktor akan menyebabkan elektron memiliki energi yang cukup untuk meloncat ke pita konduksi. Elektron bebas yang dihasilkan dan pasangan lubangnya akan mengalirkan listrik, sehingga menurunkan resistansinya.

Komponen yang menggunakan sensor cahaya berikutnya adalah Photo Transistor , secara sederhana adalah sebuah transistor bipolar yang memakai kontak (junction) base-collector yang menjadi permukaan agar dapat menerima cahaya sehingga dapat digunakan menjadi konduktivitas transistor. Secara lebih detail Photo Transistor merupakan sebuah benda padat pendeteksi cahaya yang memiliki gain internal. Hal ini yang membuat foto transistor memiliki sensivitas yang lebih tinggi dibandingkan photodiode / foto diode, dalam ukuran yang sama. Alat ini dapat menghasilkan sinyal analog maupun sinyal digital. Photo Transistor sejenis dengan transistor pada umumnya, bedanya pada Photo Transistor dipasang sebuah lensa pemfokus sinar pada kaki basis untuk memfokuskan sinar jatuh pada pertemuan PN.

1. **Sensor DHT11**

DHT11 adalah sensor Suhu dan Kelembaban (air temperature sensor), dia memiliki keluaran sinyal digital yang dikalibrasi dengan sensor suhu dan kelembaban yang kompleks. Teknologi ini memastikan keandalan tinggi dan sangat baik stabilitasnya dalam jangka panjang. Mikrokontroler terhubung pada kinerja tinggi sebesar 8 bit. Sensor ini termasuk elemen resistif dan perangkat pengukur suhu NTC. Memiliki kualitas yang sangat baik, respon cepat, kemampuan anti-gangguan dan keuntungan biaya tinggi kinerja (Budiharto:2012).

Setiap sensor DHT11 memiliki fitur kalibrasi sangat akurat dari kelembaban ruang kalibrasi. Koefisien kalibrasi yang disimpan dalam memori program OTP, sensor internal mendeteksi sinyal dalam proses, kita harus menyebutnya koefisien kalibrasi. Sistem antarmuka tunggal-kabel serial terintegrasi untuk menjadi cepat dan mudah. Kecil ukuran, daya rendah, sinyal transmisi jarak hingga 20 meter, sehingga berbagai aplikasi dan bahkan aplikasi yang paling menuntut. Produk ini 4-pin pin baris paket tunggal. Koneksi nyaman, paket khusus dapat diberikan sesuai dengan kebutuhan pengguna. Spesifikasi dari sensor DHT11 yaitu Supply Voltage: +5 V, Temperature range : 0-50 °C error of ± 2 °C, Humidity : 20-90% RH ±5% RH error, dengan sesifikasi digital interfacing system. Output yang dihasilkan merupakan baris data digital 40 bits, yang terdiri dari 16 bit data temperature, 16 bit data humidity dan 8 bit data parity. [5]

****

Gambar 2.5 Sensor DHT11

Tabel 2.4 Karakteristik Sensor DHT11

|  |  |
| --- | --- |
| Model | DHT11 |
| Power supply | 3-5.5V DC |
| Output signal | digital signal via single-bus |
| Measuring range | humidity 20-90% RH ± 5% RH error temperature 0-50 °C error of ± 2 °C |
| Accuracy | humidity +-4%RH (Max +-5%RH); temperature +-2.0Celsius |
| Resolution or sensitivity | humidity 1%RH; temperature 0.1Celsius |
| Repeatability | humidity +-1%RH; temperature +- 1Celsius |
| Humidity hysteresis | +-1%RH |
| Long-term Stability | +-0.5%RH/year |
| Sensing period | Average: 2s |
| Interchangeability | fully interchangeable |
| Dimensions size | 12\*15.5\*5.5mm |

Dari penjelasan (Tabel 2.4) diatas bahwa struktur yang merupakan cara kerja dari sensor kelembaban udara/Humidity DHT11 memiliki empat buah kaki yaitu: pada bagian kaki(VCC), dihubungkan ke bagian Vss yang bernilai sebesar 5V,pada board minimum sistem dan untuk bagian kaki GND dihubungkan ke ground (GND) pada minimum sistem, sedangkan pada bagian kaki data yang merupakan keluaran (Output) dari hasil pengolahan data analog dari sensor DHT11 yang dihubungkan ke bagian analog input mikrokontroler dan yang tak ketinggalan terdapat satu kaki tambahan yaitu kaki NC (Not Connected), yang tidak dihubungkan ke pin manapun.

1. **Motor Servo**

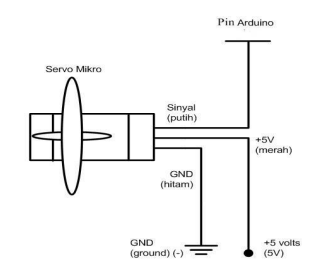
Motor servo adalah sebuah motor dengan sistem umpan balik tertutup di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor servo merupakan salah satu jenis motor DC. Berbeda dengan motor stepper, motor servo beroperasi secara close loop. Poros motor dihubungkan dengan rangkaian kendali, sehingga jika putaran poros belum sampai pada posisi yang diperintahkan maka rangkaian kendali akan terus mengoreksi posisi hingga mencapai posisi yang diperintahkan. Motor servo banyak digunakan pada peranti R/C (Remote Control) seperti mobil, pesawat, helikopter, dan kapal, penggerak pada kamera serta sebagai aktuator robot. (Fahmi, 2011). Gambar 2.6 merupakan motor servo standar.



Gambar 2.6 Motor Servo Standar Hitec HS-311

1. **Konstruksi Motor Servo**

Motor servo merupakan motor yang berputar lambat, dimana biasanya ditunjukkan oleh rate putarannya yang lambat, namun demikian memiliki torsi yang kuat karena pada internal gear-nya. Motor servo memiliki 3 kabel yaitu putih sebagai I/O pin, merah sebagai Vcc dan hitam sebagai ground. Dengan demikian, motor servo dapat dikontrol melalui kabel I/O yang berwarna putih. Pada gambar 2.40 dibawah ini merupakan pin-pin dan pengkabelan dari motor servo yang dihubungkan pada rangkaian pengontrol. (Suyadi, 2014)

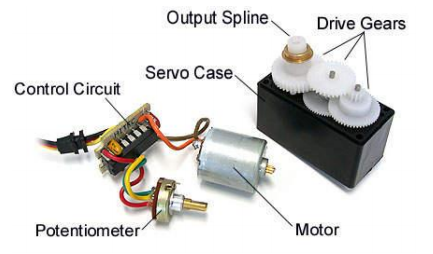


Gambar 2.7 Pin out kabel motor servo

Didalam sebuah motor servo terdapat beberapa karakteristik, yaitu :

* + 3 jalur : power, ground dan control.
  + Sinyal control mengendalikan posisi.
  + Operasional dari motor servo dikendalikan oleh pulsa sebesar 20ms, dimana lebar pulsa antara 500µs dan 2400µs menyatakan akhir dari range sudut maksimum.
  + Konstruksi didalamnya meliputi internal gear, potensiometer dan feedback control.

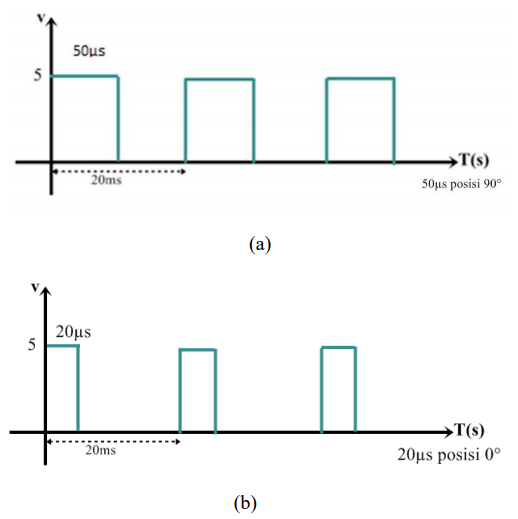
Didalam motor servo terdapat potensiometer yang digunakan sebagai sensor posisi. Potensiometer tersebut dihubungkan dengan output shaft untuk mengetahui sudut posisi dari output gear pada motor servo. Ketika motor DC (Direct Current) berputar, maka output shaft juga berputar dan sekaligus memutar potensiometer. Rangkaian control kemudian dapat membaca kondisi potensiometer tersebut untuk mengetahui posisi actual shaft. Jika posisinya sesuai dengan yang diinginkan, maka motor dc akan berhenti. Sudut operasi motor servo (operating angle) bervariasi tergantung jenis motor servo. Pada gambar 2.8 Merupakan internal gear dan kontrol elektronik untuk mengatur pergerakan dari motor.

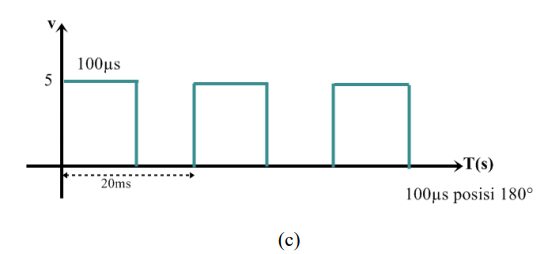


Gambar 2.8 Konstruksi motor servo

1. **Jenis – Jenis Motor Servo**
2. **Motor Servo Standar**

Motor servo jenis ini hanya mampu bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°. Pengaturan motor servo diperlukan untuk mengetahui gerakan dari motor servo dan pula yang harus kita berikan untuk bergerak ke kanan atau bergerak ke kiri. Teknik PWM ( Pulse Width Modulation ) digunakan untuk mengatur sudut motor servo jenis standard 180˚ ini dapat di lihat pada gambar 2.42. Pada motor servo jenis standard 180˚ memiliki 3 sudut yaitu pada saat kondisi sudut 0˚, 90˚ dan 180˚.

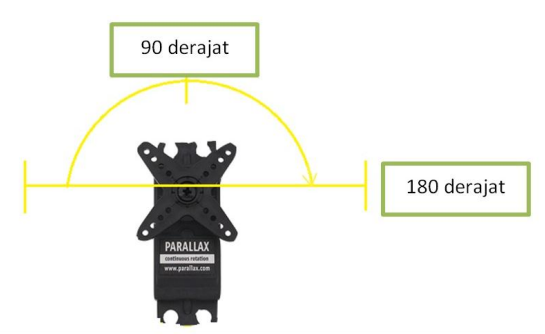




Gambar 2.9 Teknik PWM untuk mengatur sudut motor servo 180˚ (a) pulsa 50µs posisi 90˚ (b) pulsa 20µs posisi servo 0˚ (c) pulsa 100 µs posisi 180˚

Sudut dari motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Sebagai contoh, dengan pulsa 50µs pada periode delay selebar 20ms maka sudut dari motor servo akan berada pada posisi netral atau 90˚ sedangkan pada saat pulsa ≤ 20µs pada periode delay selebar 20ms maka sudut dari motor servo akan berada pada posisi 0˚ dan untuk pulsa 100µs pada periode delay sebesar 20ms maka sudut dari motor servo akan berada pada posisi 180˚.

Pada motor servo standard 180˚ semakin lebar pulsa off maka akan semakin besar gerakan sumbu kearah jarum jam dan semakin kecil pulsa off maka akan semakin besar gerakan sumbu berlawanan dengan arah jarum jam sehingga semakin besar pulsa yang masuk melalui kaki pin motor sevo maka semakin besar sudut yang dihasilkan. Arah putar motor servo dapat dilihat seperti yang ditunjukkan oleh Gambar 2.10 Dibawah ini.



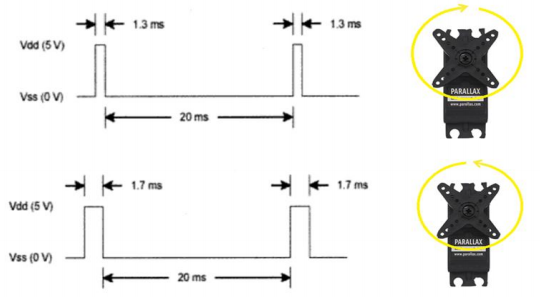
Gambar 2.10 Arah putaran motor servo standar

1. **Motor Servo Kontinu**

Motor servo kontinu merupakan motor servo yang bagian feedback-nya dilepas sehingga motor servo jenis ini mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

Prinsip kerja dari motor servo continuous sedikit berbeda dari motor servo standar. Untuk menggerakan motor servo ke kanan atau ke kiri, tergantung dari nilai delay yang diberikan. Untuk membuat servo pada posisi center, berikan pulsa 1.5ms dan untuk pemberian pulsa ≤1.3ms motor servo akan berputar searah jarum jam dengan besar putaran sumbu ditentukan oleh besar pulsa on pada motor sedangkan untuk membuat motor servo continuous berputar berlawanan dengan arah jarum jam dapat memberikan pulsa ≥ 1.7ms, dan dengan besar pulsa on yang digunakan, dapat menentukan besar putaran untuk berlawanan dengan arah jarum jam.

Teknik PWM ( Pulse Width Modulation ) pada motor servo continuous dapat dilihat pada gambar 2.11. Pada motor servo continuous mampu bergerak dua arah yaitu searah jarum jam dan berlawanan dengan arah jarum jam tanpa adanya batasan defleksi sudut putar (dapat berputar secara kontinyu sehingga motor ini berputar 360˚).



Gambar 2.11 Arah putaran motor servo continuous 360˚

1. **Karakteristik Motor Servo**

Motor Servo yang digunakan pada robot ini adalah motor servo jenis Tower Pro Micro Servo SG90. Motor servo jenis ini akan bekerja secara baik jika pada bagian pin kontrolnya diberikan sinyal PWM dengan frekuensi 50Hz dengan periode sebesar 20 ms. Pemberian besar pulsa dari mikrokontroler menentukan besar sudut yang harus dilakukan oleh motor servo. Pengaturan sudut motor servo diperlukan untuk mengetahui gerakan dari motor servo dan pulsa yang harus diberikan ke motor servo dalam pergerakan ke kanan atau ke kiri. Dari pulsa yang diberikan, kita dapat melihat gerakan motor servo. Di mana pada saat sinyal dengan frekuensi 50Hz tersebut dicapai pada kondisi Ton duty cycle 1.5 ms, maka rotor dari motor akan berhenti tepat di tengah-tengah (sudut 90° / netral).Pada saat Ton duty cycle dari sinyal yang diberikan kurang dari 1.5ms, maka rotor akan berputar ke arah kiri dengan membentuk sudut yang besarnya linier terhadap besarnya Ton duty cycle sampai batas minimum 0°, dan akan bertahan diposisi tersebut. Dan sebaliknya, jika Ton duty cycle dari sinyal yang diberikan lebih dari 1.5ms, maka rotor akan berputar ke arah kanan dengan membentuk sudut yang linier pula terhadap besarnya Ton duty cycle sampai batas sudut maksimum 180°, dan bertahan diposisi tersebut. Untuk lebih jelasnya karakteristik motor servo dapat dijelaskan oleh tabel 2.5 dibawah ini.

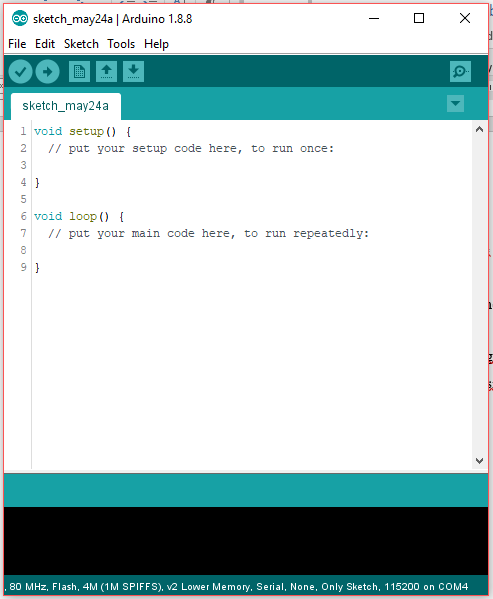
Tabel 2.5 Karakteristik motor servo tipe Tower Pro Micro Servo SG90

|  |  |
| --- | --- |
| Motor Servo | Micro Servo Sg90 |
| Dimensi | 22.6 X 21.8 X 11.4 mm |
| Berat (Hanya Motor) | 9 gram |
| Kecepatan | 0.12 S/ 60 Degree |
| Pulse Width | 500 – 2400 µs |
| PWM Period | 20 ms (50Hz) |
| Tegangan Kerja | 4,8 V – 6 V |
| Arus | Kurang Dari 500 mA |
| Temperatur Range | -30 Sampai 60˚ C |
| Panjang Kabel | 150 mm |
| Stall Torque | 1.98 Kg/Cm |
| Gear Type | Plastic |
| Limit angle | 180˚ (±10˚) |
| Neutral position | 1500 µs |

1. **IDE Arduino**

IDE Arduino adalah software yang sangat canggih ditulis dengan menggunakan java. IDE Arduino terdiri dari :

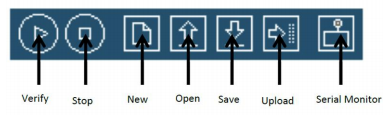
* Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa Processing.
* Compiler, sebuah modul yang mengubah kode program (bahasa processing menjadi kode biner. Bagaimanapun sebuah mikrokontroller tidak akan bisa memahami bahasa prosessing. Yang bisa dipahami oleh mikrokontroller adalah kode biner. Itulah sebabnya compiler diperlukan dalam hal ini.
* Uploader, sebuah modul yang memuat kode biner dari computer ke dalam memory dalam papan Arduino.



Gambar 2.12 Tampilan Arduino IDE

Pada gambar 2.12 merupakan tampilan software Arduino IDE, sedangkan pada gambar 2.13 dapat melihat toolbar IDE yang memberian akses instan ke fungsi-fungsi yang penting :

* Dengan tombol Verify, dapat mengkompilasi program yang saat ini di editor.
* Tombol New menciptakan program baru dengan mengosongkan isi dari jendela editor saat ini. Sebelum hal itu terjadi, IDE memberikan kesempatan untuk menyimpan semua perubahan belum disimpan.
* Dengan Open dapat membuka program yang ada dari sistem file.
* Tombol Save menyimpan program saat ini. Ketika mengklik tombol Upload, IDE mengkompilasi saat ini program dan upload ke papan Arduino yang telah pilih di IDE menu Tools > Serial port.
* Arduino dapat berkomunikasi dengan komputer melalui koneksi serial. Mengklik tombol serial monitor membuka jendela serial monitor yang memungkinkan dapat melihat yang dikirimkan oleh Arduino dan juga untuk mengirim data kembali.
* Tombol stop menghentikan serial monitor



Gambar 2.13 Toolbar Arduino IDE

Meskipun menggunakan IDE sangat mudah, mungkin dapat mengalami masalah. Dalam kasus tersebut, dapat lihat menu Help. Menu Help menunjukkan banyak sumber daya yang berguna di website Arduino yang mneyediakan solusi cepat tidak hanya untuk semua masalah khas tetapi juga untuk referensi materi dan tutorial.

1. **Bahasa C Arduino**

Arduino menggunakan pemrograman dengan bahasa C. Berikut ini adalah sedikit penjelasan singkat mengenai karakter bahasa C arduino.

1. **Struktur**

Setiap program arduino (biasa disebut *sketch*) mempunyai dua buah fungsi yang harus ada :

1. *Void setup(){}*

Semua kode didalam kurung kurawal akan dijalankan hanya satu kali ketika program arduino dijalankan untuk pertama kalinya.

1. *Void loop (){}*

Fungsi ini akan dijalankan setelah *setup* (fungsi *void setup*) selesai. Setelah dijalankan satu kali fungsi ini akan dijalankan lagi, dan lagi secara terus menerus sampai catu daya (*power*) dilepaskan.

1. **Syntax**

Berikut ini adalah elemen bahasa C yang dibutuhkan untuk format penulisan.

1. // (Komentar satu baris)

Kadang diperlukan untuk memberikan catatan pada diri sendiri apa arti dari kode-kode yang dituliskan. Cukup menuliskan dua garis miring dan apapun yang diketikkan dibelakangnya akan berwarna hijau dan diabaikan oleh program.

1. /\* \*/ (komentar banyak baris)

Jika punya banyak catatan, maka hal itu dapat dituliskan pada beberapa baris sebagai komentar. Semua hal yang terlatak diantara dua symbol tersebut akan diabaikan oleh program.

1. {} (kurung kurawal)

Digunakan untuk mendifinisikan kapan blok program mulai berakhir (digunakan juga pada fungsi dan pengulangan).

1. ;(titik koma)

Setiap baris kode harus diakhiri dengan titik koma (jika titik koma yang hilang maka program tidak bisa dijalankan).

1. **Variabel**

Sebuah program secara garis besar dapat didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas. *Variable* inilah yang digunakan untuk memindahkannya.

1. *int (integer)*

Digunakan untuk menyimpan angka dalam 2 *byte* (16 bit). Tidak mempunyai angka desimal dan menyimpan nilai dari -32,768 dan 32,767.

1. *long (long)*

Digunakan ketika *integer* tidak mencukupi lagi. Memakai 4 *byte* (32 byte) dari memori (RAM) dan mempunyai rentang dari -2,147,483,683 dan 2,147, 483,647.

1. *Boolean (Boolean)*

*Variable* sederhana yang digunakan untuk menyimpan nilai *TRUE* atau *FALSE*. Sangat berguna karena hanya menggunakan 1 bit dari RAM.

1. *Float (float)*

Digunakan untuk angka *decimal* (*floating point*). Memakai 4 *byte* (32 bit) dari RAM dan mempunyai rentang dari -3.4028235E+38 dan 3.4028235E+38.

1. *Char (character)*

Menyimpan 1 karakter menggunakan kode ASCII (misalnya „A‟= 65). Hanya memakai 1 *byte* (8 bit) dari RAM.

1. **Operator Matematika**

Operator yang digunakan untuk memanipulasi angka (bekerja seperti matematika yang sederhana).

1. =

membuat sesuatu menjadi sama dengan nilai yang lain (misalnya: x=10\*2, x sekarang sama dengan 20).

1. %

menghasilkan sisa dari hasil pembagian suatu angka dengan angka yang lain (misalnya : 12% 10, ini akan menghasilkan angka 2)

1. + Penjumlahan
2. – Pengurangan
3. \* Perkalian
4. / Pembagian
5. **Operator Pembanding**

Digunakan untuk membandingkan nilai logika.

1. ==

Sama dengan (misalnya: 12==10 adalah *FALSE* atau 12 ==12 adalah *TRUE*)

1. !=

Tidak sama dengan (misalnya 12!=10 adalah *TRUE* atau 12!=12 adalah *FALSE*).

1. <

Lebih kecil dari (misalnya:12 <10 adalah *FALSE* atau 12<12 adalah *FALSE* atau 12<14 adalah *TRUE*).

1. >

Lebih besar dari (misalnya:12 >10 adalah *TRUE* atau 12<12 adalah *FALSE* atau 12<14 adalah *FALSE*)

1. **Struktur Pengaturan**

Program sangat tergantung pada pengaturan apa yang akan dijalankan berikutnya, berikut ini adalah elemen dasar pengaturan.

1. *if..else*, dengan format seperti berikut ini:

*If (kondisi) { }*

*Else if (kondisi) { }*

*Else { }*Dengan struktur seperti diatas program akan menjalankan kode yang ada didalam kurung kurawal jika kondisi *TRUE*, dan jika *FALSE* maka akan diperiksa apakah kondisi pada *else if* dan jika kondisinya *FALSE* maka kode pada *else* yang akan dijalankan.

1. *for*, dengan format seperti berikut ini:

*For (int I =0; I < #pengulangan;i++) { }*

Digunakan bila ingin ,melakukan pengulangan kode didlam kurung kurawal beberapa kali, ganti #pengulangan dengan jumlah pengulangan yang diinginkan. Melakukan penghitungan ke atas dengan i++ atau kebawah i--.

1. **Digital**
2. *pinMode (pin,mode)*

Digunakan unutk menetapkan mode dari suatu pin, pin adalah nomor pin yang akan digunakan dari 0-9 (pin analog 0-5 adalah 14-19). Mode yang bisa digunakan adalah Input atau output.

1. *DigitalWrite (pin,value)*

Ketika sebuah pin ditetapkan sebagai output, pin tersebut dapat dijadikan *high* (ditarik menjadi 5 volt) atau *low* (diturunkan menjadi *ground*).

1. **Analog**
2. *AnalogWrite (pin, value)*

Beberapa pin pada arduino mendukung PWM yaitu 3,5,6,9,10,11. Ini dapat merubah pin hidup (*on*) atau mati (*off*) dengan sangat cepat sehingga membuatnya dapat berfungsi layaknya keluaran analog. Value (nilai) pada format kode tersebut adalah antara 0 (0% *duty cycle* ~ 0V) dan 255 (1000% *duty cycle* ~ 5V)

1. *AnalogRead(pin)*

Ketika pin ditetapkan sebagai input, pengguna dapat membaca tegangan keluarannya. Keluarannya berupa angka antara 0 (untuk 0 volt) dan 1024 (untuk 5 volt)

1. **Android**

Android adalah sebuah perangkat lunak untuk perangkat mobile berbasis linux yang mencakup sistem operasi, middleware dan aplikasi kunci. Android Standart Development Kit (SDK) merupakan tool Application Programming Interface (API) yang diperlukan untuk mengembangkan aplikasi pada platform android menggunakan bahasa pemrograman java1. (Nastiti dan Sunyoto, 2012).

Belum ada standar pabrik yang menetukan definisi telepon pintar. Bagi beberapa orang, telepon pintar merupakan merupakan telepon yang bekerja menggunakan seluruh piranti lunak sistem operasi yang menyediakan hubungan standar dan mendasar bagi pengembang aplikasi. Bagi yang lainnya, telepon pintar hanyalah merupakan sebuah telepon yang menyajikan fitur canggih seperti surel (surat elektronik), internet dan kemampuan membaca buku elektronik (ebook) atau terdapat papan ketik (baik built-in maupun eksternal) dan konektor VGA. Dengan kata lain, telepon pintar merupakan komputer mini yang mempunyai kapabilitas sebuah telepon. Pertumbuhan permintaan akan alat canggih yang mudah dibawa kemana-mana membuat kemajuan besar dalam

prosesor, memori, layar dan sistem operasi yang diluar dari jalur telepon gengam sejak beberapa tahun ini.

1. **Android Studio**

Android Studio adalah lingkungan pengembangan Android baru berdasarkan IntelliJ IDEA. Ini menyediakan fitur baru dan perbaikan atas Eclipse ADT dan akan menjadi IDE Android resmi. Android studio tersedia gratis bagi para developer dan dirancang khusus untuk pengembangan aplikasi android. Bahasa yang digunakan pada Android Studio adalah bahasa Java.

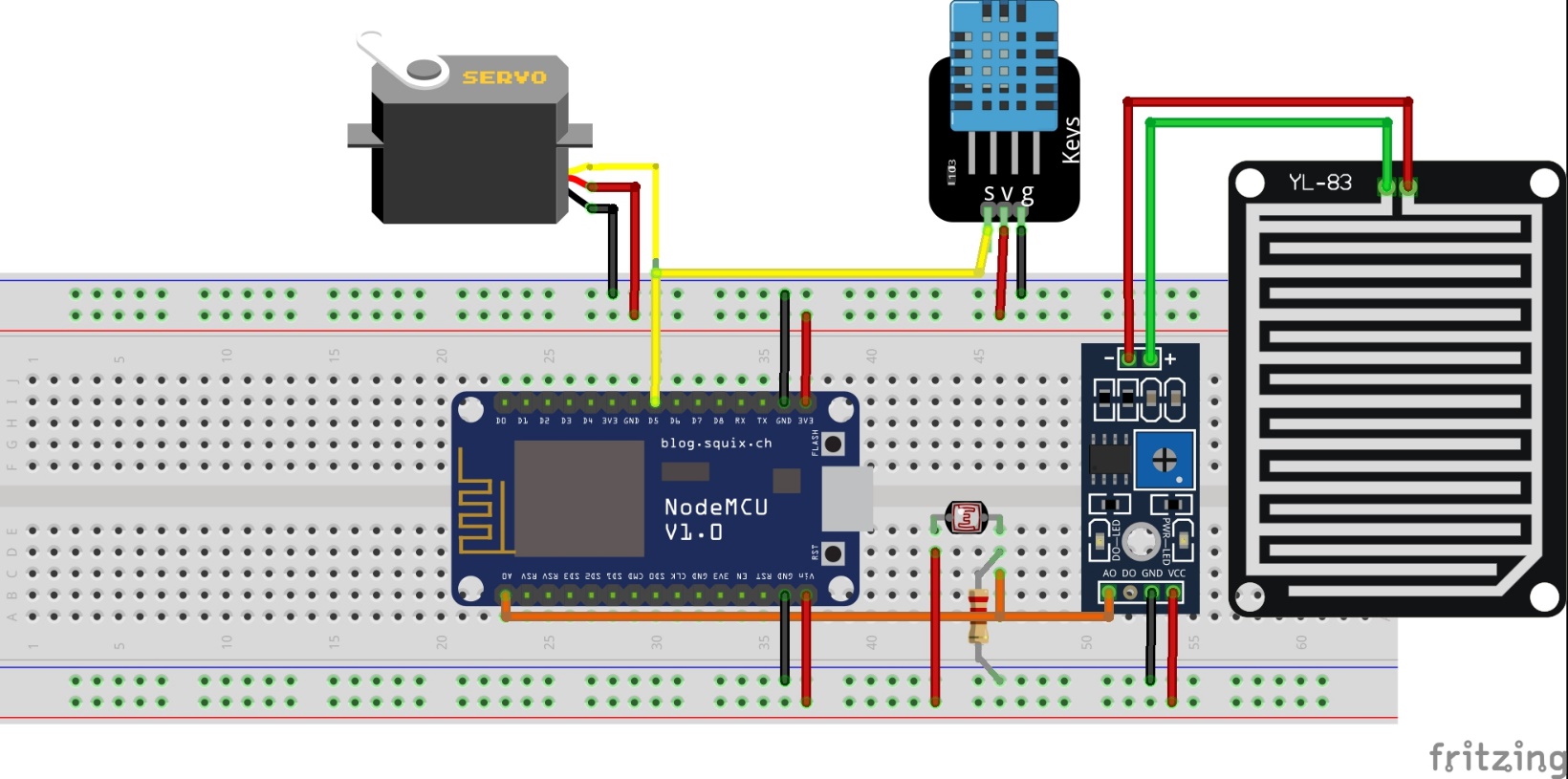
1. **Android SDK**

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk memulai pengembangan aplikasi pada platform Android menggunakan bahasa pemrograman *Java* (Safaat:2011). Android merupakan subset perangkat lunak untuk ponsel yang meliputi, sistem operasi, middleware dan aplikasi yang dirilis oleh google. Saat ini disediakan android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi platform android menggunakan bahasa pemrograman *java*. Sebagai platform aplikasi, android memberi anda kesempatan untuk membuat aplikasi yang kita butuhkan yang buka merupakan aplikasi bawaan *handphone*.

**BAB III**

**KEGIATAN PELAKSAAN**

1. **Alat dan Bahan**

****

1. **Desain Alat Jemuran Antisipasi Hujan**



1. **Flowchart Sistem**