

BAB II

DASAR TEORI

2.1 TINJAUAN PUSTAKA

Sebagai acuan penelitian, peneliti melakukan review beberapa paper salah satunya penelitian dari fenty ariani dan kawan-kawan yang berjudul “sistem monitoring suhu dan pencahayaan berbasis internet of thing (IoT) untuk penetasan telur ayam”. Pada penelitian tersebut penulis melakukan rancang bangun sistem untuk monitoring tetas telur ayam menggunakan *internet of things*, dengan metode rancangan yang digunakan yaitu mendeteksi suhu dan kelembaban yang kemudian data tersebut akan digunakan untuk acuan mengendalikan lampu dan kipas. Penulis juga melakukan metode pengumpulan data menggunakan teknik wawancara, observasi, dan studi literatur. Pengujian yang dilakukan penulis membandingkan metode tetas telur melalui induk ayam dengan tetas telur menggunakan alat yang dirancang. Hasil yang diperoleh berdasarkan lama waktu yang dibutuhkan untuk tetas telur melalui induk memakan waktu sebanyak 21 sampai 30 hari sedangkan apabila dibandingkan menggunakan alat rancang memakan waktu 21 hari. Berdasarkan tingkat keberhasilan dengan mengambil sampel 20 butir telur, hasil yang diperoleh yaitu kurang dari 14 butir yang bisa menetas sedangkan dibandingkan dengan menggunakan alat yang dirancang mampu mencapai 16 butir dapat menetas. Dari hasil pengujian yang diperoleh, penulis mengambil kesimpulan bahwa sistem monitoring tetas telur ayam berbasis *internet of things* dapat meningkatkan kuantitas penetasan telur dan juga dapat mempercepat masa panen telur [4].

Pada penelitian yang dilakukan suharto dan kawan-kawan yang berjudul “teknologi penetas telur itik otomatis menggunakan *mikrokontroler* arduino uno”. Penulis melakukan metode perancangan, pembuatan dan eksperimen mesin penetas telur berbasis *mikrokontroler* arduino uno yang dapat ditenagai dengan catu daya 220V yang memakan daya listrik 40 *Watt*. Eksperimen penetasan dilakukan terhadap 30 butir dengan set poin suhu 37-38°C dengan kelembaban 55-60%. Pengujian dilakukan dengan cara membandingkan metode manual menggunakan media induk itik dan metode penetasan menggunakan mesin

penetas telur dengan parameter yang dijadikan pembandingan adalah hasil jumlah penetasan normal, cacat dan mati yang direpresentasikan dengan nilai presentase keberhasilan. Hasil yang diperoleh menggunakan metode manual telur yang berhasil menetas dengan normal berjumlah 19 butir dengan hasil penetasan cacat sebanyak 6 butir telur dan hasil tetas yang mati mencapai 5 butir telur sehingga diperoleh persentase keberhasilan telur yang menetas 63,33%. Sedangkan penetasan menggunakan metode otomatis mesin penetas telur diperoleh jumlah telur yang menetas normal mencapai 27 butir telur dengan jumlah telur yang metetas cacat sebanyak 1 butir dan mati 2 butir yang mana persentase keberhasilan telur yang menetas diperoleh 90,0%. Dari penelitian tersebut dapat disimpulkan bahwa perancangan mesin penetas telur berbasis *mikrokontroler* arduino uno mampu meningkatkan tingkat keberhasilan tetas telur hingga mencapai 90% [5].

Penelitian oleh Ika Wahyu Kinnasih, Dzulki FLIH dengan judul “rancang bangun alat pengontrol suhu dan kelembapan pada tempat penetasan telur menggunakan sensor DHT22 dan motor swing berbasis IoT”. Pada penelitian tersebut penulis merancang sebuah alat yang dapat mengatur suhu dan kelembapan dengan sensor yang digunakan adalah DHT22 dan menggunakan aktuator motor dc untuk skema pembalik telur yang ditenagai sumber listrik 220v. Sistem dirancang untuk dapat dikendalikan menggunakan aplikasi *smartphone* namun juga dapat dipantau secara langsung melalui layar lcd. Pengujian dilakukan menggunakan metode pengumpulan data suhu dan kelembapan dan perhitungan *error* terhadap set poin, menggunakan 6 butir telur yang diletakan di dalam alat selama 21 hari. Hasil yang diperoleh adalah pada parameter suhu yang diukur selama 21 hari mengalami penurunan yang menyebabkan nilai *error* yang diperoleh semakin meningkat, sedangkan parameter kelembapan terus mengalami kenaikan sehingga nilai *error* yang diperoleh juga semakin meningkat. Untuk hasil tetas telur dari sebanyak 6 butir dalam 21 hari hanya 1 butir telur yang dapat menetas yang mana berarti tingkat keberhasilan telur menetas hanya sebesar 16,7%. Dari hasil data yang diperoleh penulis mengambil kesimpulan bahwa alat tetas telur yang dirancang memiliki kelebihan bahwa mesin menggunakan sensor dan *motor swing* untuk mengoptimalkan kinerja mesin dan material bahan yang

digunakan ekonomis serta desain yang sederhana. Namun penulis juga menganalisa beberapa kekurangan yaitu mesin yang dirancang masih berukuran kecil sehingga tidak dapat menampung kapasitas telur yang banyak, mesin tetas yang dirancang juga masih menghasilkan persentase keberhasilan yang kecil yaitu hanya mencapai 16,7 % dan parameter kelembaban hanya muncul pada aplikasi yang ada di *smartphone* [6].

Menurut penelitian yang dilakukan oleh khairul muttaqin, ahmad ihsan dan heri irawan yang berjudul “peningkatan produktivitas ternak ayam melalui teknologi inkubator mesin penetas telur berbasis *internet of thing*” yang membahas mengenai sistem inkubator mesin tetas telur yang berbasis *internet of things*. Penelitian tersebut mengembangkan teknologi mesin tetas telur menggunakan basis *mikrokontroler nodemcu* dengan parameter yang dimonitor adalah suhu dan kelembaban yang mana menggunakan sensor DHT11, motor dc linear digunakan sebagai aktuator untuk mengganti posisi telur supaya suhu dapat merata keseluruh bagian telur. Kemudian pengaturan suhu diatur oleh lampu yang mana suhu yang ditentukan adalah 37 – 38 *Celcius*. Metode pengujian dilakukan dengan melakukan survei ke peternak ayam untuk memperoleh data secara kuantitatif maupun kualitatif serta informasi terkait kondisi dan permasalahan peternak yang disurvei. Metode selanjutnya yaitu perancangan alat menggunakan *nodemcu* beserta sensor dan aktuator yang dibutuhkan dan juga mengintegrasikan alat dengan *internet of things*. Setelah proses perancangan penulis melakukan monitoring dan evaluasi terhadap alat yang dirancang. Evaluasi dilakukan dengan metode mengukur *persentase* keberhasilan dalam proses penetasan telur, yang mana hasil evaluasi akan dijadikan sebagai acuan untuk mengkategorikan apakah mesin tetas telur yang dirancang layak digunakan atau tidak. Hasil penelitian yang didapat oleh penulis pertama adalah berkaitan dengan wawancara lapangan, penulis mendapatkan informasi mengenai keadaan narasumber dan kendala yang dialami khususnya dalam penetasan telur ayam, seperti penetasan telur masih menggunakan induk ayam, sehingga perlu waktu lama untuk menetas telur dan resiko gagal telur masih sangat tinggi, kendala lain yang dihadapi narasumber adalah keterbatasan lahan dan kapasitas telur, sehingga ayam petelur di liang tidak terkendali dan jumlah telur yang menetas juga terbatas. Pada perancangan,

peneliti berhasil membuat alat tetas otomatis yang dapat dipantau melalui aplikasi berbasis internet. Kemudian alat ini diterapkan untuk menguji tingkat keberhasilan alat dalam menetas telur. Pengujian dilakukan dengan mengatur batas atas parameter suhu sebesar 38,8°C dan batas bawah sebesar 37,7°C sedangkan parameter kelembaban diatur dengan batas atas 65% dengan batas bawah 56%. Ujicoba dilakukan dilakukan selama 21 hari. Hasil yang diperoleh adalah telur berhasil menetas pada hari ke 19 [7].

Berdasarkan penelitian-penelitian yang peneliti review di atas bahwa mesin tetas menggunakan *mikrokontroler* dapat meningkatkan tingkat keberhasilan untuk menetas telur. Maka dari itu peneliti akan mengadaptasi alat dan bahan yang digunakan pada penelitian-penelitian diatas pada tahap perancangan alat sistem tetas telur entok yaitu menggunakan sensor DHT22 untuk mengukur suhu dan kelembaban sedangkan untuk mengendalikan suhu dan kelembaban pada alat menggunakan lampu. Dikarenakan menggunakan lampu maka peneliti akan menambahkan sensor LDR untuk mengukur intensitas cahaya lampu. Sistem monitoring menggunakan *internet of things* dengan protokol MQTT, dengan demikian penelitian akan menggunakan Esp8266 sebagai *mikrokontroler* yang dapat menghubungkan perangkat dengan internet.

2.2 DASAR TEORI

Pada subbab ini peneliti akan memaparkan beberapa teori-teori mendasar mengenai mesin inkubator telur dan hubungannya dengan *internet of things*, pengertian *Delay*, teori mengenai protokol MQTT, Esp8266 *development board* dan spesifikasinya, sensor DHT22 beserta cara kerja dan teori tentang sensor optik atau cahaya LDR disertai cara kerjanya.

2.2.1 Inkubator telur

Pengeraman telur dengan cara alami yaitu mengerami dengan induknya sendiri memiliki kelemahan yaitu tidak dapat menetas telur dalam jumlah banyak. Hal tersebut mengakibatkan usaha bisnis bidang peternakan yang menggunakan cara alami akan kurang efektif, sehingga penggunaan inkubator atau alat tetas telur buatan diperlukan agar bisnis dapat berjalan dengan baik.

Alat penetas telur buatan atau incubator hingga saat ini yang cukup dikenal yaitu ada dua jenis antara lain :

1. Tetas telur konvensional

Inkubator jenis ini menggunakan matahari sebagai sumber panas dan menyimpan panas pada media sekam. Mesin penetas telur ini bisa dikatakan sebagai mesin penetas telur yang sederhana. Dahulu pembenihan pada awalnya, peternak di Bali adalah pengguna pertama inkubator ini, dan kemudian penggunaannya mulai menyebar ke berbagai lokasi.



Gambar 2. 1 Alat tetas telur konvensional [8]

Pada Gambar 2.1 menunjukkan alat tetas telur bentuk konvensional. Walaupun sederhana, penggunaan inkubator ini memerlukan persiapan yang teliti. Ada dua metode yang umum digunakan untuk proses ini, yaitu metode Bali dan metode Bali Amuntai. Inkubator ini sering digunakan khususnya untuk menetas telur bebek.

2. Mesin tetas telur

Sederhananya, inkubator adalah lemari yang diatur untuk memaksimalkan panas. Suhu dan ukuran tahap pemanasan dapat diatur sesuai kebutuhan selama inkubasi.

Dalam aplikasinya, mesin tetas telur memiliki kapasitas berdasarkan ukuran box dan juga daya panas yang diterima oleh inkubator. Semakin besar kapasitasnya, semakin banyak daya pemanas yang dibutuhkan. Saat menyiapkan tempat penetasan perlu disesuaikan jumlah telur yang akan ditetaskan dan kapasitas inkubator, semakin kecil telur maka semakin besar jumlah telur yang dapat ditetaskan.

Mesin inkubator dibagi menjadi tiga jenis menurut operasi dan proses pembalikan. Yang pertama adalah inkubator manual, yang kedua adalah inkubator semi otomatis, dan yang ketiga adalah inkubator otomatis.

1. Mesin inkubator manual

Proses penetasan menggunakan alat konvensional ini memerlukan tahap pembalikan posisi telur yang mana pada alat tetas telur konvensional dilakukan secara manual satu-per-satu.

2. Mesin tetas telur semi otomatis

Mesin inkubator semi otomatis ini merupakan perkembangan dari mesin incubator manual. Berikut gambar menunjukkan alat tetas telur semi otomatis.



Gambar 2. 2 Alat tetas telur semi otomatis [8]

Pada Gambar 2.2 perbedaan dari alat tetas telur semi otomotasi dan konvensional, yaitu proses pembalikan telur pada alat semi otomatis dilakukan menggunakan tuas pemutar sehingga tidak perlu melakukan proses pembalikan satu-per-satu.

3. Mesin inkubator full otomatis

Secara umum mesin inkubator otomatis sudah dilengkapi timer dan dengan desain yang memungkinkan telur dapat berputar secara otomatis dengan waktu pemutaran yang sudah ditentukan.



Gambar 2. 3 Alat tetas telur full otomatis [8]

Pada Gambar 2.3 mesin ini juga sudah dilengkapi dengan sensor suhu untuk mengatur suhu dalam mesin inkubator. Dengan mesin tetas telur otomatis ini memungkinkan peternak atau pelaku bisnis dapat meningkatkan produktivitas.

Berbagai jenis inkubator tentunya memiliki beberapa komponen. Komponen inkubator telur ini tergolong sama baik inkubator manual, inkubator semi otomatis maupun inkubator full otomatis. Beberapa komponen tersebut antara lain:

a. Pengontrol suhu (*thermostat*)

Termostat adalah item penting yang berguna saat menyesuaikan suhu secara otomatis. Sistem operasi komponen ini mematikan daya saat suhu di inkubator memenuhi persyaratan telur. Begitu juga saat suhu turun, arus mengalir, memungkinkan inkubator beroperasi normal kembali.

b. Sumber panas

Dahulu api dari kompor minyak tanah atau kayu bakar digunakan sebagai sumber panas, namun sekarang telah berubah menjadi alat lain seperti bola lampu dan elemen *wire-coil* yang berasal dari energi listrik atau gas.

c. *Termometer* dan *higrometer*

Pemantau suhu dan kelembaban inkubator, memungkinkan operator inkubator untuk memeriksa apakah sesuai atau tidak.

d. Pengatur *humidity*

Untuk mengatur kelembapan di dalam inkubator, biasanya membutuhkan air yang ditempatkan di wadah atau wadah plastik. Air digunakan untuk mengatur kelembapan di dalam inkubator agar tetap ideal. Air tidak perlu ditambahkan atau dikurangi sampai penetasan selesai, air dikeluarkan dan diganti dengan air baru setelah selesai proses tetas telurnya.

internet. Setelah objek memiliki alamat ip dan terhubung ke internet, sensor-sensor yang terdapat pada objek tersebut berperan penting. Sensor-sensor ini berfungsi untuk mengumpulkan *input* informasi yang kemudian diolah dan terjadi pertukaran informasi di antara objek tersebut. Dengan demikian, objek tersebut dapat bekerja secara otomatis atau berdasarkan perintah yang diberikan.

Komponen pertama, semua hal yang terhubung ke internet harus disebut hal, yang sebagian besar menunjukkan bahwa semuanya pasti akan menjadi pintar dengan bantuan IoT, yang dapat membantu dalam banyak hal. Contohnya adalah *smartphone*, tablet, dan komputer yang merupakan bagian dari IoT karena memiliki banyak *input* dan *output* yang menghubungkannya ke internet.

Komponen lain dari IoT adalah kemampuan benda untuk berkomunikasi dengan orang dan benda lain secara umum melalui sensor, termasuk kamera, sistem GPS, *microchip*, dan motor – apa pun yang dapat bertindak sebagai perangkat dalam sistem IoT. Sensor mengumpulkan informasi tentang objek dan mengirimkannya secara khusus ke unit komputasi, tempat *algoritme* biasanya mengevaluasinya dan membuat keputusan.

Komponen terakhir dari IoT secara harfiah adalah bagaimana menggunakan barang sehari-hari seperti mobil atau benda sehari-hari, yang menunjukkan bahwa IoT memungkinkan setiap orang di dunia untuk berinteraksi satu sama lain melalui *email* atau *web*. [18].

2.2.3 *Wireshark*

Wireshark merupakan salah satu alat analisis paket jaringan. Fungsinya adalah untuk merekam dan menampilkan informasi yang terdapat dalam setiap paket jaringan dengan se jelas mungkin [19] .



Gambar 2. 5 *Wireshark* [11]

Berdasarkan Gambar 2.5 menunjukkan *wireshark*, pengguna dapat mengobservasi data dari jaringan yang sedang beroperasi atau dari data yang telah tersimpan di *disk*. *Wireshark* memungkinkan pengguna untuk langsung melihat dan menyortir data yang telah ditangkap, mulai dari informasi ringkas hingga rincian lengkap dari setiap paket, termasuk *header* lengkap dan isi data yang terkandung di dalamnya[20]. *Wireshark* juga merupakan sebuah alat yang *fleksibel*, yang dapat digunakan untuk memeriksa data yang terjadi pada jaringan, baik itu melalui koneksi kabel maupun *nirkabel (wireless)* [21].

2.2.4 MQTT Protocol

MQTT adalah sebuah protokol komunikasi standar yang digunakan untuk menghubungkan dan mengirimkan data antar mesin. Protokol ini sering digunakan dalam komunikasi antara sensor pintar, perangkat yang dapat dikenakan, dan perangkat *internet of things* lainnya. Penggunaan MQTT menjadi penting karena perangkat *IoT* tersebut perlu mengirim dan menerima data melalui jaringan yang memiliki sumber daya dan *bandwidth* terbatas. MQTT memungkinkan perangkat *IoT* untuk mentransfer data secara *efisien* dan mudah diimplementasikan. Selain itu, MQTT juga mendukung komunikasi antara perangkat dan *cloud*, serta dari *cloud* ke perangkat.[14]

Protokol MQTT telah menjadi standar dalam komunikasi *internet of things* karena memberikan berbagai keuntungan berikut ini:

1. Ringan dan efisien

Penerapan MQTT pada perangkat *IoT* membutuhkan sumber daya yang minimal, bahkan dapat digunakan pada *mikrokontroler* kecil. Sebagai contoh, pesan kontrol MQTT dapat memiliki ukuran data setidaknya dua *byte*. Selain itu, *header* pesan MQTT juga memiliki ukuran yang kecil, yang memungkinkan pengoptimalan penggunaan *bandwidth* jaringan.

2. Scalable

Implementasi MQTT membutuhkan kode yang minimal dan memiliki konsumsi daya yang sangat rendah saat dijalankan. Protokol ini juga telah didesain dengan kemampuan bawaan untuk mendukung komunikasi dengan

banyak perangkat IoT. Oleh karena itu, dapat menggunakan protokol MQTT untuk terhubung ke jutaan perangkat IoT dengan mudah.

3. Andal

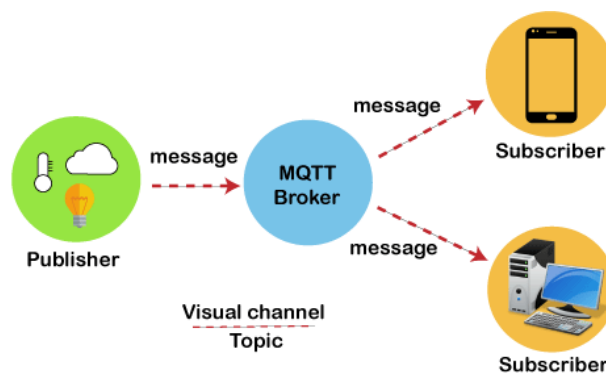
Banyak perangkat IoT terhubung melalui jaringan seluler dengan latensi rendah, *bandwidth* terbatas, dan keandalan yang kurang stabil. MQTT menyediakan fitur bawaan yang dapat mengurangi waktu yang dibutuhkan oleh perangkat IoT untuk melakukan koneksi ulang ke *cloud*. Selain itu, MQTT juga mendefinisikan tiga tingkat *quality of service* yang berbeda untuk memastikan keandalan dalam penggunaan IoT, yaitu paling banyak sekali (qos 0), setidaknya sekali (qos 1), dan tepat (qos 2).

4. Aman

Dengan MQTT, pengembang dapat dengan mudah mengenkripsi pesan dan mengautentikasi perangkat dan pengguna menggunakan protokol *otentikasi modern* seperti *oauth*, *tls1.3*, sertifikat yang dikelola pengguna, dan banyak lainnya.

5. Didukung dengan baik

Ada beberapa bahasa pemrograman, seperti *python*, yang menyediakan dukungan yang luas untuk mengimplementasikan *protocol* MQTT. Dengan demikian, pengembang dapat dengan mudah mengadopsi *python* untuk berbagai jenis aplikasi dengan usaha pemrograman minimal. [15].



Gambar 2. 6 MQTT architecture [16]

Pada Gambar 2.6 untuk memahaminya lebih jelas dapat dicontohkan yaitu suatu perangkat memiliki sensor suhu dan ingin mengirimkan data ke *server* atau *broker*. Di sisi lain, saat ponsel atau aplikasi *desktop* ingin membaca suhu

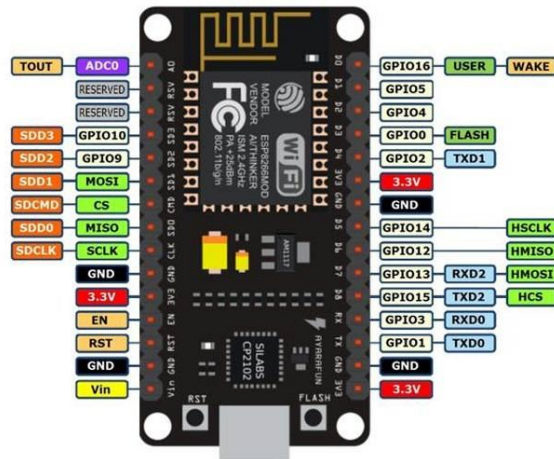
tersebut, dua hal terjadi. *Publisher* pertama-tama menentukan subjeknya misalnya, suhu kemudian menerbitkan pesan, yaitu nilai suhu. Di sisi lain, setelah mempublikasikan pesan, aplikasi seluler atau *desktop* berlangganan (*subscribe*) ke topik, yaitu suhu, dan kemudian menerima pesan yang dipublikasikan, yaitu nilai suhu. Tugas *server* atau *broker* adalah menyampaikan berita yang dipublikasikan ke aplikasi seluler atau *desktop* [16].

2.2.5 *Esp8266 Development board*

Arduino adalah platform di dunia *mikrokontroler* dan sistem tertanam. Dengan berbagai sensor dan modul yang murah, maka pengembang dapat membuat berbagai proyek, baik hobi maupun komersial. Seiring kemajuan teknologi, ide dan implementasi desain baru ikut berperan dan salah satu konsepnya adalah *internet of things* atau IoT. Ini adalah *platform* yang terhubung di mana beberapa "*things*" atau perangkat terhubung melalui internet untuk berbagi data.

Dalam komunitas *do-it-yourself*, proyek IoT terutama berfokus pada otomatisasi rumah dan aplikasi rumah pintar, tetapi proyek IoT komersial dan industri memiliki implementasi yang jauh lebih kompleks, seperti pembelajaran mesin, kecerdasan buatan, jaringan sensor nirkabel, dan sebagainya. Poin utama dalam pengantar singkat ini adalah apakah itu proyek kecil hobi diy atau proyek industri yang kompleks, setiap proyek IoT harus memiliki konektivitas internet dimana tercipta modul seperti Esp8266 dan Esp32 [13].[22]

Esp8266 adalah *mikrokontroler* yang sudah terintegrasi dengan *wi-fi* berdaya rendah yang dirancang oleh *espressif systems*. *Nodemcu* beroperasi pada kisaran suhu yang luas: -40°C hingga $+125^{\circ}\text{C}$. *Espressif* adalah perusahaan Cina yang berbasis di Shanghai. Volume produksi Esp8266 dimulai pada awal 2014 Esp8266 berupa ic yang berukuran sangat kecil dan hampir tidak mungkin bagi penghobi *embedded system* untuk memasang kabel agar dapat dicolokkan ke *breadboard*. Untuk mempermudah pengguna dapat membeli *development board* yang sudah jadi seperti *board nodemcu* [13] [23]



Gambar 2. 7 Esp8266 development board [22]

Berdasarkan Gambar 2.7 merupakan pin-pin yang terdapat pada *nodemcu* Esp8266 yang mana masing masing memiliki fungsi diantaranya:

1. Power pins ada empat pin daya. Pin vin dan tiga pin 3.3v.
 - a. Vin dapat digunakan untuk memberi pasokan daya langsung kepada *nodemcu/Esp8266* dan periferalnya. Arus yang mengalir melalui vin diatur melalui regulator yang terpasang di modul *nodemcu*. Selain itu, dapat memberikan pasokan daya teratur 5v ke pin vin.
 - b. Pin 3.3v merupakan hasil keluaran dari *regulator* tegangan yang ada pada modul tersebut dan dapat digunakan untuk menyediakan daya kepada komponen *eksternal*.
2. Gnd adalah pin *ground* dari *nodemcu/Esp8266*
3. Pin I2c digunakan untuk menghubungkan sensor dan periferall yang menggunakan protokol I2c. Pin ini mendukung baik mode I2c *master* maupun I2c *slave*. Fungsi dan fitur antarmuka I2c dapat diimplementasikan secara terprogram, dengan batasan frekuensi *clock* maksimum sebesar 100 khz. Penting untuk diingat bahwa frekuensi *clock* I2c harus lebih tinggi dari frekuensi *clock* terendah yang didukung oleh perangkat *slave* yang terhubung.
4. *Nodemcu/Esp8266* memiliki 17 pin GPIO yang dapat diatur untuk berbagai fungsi seperti I2c, I2s, UART, *pwm*, *remote control ir*, *lamp led*, dan tombol dengan menggunakan pemrograman. Setiap pin GPIO yang diaktifkan secara digital dapat dikonfigurasi dengan *pull-up* atau *pull-down internal*, atau disetel ke impedansi tinggi. Ketika diatur sebagai *input*, pin

GPIO juga dapat dikonfigurasi dengan pemacu tepi atau pemacu level untuk menghasilkan interupsi pada cpu.

5. *Nodemcu* memiliki saluran adc yang terintegrasi dengan presisi 10-bit menggunakan metode adc sar. Melalui fungsi adc ini, dapat mengukur tegangan pada pin *vdd3p3* dan juga mengukur tegangan *input* pada pin *tout*. Namun, perlu dicatat bahwa keduanya tidak dapat dilakukan secara bersamaan atau dalam waktu yang sama.
6. *Nodemcu/Esp8266* memiliki dua antarmuka UART, yaitu UART0 dan UART1, yang mendukung komunikasi asinkron seperti RS232 dan RS485. Kedua antarmuka ini dapat berkomunikasi dengan kecepatan hingga 4,5 *mbps*. UART0, yang terdiri dari pin *Txd0*, *Rxd0*, *Rst0*, dan *Cts0*, dapat digunakan untuk komunikasi secara penuh. Namun, UART1, yang hanya memiliki pin *Txd1*, biasanya digunakan hanya untuk menampilkan sinyal transmisi data seperti mencetak *log*.
7. *Nodemcu/Esp8266* memiliki pin spi yang menyediakan dua saluran spi (*spi* dan *hspi*) yang dapat berfungsi sebagai mode *slave* dan *master*. Selain itu, spi ini juga mendukung berbagai fitur spi tujuan umum, seperti berikut:
 - a. 4 mode pengaturan waktu transfer format spi
 - b. Hingga 80 mhz dan jam terbagi 80 mhz
 - c. Fifo hingga 64-byte
8. Pin *sdio* pada *nodemcu/Esp8266* memiliki fitur antarmuka *secure digital input/output* (*sdio*) yang digunakan untuk menghubungkan kartu sd secara langsung. *Nodemcu/Esp8266* mendukung *sdio v1.1* dengan mode *4-bit* dan kecepatan 25 *mhz*, serta *sdio v2.0* dengan mode *4-bit* dan kecepatan 50 *mhz*.
9. Papan ini dilengkapi dengan 4 pin *pwm* (modulasi lebar pulsa) yang dapat digunakan. *Output pwm* dapat dikonfigurasi secara terprogram dan berguna untuk menggerakkan motor *servo digital* dan mengendalikan intensitas cahaya pada led. Rentang frekuensi *pwm* dapat disesuaikan dari 1000 μ s hingga 10000 μ s (100 *hz* hingga 1 *khz*).

10. Pin kontrol digunakan untuk mengontrol *nodemcu*/Esp8266. Pin ini termasuk pin aktifkan chip (*en*), pin reset (*rst*) dan pin *wake*.
 - a. *En*: *chip* Esp8266 diaktifkan ketika pin *en* di-*trigger* high. Saat di-*trigger* rendah, chip bekerja dengan daya minimum.
 - b. *Rst*: pin *Rst* digunakan untuk mereset chip Esp8266.
 - c. *Wake*: *wake* pin digunakan untuk membangunkan chip dari deep-sleep.

Tabel 2. 1 spesifikasi *nodemcu* Esp8266

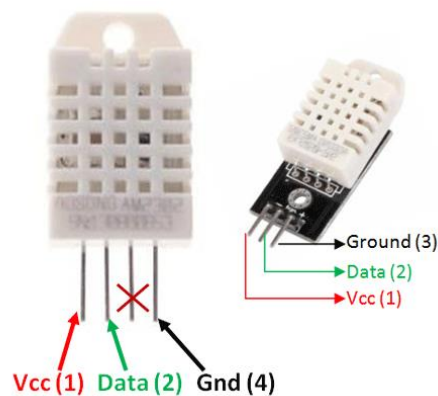
<i>Mikrokontroler</i>	Tensilica 32-bit risc cpu xtensalx106
Tegangan operasi	3,3 <i>volt</i>
Tegangan masukan	6-12 <i>volt</i>
Pin digital i/o (<i>dio</i>)	16
Pin analog <i>input</i> (<i>adc</i>)	1
<i>Uarts</i>	2
<i>Spis</i>	1
<i>I2cs</i>	1
<i>Flash memory</i>	4 mb
<i>Sram</i>	64 kb
<i>Clock speed</i>	Hz

2.2.6 *Digital Human Temperature (DHT22)*

Sensor DHT22 merupakan versi yang lebih canggih dari modul DHT11, dan dapat dibeli sebagai sensor tunggal atau dalam bentuk modul. Meskipun demikian, kinerja sensor tersebut tetap sama. Sensor ini hadir dalam paket dengan 4 pin, namun hanya menggunakan tiga pin untuk fungsionalitasnya. Sementara

itu, modul DHT22 hanya memiliki tiga pin seperti yang ditunjukkan pada *pinout* DHT22.

Satu-satunya perbedaan antara sensor dan modul adalah bahwa modul memiliki kapasitor *filter* dan *resistor pull-up* bawaan, dan sensor harus menggunakannya secara eksternal bila diperlukan. Modul ini sedikit lebih mahal daripada DHT11, tetapi memiliki jangkauan pengukuran yang lebih besar dan akurasi yang sedikit lebih baik.



Gambar 2. 8 DHT22 modul [19]

Pada Gambar 2.8 DHT22 merupakan sensor yang populer digunakan untuk mengukur suhu dan kelembaban. Sensor ini dilengkapi dengan termistor ntc terpisah untuk pengukuran suhu, serta *mikrokontroler* 8-bit untuk menampilkan data suhu dan kelembaban sebagai *output* serial. Sensor DHT22 telah dikalibrasi di pabrik, sehingga memudahkan integrasinya dengan *mikrokontroler* lain. Rentang pengukuran suhu yang dapat diukur oleh sensor ini adalah dari -40°C hingga 125°C , sedangkan rentang kelembaban yang dapat diukur adalah dari 0% hingga 100% dengan akurasi $\pm 1^{\circ}\text{C}$ dan $\pm 1\%$. [19].

DHT22 dibandingkan dengan *thermohygrometer* standar yang telah dipatenkan dan terpercaya untuk menilai perbandingan nilai hasil dari kedua perangkat. Data dihasilkan dari kedua alat akan dihitung perbandingannya menggunakan rumus persamaan sebagai berikut:

$$\text{selisih} = |\text{nilai uji} - \text{nilai standar}| \quad (1)$$

Dari persamaan diatas merupakan nilai selisih dari hasil pengurangan nilai uji sensor DHT22 yang digunakan pada *incubator* telur entok dengan standar alat yang digunakan yaitu *thermohygrometer*

$$\text{Rata - rata } (x) = \left| \frac{x_1+x_2+x_3+\dots+x_n}{n} \right| \quad (2)$$

Dari persamaan diatas, digunakan untuk mencari nilai rata-rata yaitu dengan menunjukkan seluruh data dan dibagi dengan banyaknya data.

$$\text{Error } (\%) = \left| \frac{\text{Nilai uji} - \text{Nilai Standar}}{\text{Nilai Standar}} \right| \times 100\% \quad (3)$$

Nilai *error* digunakan untuk menunjukkan pada deviasi standar dari hasil pengukuran. Untuk menghitung nilai *error* dapat menggunakan persamaan diatas. Nilai *error* digunakan untuk mengetahui seberapa sensitiv alat yang digunakan oleh DHT22 dengan alat standar *thermohygrometer* untuk menjamin kualitas sensor untuk menjaga kualitas suhu tetap baik didalam *incubator*.

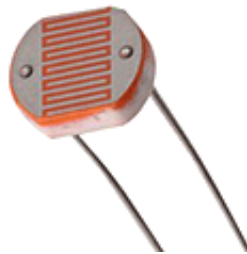
Di bawah ini merupakan spesifikasi lengkap dari sensor suhu dan kelembaban DHT22 secara umum pada tabel 2.2 berikut.

Tabel 2. 2 Spesifikasi sensor suhu dan kelembaban am2303 atau DHT22 [20]

Model	Am2303
<i>Power supply</i>	3.3 - 6v dc
<i>Output signal</i>	<i>Digital signal via single-bus</i>
<i>Sensing element</i>	<i>Polymer humidity capacitor & ds18b20 for detecting temperature</i>
<i>Measuring range</i>	<i>Humidity 0-100% rh; temperature-40~125 celsius</i>
<i>Accuracy</i>	<i>Humidity +-2% rh (max+-5%rh); temperature +-0.2 celsius</i>
<i>Resolution or sensitivity</i>	<i>Humidity 0.1%rh ; temperature 0.1 celsius</i>
<i>Repeatability</i>	<i>Humidity +-1%rh ; temperature +-0.2 celsius</i>
<i>Humidity hysteresis</i>	<i>+ -0.3%rh</i>
<i>Long-term stability</i>	<i>+ -0.5%rh/year</i>
<i>Sensing period</i>	<i>Average: 2s</i>
<i>Interchangeability</i>	<i>Fully interchangeable</i>

2.2.7 *Light Dependent Resistor (LDR)*

Photoresistor, juga dikenal sebagai *light dependent resistor (LDR)*, adalah salah satu jenis perangkat yang peka terhadap cahaya dan umum digunakan untuk mendeteksi keberadaan atau mengukur intensitas cahaya. Pada kondisi gelap, resistansinya sangat tinggi, mencapai hingga $1\text{ M}\Omega$. Namun, ketika sensor *LDR* terkena cahaya, resistansinya secara drastis menurun hingga beberapa *ohm*, tergantung pada intensitas cahaya yang terpapar. Sensitivitas *LDR* bervariasi tergantung pada panjang gelombang cahaya yang diterima, dan perangkat ini memiliki karakteristik *non-linier*. Sensor ini banyak digunakan dalam berbagai aplikasi, terutama sebagai penginderaan cahaya.



Gambar 2. 9 *light dependent resistor* [21]

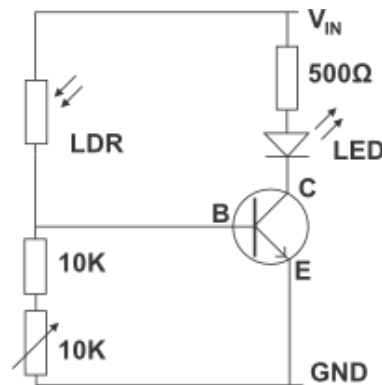
Pada Gambar 2.9 berdasarkan bahan yang digunakan, *photoresistor* dapat dibagi menjadi dua jenis yaitu internal dan eksternal.

1. *Photoresistor* internal menggunakan bahan yang tidak dilapisi bahan silikon atau germanium. *Photons* yang terpancar pada perangkat membangkitkan elektron dari pita valensi ke pita konduksi. Ini menciptakan lebih banyak elektron bebas dalam material yang tersedia untuk menghantarkan arus, sehingga menghasilkan lebih sedikit hambatan.
2. *Photoresistor* eksternal terbuat dari bahan yang telah diolah dengan *dopants*. *Dopants* menciptakan pita energi baru di atas pita valensi yang ada, yang diisi dengan elektron. Elektron ini membutuhkan lebih sedikit energi untuk bergerak ke pita konduksi karena celah energi yang lebih kecil. Hasilnya adalah perangkat yang peka terhadap panjang gelombang cahaya yang berbeda.

Namun, kedua jenis *photoresistor* tersebut sama-sama memiliki nilai resistansi yang akan berkurang saat disinari. Semakin besar intensitas cahaya, semakin besar penurunan resistansi. *Photoresistor* paling sering digunakan sebagai sensor cahaya. Sensor ini banyak digunakan ketika diperlukan untuk

mendeteksi ada dan tidak adanya cahaya atau untuk mengukur intensitas cahaya. Contohnya adalah lampu malam dan pengukur cahaya fotografi. Selain itu juga sering digunakan untuk alat yang bergantung pada cahaya seperti robot pelacak garis, yang menggunakan sumber cahaya, dua atau lebih LDR untuk menentukan perubahan jalur yang diperlukan. Sensor *photoresistor* juga sering digunakan dalam aplikasi lain seperti *kompresor audio* karena responsnya terhadap cahaya tidak seketika sehingga peran LDR adalah memberikan respons yang tertunda.

a. *Light sensor*



Gambar 2. 10 Rangkaian light sensor [21]

jika diperlukan sensor cahaya sederhana, rangkaian LDR seperti gambar 2.10 dapat digunakan. Led menyala bila intensitas cahaya pada *resistor* LDR cukup. Resistor 10 k ω yang dapat disesuaikan menentukan ambang batas di mana led menyala. Saat lampu LDR berada di bawah intensitas ambang batas, led tetap mati. Dalam aplikasi nyata, led diganti dengan *relay* atau keluarannya dapat dihubungkan ke *mikrokontroler* atau perangkat lain. Jika sensor gelap diperlukan saat led menyala tanpa cahaya, LDR dan dua *resistor* 10 k ω harus ditukar.

b. *Audio compressor*

Audio compressor adalah perangkat yang digunakan untuk mengurangi penguatan audio ketika *amplitudo* sinyal di atas nilai tertentu. Hal ini dilakukan untuk memperkuat suara pelan sekaligus mencegah suara keras menggelegar. Beberapa *audio kompresor* menggunakan LDR dan lampu kecil (*panel led* atau *electroluminescent*) yang terhubung ke sumber sinyal untuk mengubah penguatan sinyal. Teknologi ini menambahkan karakteristik sinyal yang lebih halus, karena waktu respons dan ketahanan cahaya. Waktu tunda respons aplikasi ini berada di rentang 0,1 detik [21].

2.2.8 Quality of Service (QoS)

Quality of service (qos) merupakan kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan menjamin kapasitas jaringan, mengatasi *jitter*, dan mengurangi *Delay* dalam pengiriman layanan data. *Qos* bertujuan untuk mengukur sejauh mana kinerja jaringan optimal dan melibatkan definisi karakteristik dan sifat dari suatu layanan. *Qos* digunakan untuk mengukur sekelompok atribut kinerja yang telah ditentukan dan terkait dengan layanan tersebut. Tantangan dalam *qos* terjadi saat mentransmisikan data melalui jaringan berbasis *ip dan internet*. *Qos* menghadapi beberapa kendala dalam pengiriman data melalui jaringan *ip* atau *internet*, seperti keterbatasan kapasitas *bandwidth* pada jaringan, penundaan (*Delay*), *buffering*, dan faktor lainnya [22].

2.2.9 DELAY

Delay merupakan salah satu *parameter* dalam *quality of service (qos)* yang mengindikasikan waktu total yang diperlukan oleh paket untuk melakukan perjalanan dari sumber ke tujuan. Beberapa faktor yang dapat mempengaruhi *Delay* antara lain perangkat keras yang digunakan, jarak yang harus ditempuh, dan tingkat kepadatan (*congestion*) dalam jaringan. [23]. Secara umum parameter untuk analisa qos adalah *Delay*.

Delay merujuk pada waktu yang dibutuhkan bagi data untuk melakukan perjalanan dari pengirim ke penerima. Terdapat beberapa faktor yang memengaruhi *Delay*, seperti jarak antara pengirim dan penerima, kondisi cuaca, dan faktor-faktor lainnya yang dapat mempengaruhi waktu yang diperlukan untuk mentransmisikan data.

Delay merujuk pada waktu yang diperlukan bagi data untuk bergerak dari titik asal ke titik tujuan. Faktor-faktor yang dapat mempengaruhi *Delay* meliputi jarak fisik antara perangkat pengirim dan penerima, kepadatan lalu lintas pada jaringan yang dapat menyebabkan penundaan, serta proses pemaketan data yang memerlukan waktu tambahan.[22]. Persamaan berikut ini telah ditetapkan untuk menghitung *Delay*[24]:

$$delay = \frac{\text{waktu antar paket}}{\text{jumlah paket}} \quad (4)$$

Tabel 2. 3 Delay berdasarkan perhitungan [24]

<i>Delay (ms)</i>	<i>Quality</i>
>450 ms	Buruk
300 - 450 ms	Sedang
150 - 300 ms	Bagus
<150 ms	Sangat bagus