

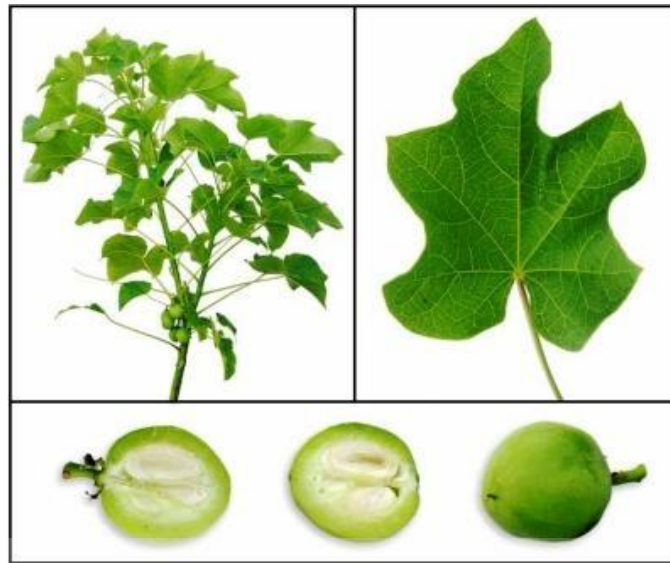
## BAB II

### LANDASAN TEORI

#### 2.1 Jarak Pagar (*Jatropha curca L.*)

Salah satu sumber minyak nabati yang sangat prospektif untuk dimanfaatkan sebagai bahan baku biodiesel adalah tanaman jarak. Beberapa jenis tanaman jarak yang tercatat di Indonesia diantaranya adalah jarak kaliki/kastor (*Ricinus communis*), jarak pagar (*Jatropha curcas*), jarak gurita (*Jatropha multifida*), dan jarak landi (*Jatropha gossypifolia*), tetapi untuk jenis tanaman jarak yang paling menghasilkan kandungan minyak terbanyak adalah jenis jarak pagar (*Jatropha Curcas L.*) karena untuk jenis tanaman ini berbuah terus menerus (tahunan).

Minyak jarak pagar tidak termasuk dalam kategori minyak makan (*edible oil*), sehingga pemanfaatannya sebagai biodiesel tidak akan mengganggu penyediaan kebutuhan minyak makan nasional. Biji (dengan cangkang) jarak pagar mengandung 20-40% minyak nabati, namun bagian inti biji (biji tanpa cangkang) dapat mengandung 45-60% minyak kasar. Untuk mendapatkan kandungan minyak yang begitu besar salah satunya yang harus diperhatikan adalah suhu dan kelembaban air disekelilingnya, maka dibutuhkan suatu sistem monitoring guna memudahkan pemantauan suhu dan kelembaban tanah pada lahan tanaman jarak. Suhu yang sesuai untuk tanaman jarak 20° – 35°C dan sedangkan untuk kelembaban tanah yang paling bagus adalah 65% dengan rentang 55%-75% (Riajaya, P. D., Kadarwati, F. T., & Yoga, S., 2007).



Gambar 2.1 Tanaman Jarak Pagar.

## 2.2 *Wireless Sensor Network* (WSN)

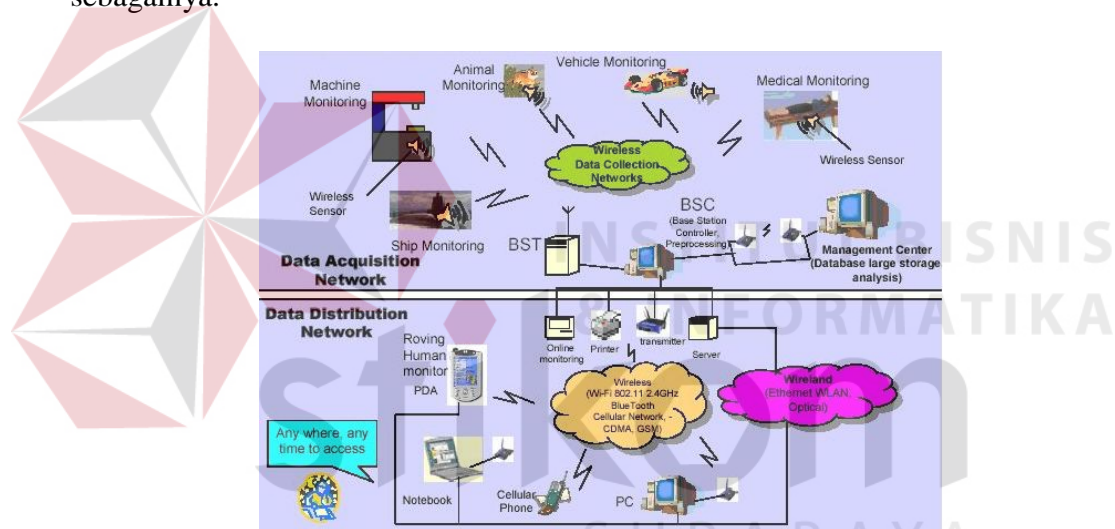
WSN adalah suatu infrastruktur jaringan *wireless* yang menggunakan sensor untuk memantau kondisi fisik atau kondisi lingkungan yang dapat terhubung ke jaringan. Masing–masing *node* dalam jaringan sensor nirkabel biasanya dilengkapi dengan radio *transceiver* atau alat komunikasi *wireless* lainnya, mikrokontroler, dan sumber energi, biasanya baterai.

Berdasarkan fakta di dunia, sekitar 98% prosesor bukan berada didalam sebuah komputer, namun terintegrasi dalam aplikasi militer, kesehatan, *remote control*, chip robotik, alat komunikasi dan mesin-mesin industri yang didalamnya telah dipasang sensor.

Perkembangan WSN dan kemajuan teknologi dapat direpresentasikan pada Gambar 2.1. Bahwa dengan berjalannya waktu, maka perkembangan teknologi semakin mengarah kepada konektivitas lingkungan fisik. Kebanyakan observasi

yang dilakukan di lapangan melibatkan banyak faktor dan parameter – parameter untuk mendapatkan hasil yang maksimal dan akurat. Jika peneliti hendak mengambil informasi langsung di lapangan, maka kendalanya adalah dibutuhkan biaya yang besar dan waktu yang lama untuk mendeteksi fenomena yang muncul sehingga menyebabkan performansi yang tidak efisien dan tidak praktis.

Dengan adanya teknologi WSN, memungkinkan peneliti untuk mendapat informasi yang maksimal tanpa harus berada di area sensor. Informasi dapat diakses dari jarak jauh melalui *gadget* seperti laptop, *remote control*, *server* dan sebagainya.



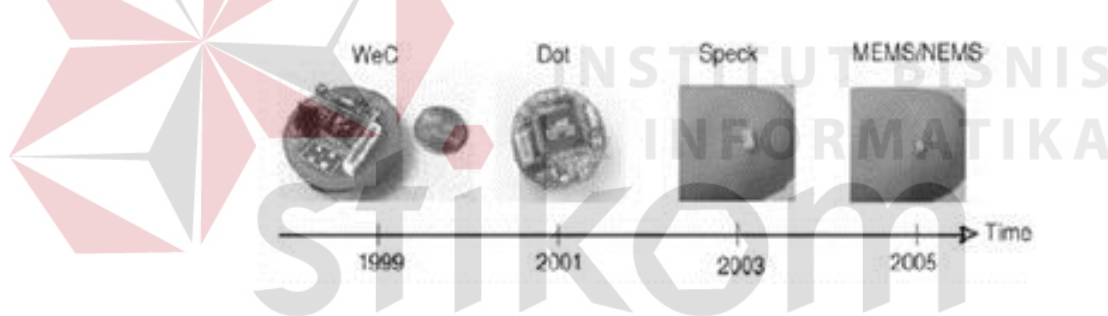
Gambar 2.2 Arsitektur WSN.

Beberapa keuntungan yang bisa diperoleh dari teknologi WSN adalah praktis dan ringkas, karena tidak perlu ada instalasi kabel yang rumit dan dalam kondisi geografi tertentu sangat menguntungkan dibanding *Wired* Sensor. Sensor menjadi bersifat *mobile*, artinya pada suatu saat dimungkinkan untuk memindahkan sensor untuk mendapat pengukuran yang lebih tepat tanpa harus khawatir mengubah desain ruangan maupun susunan kabel ruangan.

### 2.2.1 Arsitektur WSN

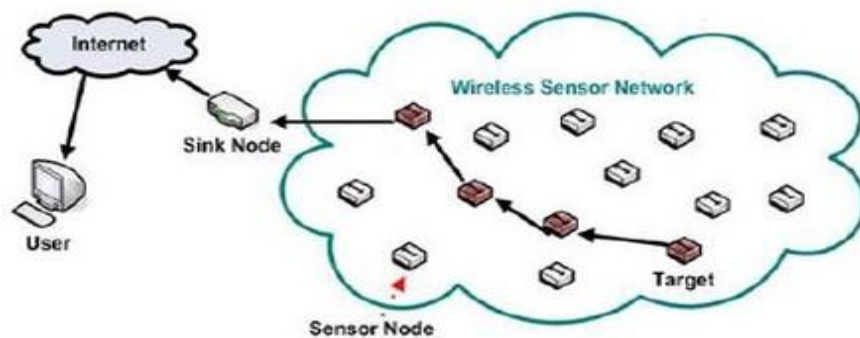
Pada WSN, *node* sensor disebar dengan tujuan untuk menangkap adanya gejala atau fenomena yang hendak diteliti. Jumlah *node* yang disebar dapat ditentukan sesuai kebutuhan dan tergantung beberapa faktor misalnya luas area, kemampuan *sensing node* dan sebagainya. Tiap *node* memiliki kemampuan untuk mengumpulkan data dan meroutingkannya kembali ke *Base Station*. *Node* sensor dapat mengumpulkan data dalam jumlah yang besar dari gejala yang timbul dari lingkungan sekitar.

Perkembangan *node* sensor mengikuti trend teknologi nano, dimana ukuran *node* sensor menjadi semakin kecil dari tahun ke tahun. *Node* sensor dapat direpresentasikan dalam Gambar 2.3 berikut:



Gambar 2.3 Perkembangan Dimensi *Node* Sensor Terhadap Waktu.

Dan untuk arsitektur WSN secara umum dapat direpresentasikan dalam Gambar 2.4 sebagai berikut:



Gambar 2.4 Gambaran Arsitek *Wireless Sensor Network*.

Pada Gambar 2.4 dapat dilihat, *node* sensor yang berukuran kecil tersebar dalam di suatu area sensor. *Node* sensor tersebut memiliki kemampuan untuk merutekan data yang dikumpulkan ke *node* lain yang berdekatan. Data dikirimkan melalui transmisi radio akan diteruskan menuju BS (*Base Station*) yang merupakan penghubung antara *node* sensor dan *user*. Informasi tersebut dapat diakses melalui berbagai *platform* seperti koneksi internet atau satelit sehingga memungkinkan *user* untuk dapat mengakses secara realtime melalui *remote server* (Rolis, 2012).

## 2.3 Perangkat Keras

### 2.3.1 Mikrokontroler AVR

Mikrokontroler adalah sebuah *computer* kecil (“*special purpose computers*”) di dalam satu IC yang berisi CPU, memori, timer, saluran komunikasi serial dan paralel, port *input/output*, ADC. Mikrokontroler digunakan untuk suatu tugas dan menjalankan suatu program.

Pada saat ini penggunaan mikrokontroler dapat kita temui pada berbagai peralatan, misalnya peralatan yang terdapat di rumah, seperti telepon *digital*, *microwave oven*, televisi, mesin cuci, *system* keamanan rumah, PDA, dan lain-lain. Mikrokontroler dapat kita gunakan untuk berbagai aplikasi misalnya untuk pengendalian, otomasi industri, akuisisi data, telekomunikasi dan lain-lain. Keuntungan menggunakan mikrokontroler yaitu harganya murah, dapat di program berulang kali, dan dapat kita program sesuai dengan keinginan kita. Saat ini keluarga mikrokontroler yang ada dipasaran yaitu intel 8048 dan 8501(MCS51), Motorola 68HC11, Microchip PIC, Hitachi H8, dan Atmel AVR.

Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler AVR.

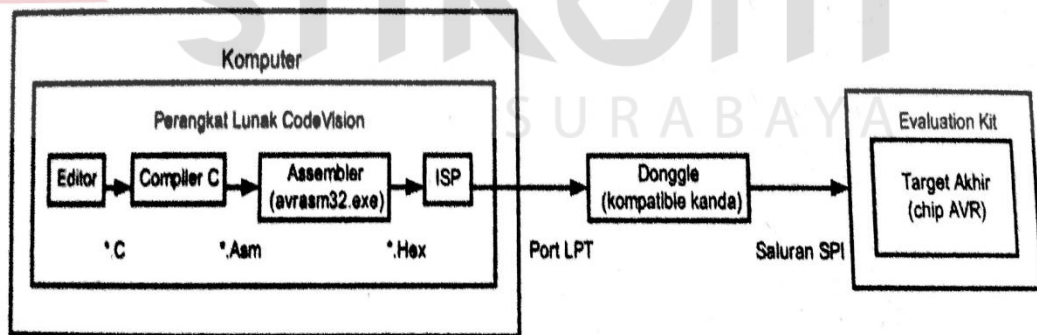
AVR adalah mikrokontroler RISC (*Reduce Instruction Set Compute*) 8 bit berdasarkan arsitektur Harvard, yang dibuat oleh Atmel pada tahun 1996. AVR mempunyai kepanjangan *Advanced Versatile RISC* atau *Alf and Vegard's Risc processor* yang berasal dari nama dua mahasiswa Norwegian Institute of Technology (NTH), yaitu Alf-Egil Bogen dan Vegard Wollan.

AVR memiliki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu AVR memiliki kecepatan eksekusi program yang lebih cepat karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock*, lebih cepat dibandingkan dengan mikrokontroler MCS51 yang memiliki arsitektur CISC (*Complex Instruction Set Computer*) di mana mikrokontroler MCS51 membutuhkan 12 siklus *clock* untuk mengeksekusi 1 instruksi. Selain itu mikrokontroler AVR memiliki fitur yang lengkap (ADC Internal, EEPROM Internal, *Timer/Counter*, *Watchdog Timer*, PWM, Port I/O, Komunikasi serial, Komparator, I2C, dan lain-lain. ), sehingga dengan fasilitas yang lengkap ini, *programmer* dan desainer dapat menggunakannya untuk berbagai aplikasi sistem elektronika seperti robot, otomasi industri, peralatan telekomunikasi, dan berbagai keperluan lain.

Pemrograman mikrokontroler AVR dapat menggunakan *low level language (assembly)* dan *high level language* (C, Basic, Pascal, JAVA, dan lain-lain) tergantung *compiler* yang digunakan. Bahasa *Assembler* mikrokontroler AVR memiliki kesamaan instruksi, sehingga jika pemrograman satu jenis mikrokontroler AVR sudah dikuasai, maka akan dengan mudah menguasai

pemrograman keseluruhan mikrokontroler jenis AVR, namun bahasa assembler relatif lebih sulit dipelajari daripada bahasa C, untuk pembuatan suatu proyek yang besar memakan waktu yang lama, serta penulisan programnya akan panjang. Sedangkan bahasa C memiliki keunggulan dibandingkan bahasa *assembler* yaitu *independent* terhadap *hardware* serta lebih mudah untuk menangani *project* yang besar. Bahasa C memiliki keuntungan-keuntungan yang dipunyai oleh bahasa mesin (*assembly*), hampir semua operasi yang dapat dilakukan oleh bahasa mesin, dapat dilakukan oleh bahasa C dengan penyusunan program yang lebih sederhana dan mudah. Bahasa C sendiri sebenarnya terletak diantara bahasa pemrograman tingkat tinggi di *assembly* (Adrianto, 2012).

Pada praktik pemrograman, mikrokontroler AVR yang digunakan yaitu ATmega16 dan *software compiler*-nya menggunakan CodeVision. Alur pemrograman mikrokontroler AVR menggunakan CodeVision AVR dapat dilihat pada Gambar 2.5 di bawah ini :



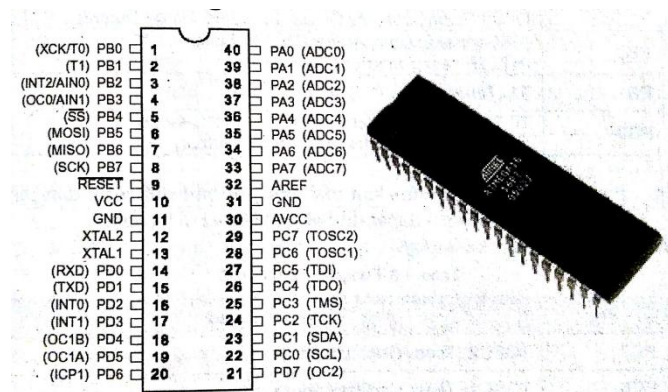
Gambar 2.5 Alur Program Code Vision

Fitur-fitur yang dimiliki ATmega16 sebagai berikut :

1. Mikrokontroler AVR 8 bit yang memiliki kemampuan tinggi, dengan daya rendah.
2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16Mhz.

3. Memiliki kapasitas Flash memori 16 KByte, EEPROM 512 Byte dan SRAM 1 KByte.
4. Saluran I/O sebanyak 32 buah, yaitu *Port A*, *Port B*, *Port C*, dan *Port D*.
5. CPU yang terdiri atas 32 buah *regiser*.
6. Unit interupsi internal dan eksternal
7. *Port* USART untuk komunikasi serial.
8. Fitur Peripheral
  - a. Tiga buah *Timer* atau *Counter* dengan kemampuan perbandingan.
    - 2 (dua) buah *Timer* atau *Counter* 8 bit dengan *Prescaler* terpisah dan mode *Compare*
    - 1 (satu) buah *Timer* atau *Counter* 16 bit dengan *Prescaler* terpisah, Mode *Compare*, dan Mode *Capture*
  - b. *Real Time Counter* dengan *Oscilaltor* tersendiri
  - c. 4 channel PWM
  - d. 8 channel 10-bit ADC
    - 8 *Singel-ended Channel*
    - 7 *Differential Channel* hanya pada kemasan TQFP
    - 2 *Differntial Channel* dengan *Progammable Gain* 1X, 10X, atau 200X
  - e. *Byte-oriented Two-wire Serial Interface*
  - f. *Programmable* Serial USART
  - g. Antarmuka SPI
  - h. *WATCHdog* Timer dengan *Oscillator internal*
  - i. *On-chip Analaog Comparator*





Gambar 2.6 Konfigurasi pin Atmega16

Konfigurasi pin ATMEGA16 dengan kemasan 40 pin DIP (*Dual In Lane Package*) dapat dilihat pada Gambar 2.6. dapat dijelaskan fungsi dari masing-masing pin ATMEGA16 sebagai berikut :

1. VCC merupakan pin yang berfungsi sebagai masukan catu daya.
2. GND merupakan pin Ground
3. Port A(PA0...PA7) merupakan pin input/output dua arah dan pin masukan ADC
4. Port B(PB0...PB7) merupakan pin input/output dua arah dan pin khusus, seperti dapat dilihat pada Tabel 2.1 di bawah ini.

Tabel 2.1 Konfigurasi Port B Mikrokontroler

Pin	Fungsi Khusus
PB7	SCK ( <i>SPI Bus Serial Clock</i> )
PB6	MISO ( <i>SPI Bus Master Input/Slave Output</i> )
PB5	MOSI ( <i>SPI Bus Master Output/Slave Input</i> )
PB4	SS ( <i>SPI Slave Select Input</i> )
PB3	AIN1 ( <i>Analog Comparator Negative Input</i> ) OC0 ( <i>Timer/Counter0 Output Compare Match Output</i> )
PB2	AIN0 ( <i>Analog Comparator Positive Input</i> ) INT2 ( <i>External Interrupt 2 input</i> )
PB1	T1 ( <i>Timer/Counter1 External Counter Input</i> )
PB0	T0 T1 ( <i>Timer/Counter0 External Counter Input</i> ) XCK ( <i>USART External Clock Input/Output</i> )

5. Port C(PC0...PC7) Merupakan *pin input/output* dua arah dan *pin* fungsi khusus, seperti dapat dilihat pada Tabel 2.2 di bawah.

Tabel 2.2 Konfigurasi Port C Mikrokontroller

<i>Pin</i>	Fungsi Khusus
PC7	TOSC2 ( <i>Timer Oscillator Pin2</i> )
PC6	TOSC1 ( <i>Timer Oscillator Pin1</i> )
PC5	TDI ( <i>JTAG Test Data In</i> )
PC4	TDO ( <i>JTAG Test Data Out</i> )
PC3	TMS ( <i>JTAG Test Mode Select</i> )
PC2	TCK ( <i>JTAG Test Clock</i> )
PC1	SDA ( <i>Two- Wire Serial Bus Data Input/Output Line</i> )
PC0	SCL ( <i>Two-Wire Serial Bus Clock LIne</i> )

6. Port D(PD0...PD7) Merupakan *pin input/output* dua arah dan *pin* fungsi khusus, seperti dapat dilihat pada Tabel 2.3 di bawah.

Tabel 2.3 Konfigurasi Port D Mikrokontroller

<i>Pin</i>	Fungsi Khusus
PD7	OC2 ( <i>Timer/Counter2 Output Compare Match Output</i> )
PD6	ICP ( <i>Timer/Counter1 Input Capture Pin</i> )
PD5	OC1A ( <i>Timer/Counter1 Output Compare A Match Output</i> )
PD4	OC1B ( <i>Timer/Counter1 Output Compare B Match Output</i> )
PD3	INT1 ( <i>External interrupt 1 Input</i> )
PD2	INT2 ( <i>External interrupt 0 Input</i> )
PD1	TXD ( <i>USART Output Pin</i> )
PD0	RXD ( <i>USART Input Pin</i> )

7. RESET merupakan *pin* yang digunakan untuk me-*reset* mikrokontroler
8. XTAL1 dan XTAL2 merupakan *pin* masukan *clock* eksternal.
9. AVCC merupakan *pin* masukan tegangan untuk ADC.
10. AREFF merupakan *pin* masukan tegangan referensi ADC.

### 2.3.2 Xbee

Xbee merupakan perangkat yang menunjang komunikasi data tanpa kabel (*wireless*). Ada 2 jenis xbee yaitu :

1. Xbee 802.15.4 (*Xbee Series 1*)

*Xbee series 1* hanya dapat digunakan untuk komunikasi *point to point* dan topologi *star* dengan jangkauan 30 meter *indoor* dan 100 meter *outdoor*.

## 2. *Xbee ZB Series 2*

*Xbee series 2* dapat digunakan untuk komunikasi *point to point*, *point to multipoint* dan topologi *star*, dan topologi *mesh* dengan jangkauan 40 meter *indoor* dan 100 meter *outdoor*.

*Xbee series 1* maupun *series 2* tersedia dalam 2 bentuk berdasarkan kekuatan transmisinya yaitu *xbee reguler* dan *xbee-pro*. *Xbee reguler* biasa disebut dengan *xbee* saja (Robosoccer, 2012). *Xbee-PRO* mempunyai kekuatan transmisi lebih kuat, ukuran perangkatnya lebih besar, dan harganya lebih mahal. *Xbee-PRO* mempunyai jangkauan *indoor* mencapai 60 meter dan *outdoor* mencapai 1500 meter. *Xbee* ini dapat digunakan sebagai pengganti serial atau USB atau dapat memasukkannya ke dalam *command mode* dan mengkonfigurasinya untuk berbagai macam jaringan *broadcast* dan *mesh*. *Shield* membagi setiap pin *Xbee*. *Xbee* juga menyediakan header pin *female* untuk penggunaan pin digital 2 sampai 7 dan input analog, yang discover oleh *shield* (pin digital 8 sampai 13 tidak tercover oleh *shield*, sehingga dapat menggunakan *header* pada papan itu sendiri (Arduino, 2011).



Gambar 2.7 *Xbee* dan *Xbee Shield*.

Berikut parameter untuk mengkonfigurasi modul *Xbee* S2 dengan mode AT.

Tabel 2.4 Parameter *Xbee*

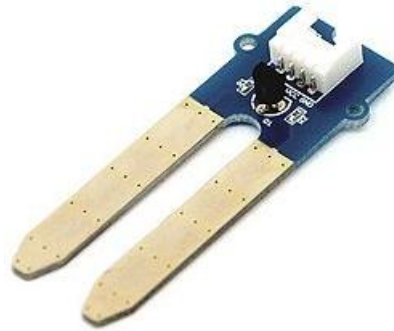
Perintah	Keterangan	Nilai <i>valid</i>	Nilai <i>Default</i>
ID	<i>Id</i> jaringan modul <i>Xbee</i>	0-0Xffff	3332
CH	Saluran dari modul <i>Xbee</i> .	0x0B-0x1A	0x0C
Sh dan SL	Nomor seri modul <i>Xbee</i> (SH memberikan 32bit tinggi, SL32 bit rendah). <i>Read-only</i> .	0-0xFFFFFFFF	berbeda untuk setiap modul
MY	Alamat 16-bit dari modul.	0-0xFFFF	0
Dh dan DL	Alamat tujuan untuk komunikasi nirkabel(DH adalah 32bit tinggi, DL32 low).	0-0xFFFFFFFF (untuk kedua DH dan DL)	0(untuk kedua DH dan DL)
BD	<i>baud rate</i> yang digunakan untuk komunikasi serial dengan papan Arduino atau komputer.	0 (1200 bps) 1 (2400 bps) 2 (4800 bps) 3 (9600 bps) 4 (19200 bps) 5 (38400 bps) 6 (57600 bps) 7 (115200 bps)	3 (9600 baud)

Catatan: meskipun nilai-nilai yang *valid* dan standar dalam Tabel 2.4 ditulis dengan awalan "0x" (untuk menunjukkan bahwa mereka adalah nomor heksadesimal), modul tidak akan mencakup "0x" ketika melaporkan nilai parameter, dan harus menghilangkan ketika menetapkan nilai-nilai (Arduino, 2011).

### 2.3.3 Soil moisture sensor (SEN0114)

*Soil Moisture Sensor* adalah sensor yang dapat mendeteksi kelembaban tanah disekitarnya. Sensor ini terdiri dari dua *probe* untuk melewati arus listrik dalam tanah, kemudian membaca resistansinya untuk mendapatkan nilai tingkat

kelembaban. Semakin banyak air membuat tanah lebih mudah menghantarkan listrik (resistansi kecil), sedangkan tanah yang kering sangat sulit menghantarkan listrik (resistansi besar).



Gambar 2.8 *Soil Moisture Sensor*.

Sensor ini sangat membantu mengingatkan tingkat kelembaban pada tanaman atau untuk memantau kelembaban tanah untuk pertanian. *IO Expansion Shield* adalah *shield* untuk menghubungkan sensor dengan Arduino (DFRobot, 2012). Pada sensor kelembaban mempunyai 3 macam *output* kondisi untuk dapat mencari nilai dalam satuan %R H, yaitu kering= 0~358, lembab= 359~460 dan basah= 461~495 yang seperti ditunjukkan pada spesifikasi berikut:

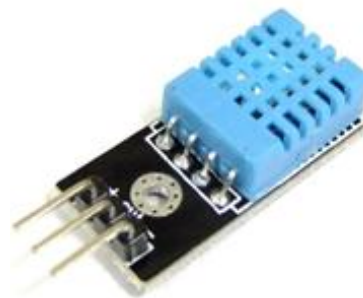
1. *Power supply*: 3.3v or 5v
2. *Output voltage signal*: 0~4.2v
3. *Current*: 35mA
4. *Pin definition*:
  - a. Analog *output*(Blue wire).
  - b. GND(*Black wire*).
  - c. *Power*(Red wire).
5. *Size*: 60x20x5mm.

- a. *Value range: 0~358 : dry soil*
- b. *359~460 : humid soil*
- c. *461~495 : in water*

#### **2.3.4 DHT11 Temperature and humidity sensor (DFR0067)**

DHT11 adalah sensor suhu dan kelembaban, dia memiliki *output* 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.

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.



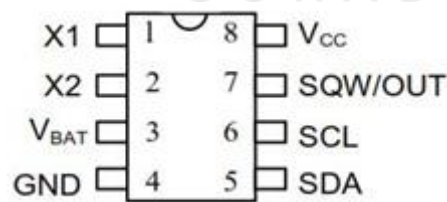
Gambar 2.9 Sensor DHT11.

Sensor ini memiliki 4 *pin* baris paket tunggal (DFRobot, 2010). Sensor ini memiliki spesifikasi sebagai berikut:

1. *Power supply*: 5 V.
2. Rentang temperatur :0-50 ° C kesalahan  $\pm 2$  ° C.
3. Kelembaban :20-90% RH
4. Toleransi  $\pm 5\%$  RH *error*.
5. *Interface*: Digital.

### 2.3.5 RTC DS1307

DS1307 merupakan *Real-time clock* yang memiliki antarmuka serial *Two-wire I2C (Inter Integrated Circuit)*. Beberapa keistimewaan RTC DS1307 yaitu sinyal keluaran berupa gelombang kotak terprogram (*programmable squarewave*), deteksi otomatis kegagalan-daya (*power -fail*), konsumsi daya kurang dari 500nA menggunakan mode baterai cadangan dengan operasional osilator. Tersedia untuk industri dengan ketahanan suhu: -40C hingga +85C. Tersedia dalam kemasan 8 pin DIP atau SOIC.



Gambar 2.10 RTC DS1307

Penjelasan dari masing-masing kaki adalah sebagai berikut:

1. X1 dan X2 adalah *pin* yang dihubungkan dengan Kristal 32.768 KHz.
2. VBAT adalah *pin* yang dihubungkan masukan baterai +3V.
3. GND adalah *pin* yang dihubungkan *Ground*.

4. SDA adalah *pin* yang difungsikan sebagai jalur data.
5. SCL adalah *pin* yang difungsikan sebagai jalur *clock*.
6. SQW/OUT adalah *pin* yang digunakan sebagai keluaran sinyal kotak.
7. VCC adalah *pin* untuk mencatu tegangan 5V.

DS1307 memiliki akurasi hingga tahun 2100. Sistem RTC DS1307 memerlukan baterai eksternal 3 volt yang terhubung ke *pin* Vbat dan ground. *Pin* X1 dan X2 dihubungkan dengan Kristal osilator 32.768 KHz. Sedangkan *pin* SCL, SDA, dan SQW/OUT di-*pull-up* dengan resistor (nilainya 1k sampai dengan 10k) ke Vcc. Untuk membaca data tanggal dan waktu yang tersimpan di memori RTC DS1307 dapat dilakukan melalui komunikasi serial I2C (Adrianto, 2012).

## **2.4 Perangkat Lunak**

### **2.4.1 Code Vision AVR**

Ada beberapa program yang dapat digunakan sebagai *editor* dan *compiler* untuk mikrokontroler AVR, salah satunya yaitu CodeVision. CodeVisionAVR adalah salah satu alat bantu pemrograman (*programming tool*) yang bekerja dalam lingkungan pengembangan perangkat lunak yang terintegrasi (*Integrated Development Environment, IDE*). Seperti aplikasi IDE lainnya, CodeVisionAVR dilengkapi dengan *source code editor, compiler, linker*, dan dapat memanggil Atmel AVR Studio untuk debugger nya (Adrianto, 2012).

### **2.4.2 Sleep Mode Idle**

*Mode sleep* adalah *mode* kerja yang mematikan modul AVR yang tidak diperlukan supaya dapat menghemat pemakaian daya. *Sleep mode idle* akan



menghentikan CPU. Tetapi mengizinkan modul-modul SPI, USART, *Analog, Comparator, TWI, Timer* atau *Counter, Watchdog*, dan sistem interupsi, untuk terus bekerja. *Mode sleep* ini menghentikan *clock* clkCPU dan clkFlash, tetapi tetap menjalankan *clock* yang lain.

*Mode idle* dapat dibangunkan, baik oleh interupsi eksternal maupun internal seperti interupsi timer *overflow* dan USART *transmit complete*. Jika AVR tidak perlu dibangunkan dari *idle* melalui interupsi analog comparator maka sebaliknya analog *comparator* dimatikan dengan cara memberi logika 1 pada bit ACD supaya dapat lebih menghemat daya AVR dalam *mode idle*. Jika ADC diizinkan maka konversi dimulai secara otomatis ketika *mode idle* dimasuki (Nurchahyo, 2012).

### 2.4.3 X-CTU

X-CTU adalah sebuah aplikasi yang disediakan oleh DIGI, dimana program ini dirancang oleh Digi untuk berinteraksi dengan *Xbee*. Pada aplikasi ini *user* bisa mengupdate *firmware xbee* dari *coordinator* menjadi *Router/End Device(endpoint)* ataupun sebaliknya (DIGI, 2008).