

BAB II

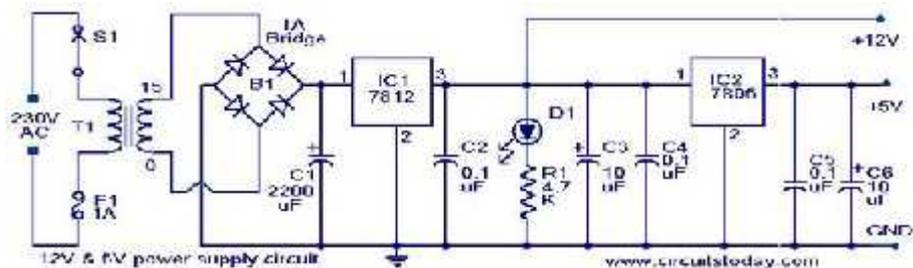
TINJAUAN PUSTAKA

2.1 Catu Daya (*Power Supply*)

Catu daya adalah sebuah peralatan penyedia tegangan atau sumber daya untuk peralatan elektronika dengan prinsip mengubah tegangan listrik yang tersedia dari jaringan distribusi transmisi listrik menuju level yang diinginkan sehingga berimplikasi pada perubahan daya listrik. Dalam sistem perubahan daya.

Jika suatu catu daya bekerja dengan beban maka terdapat keluaran tertentu dan jika beban tersebut dilepas maka tegangan keluar akan naik, persentase kenaikan tegangan dianggap sebagai regulasi dari catu daya tersebut. Regulasi adalah perbandingan perbedaan tegangan yang terdapat pada tegangan beban penuh.

Agar tegangan keluaran catu daya lebih stabil, dapat digunakan suatu komponen IC yang disebut IC regulator, misalnya IC Regulator 7812 atau IC Regulator 7805. Hal ini memungkinkan keluaran DC catu daya dapat dibentuk sesuai kebutuhan. Gambar 2.1 menunjukkan rangkaian catu daya menggunakan IC regulator.



Gambar 2.1 rangkaian catu daya

(<http://www.circuitstoday.com/12v-5v-combo-power-supply>)



2.1.1 Baterai Li-po

Baterai Li-po terletak pada penghantar arus listrik (*elektrolit*) yang ada pada kedua jenis baterai tersebut. Baterai Li-Po adalah singkatan *Lithium Polymer*, baterai ini bersifat cair (*Liquid*), menggunakan elektrolit polimer yang padat, dan mampu menghantarkan daya lebih cepat dan jenis baterai ini adalah hasil pengembangan dari *Lithium Ion*. Baterai Li-Po ini disebut sebagai baterai ramah lingkungan. Berikut ini adalah kelebihan dan kekurangan Baterai Li-Po, yaitu:

- **Kelebihan Baterai Li-Po**

1. Ramah Lingkungan
2. Fleksibel bisa dibuat berdasarkan kebutuhan
3. Lebih Aman
4. Lebih ringan

- **Kekurangan Baterai Li-Po**

1. Biaya manufaktur mahal
2. Harga baterai juga mahal karena cost untuk energi ini juga mahal
3. Butuh perawatan khusus untuk isi ulang, seperti jangan sampai baterai habis baru di isi ulang
4. Usia baterai lebih pendek



Gambar 2.2 contoh baterai Li-po

(<http://jogjarobotika.com/sistem-energi/48-turnigy-nano-tech-1300mah-3s-111v.html>)



2.1.2 UBEC (*Universal Battery Elimination Circuit*)

UBEC merupakan rangkaian untuk mengubah tegangan, tinggi ke rendah atau sebaliknya, memerlukan rangkaian yang tepat, agar daya dapat di-*deliver* dengan tingkat efisiensi setinggi mungkin. Namun ada juga SBEC (*Switching Battery Elimination Circuit*) dimana secara keseluruhan kegunaannya sama dengan UBEC, hanya saja SBEC memiliki kualitas dibawah UBEC

Untuk menurunkan tegangan dengan menggunakan IC regulator seperti 7805, sangat umum digunakan. Regulator ini memiliki kemampuan menangani arus hingga 1A, dengan V_{in} minimal sama dengan 7V, untuk menghasilkan output 5V. Dengan perhitungan sederhana, bila $V_{in} = 9V$, maka disipasi daya 4 Watt, satu nilai yang cukup besar (panas) atau menggunakan regulator linier tipe LDO, seperti 2940, yang juga memiliki kemampuan menangani arus hingga 1A, dengan V_{in} minimal sama dengan 5.5V, untuk menghasilkan output 5V.

Pilihan lain adalah regulator switching. Untuk kebutuhan mencatu motor servo atau rangkaian lain yang bekerja pada tingkat tegangan 5V – 6V, dapat menggunakan UBEC. UBEC adalah rangkaian elektronik yang mengambil daya dari battery pack atau sumber DC lainnya, dan menurunkannya ke level tegangan 5V atau 6V. Tegangan input maksimum tergantung pada spesifikasi UBEC.



Gambar 2.3 UBEC (*Universal Battery Elimination Circuit*)

(<http://www.evolution-models.com/speed-controller-electronic-speed-controller-brushless-esc/turnigy-5a-sbec.html>)



2.2 Sensor Jarak SHARP GP2Y0A21

Sensor jarak Sharp pilihan populer untuk banyak proyek yang membutuhkan pengukuran jarak akurat. Sensor IR ini lebih ekonomis daripada pengukur jarak sonar, namun memberikan kinerja yang jauh lebih baik daripada alternatif lain IR. Interfacing ke mikrokontroler yang paling mudah adalah output analog tunggal dapat dihubungkan ke sebuah konverter analog-ke-digital untuk melakukan pengukuran jarak, atau output dapat dihubungkan ke komparator untuk deteksi ambang batas. Jangkauan deteksi versi ini adalah sekitar 10 cm sampai 80 cm (4 "sampai 32") plot jarak terhadap tegangan output ditunjukkan di bawah ini.

Sensor ini termasuk pada sensor jarak kategori optik. Pada dasarnya, sensor ini sama seperti sensor *Infra Red* (IR) konvensional, GP2Y0A21 memiliki bagian transmitter/emitter dan receiver (detektor). Bagian transmitter akan memancarkan sinyal IR, sedangkan pantulan dari IR (apabila mengenai sebuah objek) akan ditangkap oleh bagian detektor yang terdiri dari lensa pem-fokus dan sebuah linear CCD array. Linier CCD array terdiri atas sederetan elemen peka cahaya yang disebut piksel (*picture element*). Dalam Gambar 2.4 memperlihatkan bentuk fisik dari sensor sharp GP2Y0A21.

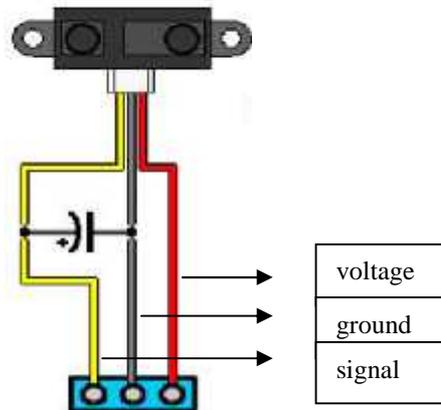


Gambar 2.4 Bagian Depan dan Belakang Sensor SHARP GP2Y0A21

(<http://www.ediy.com.my/index.php/blog/item/92-sharp-gp2y0a21-ir-distance-sensors>)

2.2.1 Pin Out Sensor SHARP GP2Y0A21

Sensor ini memiliki 3-pin, **Voltage**, **Ground**, **Signal**. Output sensor ini adalah analog tunggal, dapat terhubung ke sebuah konverter analog ke-digital untuk mengambil pengukuran jarak, atau output dapat dihubungkan ke comparator untuk deteksi ambang batas. Untuk menghubungkan sensor ke mikrokontroler, sensor Sharp GP2Y0A21 menggunakan konektor JST 3-pin yang terhubung ke kabel 3-in JST untuk sensor jarak itu sendiri.



Gambar 2.5 Pin out pada sensor Sharp GP2Y0A21

(<http://www.mhobbies.com/free-shipping-sharp-gp2y0a21-ir-distance-sensor-module-10-80cm.html>)



Gambar 2.6 JST connector 3-pin

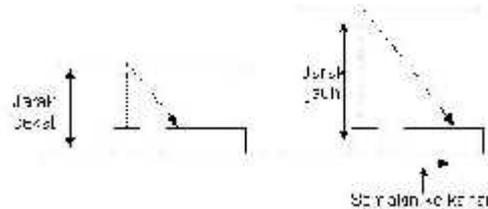
(www.robot-electronics.co.uk)

2.2.2 Prinsip Kerja Sensor SHARP GP2Y0A21

Bila dibandingkan dengan sensor jarak ultrasonik, dimana dapat mengatasi tipuan-tipuan dalam bentuk cermin, namun sensor ini memiliki kelemahan apabila obyek yang dideteksi berupa dinding yang bergelombang di mana sinyal sonar akan dipantulkan ke arah lain sehingga jarak tidak terdeteksi. Untuk mengatasi hal ini, sensor inframerah sebagai pendukung sistem pengukuran jarak adalah alternatif yang baik. Berbeda dengan sensor ultrasonik, sensor inframerah tidak menghitung waktu pancaran sinar melainkan menghitung di bagian mana sinar inframerah yang dikembalikan diterima oleh rangkaian phototransistor. Semakin jauh jarak maka semakin ke kanan sinar inframerah yang diterima pada rangkaian



phototransistor dan semakin kecil tegangan outputnya. Hasil output ini akan diterima oleh adc terlebih dahulu sebelum diambil oleh mikrokontroler.

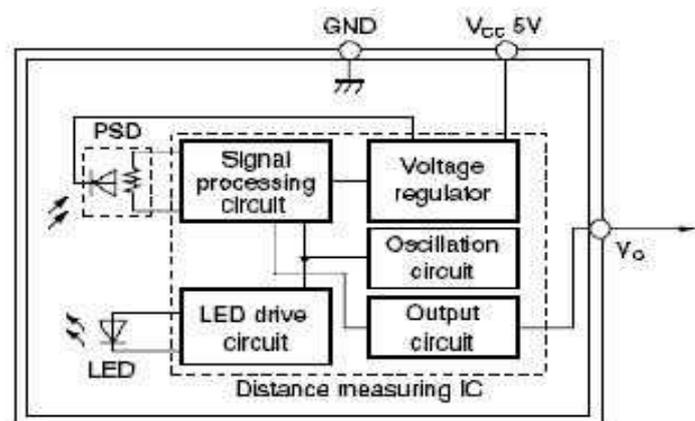


Gambar 2.7 Sudut pantul sharp GP2Y0A21

(<http://deltaelectronic.com/article/kri,krci,%20peraturan%20krci%202012,%20panduan%20krci%202012,%20peraturan%20krci,%20panduan%20krci/sensor/>)

Sharp GP2Y0A21 merupakan versi terbaru dari Sharp GP2D12 . Bagian LED Drive circuit akan memancarkan cahaya inframerah ke obyek dan memantulkan dalam sudut yang sama. Apabila obyek menjauh maka sinar akan diterima semakin ke kanan dan tegangan keluaran akan semakin mengecil.

Sinar diterima pada phototransistor yang ada di dalam bagian *signal processing circuit* dan menghasilkan tegangan analog yang dikeluarkan ke bagian output

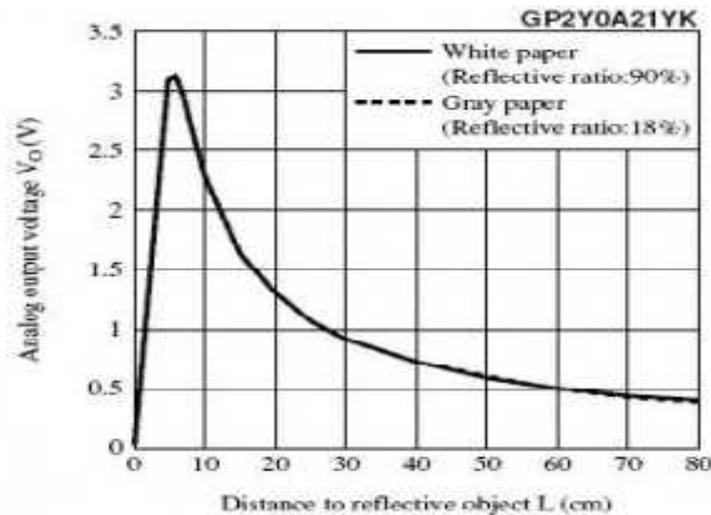


Gambar 2.8 GP2Y0A21 Block Diagram

(<http://deltaelectronic.com/article/kri,krci,%20peraturan%20krci%202012,%20panduan%20krci%202012,%20peraturan%20krci,%20panduan%20krci/sensor/>)



Hasil output tegangan tersebut tidaklah linier melainkan membentuk kurva seperti pada gambar berikut. Sensor mulai menampilkan jarak yang valid saat berada di jarak sekitar 4 cm dan menurun hingga 80 cm.



Gambar 2.9 Kurva Sharp GP2Y0A21

(<http://deltaelectronic.com/article/kri,krci,%20peraturan%20krci%202012,%20panduan%20krci%202012,%20peraturan%20krci,%20panduan%20krci/sensor/>)

Untuk menghitung jarak maka dapat dilakukan dengan dua cara yaitu *look up table* dan interpolasi. Teknik paling sederhana adalah *look up table* yaitu dengan menyimpan di memori jarak-jarak.

Berikut ini adalah beberapa sensor inframerah untuk pengukur jarak yang ada.

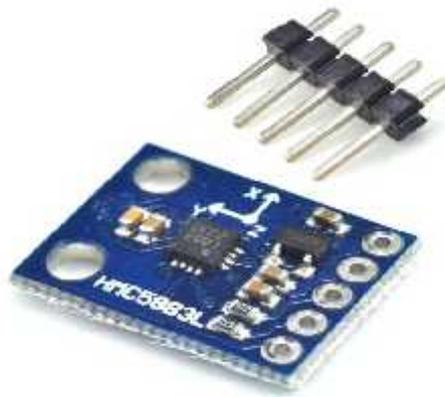
- Sharp GP2D12, versi lama dengan jarak maksimum 80 cm
- Sharp GP2Y0A21, versi terbaru dari Sharp GP2D12 dengan jarak maksimum 80 cm
- Sharp GP2Y0A02, untuk jarak maksimum 150 cm.

2.3 Sensor Kompas Magnetometer 3 Axis HMC5883L

Magnetometer 3 Axis HMC5883L adalah sensor magnet terkemas dalam *surface mount* 3.0x3.0x0.9 mm 16-pin *Leadless Chip Carrier* (LCC). Magnetometer 3 Axis HMC5883L tersusun atas sensor resistif magnet beresolusi



tinggi dengan demagnetisasi otomatis, penghilang offset dan ADC 12-bit untuk pengukuran medan magnet bumi dengan resolusi tinggi. Menggunakan teknologi *anisotropic magneto-resistive* (AMR) Honeywell, *Magnetometer 3 Axis HMC5883L* menyediakan kepresisian lebih pada sensitifitas dan linieritas sumbu dan dirancang untuk mengukur kedua arah dan medan magnet bumi.



Gambar 2.10 Sensor Kompas *Magnetometer 3 Axis HMC5883L*

(http://www.bengkel-elektro.com/product.php?category=93&product_id=217)

2.3.1 Spesifikasi Sensor Kompas *Magnetometer 3 Axis HMC5883L*

Sensor kompas *Magnetometer 3 Axis HMC5883L* memiliki beberapa spesifikasi, antara lain :

1. ADC 12-bit ADC terkopling dengan sensor AMR low noise yang akan menghasilkan resolusi 2-mili gauss pada medan ± 8 gauss
2. Mengijinkan akurasi kompas 1 sampai 2 derajat
3. Tersedia self-test yaitu fitur tambahan yang dikemas dalam ASIC
4. Tegangan kerjanya rendah (2.16 ~ 3.6V) dan konsumsi daya rendah (100uA).
Cocok untuk aplikasi yang dicatu menggunakan baterai
5. Tersedia rangkaian drive strap
6. Menyediakan demagnetisasi sensor untuk setiap pengukuran, dan juga kompensasi offset untuk mendapatkan pengukuran yang konsisten dengan akurasi hingga 1 sampai 2 derajat dan mereduksi perlunya kalibrasi ulang
7. Antarmuka digital I2C



8. Range medan magnet yang dapat diukur cukup lebar (+/-8 Oe)
9. Sensor bisa digunakan pada lingkungan dengan medan magnet yang kuat dengan akurasi kompas 1 sampai 2 derajat.

Selain itu sensor ini juga memiliki software dan algoritma yang didukung oleh:

1. Heading, hard iron, soft iron dan auto calibration libraries
2. Laju keluaran cepat maksimal 160 Hz
3. Ideal untuk navigasi navigasi pejalan kaki dan aplikasi LBS

2.4 I²C BUS (*Inter-Integrated Circuit Bus*)

I²C merupakan bus standar yang didesain oleh Philips pada awal tahun 1980an untuk memudahkan komunikasi antar komponen yang tersebar pada papan rangkaian. I²C merupakan singkatan dari Inter IC atau komunikasi antar IC, sering disebut juga IIC atau I2C. Pada awalnya, kecepatan komunikasi maksimumnya diset pada 100kbps karena pada awalnya kecepatan tinggi belum dibutuhkan pada transmisi data. Untuk yang membutuhkan kecepatan tinggi, ada mode 400kbps dan sejak 1998 ada mode kecepatan tinggi 3,4Mbps. I²C tidak hanya digunakan pada komponen yang terletak pada satu board, tetapi juga digunakan untuk mengkoneksikan komponen yang dihubungkan melalui kabel.

Kesederhanaan dan fleksibilitas merupakan ciri utama dari I²C, kedua hal tersebut membuat bus ini mampu menarik penggunaannya dalam berbagai aplikasi. Fitur-fitur signifikan dari bus ini adalah :

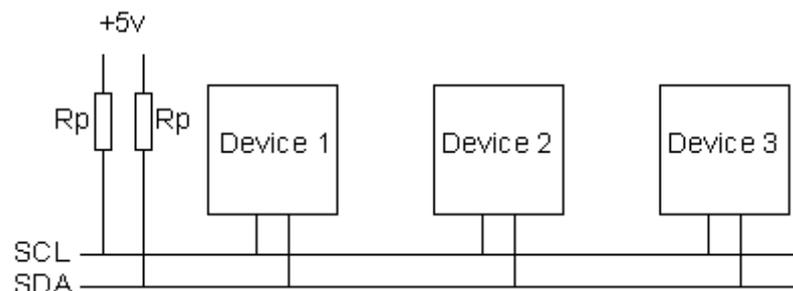
- hanya 2 jalur/kabel yang dibutuhkan
- tidak ada aturan baud rate yang ketat seperti pada RS232, di bus ini IC yang berperan sebagai master akan mengeluarkan bus clock
- Hubungan master/slave berlaku antara komponen satu dengan yang lain, setiap perangkat yang terhubung dengan bus mempunyai alamat unik yang diset melalui software
- IC yang berperan sebagai master mengontrol seluruh jalur komunikasi dengan mengatur clock dan menentukan siapa yang menggunakan jalur komunikasi. Jadi IC yang berperan sebagai slave tidak akan mengirim data kalau tidak diperintah oleh Master.



- I²C merupakan bus yang mendukung multi-master yang mempunyai kemampuan arbitrase dan pendeteksi tabrakan data.

2.4.1 Mekanisme Hubungan Antar Komponen

I²C hanya membutuhkan dua jalur/kabel yaitu SDA dan SCL. SCL/serial clock merupakan jalur clock, digunakan untuk mensinkronkan data transfer antara Master dan Slave dalam I²C bus. SDA/Serial Data merupakan jalur komunikasi data dua arah. SDA dan SCL dihubungkan ke seluruh komponen dalam bus I²C. Selain kedua jalur/kabel masih ada jalur/kabel ketiga yaitu Ground/0 volt serta jalur Vcc/5v untuk menghidupkan perangkat/komponen. Baik SDA maupun SCL merupakan tipe open drain. Maksud dari open drain adalah chip bisa membuat outputnya berlogika 0, tapi tidak bisa membuat outputnya berlogika 1. Sehingga agar mampu memberikan output 1, diperlukan pull up resistor yang dihubungkan ke suplai 5v. Jadi kita harus memberikan resistor pull up pada jalur SDA dan SCL yang ujung satunya dihubungkan ke 5v. Kita hanya memerlukan 1 buah resistor per jalur untuk seluruh I²C bus, tidak perlu satu-persatu untuk tiap perangkat. Ilustrasinya adalah sebagai berikut:



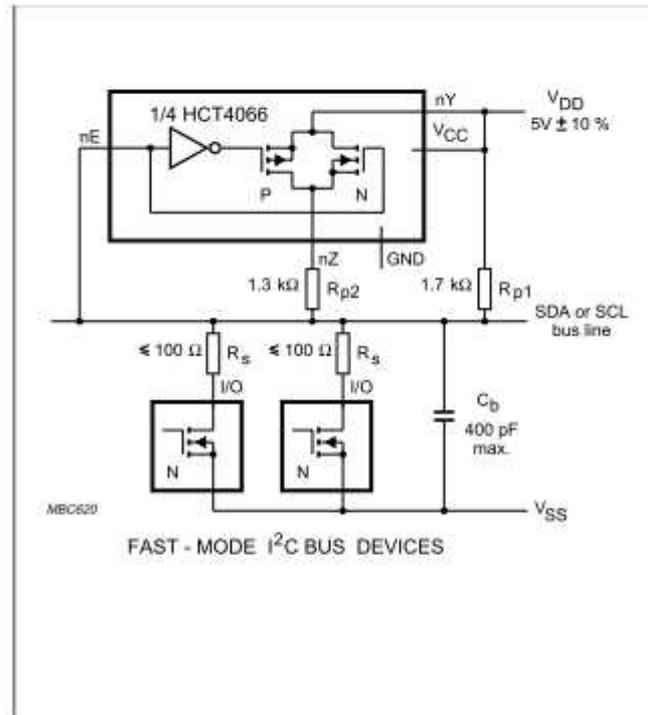
Gambar 2.11 Ilustrasi Rangkaian Pull Up

(<http://pima.kandangbuaya.com/2011/04/i2c-inter-ic-bus-twi/>)

Nilai dari resistor tidak begitu berpengaruh. Dari referensi di internet, didapat bahwa ada yang memakai resistor dengan nilai mulai 1k8 (18000 ohms) sampai 47k (47000 ohms), untuk penulis sendiri, berdasar pengalaman cenderung memilih nilai 4k7 (4700ohm). Pemilihan nilai lain pun tidak masalah, asal masih dalam range tersebut karena berdasar I²C bus specification yang dirilis oleh NXP pada 2001, pull up device tergantung pada bus loads. Jika bus load mencapai



200pF, maka pull up deviceny bisa berupa resistor, juka bus load antara 200-400pF (fast mode) maka pull up device bisa berupa sumber arus (max 3mA) atau switched resistor circuit (silahkan lihat gambar dibawah untuk referensi). Jika kita tidak memberikan pull up device (resistor/sumber arus) maka jalur SDA dan SCL akan selalu berlogika rendah, hampir 0 volt, dan I²C bus tidak akan bekerja.



Gambar 2.12 Rangkaian Switched-Pull Up

(<http://pima.kandangbuaya.com/2011/04/i2c-inter-ic-bus-twi/>)

2.4.2 Masters dan Slave

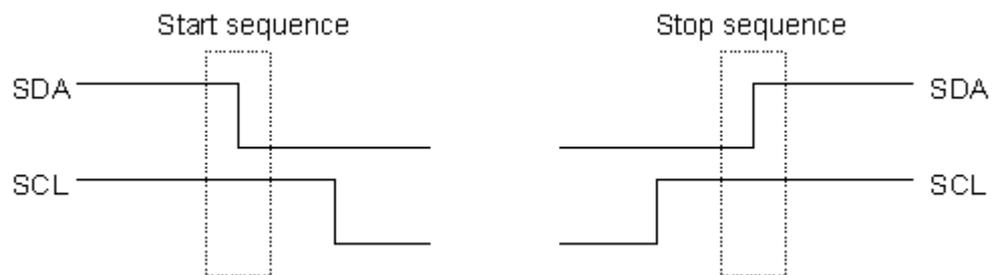
Perangkat/komponen dalam bus I²C akan berperan sebagai Master atau Slave. Master merupakan komponen/perangkat yang mengontrol jalur clock dan siapa yang berhak menggunakan jalur data. Slave adalah komponen/perangkat yang merespon pada perintah Master dan tidak bisa mengambil inisiatif komunikasi. Jadi hanya master saja yang bisa mengambil inisiatif transfer data dalam I²C bus. I²C bus mendukung multiple slave dalam satu bus dengan hanya 1 master. Pada teorinya, sistem ini mendukung multiple master, tapi tidak akan dibahas karena jarang digunakan. Pada sistem seperti robot, Master adalah kontrolernya (mikrokontroler/mikroprosesor) dan slave adalah modul sensor



seperti SRF08, CMPS03 atau modul RTC DS1307. Baik master maupun slave bisa mentransfer data melalui I²C bus, tetapi transfer hanya dikontrol oleh master.

2.4.3 Protokol Fisik I²C

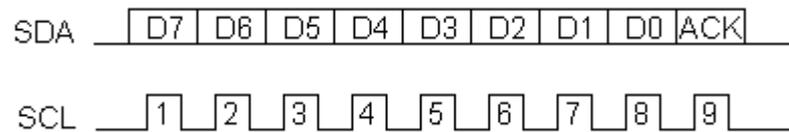
Ketika master (kontroler) ingin berkomunikasi dengan slave, master akan mulai mengirim start sequence pada bus I²C. Start sequence adalah salah satu dari dua sequence spesial pada bus I²C, sekuens spesial lainnya adalah stop. Start sequence dan stop sequence merupakan tahap spesial dimana merupakan kondisi dimana SDA (jalur data) boleh berubah ketika SCL (jalur clock) dalam kondisi high (1). Start sequence menandai awal dari transaksi data dengan perangkat slave. Stop sequence menandai akhir transaksi data dengan perangkat slave.



Gambar 2.13 Start Sequence dan Stop Sequence

(<http://pima.kandangbuaya.com/2011/04/i2c-inter-ic-bus-twi/>)

Data ditransfer dalam rangkaian/sequence 8bit. Bit dikirim lewat jalur SDA dimulai dengan MSB (Most Significant Bit). Jalur SCL kemudian dipulsakan high(1) kemudian low(0). Tanpa pull-up device, chip tidak bisa mengubah line ke kondisi high (1). Untuk setiap 8 bit yang ditransfer, perangkat penerima data mengirim balik bit *acknowledge* (ack), jadi pada kenyataannya ada 9 clock SCL untuk tiap transfer dari 8 bit data. Jika perangkat penerima mengirim balik sebuah low ACK bit (ACK=0) maka perangkat tersebut telah menerima data dan siap menerima data berikutnya. Jika perangkat penerima mengirim balik high ACK bit (ACK=1) berarti perangkat tersebut tidak bisa lagi menerima data selanjutnya, sehingga master harus mengakhiri transfer data dengan mengirim stop sequence.



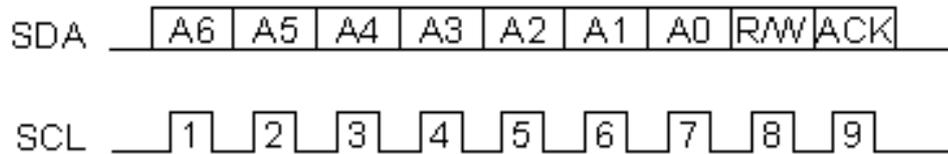
Gambar 2.14 Data SDA dan SCL

(<http://pima.kandangbuaya.com/2011/04/i2c-inter-ic-bus-twi/>)

Kecepatan serial clock (SCL) untuk I²C mencapai 100KHz. Untuk yang membutuhkan kecepatan tinggi, ada fast mode dimana clock bisa mencapai 400Khz dan sejak 1998 ada mode kecepatan tinggi dimana clock bisa mencapai 3,4MHz. Untuk aplikasi seperti robot dan modul embedded yang non vital biasanya clock yang digunakan masih dibawah 100KHz. Berdasarkan I²C Specification terbaru yang dikeluarkan NXP/Phillips, untuk transmisi data pada clock diatas 1Mhz dibutuhkan delay beberapa uS diantara setiap byte yang ditransfer.

2.4.4 Pengalamatan Perangkat pada Bus I²C

Teorinya, alamat perangkat pada bus I²C bisa dalam format 7bit maupun 10bit. Namun, pengalamatan 10bit jarang digunakan dan tidak akan dibahas dalam makalah ini. Modul yang umum digunakan biasanya menggunakan alamat 7bit. Ini berarti anda bisa menggunakan sampai 128 perangkat dalam satu bus I²C, karena alamat 7 bit adalah angka dari 0 sampai 127. Ketika kita mengirim alamat 7 bit, kita tetap akan mengirim data 8 bit sesuai format pengiriman data dalam bus I²C. Bit sisa digunakan untuk menginformasikan pada slave apakah master akan menulis atau membaca ke perangkat slave tersebut. Jika bit sisa tersebut bernilai 0 maka master akan menulis ke slave. Jika bit sisa tersebut bernilai 1 maka master akan membaca data dari slave. Tujuh bit alamat akan ditempatkan pada 7 bit pertama dari data 8 bit sehingga bit Read/Write (R/W) terletak di LSB (Least Significant Bit).



Gambar 2.15 Alamat 7 bit

(<http://pima.kandangbuaya.com/2011/04/i2c-inter-ic-bus-twi/>)

Peletakan 7 bit alamat pada 7 bit awal dari data 8 bit biasanya menjadi sumber kebingungan pemula dalam memahami mekanisme pengalamatan dalam I²C. Contohnya jika kita ingin menulis pada alamat 21, kita harus mengirim 42 yang berarti 21 digeser 1 bit ke depan. Maka dari itu lebih mudah bagi kita untuk mengumpamakan bus I²C memiliki alamat 8 bit dimana alamat genap untuk menulis dan ganjil untuk membaca dari perangkat tersebut. Contohnya adalah pada CMPS03, untuk menulis digunakan alamat 0xC0 (\$C0) dan 0xC1 (\$C1) untuk membaca data darinya. Jadi bit R/W hanya akan membuat alamatnya hanya berbeda genap dan ganjil.

2.4.5 Protokol Software pada Bus I²C

Pada software (pemrograman) misal lewat bascom, codevision, win avr, ada perintah-perintah khusus untuk protokol I²C. Yang pertama dilakukan adalah master akan mengirim start sequence. Start sequence akan membuat semua perangkat slave bersiap mendengar perintah yang datang dari master. Kemudian master akan mengirim alamat perangkat yang dituju. Slave yang cocok dengan address yang dikirim oleh master akan melanjutkan ke transmisi berikutnya. Slave selain yang alamatnya cocok akan mengabaikan transmisi dari mater dan menunggu sampai alamatnya cocok. Setelah mendapatkan slave yang alamatnya cocok, master sekarang harus mengirim lokasi internal dari nomor register pada slave yang akan dibaca atau ditulis oleh master. Kita dapat mengetahui nomor registernya pada datasheet perangkat/modul. Contohnya CMPS03 mempunyai 16 lokasi (0-15) dan SRF08 mempunyai 36.

Setelah mendapat slave yang alamatnya cocok dan alamat internal register yang dituju, maka master dapat mulai mengirim data per 8bit (1 byte). Jika slave sudah mengacknowledged (ACK=0) setelah pengiriman 1 byte data, maka master



dapat mengirim 1 byte data selanjutnya dan proses akan berulang. Data secara otomatis akan ditempatkan pada internal register selanjutnya (tidak menumpuk data yang sudah terkirim) karena slave akan mengincrement alamat register internal setelah menerima 1 byte data. Ketika master sudah selesai menulis seluruh data ke slave, maka master akan mengirim stop sequence yang mengakhiri transaksi data.

Jadi untuk menulis ke perangkat slave, urutannya adalah:

1. Mengirim start sequence
2. Mengirim alamat I²C dari slave dengan bit R/W low (alamat genap)
3. Mengirim alamat dari internal register yang ingin ditulisi data
4. Mengirim byte data
5. Jika data belum tertulis semua, setelah Slave mengirim ACK=0 maka byte data berikutnya dikirim
6. Mengirim stop sequence

Untuk contohnya adalah mengirim data ke SRF08, untuk memulai pengukuran jarak kita harus mengirim perintah 0x51 ke register perintah di alamat 0x00.

Sehingga urutannya:

1. Mengirim start sequence
2. Mengirim 0xE0 (alamat I²C dari SRF08 dengan bit R/W low)
3. Mengirim 0x00 (alamat dari register perintah internal)
4. Mengirim 0x51 (perintah untuk memulai pengukuran jarak SRF08)
5. Mengirim stop sequence

2.4.6 Membaca dari Slave

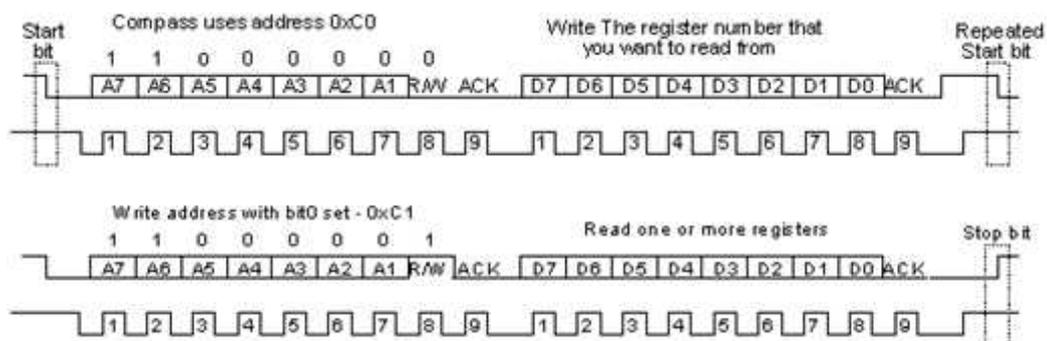
Untuk membaca dari slave, prosesnya sedikit lebih rumit. Sebelum membaca dari perangkat slave, kita harus memberitahu alamat register internal mana yang ingin kita baca. Jadi proses pembacaan dari slave juga akan diawali dengan menulis ke slave dahulu. Jadi awalnya sama saja, kita mengirim start



sequence, alamat I²C dari slave dengan bit R/W low dan alamat internal register yang kita tulisi. Kemudian kita mengirim kembali sebuah start sequence (restart), kembali mengirim alamat I²C, kali ini dengan bit terakhir=1 (bit read) dan alamat internal register yang ingin kita baca. Kemudian kita membaca byte data sebanyak yang kita inginkan sampai kita mengakhiri transmisi dengan stop sequence.

Jadi untuk membaca arah kompas dalam bentuk byte dari modul CMPS03 urutannya adalah:

1. Mengirim start sequence
2. Mengirim 0xC0 (alamat I²C dari CMPS03 dengan bit R/W low atau menulis)
3. Mengirim 0x01 (alamat register internal dari arah kompas)
4. Mengirim start sequence lagi
5. Mengirim 0xC1 (alamat I²C dari CMPS03 dengan bit R/W high atau baca)
6. Membaca byte data dari CMPS03
7. Mengirim stop sequence



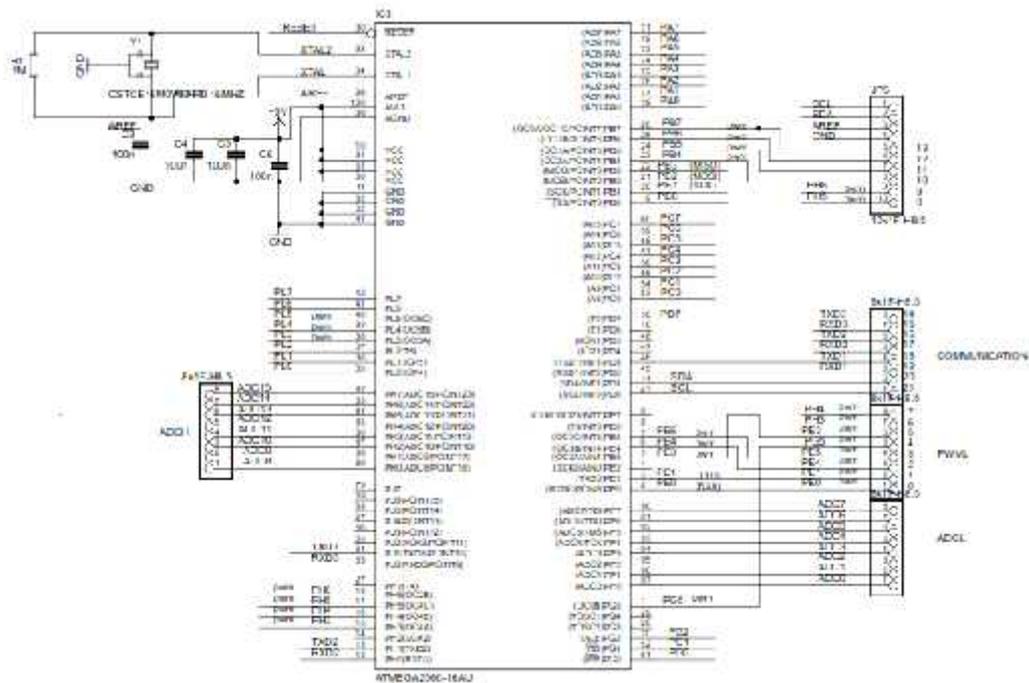
Gambar 2.16 Penulisan dan Pembacaan Data I2C

(<http://pima.kandangbuaya.com/2011/04/i2c-inter-ic-bus-twi/>)



2.5 Arduino Mega

Sistem Minimum ATMEGA 2560 :



Gambar 2.17 Sistem Minimum ATMEGA 2560

Arduino adalah *open-source platform* komputasi didasarkan pada i/o board sederhana dan lingkungan pengembangan yang mengimplementasikan bahasa Pemrograman. Arduino dapat digunakan untuk mengembangkan sendiri objek interaktif atau dapat dihubungkan ke perangkat lunak pada komputer (misalnya Flash, Processing, MaxMSP).

Arduino Mega 2560 merupakan board mikrokontroler berbasis ATMEGA2560. Modul ini memiliki 54 digital input/output di mana 14 digunakan untuk PWM output dan 16 digunakan sebagai analog input, 4 untuk UART, 16 MHz osilator kristal, koneksi USB, power jack, ICSP Header, dan tombol reset. Modul ini memiliki segalanya yang dibutuhkan untuk memrogram mikrokontroler seperti kabel USB dan sumber daya melalui adaptor ataupun battery.



Gambar 2.18 Arduino Mega

(<http://www.electroschematics.com/7963/arduino-mega-2560-pinout/>)

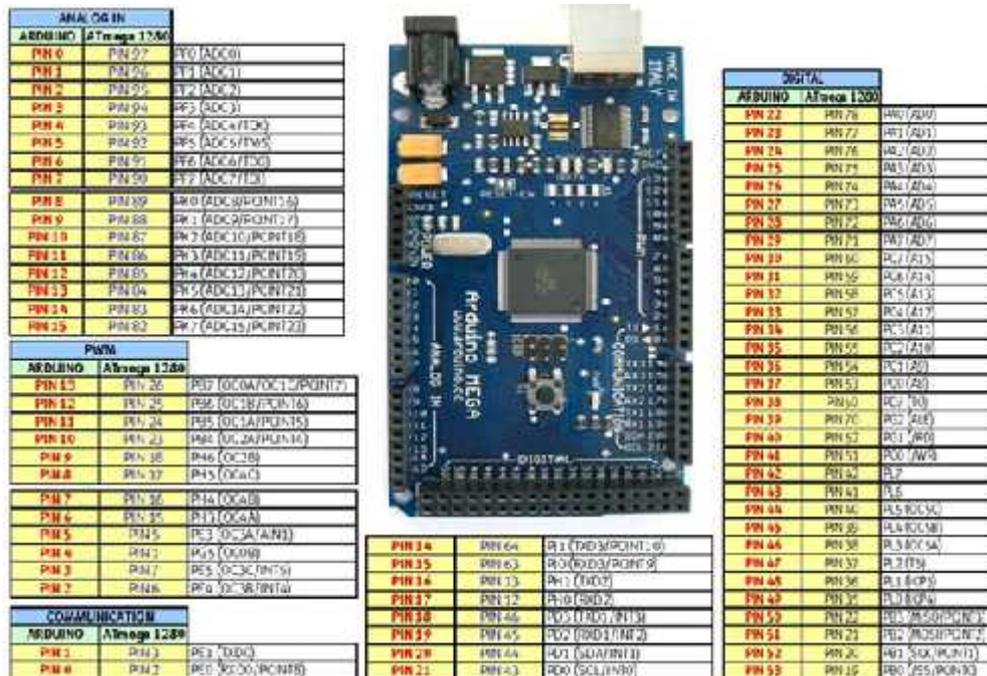
Board Arduino Mega memiliki fitur-fitur sebagai berikut:

- *Pinout 1.0*: ditambah pin SDA dan SCL yang dekat dengan pin AREF dan dua pin baru lainnya yang diletakkan dekat dengan pin *Reset*, IOREF yang memungkinkan *shield-shield* untuk menyesuaikan tegangan yang disediakan dari *board*. Untuk ke depannya, *shield* akan dijadikan kompatibel/cocok dengan *board* yang menggunakan AVR yang beroperasi dengan tegangan 5V dan *Arduino Mega* yang beroperasi dengan tegangan 3.3V.
- *Circuit RESET* yang lebih kuat.
- Atmega 16U2 menggantikan 8U2.



Tabel 2.1. Spesifikasi Arduino Mega

Mikrokontroler	Atmega2560
Tegangan pengoperasian	5V
Tegangan input	7-12V
Batas tegangan input	6-20V
Jumlah pin I/O digital	54 (15 di antaranya menyediakan keluaran PWM)
Jumlah pin input analog	16
Arus DC tiap pin I/O	40 mA
Arus DC untuk pin 3.3V	50 Ma
Memori Flash	256 KB (ATmega328), sekitar 8 KB digunakan oleh bootloader
SRAM	8 KB (Atmega2560)
EEPROM	4 KB (Atmega2560)
Clock Speed	16 MHz



Gambar 2.19 konfigurasi pin I/O Arduino Mega

(<http://forum.arduino.cc/index.php?topic=84002.0>)



2.5.1 Daya (Power)

Arduino Mega dapat diaktifkan melalui koneksi USB atau dengan sebuah *power supply external*. Sumber daya dipilih secara otomatis. *Supply external* (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau *battery*. Adaptor dapat dihubungkan dengan mencolokkan sebuah *center-positive plug* yang panjangnya 2,1 mm ke *power jack* dari *board*. Kabel *lead* dari sebuah *battery* dapat dimasukkan dalam *header/kepala pin Ground (Gnd)* dan pin *Vin* dari konektor *Power*.

Board Arduino Mega dapat beroperasi pada sebuah *supply external* 6 sampai 20 Volt. Jika disupply dengan yang lebih kecil dari 7 V, kiranya pin 5 Volt mungkin mensupply kecil dari 5 Volt dan *board Arduino Mega* bisa menjadi tidak stabil. Jika menggunakan *supply* yang lebih dari besar 12 Volt, *voltage regulator* bisa kelebihan panas dan membahayakan *board Arduino Uno*. Range yang direkomendasikan adalah 7 sampai 12 Volt.

Pin-pin daya yaitu sebagai berikut :

- **VIN.** Tegangan input ke *board Arduino Mega* ketika *board* sedang menggunakan sumber *supply external* (seperti 5 Volt dari koneksi USB atau sumber tenaga lainnya yang diatur). Kita dapat mensupply tegangan melalui pin ini, atau jika *supply* tegangan melalui *power jack*, aksesnya melalui pin ini.
- **5V.** Pin *output* ini merupakan tegangan 5 Volt yang diatur dari *regulator* pada *board*. *Board* dapat disuplai dengan salah satu *supply* dari DC *power jack* (7-12V), USB *connector* (5V), atau pin *VIN* dari *board* (7-12). *Supply* tegangan melalui pin 5V atau 3,3V *bypass regulator*, dan dapat membahayakan *board*. Hal itu tidak dianjurkan.
- **3V3.** Sebuah *supply* 3,3 Volt dihasilkan oleh *regulator* pada *board*. Arus maksimum yang dapat dilalui adalah 50 mA.
- **GND.** Pin *ground*.
- **I/OREF.** Pin ini di papan *Arduino* memberikan tegangan referensi dengan yang dioperasikan mikrokontroler. Sebuah perisai dikonfigurasi dengan benar dapat membaca pin tegangan *I/OREF* dan pilih sumber daya yang



tepat atau mengaktifkan penerjemah tegangan pada output untuk bekerja dengan 5V atau 3.3V.

2.5.2 Memori

Atmega2560 mempunyai 256 KB dengan 8 KB digunakan untuk *bootloader*.

2.5.3 Input dan Output

Setiap 54 pin digital pada *Arduino Mega* dapat digunakan sebagai *input* dan *output* yaitu menggunakan fungsi pin `Mode()`, `digital Write()`, dan `digital Read()`. Fungsi-fungsi tersebut beroperasi di tegangan 5 Volt. Setiap pin dapat memberikan atau menerima suatu arus maksimum 40 mA dan mempunyai sebuah *resistor pull-up* (terputus secara *default*) 20-50 K . Selain itu, beberapa pin mempunyai fungsi-fungsi spesial:

- **Serial: 0 (RX) dan 1 (TX).** Digunakan untuk menerima (RX) dan memancarkan (TX) serial data TTL (*Transistor-Transistor Logic*). Kedua pin ini dihubungkan ke pin-pin yang sesuai dari *chip* serial Atmega16U2 USB ke TTL.
- **External Interrupts: 2 dan 3.** Pin-pin ini dapat dikonfigurasi untuk dipicu sebuah *interrupt* (gangguan) pada sebuah nilai rendah, suatu kenaikan atau penurunan yang besar, atau suatu perubahan nilai.
- **PWM: 2-13 dan 44-46.** Memberikan 8-bit PWM *output* dengan fungsi `analogWrite()`.
- **SPI: 53 (SS), 51 (MOSI), 50 (MISO), 52 (SCK).** Pin-pin ini mendukung komunikasi SPI menggunakan SPI *library*.
- **LED: 13.** Ada sebuah LED yang terpasang, ketika pin bernilai *HIGH* dan led menyala, ketika pin bernilai *LOW* dan led mati.

Arduino Mega2560 mempunyai 16 input analog, diberi label A0 sampai A5, setiapnya memberikan 10 bit resolusi (contohnya 1024 nilai yang berbeda). Secara *default*, 6 input analog tersebut mengukur dari *ground* sampai tegangan 5 Volt, dengan itu mungkin untuk mengganti batas atas dari *range* dengan menggunakan pin AREF dan fungsi `analogReference()`.



2.5.4 Komunikasi

Arduino Mega mempunyai sejumlah fasilitas untuk komunikasi dengan sebuah komputer, *Arduino* lainnya atau mikrokontroler lainnya. Atmega 2560 menyediakan serial komunikasi UART TTL (5V), yang tersedia pada pin digital 0 (RX) dan 1 (TX). Sebuah Atmega 16U2 pada *channel board* serial komunikasinya melalui USB dan muncul sebagai sebuah port virtual ke *software* pada komputer. *Firmware* 16U2 menggunakan *driver* USB COM standar, dan tidak ada *driver* eksternal yang dibutuhkan. *Software Arduino Mega* mencakup sebuah serial *monitor* yang memungkinkan data tekstual terkirim ke *board Arduino Mega*. LED RX dan TX pada *board* akan menyala ketika data sedang ditransmit melalui *chip USB to serial* dan koneksi USB pada komputer.

2.5.5 Programming

Arduino Mega dapat diprogram dengan *software Arduino*. Pilih “*Arduino Mega*” dari menu **Tools** > **Board** (termasuk mikrokontroler pada *board*). Atmega2560 pada *Arduino Mega* dengan sebuah *bootloader* yang memungkinkan untuk mengupload kode baru ke Atmega2560 tanpa menggunakan pemrogram *hardware external*. Atmega2560 berkomunikasi menggunakan protokol STK500 asli. Dapat juga sebagai *bypass bootloader* dan program mikrokontroler melalui kepala/header ICSP (*In-Circuit Serial Programming*). ATmega16U2/8U2 diload dengan sebuah *bootloader* DFU, yang dapat diaktifkan dengan:

- Pada *board* Revisi 1: Dengan menghubungkan *jumper* solder pada belakang *board* dan kemudian mereset 8U2
- Pada *board* Revisi 2 atau setelahnya: Ada sebuah *resistor* yang menarik garis HWB 8U2/16U2 ke *ground*, dengan itu dapat lebih mudah untuk meletakkan ke dalam mode DFU. Kita dapat menggunakan *software* Atmel’s FLIP (Windows) atau pemrogram DFU (Mac OS X dan Linux) untuk meload sebuah *firmware* baru atau dapat menggunakan *header* ISP dengan sebuah pemrogram eksternal.



2.5.6 Reset Otomatis (*Software*)

Arduino Mega didesain pada sebuah cara yang memungkinkannya untuk direset dengan *software* yang sedang berjalan pada komputer yang sedang terhubung. Salah satu garis kontrol aliran *hardware* (DTR) dari ATmega8U2/16U2 dihubungkan ke garis reset dari Atmega2560 melalui sebuah kapasitor 100 nF. Ketika saluran ini dipaksakan (diambil rendah), garis reset jatuh cukup panjang untuk mereset chip. *Software Arduino Mega* menggunakan kemampuan ini untuk memungkinkan kita untuk mengupload kode dengan mudah menekan tombol *upload* di *software Arduino Mega*. Ini berarti bahwa *bootloader* dapat mempunyai sebuah batas waktu yang lebih singkat, sebagai penurunan dari DTR yang dapat menjadi koordinasi yang baik dengan memulai penguploadan. Pengaturan ini mempunyai implikasi, ketika *Arduino Mega* dihubungkan ke sebuah komputer lain yang sedang *running* menggunakan OS Mac X atau Linux, *Arduino Uno* mereset setiap kali sebuah koneksi dibuat dari *software* melalui USB. Untuk berikutnya, setengah detik atau lebih, *bootloader* sedang berjalan pada *Arduino Mega*. Ketika *Arduino Mega* diprogram untuk mengabaikan data yang cacat/salah untuk menahan beberapa bit pertama dari data yang dikirim ke *board* setelah sebuah koneksi dibuka. Jika sebuah *sketch* sedang berjalan pada *board* menerima satu kali konfigurasi atau data lain ketika *sketch* pertama mulai, memastikan bahwa *software* yang berkomunikasi menunggu satu detik setelah membuka koneksi dan sebelum mengirim data ini.

2.6 MOSFET

MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*) adalah suatu transistor dari bahan semikonduktor (silikon) dengan tingkat konsentrasi ketidakmurnian tertentu. Tingkat dari ketidakmurnian ini akan menentukan jenis transistor tersebut, yaitu transistor MOSFET tipe-N (NMOS) dan transistor MOSFET tipe-P (PMOS). Bahan silicon digunakan sebagai landasan (*substrat*) dari penguras (*drain*), sumber (*source*), dan gerbang (*gate*). Selanjutnya transistor dibuat sedemikian rupa agar antara substrat dan gerbangnya dibatasi oleh oksida silikon yang sangat tipis. Oksida ini diendapkan



di atas sisi kiri dari kanal, sehingga transistor MOSFET akan mempunyai kelebihan dibanding dengan transistor BJT (*Bipolar Junction Transistor*), yaitu menghasilkan disipasi daya yang rendah.

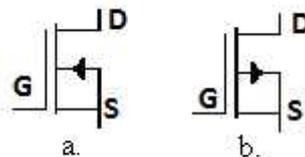
2.6.1 Jenis-jenis MOSFET

MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*) adalah suatu transistor dari bahan semikonduktor (silikon) dengan tingkat konsentrasi ketidakmurnian tertentu. Tingkat dari ketidakmurnian ini akan menentukan jenis transistor tersebut, yaitu transistor MOSFET tipe-N (NMOS) dan transistor MOSFET tipe-P (PMOS). Bahan silicon ini yang akan digunakan sebagai landasan (*substrat*) penguras (*drain*), sumber (*source*), dan gerbang (*gate*). Selanjutnya transistor ini dibuat sedemikian rupa agar antara substrat dan gerbangnya dibatasi oleh oksida silicon yang sangat tipis. Oksida ini diendapkan di atas sisi kiri kanal, sehingga transistor MOSFET akan mempunyai kelebihan dibanding dengan transistor BJT (*Bipolar Junction Transistor*), yaitu menghasilkan disipasi daya yang rendah.

Bila dilihat dari cara kerjanya, transistor MOS dapat dibagi menjadi dua, yaitu:

1) Transistor Mode Pengosongan (*Transistor Mode Depletion*)

Pada transistor mode depletion, antara drain dan source terdapat saluran yang menghubungkan dua terminal tersebut, dimana saluran tersebut mempunyai fungsi sebagai saluran tempat mengalirnya elektron bebas. Lebar dari saluran itu sendiri dapat dikendalikan oleh tegangan gerbang. Transistor MOSFET mode pengosongan terdiri dari tipe-N dan tipe-P, simbol transistor ditunjukkan dalam Gambar .



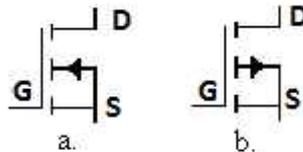
Gambar 2.20 Simbol Transistor MOSFET Mode Depletion

(a). N-Channel Depletion (b). P-Channel Depletion



2) Transistor Mode peningkatan (Transistor Mode Enhancement)

Transistor mode enhancement ini pada fisiknya tidak memiliki saluran antara drain dan sourcena karena lapisan bulk meluas dengan lapisan SiO₂ pada terminal gate. Transistor MOSFET mode peningkatan terdiri dari tipe-N dan tipe-P, simbol transistor ditunjukkan dalam Gambar 2.21.



Gambar 2.21 Simbol Transistor MOSFET Mode *Enhancement*

(a). *N-Channel Enhancement* (b). *P-Channel Enhancement*

2.7 Motor DC

Motor DC merupakan motor yang paling sederhana untuk pengaktifannya. Hanya dengan memberikan tegangan DC, motor ini akan berutar secara kontinyu kearah tertentu. Membalik arah putaran motor dapat dilakukan dengan mengubah polaritas arus yang mengalir pada motor. Motor DC biasanya mempunyai kecepatan putar yang cukup tinggi dan sangat cocok digunakan untuk roda robot yang membutuhkan kecepatan gerak yang tinggi.

Motor DC memerlukan suplai tegangan arus searah ppada kumparan medan untuk diubah menjadi energi gerak ekanik. Kumparan medan pada motor DC disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Motor arus searah, sebagaimana namanya, menggunakan arus langsung yang tidak langsung/direct-undirectional.



Gambar 2.22 Motor DC

(http://www.mind.ilstu.edu/curriculum/medical_robotics/motors.php)

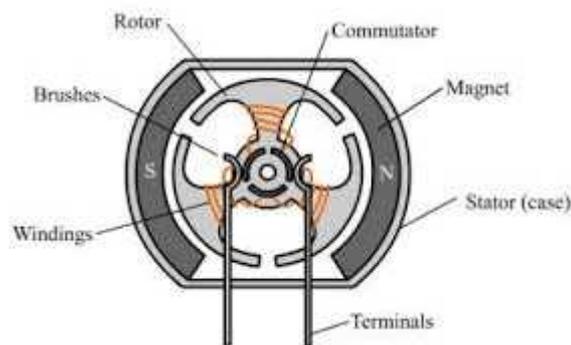


Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur tegangan *dynamo* yaitu dengan meningkatkan tegangan dinamo akan meningkatkan kecepatan dan mengatur arus medan dengan menurunkan arus medan akan meningkatkan kecepatan.

2.7.1 Konstruksi Motor DC.

Bagian-bagian yang penting dari motor dc dapat ditunjukkan pada gambar 2.23, dimana stator mempunyai kutub yang menonjol dan dililit oleh kumparan medan. Pembagian dari fluks yang terdapat pada daerah celah udara yang dihasilkan oleh lilitan medan secara simetris yang berada disekitar daerah tengah kutub kumparan medan.

Kumparan penguat dihubungkan secara seri, letak kumparan jangkar berada pada slot besi yang berada disebelah luar permukaan jangkar. Pada jangkar terdapat komutator yang berbentuk silinder dan isolasi sisi kumparan yang dihubungkan dengan komutator pada beberapa bagian yang berbeda sesuai dengan jenis belitan.



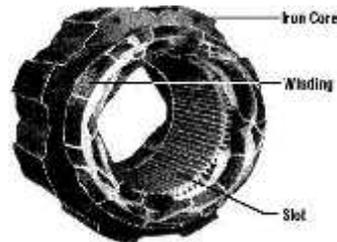
Gambar 2.23 Konstruksi Motor DC

(Sumber : <http://depokinstruments.com/>)



a. Stator Motor DC

Stator merupakan bagian dari motor yang permanen atau tidak berputar. Bagian ini menghasilkan medan magnet, baik yang dihasilkan dari koil (elektromagnetik), maupun dari magnet.



Gambar 2.24 Konstruksi Bagian Stator Motor

(Sumber : <http://www.wisdompage.com/>)

b. Rotor Motor DC

Fungsi dari rotor yaitu untuk mengubah energi listrik menjadi energi mekanik dalam bentuk gerak putar. Rotor terdiri dari poros baja dimana tumpukan keping-keping inti yang berbentuk silinder dijepit. Pada inti terdapat alur-alur dimana lilitan rotor diletakkan. Suatu kumparan motor akan berfungsi apabila mempunyai kumparan medan, kumparan tersebut berfungsi sebagai penghasil medan magnet..



Gambar 2.25 Konstruksi Rotor Motor DC

(Sumber : <http://www.diracdelta.co.uk>)



c. Komutator

Konstruksi dari komutator terdiri dari lamel-lamel, antar lamel dengan lamel lainnya diisolasi dengan *mica*. Gambar 2.26 merupakan gambar komutator pada motor DC.



Gambar 2.26 Konstruksi Komutator Motor DC

(Sumber : <http://1.bp.blogspot.com>)

2.8 Motor Servo

Motor servo adalah sebuah motor dengan sistem closed feedback di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor, serangkaian gear, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Tampak pada gambar dengan pulsa 1.5 mS pada periode selebar 2 mS maka sudut dari sumbu motor akan berada pada posisi tengah. Semakin lebar pulsa OFF maka akan semakin besar gerakan sumbu ke arah jarum jam dan semakin kecil pulsa OFF maka akan semakin besar gerakan sumbu ke arah yang berlawanan dengan jarum jam.

2.8.1 Motor Servo Continuous

Motor servo jenis ini mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).



Gambar 2.27 Motor Servo Continuous

(<http://www.scribd.com/doc/225500394/Jurnal>)

2.8.2 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° .



Gambar 2.28 Motor Servo Standar Anonim

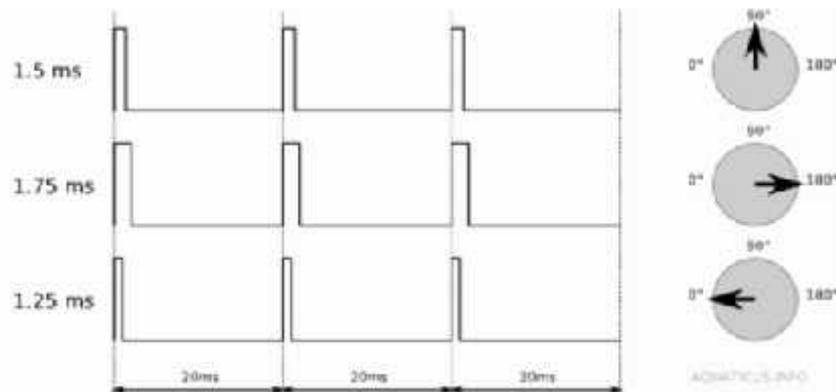
(<http://electrosome.com/interfacing-servo-motor-with-atmega32-microcontroller/>)

2.8.3 Prinsip Kerja Motor Servo

Prinsip utama dari pengendalian motor servo adalah pemberian nilai PWM pada kontrolnya. Frekuensi PWM yang digunakan pada pengontrol motor servo selalu 50 Hz sehingga pulsa dihasilkan setiap 20 ms. Lebar pulsa akan menentukan posisi servo yang dikehendaki. Pemberian lebar pulsa 1,5 ms akan membuat



motor servo berputar ke posisi netral (90 derajat), lebar pulsa 1,75 ms akan membuat motor servo berputar mendekati posisi 180 derajat, dan dengan lebar pulsa 1,25 ms motor servo akan bergerak ke posisi 0 derajat. Gambar 2.29 berikut memperlihatkan hubungan antara lebar pulsa PWM dengan arah putaran motor servo.



Gambar 2.29 Hubungan Lebar Pulsa PWM dengan Arah Putaran Motor Servo

(<http://dosen.narotama.ac.id/wp-content/uploads/2012/01/Sejarah-Robot.pdf>)