

BAB II DASAR TEORI

2.1. Tinjauan Pustaka

Tinjauan Pustaka ini mencakup beberapa referensi yang telah dievaluasi terkait perancangan sistem lengan robot 4 DOF (Degree Of Freedom) untuk soldering iron berbasis Arduino Mega 2560 dengan sistem kendali melalui HMI Nextion dan KY-040 dalam rangka pembuatan Proyek Tugas Akhir. Hasil tinjauan Pustaka dapat dilihat pada tabel 2.1, sebagai berikut.

Tabel 2. 1 Tinjauan Pustaka

No.	Penulis	Judul	Keterangan
1	Jyotindra Narayan dan Ashish Singla[8]	<i>ANFIS Based Kinematic Analysis of a 4-DOFs SCARA Robot</i>	Referensi metode <i>forward kinematic</i>
2	Tirsi & Çetin[9]	<i>Model-based Nonlinear Control with Kinematic and Dynamic Analysis of 4-DOF SCARA Robot</i>	Referensi untuk sistem control untuk posisi x, y, z, dan <i>end-effector</i>
3	Sanjeewa et al.[10]	<i>Real-Time Coordinate Estimation for SCARA Robots in PCB Repair Using Vision and Laser Triangulation</i>	Referensi untuk sistem kontrol berbasis <i>real-time coordinate estimation</i> untuk robot SCARA

2.2. SCARA (*Selective Compliance Assembly Robot Arm*)

Robot SCARA (*Selective Compliance Assembly Robot Arm*) merupakan salah satu jenis robot industri yang dirancang khusus untuk aplikasi perakitan dan pemindahan komponen secara presisi. SCARA memiliki keunikan pada strukturnya yang kaku terhadap gerakan vertikal (sumbu Z) namun fleksibel pada bidang horizontal (sumbu X dan Y), sehingga sangat ideal untuk tugas-tugas *pick and place* berkecepatan tinggi, perakitan komponen elektronik, dan proses inspection dalam jalur produksi. Konsep SCARA pertama kali diperkenalkan di Jepang pada tahun

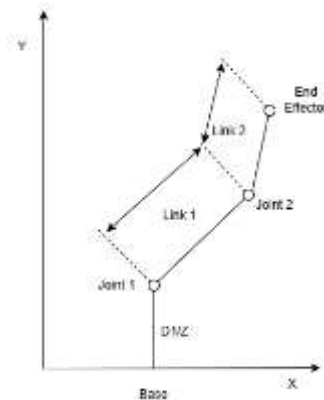
1979 dan menjadi solusi inovatif dalam dunia otomasi industri, khususnya dalam lini perakitan elektronik dan manufaktur ringa [7].

SCARA umumnya memiliki empat derajat kebebasan (4 *DOF*), yang terdiri dari dua sumbu rotasi di bidang *horizontal*, satu aktuator *linier vertikal*, dan satu sumbu rotasi di *end-effector*. Keempat sumbu ini dikendalikan oleh sistem aktuator berbasis servo atau *stepper motor*, yang diatur melalui kontroler seperti PLC, mikrokontroler, atau sistem komputer industri.

2.3. Metode *Forward Kinematik*

Kinematika robot adalah studi geometris mengenai hubungan antara parameter sendi (sudut dan translasi) dan posisi *end-effector*, tanpa memperhitungkan gaya atau massa. *SCARA* memiliki struktur dua sendi rotasi horizontal (θ_1, θ_2), satu sendi rotasi tambahan (θ_3), dan satu sendi *liniar vertikal* (D_1), sehingga cocok untuk aplikasi perakitan yang menuntut presisi tinggi pada bidang *horizontal* serta kekakuan *vertikal*[11].

Forward kinematic adalah metode untuk menghitung posisi dan orientasi *end-effector* robot berdasarkan parameter sendi yang diketahui seperti sudut rotasi dan panjang lengan. Metode ini sangat penting dalam sistem robot *SCARA* karena memungkinkan sistem kendali untuk memetakan gerakan motor ke koordinat spasial yang diinginkan[12]. *SCARA* memiliki struktur dua sendi rotasi sejajar (di bidang X-Y) dan satu translasi vertikal, sehingga cocok untuk aplikasi perakitan berkecepatan tinggi dengan gerakan planar yang stabil [10]. *Forward kinematics* biasanya dirumuskan dengan pendekatan Denavit–Hartenberg (DH), yang menyusun parameter geometri antar link secara sistematis untuk menghasilkan transformasi homogen 4x4. Pada robot *SCARA 4-DOF*, posisi *end-effector* dalam koordinat kartesian (x,y,z) dapat dilihat pada Gambar 2.1, sebagai berikut:



Gambar 2.1 Skema Geometri Antar Link SCARA 4 DOF

Pada **Gambar 2.1**, menunjukkan skema geometri antar *link*, untuk konfigurasi pergerakan *link* didapati persamaan sebagai berikut :

$$x_1 = L1 \cos\theta_2 \dots\dots\dots(2.1)$$

$$z_1 = L1 \sin\theta_2 \dots\dots\dots(2.2)$$

Dan pada link 2, konfigurasi pergerakannya mengikuti pergerakan dari *link* 1 yaitu menggunakan sudut total $(\theta_2 + \theta_3)$, maka didapati persamaan pada *link* 2 sebagai berikut :

$$x_2 = L2 \cos(\theta_2 + \theta_3) \dots\dots\dots(2.3)$$

$$z_2 = L2 \sin(\theta_2 + \theta_3) \dots\dots\dots(2.4)$$

Setelah mendapati persamaan kedua *link*, maka kedua persamaan tersebut dijumlahkan untuk menentukan persamaan posisi total pada bidang kerja lengan robot sebagai berikut:

$$R = L1 \cos\theta_2 + L2 \cos(\theta_2 + \theta_3) \dots\dots\dots(2.5)$$

$$Z = L1 \sin\theta_2 + L2 \sin(\theta_2 + \theta_3) \dots\dots\dots(2.6)$$

Dikarenakan R merupakan jarak radial, maka R akan dirubah terlebih dahulu ke *koordinat kartesian* sebagai berikut:

$$x = L1 \cos\theta_2 + L2 \cos(\theta_2 + \theta_3)\cos\theta_1 \dots\dots\dots(2.7)$$

$$y = L1 \cos\theta_2 + L2 \cos(\theta_2 + \theta_3)\sin\theta_1 \dots\dots\dots(2.8)$$

Dengan perumusan persamaan mengikuti persamaan trigonometri , maka didapati persamaan yang diterapkan pada metode *forward kinematic* sebagai berikut:

$$x = (L_1 \cos\theta_2 + L_2 \cos(\theta_2 + \theta_3)) \cos\theta_1 \dots\dots\dots (2.9)$$

$$y = (L_1 \cos\theta_2 + L_2 \cos(\theta_2 + \theta_3)) \sin\theta_1 \dots\dots\dots (2.10)$$

$$z = D1 + L_1 \sin\theta_2 + L_2 \sin\theta_2 + \theta_3) \dots\dots\dots (2.11)$$

Persamaan ini menunjukkan bahwa posisi *horizontal* (x,y) bergantung pada panjang dan sudut rotasi kedua lengan, serta arah dasar rotasi dari sendi pertama θ_1 [8]. Sementara itu, nilai koordinat vertikal z dipengaruhi oleh elevasi akibat rotasi sendi dan translasi aktuator linier $D1$ [10]. Hasil dari *forward kinematics* ini digunakan untuk memverifikasi lintasan gerakan robot atau untuk mengecek validitas solusi dari *inverse kinematics* dalam simulasi atau kontrol *real-time* [12].

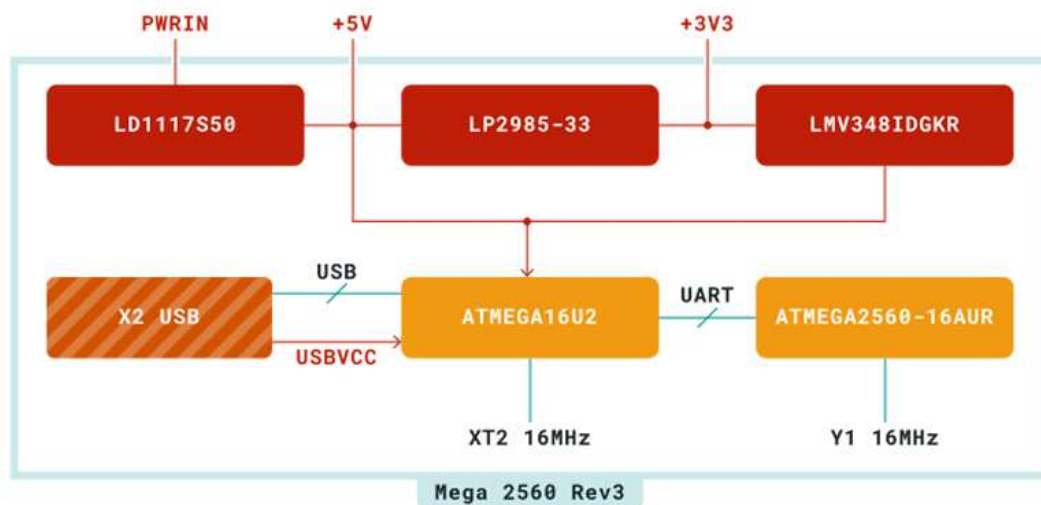
2.4. Arduino Mega 2560

Arduino Mega 2560 adalah mikrokontroler yang diperuntukkan untuk membangun sistem komunikasi yang kompleks jika dibandingkan dengan mikrokontroler lainnya oleh Arduino. Mikrokontroler berbasis ATmega2560-16AU, yang beroperasi pada frekuensi 16 MHz. dengan isolator. Catu daya dikontrol melalui regulator LD1117S50 untuk +5V dan LP298522 untuk 3,3V, memungkinkan *input* dari USB atau sumber eksternal untuk komunikasi dengan mikrokontroler ini. Pada **Gambar 2.1** dapat dilihat mikrokontroler ini menggunakan ATmegaU2 sebagai antarmuka USB to UART, yang melakukan penerjemahan data dari sinyal USB menjadi komunikasi serial (UART). Dan mikrokontroler ini memiliki 54 pin *input/output* digital, 16 *input analog*, 4 UART (port serial perangkat keras), koneksi USB, *input* daya, *header* ICSP, dan tombol reset. [13]



Gambar 2. 2 Arduino Mega 2560 [14]

Pada gambar 2.4 merupakan struktur dari Arduino Mega 2560, pada Arduino Mega 2560 ini terdapat sistem yang digunakan untuk melakukan komunikasi antara mikrokontroler dengan aktuator maupun sensor yang dapat dilihat pada **Gambar 2.3** berikut.



Gambar 2. 3 Diagram Blok Arduino Mega 2560[15]

Pada **Gambar 2.3** mikrokontroler Arduino Mega 2560 merupakan *Integrated Circuit (IC)* yang terdiri dari *Central Processing Unit (CPU)*, *Read Only Memory (ROM)*, *Random Access Memory (RAM)*, dan *Input/Output*. Sehingga mikrokontroler ini dapat melakukan pemrosesan data melalui CPU dengan memory yang telah diintegrasikan pada sistem yang akan dijalankan.

Selain itu, heatsink besar juga ditambahkan pada MOSFET untuk heatbed, guna mendukung beban daya tinggi hingga 270 W untuk menghindari *overheating* [16].

Secara spesifikasi, RAMPS 1.6 mendukung hingga lima modul driver motor stepper seperti A4988, yang digunakan untuk mengontrol motor-motor penggerak pada sumbu X, Y, Z (ganda), dan extruder. Sistem ini mendukung konfigurasi microstepping hingga 1/128, tergantung jenis driver yang digunakan, sehingga dapat menghasilkan pergerakan yang halus dan presisi tinggi. Di samping itu, board ini menyediakan konektor untuk 6 buah endstop digital, 3 sensor suhu termistor, serta 3 jalur *output* MOSFET untuk mengontrol pemanas *hotend*, fan pendingin, dan heatbed yang dapat dilihat pada **Tabel 2.2**. Board juga kompatibel dengan berbagai *firmware open-source* seperti Marlin, Repetier, dan Klipper, menjadikannya fleksibel untuk berbagai konfigurasi dan kebutuhan pengembangan[17].

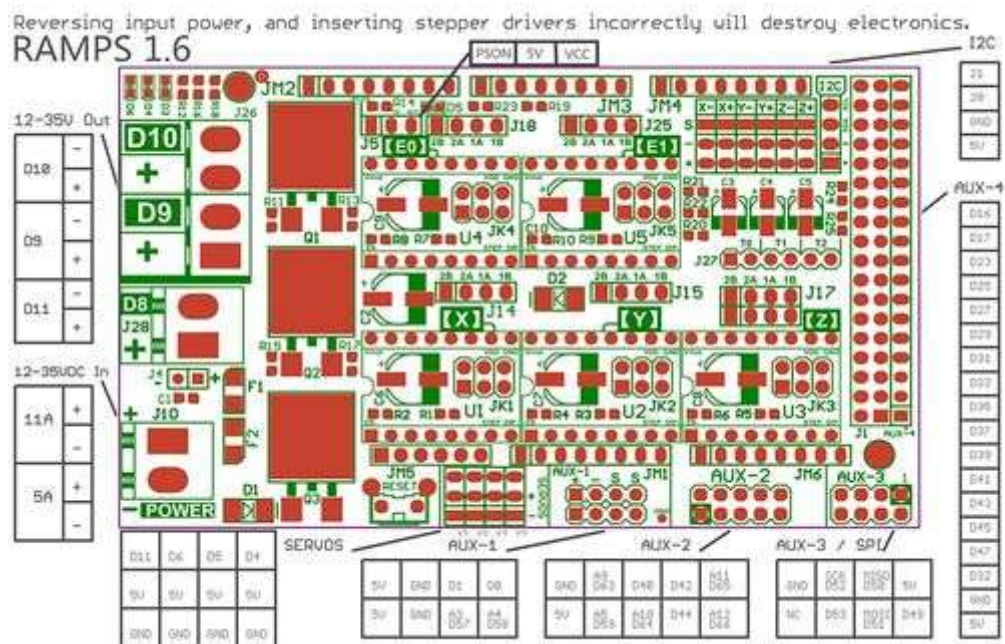
Tabel 2. 2 Spesifikasi RAMPS 1.6[16]

Spesifikasi	Keterangan
Tegangan <i>Input</i>	12V DC
Daya Maksimal	+/- 200W
Jumlah Motor Driver	Maksimal 5 Driver
Microstep	1/128 Step
Enstop/Sensor	6 <i>limit switch</i> , 3 Termistor
<i>Output</i> Mosfet	3 Channel (<i>HotEnd</i> , <i>Fan</i> , <i>Heatbed</i>)
Fuse Proteksi	5A (Sirkuit Umum, 20A (<i>Headbed</i>))
Dimensi	102 X 68 X 29 mm
Kompatibilitas	Arduino Mega2560

Sistem kerja RAMPS 1.6 dirancang untuk dapat terhubung dengan Arduino mega 2560, di mana Arduino Mega menangani pemrosesan data, sedangkan RAMPS bertugas mendistribusikan sinyal dan daya ke komponen utama seperti motor, sensor, dan aktuator. Jalur komunikasi antara Arduino Mega dan RAMPS terdiri dari pin digital dan *analog*. *Input* tegangan disuplai melalui terminal utama 12–24V DC, yang kemudian dibagi ke jalur-jalur MOSFET melalui fuse pelindung sebesar 5 A untuk sirkuit logika dan hingga 20 A untuk sirkuit pemanas *heatbed*. Hal ini meningkatkan keamanan sistem, khususnya dalam aplikasi berbeban besar

seperti pemanasan simultan pada *nozzle* dan *bed*. Selain itu, RAMPS 1.6 juga menyediakan header ekspansi untuk modul LCD, sensor berbasis I2C/SPI, dan servo motor, memungkinkan pengembangan lebih lanjut untuk sistem otomatisasi seperti robot SCARA berbasis Arduino[18].

Dalam implementasinya, RAMPS 1.6 banyak digunakan oleh peneliti dan pengembang sistem robotika karena kemudahan konfigurasi, ketersediaan dokumentasi luas, serta kompatibilitas tinggi dengan sistem kontrol berbasis G-code. Dalam jurnal oleh Zavar et al. (2023), RAMPS digunakan untuk mengatur gerakan presisi pada lengan robot SCARA 4 DOF berbasis Arduino dengan motor stepper A4988. Penelitian tersebut menunjukkan bahwa board ini mampu menangani beban sistem sensor dan aktuator dengan stabil.



Gambar 2.5 Pin I/O RAMPS 1.6[19]

Pada **Gambar 2.5** merupakan struktur yang terdapat pada RAMPS 1.6 yang berfungsi sebagai *input* dan *output* guna menjalankan sistem yang terdapat pada RAMPS 1.6. RAMPS 1.6 terdiri dari beberapa *output* sensor maupun *actuator*, dengan *input* sebesar 12V DC yang dapat dilihat pada **Tabel 2.3** sebagai berikut.

Tabel 2. 3 Pin I/O Pada RAMPS 1.6[19]

Fungsi	Pin RAMPS 1.6	Pin Arduino Mega
X-Step	Step 1	D26
X-DIR	DIR 1	D28
X-Enable	Enable 1	D24
Y-Step	Step 2	D38
Y-DIR	DIR 2	D40
Y-Enable	Enable 2	D36
Z-Step	Step 3	D44
Z-DIR	DIR 3	D46
Z-Enable	Enable 3	D42
<i>E0-Axis</i>		
E0-Step	Step 4	D34
E0-DIR	DIR 4	D36
E0-Enable	Enable 4	D30
<i>E1-Axis</i>		
E1-Step	Step 5	D37
E1-DIR	DIR 5	D35
E1-Enable	Enable 5	D31
Hotend Heater	HEATER0	D10
Heated Bed	HEATER1	D8
Cooling Fan	FAN	D9
Thermistor 0	T0	A0
Thermistor 1	T1	A1
Thermistor 2	T2	A2
Thermistor 3	T3	A3
X-Min	X-Stop	D3
X-Max	X-Stop (Shared)	D3
Y-Min	y-Stop	D14
Y-Max	y-Stop (Shared)	D14
Z-Min	Z-Stop	D18
Z-Max	Z-Stop (Shared)	D18
Servo (Z-Probe)	Servo0	D11
LCD	EXP1 dan EXP2	D16, D17
<i>Thermocouple</i>	<i>Thermocouple</i>	D50, D51, D52, D53
Power	Power In	12V/24V
USB	-	Via Arduino Mega
Serial	TX0/RX0	D0/D1

2.6. HMI Nextion

HMI Nextion NX8048P050 merupakan HMI (*Human Machine Interface*) berbasis TFT LCD yang telah dilengkapi dengan mikrokontroler internal untuk pengelolaan tampilan dan interaksi. Nextion merupakan HMI yang menggabungkan prosesor internal dan layer sentuh memori dengan perangkat lunak Nextion Editor untuk pengaturan *interface* HMI GUI (*Graphical User Interface*)nya. Pada Nextion menggunakan sistem *drag and drop* komponen dan interaksi teks ASCII untuk konfigurasi komponen berintegrasi pada antarmuka HMI. HMI Nextion terhubung ke MCU peripheral melalui UART untuk melakukan pemrosesan data[20].

Fungsi HMI Nextion ini dalam sistem mikrokontroler adalah sebagai antarmuka tampilan interaktif untuk menampilkan data sensor, status sistem, dan memberikan perintah *input* dari pengguna ke sistem kontrol. Nextion NX8048T070 memiliki ukuran layar 5 inci dengan resolusi 800X480 *pixel*, dengan kecepatan *baud rate* mulai dari 9600 hingga 115200 bps[20]. HMI ini bekerja pada tegangan 5V DC dan dilengkapi dengan slot micro *SD Card* yang digunakan untuk membuat tampilan antarmuka menggunakan *software Nextion Editor*. Terkait spesifikasi HMI Nextion dapat dilihat pada **Tabel 2.4** sebagai berikut.

Tabel 2. 4 Spesifikasi HMI NEXTION[20]

Spesifikasi	Data	Keterangan
Warna	65K 65536 warna	16 bit 565, 5R-6G-5B
Ukuran	136.9mm(L)×84mm(W)×5.8mm(H)	NX8048P050
Area visual	120.7mm(L)×75.8mm(W)	
Area aktif	108.00mm(L)×64.80mm(W)	
Resolusi	800X480 <i>pixel</i>	
Jenis sentuh	Resistif	
<i>Touches</i>	>1 juta	
<i>Backlight</i>	LED	
<i>Backlight Lifetime</i> (rata-rata)	>30.000 jam	
Kecerahan	300 nit	0% - 100%, interval penyesuaian 1%
Berat	130g	

2.7. Stepper Motor Nema 17HS3401

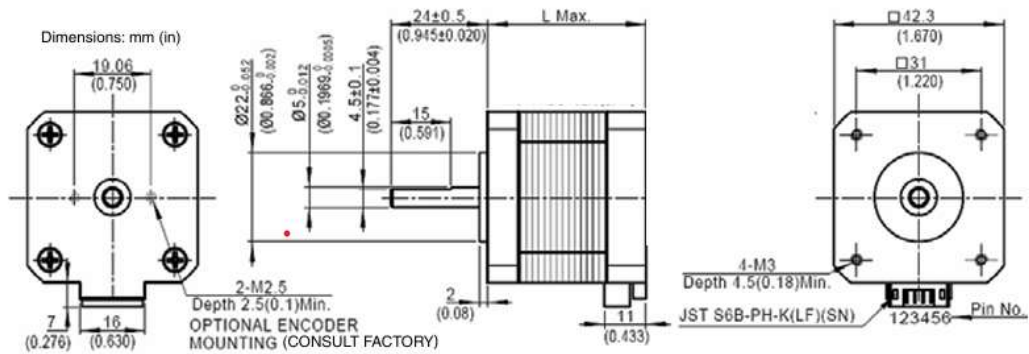
Stepper motor Nema 17HS3401 merupakan *motor stepper nema 17* tipe *bipolar hybrid*. Stepper motor ini merupakan jenis motor listrik yang bekerja berdasarkan prinsip elektromagnetik untuk menghasilkan gerakan sudut (rotasi) per Langkah. Stepper motor 17HS3401 memiliki dua coil yang bekerja untuk mengalirkan arus bolak balik ke masing-masing coil untuk memutar rotor. Stepper motor ini diatur menggunakan driver motor A4988 untuk memberikan sinyal control dan pengaturan arus ke motor[21].

Penamaan stepper motor ini sesuai dengan spesifikasi motor yang digunakan, yang mana penamaan stepper motor 17HS3401 yaitu “17” berarti stepper motor memiliki ukuran 1.7 inch atau 42mm, “HS” berarti hybrid stepper, “34” mengartikan Panjang bodi motor sebesar 34mm, dan 01 konfigurasi merupakan kode konfigurasi motor[21]. Terkait spesifikasi motor keseluruhan dapat dilihat pada **Tabel 2.5** sebagai berikut:

Tabel 2. 5 Spesifikasi Stepper Motor Nema 17HS3401[21]

Spesifikasi	Keterangan
Jenis motor	Bipolar hybrid stepper
Ukuran flense	42 X 42 mm
Panjang bodi	34 mm
Sudut Langkah	1,8° per Langkah (200 langkah/putaran)
Arus per fase	1,3A
Tegangan per fase	3,1V
Resistansi per fase	2.4Ω
Induktansi per fase	3,2Mh
Torsi tahan	±26Ncm (≈0,26 Nm atau 2.6 kg.cm)
Torsi detent	±1,1 N.cm
Jumlah kabel	4 kabel (bipolar)

Stepper motor ini memiliki panjang bodi sebesar 32mm, dengan sudut langkah 1,8°, yang mana menciptakan 200 langkah untuk satu putaran penuh (360°). Stepper motor nema 17HS3401 dengan dimensi 42mm X 42mm pada bagian flensanya yang dapat dilihat pada **Gambar 2.6** sebagai berikut :



Gambar 2. 6 Dimensi Stepper Motor Nema 17HS3401[21]

Stepper motor bekerja dengan mengeluarkan torsi dari putar rotor motor stepper. Torsi merupakan besaran kemampuan motor untuk memutar beban terhadap poros rotor. Besaran torsi dipengaruhi oleh desain motor, arus *input*, dan metode pengendalian (*full step*, *half step*, dan *mikrostepping*). Pada stepper motor nema 17HS3401 memiliki *holding torque* sebesar 0,26 Nm, sehingga motor mampu menahan beban sebanding dengan gaya 2,6kg yang bekerja pada rotor sepanjang 1 cm dari pusat porosnya.

Dalam sistem aktuator untuk lengan robot, perhitungan torsi diperlukan guna memastikan kapasitas kemampuan motor dalam menggerakkan lengan robot dengan beban yang aman dan efisien. Salah satu metode transmisi yang biasa digunakan adalah sistem pully, dimana gaya poros motor diteruskan melalui perbandingan diameter roda penggerak (*pulley driver*) dengan roda yang digerakan (*pulley driven*). Perhitungan torsi motor dengan menggunakan transmisi metode pully adalah menghitung rasio transmisi pully menggunakan persamaan sebagai berikut.

$$n = \frac{r_{pully\ driven}}{r_{pully\ driver}} \dots\dots\dots (2.12)$$

Rasio ini menunjukkan seberapa banyak torsi yang akan dikalikan atau kecepatan yang akan dibagi oleh sistem. Setelah itu, torsi *output* dari sistem transmisi pully dihitung dengan memperhatikan efisiensi mekanis dari tiap tahap transmisi menggunakan persamaan sebagai berikut:

$$\tau_{ouput} = \tau_m \times n \times \eta \dots\dots\dots (2.13)$$

Dimana τ_m adalah torsi awal motor, n adalah rasio pully, dan η adalah efisiensi transmisi. Nilai efisiensi biasanya berada pada 0.8 hingga 0.95 tergantung pada gesekan dan keausan sistem.

Untuk menentukan apakah torsi *output* mencukupi, maka dihitung terlebih dahulu gaya total yang bekerja pada beban. Sama seperti sistem linier lainnya, gaya total terdiri dari gaya gesekan dan gaya percepatan, yang dihitung menggunakan persamaan sebagai berikut:

$$F = F_{\text{gesek}} + F_{\text{akselerasi}} \dots \dots \dots (2.14)$$

Dengan gaya gesek dihitung menggunakan persamaan sebagai berikut:

$$F_{\text{gesek}} = \mu \times m \times g \dots \dots \dots (2.15)$$

Dan gaya percepatan dihitung menggunakan menggunakan persamaan (18) sebagai berikut:

$$F_{\text{akselerasi}} = m \times a \dots \dots \dots (2.16)$$

Setelah mendapatkan total gaya (F), maka torsi yang dibutuhkan oleh sistem untuk menggerakkan beban dapat dihitung menggunakan persamaan sebagai berikut:

$$\tau_{\text{dibutuhkan}} = F \times r \dots \dots \dots (2.17)$$

Dengan r adalah jari-jari *pulley driver* atau titik aplikasi gaya terhadap poros rotasi. Sebagai langkah akhir, dilakukan analisis margin keamanan terhadap sistem, dengan membandingkan torsi *output* sistem terhadap torsi minimum yang dibutuhkan, yang dihitung menggunakan persamaan sebagai berikut:

$$\text{Margin} = \frac{\tau_{\text{dibutuhkan}}}{\tau_{\text{output}}} \dots \dots \dots (2.18)$$

Jika margin yang diperoleh lebih besar dari satu, sistem dinyatakan aman dan motor memiliki cadangan torsi untuk mengatasi beban dinamis, gesekan tambahan, atau variasi massa. Margin yang lebih tinggi juga mengindikasikan motor akan bekerja pada kondisi tidak jenuh, yang dapat memperpanjang umur operasional motor dan sistem secara keseluruhan.

Dan ketika penggunaan stepper motor untuk menggerakkan *lead screw*, yang pertama dilakukan adalah mengukur torsi pada *lead screw* menggunakan persamaan sebagai berikut:

$$\tau = \frac{F \times \text{Lead}}{2\pi \times \eta} \dots \dots \dots (2.19)$$

Setelah melakukan pengukuran torsi pada *lead screw*, selanjutnya mengukur momen inersia pada *lead screw* menggunakan persamaan berikut:

$$J_{\text{screw}} = \frac{1}{2} m_{\text{screw}} \times r^2 \dots \dots \dots (2.20)$$

Selanjutnya melakukan pengukuran pada percepatan linear dengan sudut pada *lead screw* menggunakan persamaan berikut:

$$\alpha = \frac{a}{\text{lead}/2\pi} \dots \dots \dots (2.21)$$

Setelah mengetahui percepatan linear pada sudut *lead screw*, maka dapat dihitung torsi inersia menggunakan persamaan berikut:

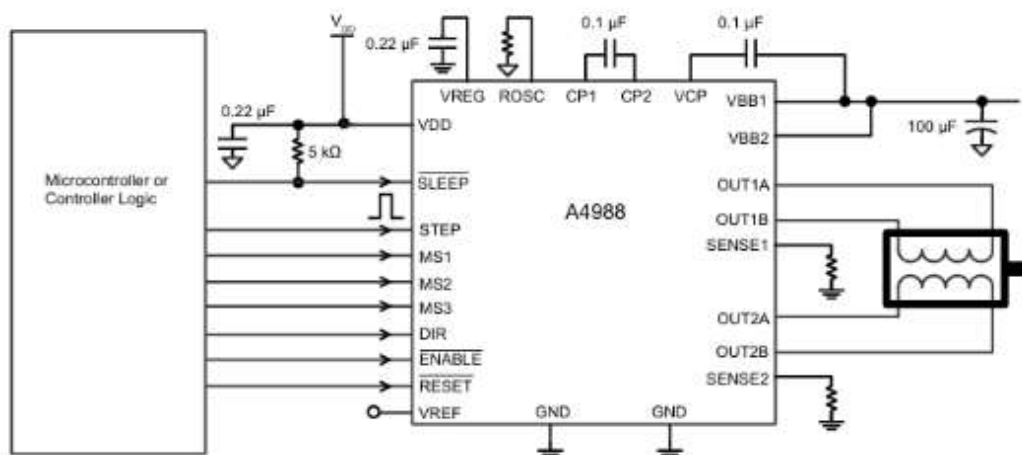
$$\tau_{\text{rotasi}} = J_{\text{screw}} \times \alpha \dots \dots \dots (2.22)$$

Setelah mengetahui besaran torsi linear dan torsi rotasi pada *lead screw* maka dapat dilakukan perhitungan torsi motor untuk menggerakan *lead screw* menggunakan persamaan sebagai berikut:

$$\tau = \tau_{\text{linear}} + \tau_{\text{rotasi}} \dots \dots \dots (2.23)$$

2.8. Driver A4988

Dirancang untuk mengoperasikan motor stepper dalam mode step penuh, setengah, seperempat, seperdelapan, dan seperenambelas, dengan kapasitas drive *output* hingga 35V dan $\pm 2A$. driver A4988 terdapat *mode mikrostepping* hingga 1/16 langkah, yang memungkinkan pengguna dapat mengatur gerakan motor menjadi lebih halus dan akurat[22].



Gambar 2. 7 Driver A4988[22]

Driver A4988 terdiri dari beberapa bagian yang berfungsi untuk melakukan pemrosesan data *input* dari mikrokontroler yang digunakan untuk menggerakkan motor. Berikut merupakan penjelasan terkait pin yang ada pada Driver A4988 yang dapat dilihat pada **Tabel 2.6**.

Tabel 2. 6 Pin-out Driver A4988[22]

Nama	Fungsi
VDD	Tegangan <i>input</i> (3.3-5V)
GND	Ground
VMOT	Tegangan motor (8-35V)
STEP	<i>Input</i> step motor
DIR	<i>Input</i> arah putar motor (HIG/LOW)
ENABLE	<i>Input</i> logika (LOW = motor aktif, HIGH = motor non-aktif)
MS1, MS2, MS3	<i>Input</i> resolusi mikrostep
SLEEP	Mode hemat daya
RESET	Reset translator
OUT1A/B	Ouput motor 1
OUT2A/B	<i>Output</i> motor 2
VREF	<i>Input</i> tegangan referensi

Driver A4988 bekerja dengan menerima sinyal STEP dan DIR dari mikrokontroler. Seperti pada **Table 2.7** sinyal STEP berfungsi sebagai penentu gerak motor, dan DIR sebagai penentu arah gerak dari motor. Pada MS1, MS2, dan MS3 motor dapat dikonfigurasi untuk bergerak mengikuti *input*kan step yang diberikan oleh mikrokontroler, konfigurasi tersebut dapat dilihat pada **Tabel 2.7** berikut.

Tabel 2. 7 Mode Mikrostepping[22]

MS1	MS2	MS3	Microstep Resolution
L	L	L	Full Step
H	L	L	Half Step
L	H	L	Quarter Step
H	H	L	Eighth Step
H	H	H	Sixteen Step

Driver A4988 menggunakan sistem pengaturan arus berdasarkan referensi tegangan (VREF) dan resistor sense (Rsense). Dengan persamaan (21) sebagai berikut :

$$I_{max} = \frac{V_{REF}}{8 \times R_{sense}} \dots \dots \dots (2.16)$$

Dengan menggunakan persamaan diatas kita dapat menentukan arus maksimum yang digunakan untuk pemrosesan data pada driver A4988.

2.9. Motor Servo MG90s

Servo motor MG90S merupakan aktuator elektromekanik mikro yang banyak diaplikasikan dalam sistem robotika, kendaraan RC (Remote Control), drone, dan berbagai proyek otomasi presisi. Komponen ini termasuk kategori servo motor posisional dengan rentang rotasi $\pm 90^\circ$ (total 180°) yang dilengkapi dengan roda gigi logam (metal gear) untuk meningkatkan kekuatan dan ketahanan mekanis[23]. Servo motor adalah jenis elektromotor yang poros keluarannya dapat dikendalikan dengan presisi tinggi baik pada sudut rotasi tertentu maupun pada kecepatan rotasi yang konstan, sehingga servo motor banyak digunakan dalam sistem robotika untuk meningkatkan presisi dan kehalusan gerak suatu mekanisme[24].

Tabel 2. 8 Spesifikasi MG90S[24]

Parameter	Nilai
Berat	13,4 g
Dimensi	$22,5 \times 12 \times 35,5$ mm
Stall Torque	1,8 kgf·cm (4,8V) / 2,2 kgf·cm (6V)
Kecepatan Operasi	0,1 s/60° (4,8V) / 0,08 s/60° (6V)
Tegangan Operasi	4,8V – 6,0V
Dead Band Width	5 μ s
Rentang Rotasi	$\pm 90^\circ$ (total 180°)
Frekuensi PWM	50 Hz (periode 20 ms)

Servo motor MG90S beroperasi pada rentang tegangan 4,8V hingga 6,0V. Torsi stall yang dihasilkan adalah 1,8 kgf·cm pada tegangan 4,8V dan meningkat menjadi 2,2 kgf·cm pada tegangan 6V, menunjukkan bahwa semakin tinggi tegangan operasi maka torsi yang dihasilkan semakin besar. Kecepatan operasi mencapai 0,1 s/60° pada 4,8V dan 0,08 s/60° pada 6V, sehingga pada tegangan lebih tinggi servo bergerak lebih cepat. Parameter dead band width sebesar 5 μ s merupakan rentang lebar pulsa di mana servo tidak merespons perubahan sinyal, berfungsi sebagai zona mati untuk mencegah getaran dan osilasi pada posisi yang

diinginkan. Semakin kecil nilai dead band, semakin sensitif dan responsif servo terhadap perubahan sinyal kontrol.

Servo motor terdiri atas motor DC, gearbox dengan poros keluaran, dan kontroler dengan sensor-sensor yang diperlukan seperti encoder atau sensor posisi. Gearbox pada servo motor berfungsi untuk mereduksi kecepatan dan meningkatkan torsi pada poros keluaran, sedangkan potensiometer atau encoder digunakan untuk melacak sudut rotasi atau kecepatan poros, sehingga terbentuklah sistem kendali closed-loop dengan umpan balik (feedback)[23]. Servo motor MG90S bekerja berdasarkan prinsip kendali loop tertutup (closed-loop control). Rangkaian elektronik di dalam servo mengukur lebar setiap pulsa masukan dan menggunakan umpan balik dari potensiometer internal untuk menggerakkan motor hingga posisi poros keluaran sesuai dengan lebar pulsa yang diperintahkan[25]. Potensiometer yang terpasang di dalam servo berfungsi sebagai sensor posisi. Ketika motor berputar, hambatan potensiometer berubah secara proporsional terhadap sudut rotasi, menghasilkan tegangan umpan balik yang dibandingkan dengan sinyal referensi. Jika terdapat selisih antara posisi aktual dan posisi yang diperintahkan, rangkaian kontrol akan menggerakkan motor ke arah yang tepat hingga selisih tersebut menjadi nol.

Pulse Width Modulation (PWM) adalah metode pengendalian motor penentuan sudut rotasi atau kecepatan poros keluaran berdasarkan panjang pulsa pada frekuensi tertentu. Penggunaan PWM untuk mengendalikan servo motor didasarkan pada prinsip bahwa PWM menghasilkan pulsa dengan lebar (durasi) yang bervariasi[23]. Sinyal PWM yang digunakan memiliki karakteristik sebagai berikut: frekuensi 50 Hz dengan periode 20 ms, dengan lebar pulsa (on-time) bervariasi antara 1 ms hingga 2 ms[24]. Pemetaan antara lebar pulsa dan posisi sudut servo adalah sebagai berikut:

Tabel 2. 9 Hubungan Lebar Pulsa PWM dengan Posisi Servo MG90S [24]

Lebar Pulsa (ms)	Posisi / Perilaku
~1,0 ms	Posisi -90° (kiri penuh / CCW)
1,5 ms	Posisi 0° (netral / tengah)

~2,0 ms	Posisi +90° (kanan penuh / CW)
---------	--------------------------------

Arah rotasi servo MG90S ditentukan sepenuhnya oleh lebar pulsa PWM. Berdasarkan datasheet MG90S, posisi "0" (pulsa 1,5 ms) adalah posisi tengah, pulsa ~2 ms memutar servo ke kanan (CW / *clockwise*), dan pulsa ~1 ms memutar servo ke kiri (CCW / *counter-clockwise*), lebar pulsa antara 1,0 ms hingga 1,5 ms menggerakkan servo ke arah CW, sedangkan antara 1,5 ms hingga 2,0 ms menggerakkan ke arah CCW[26]. Untuk mengubah posisi servo sebesar 1°, diperlukan perubahan lebar pulsa sekitar 5,5 μ s. Pengendalian arah ini dapat dilakukan menggunakan library *Servo.h* pada Arduino, di mana nilai sudut 0°–180° secara otomatis dikonversi menjadi lebar pulsa yang sesuai[24].

2.10. DC Power Supply 12V 20A

Power supply merupakan komponen yang berfungsi untuk menyuplai energi listrik ke komponen elektronik dengan tegangan dan arus yang sesuai. *Power supply* terdapat dua jenis dengan sistem kerja yang berbeda, yaitu *power supply linear* dan *power supply switching* (SMPS-*Switched Mode Power Supply*). *Power supply linear* menggunakan transformator dan regulator linier untuk menghasilkan tegangan stabil, tetapi efisiensi rendah. Sedangkan untuk *power supply switching* menggunakan transformator dan regulator switching untuk menghasilkan tegangan yang stabil, dengan proses *switching* frekuensi tinggi[27].

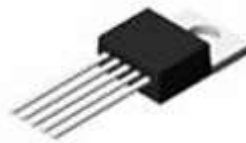
Pada proyek tugas akhir ini, penulis menggunakan *power supply* 12V/20A sebagai penyuplai daya ke mikrokontroler, sensor, dan *actuator*. Pemilihan tegangan 12V dikarenakan Arduino Mega 2560 dengan RAMPS 1.6 Board membutuhkan tegangan minimal sebesar 12V untuk menjalankan sistem untuk menggerakkan maksimal lima *stepper motor*, sensor, dan *actuator* yang dibutuhkan untuk proyek tugas akhir ini, sedangkan arus maksimal 20A dipilih guna memastikan empat *stepper motor* yang digunakan dapat bekerja dengan optimal tanpa penurunan performa. Dan bentuk hardware *power supply* 12V/10A dapat dilihat pada spesifikasi pada **Tabel 2.10** sebagai berikut.

Tabel 2. 10 Spesifikasi Power Supply 12V 10A[27]

Spesifikasi	Keterangan
<i>Input Voltage</i>	85-264V AC
<i>Output Voltage</i>	12V DC
<i>Output Arus</i>	20A (maksimal)
<i>Ouput Daya</i>	240W

2.11. Step Down XL4016 10A

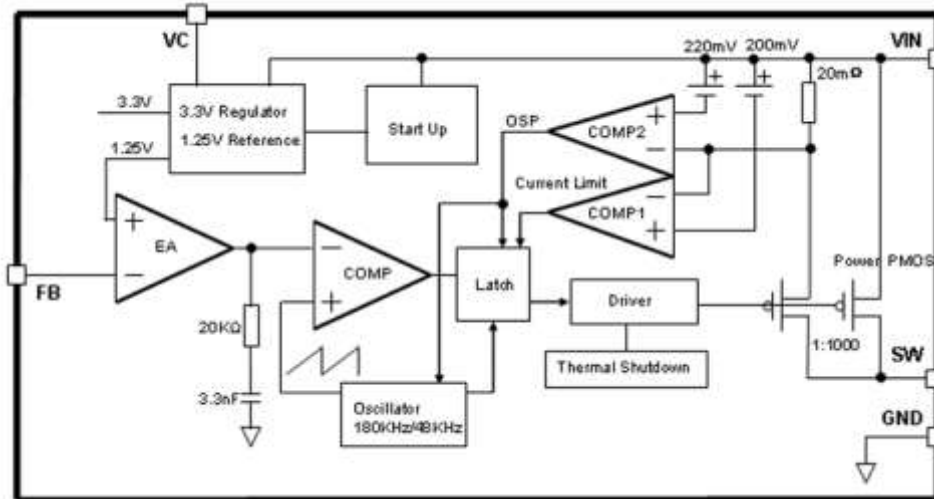
XL4016 merupakan modul DC-DC *buck converter* yang dirancang untuk menurunkan tegangan *input* DC menjadi tegangan *output* yang lebih rendah dengan efisiensi tinggi. Komponen ini bekerja menggunakan metode *Pulse Width Modulation* (PWM) dengan frekuensi 180kHz[28]. Dengan spesifikasi yang dapat dilihat pada **Tabel 2.11** berikut.

**Gambar 2. 8** XL4016[28]**Tabel 2. 11** Spesifikasi Step Down XL4016[28]

Spesifikasi	Keterangan
<i>Input Voltage</i>	8V – 40V
<i>Output Voltage</i>	1.25V – 36V
<i>Output Arus</i>	10A (maksimal)
Efisiensi	+96%
<i>Frekuensi Switching</i>	180 kHz

XL4016 bekerja dengan menurunkan *input* tegangan (8-40V) menjadi tegangan *output* yang lebih rendah (1,25-36V) menggunakan *step-down*. Sistem kerja dari *step-down* ini yaitu tegangan *input* masuk ke regulator internal menghasilkan tegangan referensi sebesar 1.25V. tegangan referensi ini dibandingkan dengan tegangan umpan balik (FB) dari *output* melalui pembagi resistor. Jika *output* lebih rendah dari target yang diinginkan, maka sistem akan memperbesar *duty cycle* pada sinyal PWM, begitu pun sebaliknya, jika *output* terlalu tinggi, maka lebar *duty cycle* akan diperkecil[28].

Sinyal PWM digunakan untuk mengontrol MOSFET Internal. MOSFET mengirimkan energi ke induktor dan kapasitor *output*, yang kemudian menghaluskan pulsa menjadi tegangan DC stabil. Proses ini akan berlangsung terus menerus hingga *output* mencapai nilai yang diinginkan[28], yang dapat dilihat pada **Gambar 2.11** berikut.



Gambar 2.9 Diagram Blok Sistem Step Down XL4016

2.12. Sensor *Thermocouple Type K*

Thermocouple merupakan sensor suhu yang bekerja berdasarkan efek *feedback*, yaitu tegangan listrik pada dua poros sambungan dari dua logam berbeda memiliki perbedaan suhu. Tegangan termoelektrik yang dihasilkan bersifat proporsional terhadap selisih suhu antara sambungan panas dan sambungan dingin.

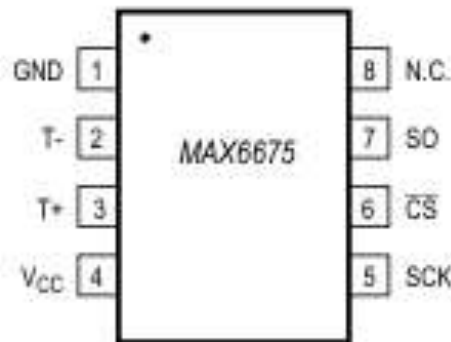
Thermocouple type k ini terdiri dari dua jenis logam, yaitu chromel sebagai kawat positif dan alumel sebagai kawat negatif. Kombinasi ini menghasilkan sensitivitas sekitar $41 \mu\text{V}/^\circ\text{C}$, dengan rentang suhu yang dapat dideteksi antara -40 hingga 600°C . Untuk sensor *thermocouple* tipe k ini dapat mendeteksi suhu dengan rentang $0-600^\circ\text{C}$, dengan resolusi suhu converter adalah $0-0.25^\circ\text{C}$. Terkait penjelasan spesifikasi sensor *thermocouple type-k* dapat dilihat pada **Tabel 2.12** berikut.

Tabel 2. 12 Spesifikasi Sensor Thermocouple Tipe-K[29]

Spesifikasi	Keterangan
Rentang Suhu	-40°C hingga 600°C
Sensitivitas	41 μ V/°C
Akurasi	\pm 1.5°C atau \pm 0.4%

2.13. MAX6675

MAX6675 merupakan konverter *thermocouple* tipe-K ke digital menggunakan sistem SPI, dengan mengubah sinyal *analog* menjadi data digital 12-bit dengan resolusi 0,25°C, dengan maksimal pengukuran suhu hingga 1024°C[30]. MAX6675 terdapat 8 pin yang digunakan dalam pemrosesan data dari [22]sensor *thermocouple* yang dapat dilihat pada **Gambar 2.10**.

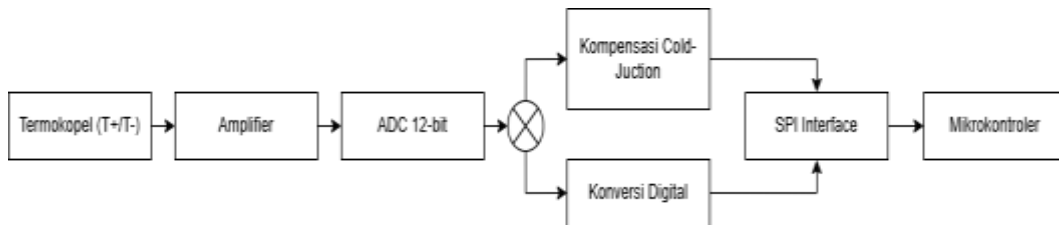
**Gambar 2. 10** Pin I/O MAX6675[30]

Pada **Gambar 2.10**, merupakan struktur pin-out MAX6675 yang berfungsi untuk melakukan pemrosesan data dari sensor *thermocouple* ke mikrokontroler, yang dapat dilihat fungsinya pada **Tabel 2.13**, sebagai berikut.

Tabel 2. 13 Pin I/O MAX6675[30]

PIN	Nama	Fungsi
1	GND	Ground
2	T-	Kabel alumel negatif thermocouple
3	T+	Kabel Chromel (positif) thermocouple
4	Vcc	Catu Daya (3V – 5.5V)
5	SCK	Input Clock SPI
6	CS	Chip Select
7	SO	Output Data SPI
8	N.C.	No Connection

Sistem komunikasi dalam pemrosesan data suhu pada MAX6675 dapat dilihat pada **Gambar 2.11** sebagai berikut :



Gambar 2. 11 Diagram Blok Sistem MAX6675[30]

Pada diagram blok diatas terdapat sistem komunikasi dalam pemrosesan data oleh MAX6675 dari sensor thermocouple, yang mana sinyal dari sensor termokopel akan diubah dengan ADC 12-bit dengan menggunakan dua prinsip kerja yaitu kompensasi cold junction dan konversi digital, selanjutnya data akan dikirim ke mikrokontroler dengan sistem SPI[30]. Berikut penjelasan terkait prinsip kerja pemrosesan data pada MAX6675 :

a. Kompensasi Cold-Junction:

Sistem yang digunakan yaitu dengan melakukan pengukuran suhu ambient disekitar IC dan menyesuaikan pembacaan dari sinyal *thermocouple* secara otomatis, dengan rumus pada persamaan berikut.

$$V_{OUT} = 41\mu V/^{\circ}C \times (T_{Hot} - T_{Cold}) \dots \dots \dots (2.24)$$

Keterangan :

T_{Hot} = Suhu ujung sensor *thermocouple*

T_{Cold} = Suhu ambient

b. Konversi Digital :

Sinyal *thermocouple* diubah ke data digital dengan ADC 12-bit. Dan pembacaannya akan dikirim melalui SPI dalam format bilangan biner. Dengan contoh sebagai berikut :

Jika *output* SPI = 0000111111 (bilangan biner)

Konversi ke desimal = 63 (bilangan decimal)

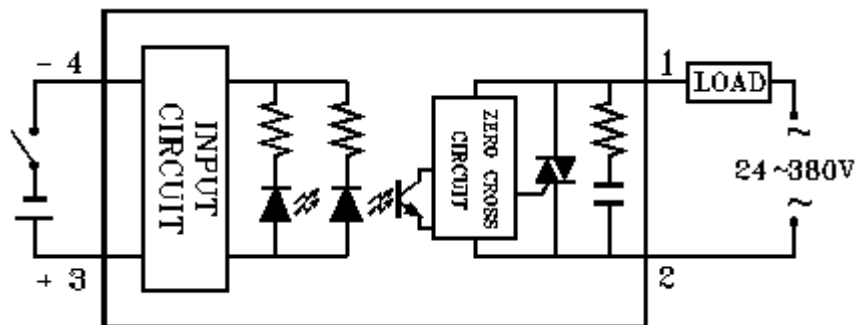
Setelah dikonversi ke nilai desimal maka nilai tersebut akan dikalikan dengan resolusi untuk data digital sebesar 0,25°C. Sehingga pembacaan suhu yang terukur adalah $63 \times 0,25^{\circ}C = 15.75^{\circ}C$.

2.14. Solid State Relay 25DA

Solid State Relay 25DA merupakan *relay* yang digunakan untuk mengendalikan beban AC berarus tinggi menggunakan sinyal control DC bertegangan rendah, dengan tegangan *output* maksimal 380V. Penamaan 25DA sendiri memiliki makna sesuai spesifikasi dari *Solid State Relay*, yang mana “25” berarti arus *output* maksimal dari *Solid State Relay* ini adalah 25A, “D” berarti *Solid State Relay* ini menggunakan *input* DC, dan “A” berarti *Solid State Relay* ini menghasilkan *output* berupa tegangan AC. Spesifikasi *Solid State Relay* ini dapat dilihat secara spesifik pada **Tabel 2.14**.

Tabel 2. 14 Spesifikasi SSR 25DA[31]

Parameter	Keterangan
Tipe	Single phase (DC to AC)
Tegangan <i>Input</i>	3-32VDC
Tegangan <i>output</i>	24-380VAC
Arus <i>Output</i>	25A
Arus Trigger	<7,5mA
<i>Surge Current</i>	250A
Metode <i>switching</i>	<i>Zero Crossing Trigger</i>



Gambar 2. 12 Diagram Sistem SSR 25DA[31]

Pada diagram sistem SSR 25DA diatas terdapat beberapa bagian dan fungsinya sebagai berikut :

a. *Input Circuit*

Terdapat terminal positif (+) dan negative (-) yang digunakan sebagai pengendali SSR. Dan juga terdapat LED yang akan menyala ketika terdapat

tegangan *input* DC, dengan dibatasi oleh resistor. Serta Dioda yang berfungsi untuk melindungi polaritas terbalik.

b. *Optocoupler*

Ketika LED menyala maka optocouple akan aktif, sehingga memicu terjadinya phototransistor disisi *output*. Yang berfungsi sebagai isolasi galvanic pada *input* dan *output* SSR.

c. *Zero Cross Circuit*

Berfungsi untuk mengurangi lonjakan arus dan gangguan elektromagnetik (EMI) saat *relay* ON/OFF.

d. *Output Sircuit*

Pada bagian ini merupakan keluaran atau out yang dihasilkan oleh SSR berupa tegangan AC .

Tabel 2. 15 Pin dan Koneksi SSR 25DA[31]

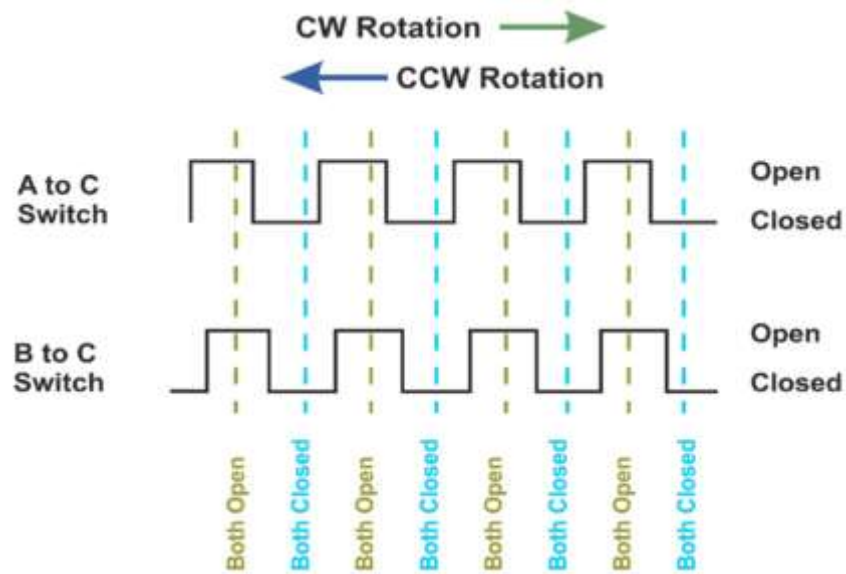
Pin	Funsgi
<i>Input</i> DC +	Tegangan control DC (3-32V)
<i>Input</i> DC -	Ground control DC
<i>Output</i> AC 1	Terminal beban AC (24-380VAC)
<i>Output</i> AC 2	Terminal beban AC (24-380VAC)

Pada alat ini, SSR 40DA berfungsi sebagai saklar yang mengontrol aliran daya listrik AC ke *soldering iron*, dengan kontrol ON/OFF untuk mempertahankan kestabilan suhu pada *soldering iron*.

2.15. *Rotary Encoder KY-040*

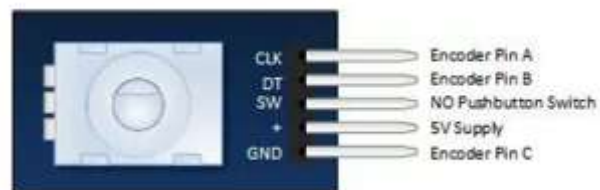
Rotary Encoder KY-040 merupakan *rotary encoder* mekanik yang berfungsi sebagai alat *input* putar (KNOP) untuk mendeteksi jumlah dan arah putaran pada kontrol motor *stepper*, servo, dan sebagainya. *Rotary encoder* ini bekerja menggunakan tegangan sebesar 5V DC dengan keluaran berupa siny digital.

Pada *rotary encoder* ini terdapat dua saklar (*switch*) yang terhubung ke pin A (CLK) dan pin B (DT). Ketika *encoder* diputar, maka kedua *switch* akan membuka dan menutup secara bergantian. Pin CLK sendiri berputar mengikuti arah putaran jarum jam (CW), sedangkan pin DT berputar berlawanan arah jarum jam (CCW), yang dapat dilihat pada **Gambar 2.15**.



Gambar 2. 13 Sistem kerja Switch pada Rotary Encoder KY-040[32]

Rotary Encoder KY-040 terdapat 5 pin yang digunakan dalam proses komunikasi data yang dapat dilihat pada **Gambar 2.16**.



Gambar 2. 14 Pin-out Rotary Encoder KY-040[32]

Tabel 2. 16 Pin-out Rotary Encoder KY-040[32]

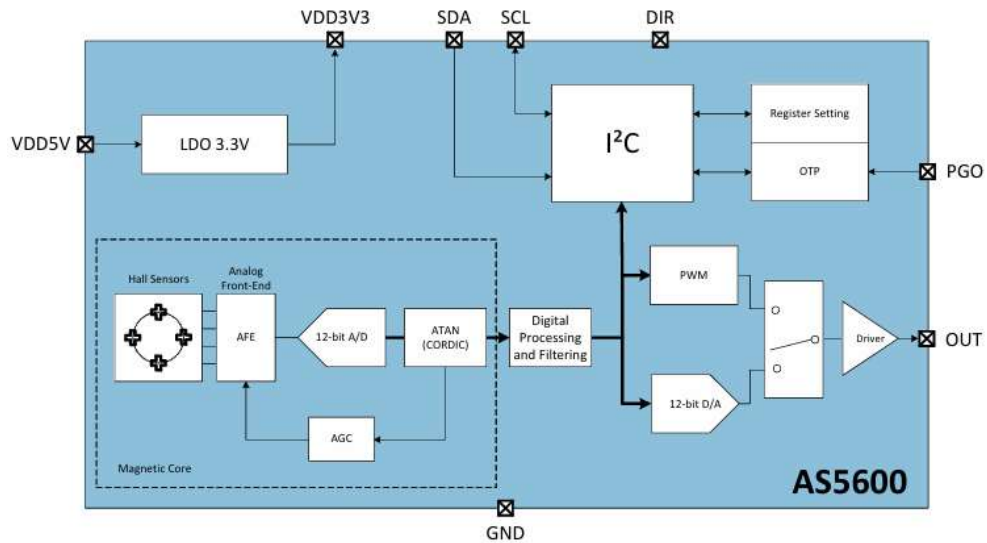
Pin	Fungsi
CLK	Pin A
DT	Pin B
SW	Switch (NO Pushbutton)
+	VCC (5V)
GND	Pin C

2.16. AS5600 Magnetic Encoder

AS5600 adalah sensor posisi sudut tanpa kontak berbasis efek Hall buatan ams AG. Sensor ini membaca posisi sudut magnet diametral yang ditempatkan di atasnya, menghasilkan *output* 12-bit (4096 langkah per putaran penuh) melalui antarmuka I²C, *analog*, atau PWM[33].

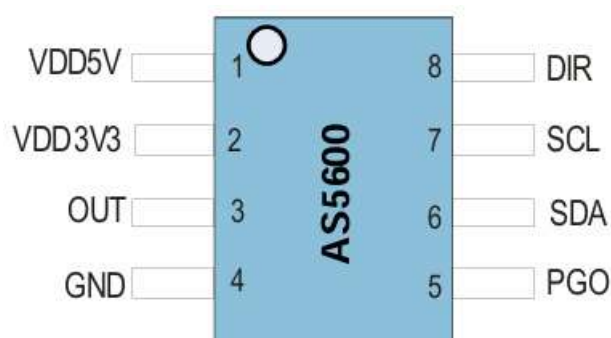
Prinsip kerja dari AS5600 yaitu Magnet diametral (kutub N dan S berada di sisi berlawanan secara horizontal) diputar di atas chip. Elemen Hall planar pada AS5600 mendeteksi komponen medan magnet B_z yang tegak lurus terhadap permukaan chip. Sinyal dari dua pasang sensor Hall diolah oleh algoritma CORDIC (Coordinate Rotation Digital Computer) untuk menghitung sudut dan besar medan secara bersamaan, menghasilkan nilai sudut 12-bit (0–4095) yang mewakili 0° – 360° . Karena kekuatan medan magnet dapat berubah akibat suhu, jarak airgap, atau degradasi magnet, AS5600 menggunakan AGC tertutup untuk menyesuaikan penguatan secara otomatis. Nilai register AGC yang berada di tengah rentangnya menandakan kondisi operasi optimal[33].

Secara fungsional, diagram blok AS5600 mengintegrasikan teknologi sensor Hall dengan pemrosesan sinyal digital yang canggih untuk menghasilkan pembacaan sudut yang presisi. Proses dimulai pada elemen Hall yang menangkap fluks magnetik dari magnet eksternal, kemudian sinyal *analog* tersebut dikonversi oleh ADC menjadi data digital. Unit pemrosesan pusat yang menggunakan algoritma CORDIC (*Coordinate Rotation Digital Computer*) bertugas menghitung posisi sudut absolut 12-bit dengan mengolah vektor magnetik tersebut yang dapat dilihat pada **Gambar 2.17**. Diagram ini juga mencakup blok memori OTP (*One-Time Programmable*) yang memungkinkan penyimpanan parameter konfigurasi secara permanen, serta unit manajemen daya cerdas yang mengatur perpindahan ke mode hemat daya secara otomatis[33].



Gambar 2. 15 Diagram Blok AS5600 Magnetic Encoder[33]

Konfigurasi pin pada AS5600 dirancang agar sensor ini dapat menggantikan potensiometer konvensional dengan fungsionalitas yang lebih luas. Pin VDD5V dan VDD3V3 memberikan opsi catu daya ganda sesuai dengan standar logika mikrokontroler yang digunakan, sementara pin OUT memberikan fleksibilitas *output* berupa sinyal *analog* ratiometrik atau modulasi lebar pulsa (PWM). Untuk kontrol arah, pin DIR memungkinkan pengguna menentukan polaritas peningkatan nilai sudut terhadap arah putaran magnet (searah atau berlawanan jarum jam) yang dapat dilihat pada **Gambar 2.18**. Antarmuka komunikasi melalui pin SCL dan SDA tidak hanya berfungsi untuk pembacaan data digital, tetapi juga sebagai jalur akses ke register internal untuk keperluan kalibrasi dan pemrograman rentang sudut kerja[33], yang dapat dilihat secara rinci pada **Tabel 2.17**.



Gambar 2. 16 Pin-out Pin AS5600 Magnetic Encoder[33]

Tabel 2. 17 Pin-out Pin AS5600 Magnetic Encoder[33]

Pin	Tipe	Keterangan
VDD5V	Supply	5Vdc
VDD3V3	Supply	3.3 Vdc

OUT	<i>Analog / Digital Output</i>	<i>PWM output</i>
GND	Supply	Ground
PGO	<i>Digital Input</i>	Program option
SDA	<i>Digital Input</i>	I2C Data
SCL	<i>Digital Input</i>	I2C Clock
DIR	<i>Digital Input</i>	Direction Polarity

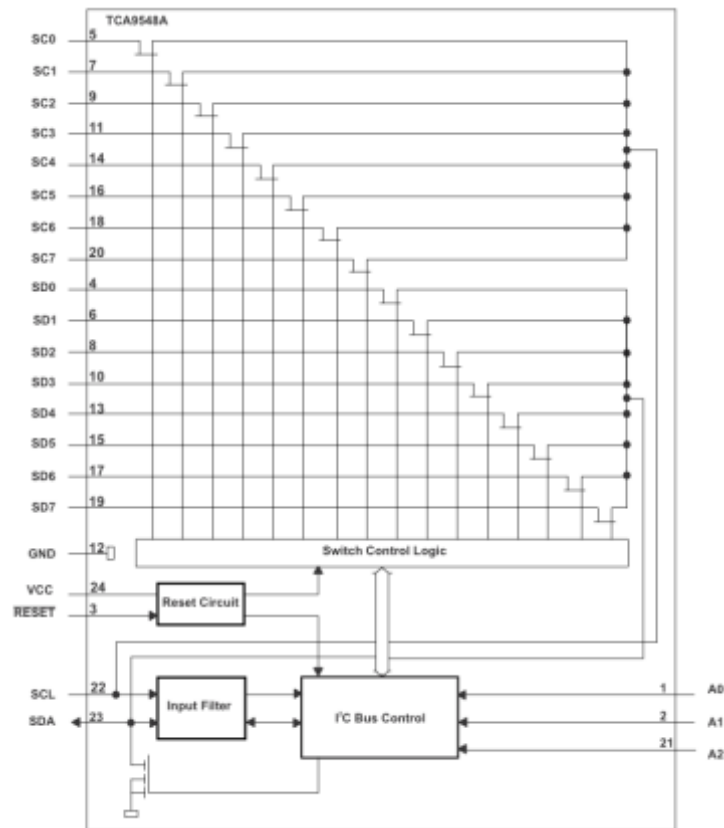
2.17. TCA9548A

TCA9548A adalah IC switch I²C yang memungkinkan satu mikrokontroler berkomunikasi dengan hingga delapan perangkat I²C secara terpisah, bahkan jika perangkat-perangkat tersebut memiliki alamat yang sama. Secara prinsip kerja, TCA9548A bertindak sebagai "jalur switching" antara satu bus I²C upstream (dari mikrokontroler) dengan delapan bus downstream. Hanya satu atau beberapa kanal yang diaktifkan pada satu waktu, ditentukan oleh nilai yang ditulis ke register kontrol 8-bit melalui I²C itu sendiri[34].

Pada TCA9548A sistem komunikasi yang digunakan adalah dengan mikrokontroler mengirim satu byte ke alamat I²C TCA9548A (0x70–0x77, ditentukan oleh pin A0–A2). Byte tersebut adalah register kontrol 8-bit di mana setiap bit mewakili satu kanal — bit bernilai 1 berarti kanal aktif. Contohnya, mengirim 0x01 (binary: 00000001) mengaktifkan hanya Channel 0, sedangkan 0x03 (00000011) mengaktifkan Channel 0 dan Channel 1 secara bersamaan. TCA9548A juga dapat menjembatani bus I²C dengan tegangan berbeda (1.8V, 2.5V, 3.3V, dan 5V) tanpa komponen tambahan — cukup dengan menyesuaikan tegangan pull-up di masing-masing kanal[34].

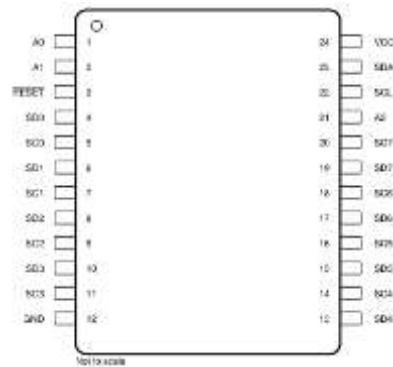
Diagram blok TCA9548A berpusat pada sebuah gerbang logika kontrol yang mengelola sakelar dua arah (*bidirectional*) berbasis teknologi CMOS dengan hambatan rendah. Struktur internalnya menghubungkan bus utama (SCL/SDA) ke delapan saluran *downstream* melalui matriks sakelar yang memungkinkan data mengalir tanpa distorsi sinyal secara signifikan. Komponen ini dilengkapi dengan logika pemilih alamat yang dihubungkan ke pin eksternal A0, A1, dan A2, yang secara kolektif menentukan identitas perangkat di bus I²C[34]. Selain itu, terdapat mekanisme proteksi berupa fitur *Power-On Reset* yang memastikan semua saluran

dalam keadaan terputus saat pertama kali dinyalakan untuk mencegah tabrakan data secara tidak sengaja yang dapat dilihat pada **Gambar 2.17** sebagai berikut.



Gambar 2. 17 Diagram Blok TCA9548A[34]

Pinout TCA9548A dirancang untuk memudahkan manajemen jalur komunikasi yang kompleks. Pin SDA dan SCL pada sisi master berfungsi sebagai gerbang masuk data utama, sementara pin SD0/SC0 hingga SD7/SC7 merupakan titik distribusi menuju perangkat target yang dapat dilihat pada **Gambar 2.17**. Pin reset memiliki peran krusial sebagai masukan aktif-rendah (*active-low*) untuk memulihkan sistem dari kondisi *bus hang* tanpa harus memutus daya utama. Fleksibilitas tegangan operasional dari 1.65v hingga 5.5v pada pin VCC mendukung fitur translasi level tegangan, sehingga memungkinkan integrasi perangkat dengan perbedaan logika tegangan dalam satu sistem kontrol, yang dapat dilihat pada **Tabel 2.18**.



Gambar 2. 18 Pin-out Pin TCA9548A[34]

Tabel 2. 18 Pin-out TCA9548A[34]

PIN	PIN		Type	Keterangan
	TSSOP, VSSOP	VQFN (RGE)		
A0	1	22	I	Address <i>input</i> 0 (VCC/GND)
A1	2	23	I	Address <i>input</i> 1 (VCC/GND)
A2	21	18	I	Address <i>input</i> 2 (VCC/GND)
GND	12	9	-	Ground
RESET	3	24	I	Active-low reset <i>input</i> .
SD0	4	1	I/O	Serial Data 0
SC0	5	2	I/O	Serial Clock 0
SD1	6	3	I/O	Serial Data 1
SC1	7	4	I/O	Serial Clock 1
SD2	8	5	I/O	Serial Data 2
SC2	9	6	I/O	Serial Clock 2
SD3	10	7	I/O	Serial Data 3
SC3	11	8	I/O	Serial Clock 3
SD4	13	10	I/O	Serial Data 4
SC4	14	11	I/O	Serial Clock 4
SD5	15	12	I/O	Serial Data 5
SC5	16	13	I/O	Serial Clock 5
SD6	17	14	I/O	Serial Data 6
SC6	18	15	I/O	Serial Clock 6
SD7	19	16	I/O	Serial Data 7
SC7	20	17	I/O	Serial Clock 7
SCL	22	19	I/O	Serial Clock Bus
SDA	23	10	I/O	Serial Data Bus
VCC	24	21	Daya	5Vdc

Pada robot *SCARA* 4 DOF ini TCA9548A berfungsi sebagai jalur komunikasi 4 channel sensor AS5600 *Magnetic Encoder* dalam pembacaan sudut posisi gerak stepper motor.

2.18. Protokol Komunikasi SPI

SPI adalah mode komunikasi serial kecepatan tinggi. Sinyal clock dialirkan dari master ke slave yang berfungsi untuk sinkronisasi. Master dapat memilih slave mana yang akan dikirim data melalui jalur slave select, kemudian data dikirimkan dari master ke slave melalui MOSI. Bus SPI mencapai kecepatan transfer data yang lebih tinggi karena dua perangkat yang saling terhubung berbagi clock yang sama, sehingga tidak perlu menunggu respons sebelum melakukan operasi lainnya. Dalam keadaan idle atau tidak sedang mengirim data, jalur slave select dijaga pada level tegangan tinggi, dan master baru akan mengaktifkan slave yang dituju dengan menurunkan jalur SS ke level tegangan rendah. Salah satu keunggulan utama antarmuka SPI adalah kemampuannya menyediakan kecepatan transfer data yang jauh lebih tinggi dibandingkan antarmuka UART dan I2C, termasuk digunakan pada sistem yang berinteraksi dengan penyimpanan non-volatile berkapasitas besar seperti SD dan microSD, serta digunakan oleh PC modern untuk memuat BIOS [35].

Koneksi SPI standar melibatkan satu perangkat master yang terhubung ke satu atau lebih perangkat slave melalui empat jalur utama. Sinyal SCLK, MOSI, dan MISO dapat digunakan bersama oleh seluruh perangkat *slave*, sementara setiap slave memiliki jalur SS tersendiri yang berfungsi untuk mengaktifkan komunikasi dengan slave yang bersangkutan[36]. Empat mode SPI dicapai dengan mengombinasikan dua properti sinyal clock, yaitu *Clock Polarity* (CPOL) dan *Clock Phase* (CPHA), di mana kombinasi keduanya menentukan edge mana dari sinyal clock yang digunakan untuk mengirimkan dan mengambil sampel data. CPOL mengatur level *clock* saat kondisi idle, sedangkan CPHA mengatur edge clock pada saat data disampling. Jika CPOL bernilai 1, maka SCK akan tetap berada pada kondisi high ketika data tidak sedang dikirimkan; sebaliknya jika CPOL

bernilai 0, SCK tetap berada pada kondisi *low* saat idle[37], yang dapat dilihat pada **Tabel 2.19**.

Tabel 2. 19 Mode SPI[37]

Mode	CPOL	CPHA	Keterangan Sampling
Mode 0	0	0	Data disampling pada <i>rising edge</i>
Mode 1	0	1	Data disampling pada <i>falling edge</i>
Mode 2	1	0	Data disampling pada <i>falling edge</i>
Mode 3	1	1	Data disampling pada <i>rising edge</i>

2.19. Protokol Komunikasi UART

Universal Asynchronous Receiver Transmitter atau yang biasa disingkat dengan UART merupakan protokol komunikasi secara serial yang digunakan untuk mengirimkan dan menerima data di perangkat komputer. UART sangat umum digunakan dalam pengiriman data serial antara perangkat satu dengan perangkat lainnya, dan pada saat sekarang UART sudah dipakai oleh banyak mikrokontroler. UART bukan sekadar protokol komunikasi seperti SPI dan I2C, melainkan juga merupakan rangkaian fisik dalam sebuah mikrokontroler, single board computer, atau integrated circuit yang tersendiri, yang terdiri dari pembangkit clock, shift register, buffer, dan logika kontrol[38].

UART hanya memerlukan dua kabel untuk beroperasi, yaitu TX (transmit) untuk mengirim data dan RX (receive) untuk menerima data. Selain itu, ditambahkan pula satu jalur GND sebagai referensi tegangan bersama. UART mendukung tiga konfigurasi komunikasi, yaitu simplex (transmisi satu arah), half duplex (transmisi dua arah namun tidak bersamaan), dan full duplex (transmisi dua arah secara bersamaan)[39].

Transmisi pada UART berlangsung pada baud rate tertentu, misalnya 9600 bps atau 115200 bps, yang menentukan kecepatan transfer data. Kedua perangkat yang berkomunikasi, baik pengirim maupun penerima, harus menyepakati baud rate yang sama agar komunikasi dapat berjalan dengan benar. Selisih baud rate yang diizinkan adalah hingga 10% sebelum timing bit menjadi terlalu jauh menyimpang dan menyebabkan kesalahan dalam pembacaan data[40], yang dapat dilihat pada

Tabel 2.20

Tabel 2. 20 Baud Rate Standart UART[40]

Baud Rate (bps)	Kecepatan	Umum Digunakan
9600	Lambat	Debug, sensor sederhana
19200	Sedang	Komunikasi serial dasar
38400	Sedang	Modul GPS, Bluetooth
57600	Cepat	Komunikasi jarak dekat
115200	Sangat Cepat	Arduino, ESP32, debug

2.20. Protocol Komunikasi I2C

I2C merupakan protokol komunikasi yang kuat dan fleksibel sehingga memungkinkan berbagai perangkat mikrokontroler seperti sensor dan modul lainnya untuk berkomunikasi secara efisien melalui jalur data tunggal, membuka peluang untuk mengembangkan sistem yang lebih canggih dan terintegrasi. Pada tahun 1982, I2C dikembangkan oleh Philips (sekarang NXP Semiconductors) dan telah menjadi protokol komunikasi yang umum digunakan dalam banyak aplikasi elektronik. I2C adalah protokol komunikasi serial sinkron dua arah yang hanya membutuhkan dua kabel, yaitu SDA (Serial Data Line) dan SCL (Serial Clock Line). I2C menggunakan konsep master-slave, di mana satu atau beberapa master seperti mikrokontroler mengendalikan bus, sedangkan banyak slave seperti sensor atau memori terhubung ke bus yang sama. Setiap perangkat memiliki alamat unik sehingga master dapat memilih perangkat mana yang akan berkomunikasi[41].

Koneksi I2C standar hanya melibatkan dua jalur bidireksional, yaitu SDA dan SCL, ditambah power dan ground. Kedua jalur tersebut memerlukan resistor pull-up yang terhubung ke tegangan sumber positif. Fungsi SCL adalah membawa sinyal clock yang menyinkronkan transfer data dan dihasilkan oleh master I2C, sementara slave dapat menahan SCL pada kondisi low untuk menjeda komunikasi dalam mekanisme yang disebut clock stretching. Data pada jalur SDA harus stabil selama periode HIGH dari SCL, dan SDA hanya boleh berubah ketika SCL dalam kondisi LOW[42].

Protokol komunikasi I2C memungkinkan beberapa perangkat untuk berkomunikasi melalui jalur yang sama dengan menggunakan alamat unik yang diberikan pada setiap perangkat. Desain referensi I2C memiliki ruang alamat 7-bit, dengan perluasan 10-bit yang jarang digunakan. Beberapa vendor melaporkan

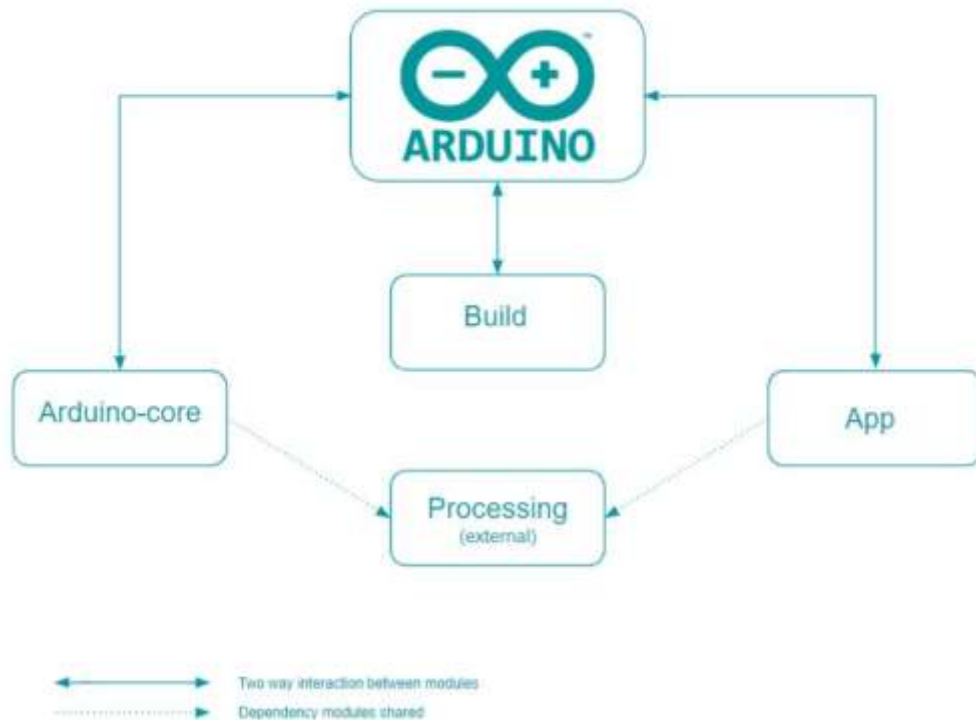
alamat 8-bit karena bit paling rendah dalam byte alamat pertama digunakan untuk membedakan antara perintah write dan read. Slave device dapat diidentifikasi dengan alamat 7-bit yang unik sehingga master dapat mengidentifikasi sensor atau perangkat apa yang terhubung di bus[40].

Dalam protokol I2C terdapat lima kategori kecepatan yang berkisar dari 100 kHz hingga 5 MHz. Kecepatan merupakan salah satu faktor yang membatasi aplikasi bus I2C. Penggunaan resistor pull-up yang menetapkan logika 1 membatasi kecepatan transfer maksimum bus, itulah mengapa high-speed mode dengan kecepatan 3,4 Mbit/s diperkenalkan[43], yang dapat dilihat pada **Tabel 2.21**.

Tabel 2. 21 Mode Kecepatan Protocol I2C[43]

Mode	Kecepatan Maksimal	Keterangan
Standard Mode (Sm)	100 kHz	Mode dasar, kompatibel semua perangkat I2C
Fast Mode (Fm)	400 kHz	Didukung oleh sebagian besar perangkat I2C
Fast Mode Plus (Fm+)	1 MHz	Perlu perangkat yang mendukung Fm+
High-Speed Mode (Hs)	3,4 MHz	Perlu pertimbangan khusus pada sinyal
Ultra-Fast Mode (UFm)	5 MHz	Unidireksional, hanya write

2.21. Arduino IDE



Gambar 2. 19 Arsitektur Development View Arduino IDE[44]

Arduino IDE merupakan *software* pengembangan terintegrasi berbahasa C/C++ yang digunakan untuk melakukan perintah pada *actuator* atau sensor melalui mikrokontroler dengan merekayasa, mengompilasi dan mengunggah kode ke dalam mikrokontroler seperti Arduino dan ESP32. Pada **Gambar 2.19**, terdapat beberapa komponen yang digunakan untuk mendukung kinerja Arduino ide yaitu Arduino-core sebagai penyedia *library* pengontrol *hardware* seperti *actuator* dan sensor, App sebagai penyedia *GUI* untuk mengunggah kode, *build* sistem berfungsi untuk mengelola proses kompilasi, dan *processing* sebagai basis pemrosesan kode[44].