

## PENGEMBANGAN MESIN CNC LASER DIODA TERINTEGRASI ROBOT SCARA MENGGUNAKAN APLIKASI PYTHON

<sup>1</sup>Rima Nur Badriyah, <sup>1</sup>Afzeri, dan <sup>1</sup>Deni Kurnia

<sup>1</sup>Teknologi Rekayasa Mekatronika, Politeknik Enjineri Indorama, Purwakarta, Indonesia

e-mail: [afzeri@pei.ac.id](mailto:afzeri@pei.ac.id)

### Abstrak

Penelitian ini membahas pengembangan mesin CNC laser dioda yang terintegrasi dengan robot SCARA menggunakan aplikasi berbasis Python untuk kebutuhan industri skala kecil dan menengah (UMKM). Tujuan utama dari penelitian ini adalah menciptakan sistem otomatisasi yang efisien, akurat, dan hemat biaya dengan mengintegrasikan dua perangkat utama CNC laser dioda dan robot SCARA dalam satu kesatuan yang saling berkoordinasi. Proses integrasi dilakukan melalui modifikasi firmware grbl.h pada masing-masing mikrokontroler untuk membedakan alamat komunikasi serial. Aplikasi Python dirancang untuk mengendalikan kedua perangkat secara sinkron, memungkinkan pengiriman instruksi secara otomatis dan real-time. Hasil pengujian menunjukkan bahwa sistem mampu melakukan identifikasi perangkat melalui port serial, mengirimkan perintah NC secara akurat, dan menjaga sinkronisasi gerakan antara CNC laser dan robot SCARA. Selain itu, pengujian performa menunjukkan bahwa sistem ini dapat beroperasi dengan waktu operasional melebihi 8 jam, serta dapat melanjutkan pekerjaan setelah gangguan atau penghentian darurat. Dengan menggunakan komponen yang tersedia secara luas dan biaya produksi yang terjangkau, sistem ini menawarkan solusi inovatif dan efisien bagi UMKM dalam meningkatkan produktivitas dan kualitas hasil produksi.

**Kata kunci:** CNC laser dioda, robot SCARA, Python, integrasi sistem, UMKM.

### Abstract

*This research focuses on developing a diode-laser CNC machine integrated with a SCARA robot, using a Python-based application for small and medium-sized enterprises. The primary objective of this study is to create an efficient, accurate, and cost-effective automation system by integrating two main devices, the diode laser CNC and the SCARA robot, into a synchronized operation. The integration process involves modifying the grbl.h firmware on each microcontroller to distinguish serial communication addresses. A Python application is designed to control both devices synchronously, enabling automatic and real-time instruction delivery. The test results demonstrate that the system can accurately identify devices via serial ports, transmit NC commands precisely, and maintain motion synchronization between the CNC laser and the SCARA robot. Furthermore, performance tests reveal that the system remains stable during extended operational periods and can resume tasks after interruptions or emergency stops. Using widely available components and low-cost production, this system provides an innovative and efficient solution for UMKM to enhance productivity and production quality.*

**Keywords:** diode laser CNC, SCARA robot, Python, system integration, UMKM.

## 1. PENDAHULUAN

Dalam beberapa tahun terakhir, perkembangan teknologi di bidang otomasi dan manufaktur mengalami peningkatan pesat seiring dengan kebutuhan industri akan efisiensi dan presisi. Salah satu teknologi yang menjadi perhatian utama adalah *Computer Numerical Control* (CNC) yang

memungkinkan otomatisasi proses pemrosesan berbasis komputer. Teknologi ini telah banyak diterapkan di berbagai sektor industri, termasuk dalam proses pemotongan, pengukiran, dan pencetakan berbasis laser. CNC berbasis laser dioda semakin diminati karena kemampuannya untuk melakukan pemrosesan material dengan tingkat akurasi yang tinggi dan waktu pengerjaan yang efisien [1]. Menurut data dari International Federation of Robotics (IFR), jumlah pemasangan robot industri di seluruh dunia mencapai lebih dari 500.000 unit pada tahun 2023, meningkat 12% dari tahun sebelumnya. SCARA robot menyumbang sekitar 25% dari total pemasangan karena kemampuannya dalam menangani tugas-tugas presisi tinggi seperti laser cutting dan pemrosesan komponen elektro.

Integrasi teknologi CNC dengan *Selective Compliance Assembly Robot Arm* (SCARA) menawarkan potensi besar dalam meningkatkan fleksibilitas dan produktivitas di industri manufaktur. Robot SCARA merupakan salah satu jenis robot industri yang sering diintegrasikan dalam proses produksi karena memiliki kecepatan dan ketepatan gerak tinggi untuk aplikasi seperti pemindahan material dan operasi ukir laser [2]. Robot SCARA dikenal memiliki keunggulan dalam tugas-tugas presisi seperti perakitan komponen kecil dan pengelasan laser dioda. Dengan dikombinasikannya teknologi ini dengan aplikasi berbasis **Python**, dapat meningkatkan kemampuan adaptasi sistem terhadap berbagai kebutuhan produksi secara *real-time* [3].

Meskipun teknologi CNC dan robot SCARA telah banyak digunakan, terdapat beberapa tantangan yang dihadapi dalam integrasi keduanya. Salah satu tantangan utama adalah sinkronisasi gerakan presisi antara sistem CNC dan robot SCARA untuk memastikan akurasi pemotongan laser diode [4]. Selain itu, kesulitan dalam pemrograman dan pengaturan jalur gerak (*motion planning*) menjadi kendala dalam pengembangan sistem yang adaptif dan mudah digunakan di berbagai lingkungan produksi [5]. Selain itu, pengembangan antarmuka berbasis *Python* menghadapi kendala dalam mengelola data besar secara *real-time* dan menyesuaikan parameter pemotongan laser secara dinamis [6].

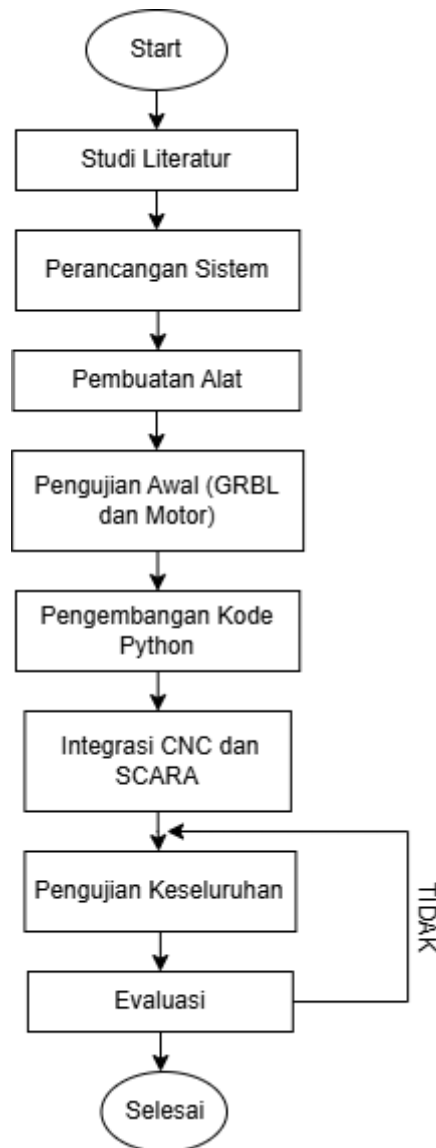
Penelitian terdahulu telah membahas pengembangan berbagai model CNC dan aplikasi robotik berbasis SCARA. Mungabusi Sisa (2018) meneliti implementasi robot kartesian untuk kebutuhan produksi berbasis CNC, namun penelitian tersebut belum sepenuhnya membahas integrasi dengan laser diode [7]. Penelitian lain oleh Yip (2022) mengulas pemanfaatan laser dioda dalam otomatisasi industri, tetapi kurang membahas aspek pemrograman adaptif berbasis *Python* yang memungkinkan fleksibilitas yang lebih besar dalam pengendalian robot [8]. Penelitian oleh Li et al. (2020) mengkaji sistem CNC berbasis *Python* untuk pemotongan presisi menggunakan laser dioda, tetapi memiliki keterbatasan dalam kecepatan proses dan akurasi pengendalian gerakan [9].

Kesenjangan dalam penelitian yang ada menunjukkan perlunya pengembangan sistem CNC berbasis laser dioda yang terintegrasi dengan robot SCARA menggunakan aplikasi *Python*. Penelitian ini bertujuan untuk mengembangkan dan menguji kinerja sistem CNC laser dioda yang terintegrasi dengan robot SCARA menggunakan aplikasi *Python*. Fokus utama penelitian adalah memastikan akurasi, efisiensi, dan fleksibilitas sistem dalam berbagai skenario produksi. Penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam pengembangan teknologi otomatisasi industri di Indonesia dan mendukung tren Industri 4.0. Berbeda dengan penelitian sebelumnya yang fokus pada kontrol terpisah, penelitian ini menawarkan solusi integrasi penuh (*full-integration*) dengan mekanisme sinkronisasi dua arah (*two-way handshake*) berbasis *Python* yang belum banyak dikaji. Kontribusi utama penelitian ini adalah pengembangan protokol komunikasi sederhana dan mudah untuk mengoordinasikan dua kontroler GRBL secara *real-time*, sehingga memungkinkan sistem CNC-SCARA yang *compact* dan terjangkau bagi UMKM.

## 2. METODE PENELITIAN

Metode penelitian ini disusun untuk memastikan semua tahap pengembangan berjalan sesuai dengan prosedur yang telah direncanakan. Setiap tahapan dalam proses ini memiliki peran penting untuk memastikan integrasi antara perangkat keras (mesin CNC dan robot SCARA) dan perangkat lunak berbasis *Python* berjalan secara optimal. Aplikasi *Python* digunakan sebagai pengendali utama untuk memfasilitasi komunikasi antara komponen

hardware dan memungkinkan otomatisasi yang lebih fleksibel dan efisien yang ditunjukkan pada Gambar 1.



Gambar 1. Alur Penelitian

Gambar 1 menjelaskan alur penelitian. Ada beberapa tahap yang akan dijelaskan di bawah ini.

Tahap awal dalam penelitian ini adalah melakukan studi literatur yang berfungsi sebagai dasar teoretis dan acuan dalam memahami konsep dan teknologi yang digunakan. Studi ini mencakup pemahaman tentang prinsip kerja mesin CNC, sistem kendali robot SCARA, serta penerapan bahasa pemrograman Python dalam mengintegrasikan kedua perangkat tersebut. Dengan mempelajari berbagai referensi, peneliti dapat menentukan metode yang paling sesuai dalam mengembangkan

sistem, termasuk memahami protokol komunikasi seperti GRBL yang digunakan untuk mengendalikan mesin CNC melalui instruksi berbasis *G-code* [10].

Setelah memperoleh landasan teori yang kuat, dilakukan proses perancangan sistem yang mencakup pemilihan komponen perangkat keras dan perangkat lunak. Pada tahap ini, peneliti menggunakan motor stepper Nema 23, driver TB6600, sensor, dan laser dioda dengan daya 50 watt yang dibutuhkan untuk membangun mesin CNC dan robot SCARA. Selain itu, desain perangkat lunak dikembangkan menggunakan *Python* untuk mengatur koordinasi gerakan antara kedua perangkat. Rancangan ini disusun secara rinci untuk memastikan seluruh komponen dapat berfungsi secara harmonis dan mendukung integrasi sistem yang efisien [11].

Selanjutnya, dilakukan pembuatan alat berdasarkan desain yang telah dirumuskan sebelumnya. Proses ini melibatkan perakitan fisik mesin CNC dan robot SCARA, pemasangan komponen elektronik, dan pengaturan koneksi antarperangkat. Selain itu, lingkungan kerja berbasis *Python* dikonfigurasi untuk menghubungkan sistem kontrol dengan perangkat keras. Pembuatan alat memerlukan ketelitian tinggi agar setiap komponen dapat berfungsi sesuai rencana dan siap untuk diuji pada tahap berikutnya [12].

Tahap pengujian awal dilakukan untuk memastikan bahwa motor yang dikendalikan oleh GRBL merespons perintah dari kode *Python* dengan benar. Pengujian ini penting untuk memverifikasi fungsi dasar dari sistem dan mengidentifikasi potensi kesalahan sebelum melanjutkan ke tahap integrasi lebih lanjut. Jika terdapat kendala atau malfungsi pada tahap ini, perbaikan dilakukan hingga sistem berjalan sesuai ekspektasi. Pengujian awal menjadi fondasi penting sebelum mengembangkan kontrol yang lebih kompleks di tahap berikutnya.

Setelah pengujian awal berhasil, penelitian berlanjut ke tahap pengembangan kode *Python*. Pada tahap ini, peneliti menulis program menggunakan *Python* untuk mengatur dan mengendalikan gerakan mesin CNC serta robot SCARA secara terintegrasi. Kode *Python* dikembangkan secara bertahap dan diuji untuk memastikan keakuratan instruksi yang dikirimkan. Program ini berfungsi sebagai penghubung utama yang menerjemahkan instruksi ke dalam format yang dimengerti oleh perangkat keras, seperti perintah *G-code* untuk mesin CNC dan kontrol gerakan robot SCARA.

Proses integrasi dilakukan setelah kode *Python* dikembangkan dan diverifikasi. Integrasi ini bertujuan menggabungkan fungsi mesin CNC dan robot SCARA menjadi satu sistem yang bekerja secara bersamaan. Peneliti memastikan bahwa kedua perangkat mampu berkomunikasi secara sinkron dan merespons perintah dari perangkat lunak dengan akurat. Proses ini melibatkan pengaturan kecepatan gerak, koordinasi tugas, dan sinkronisasi antara CNC dan SCARA untuk menjalankan proses otomatisasi yang kompleks.

Setelah integrasi selesai, dilakukan pengujian menyeluruh terhadap sistem. Pengujian ini mencakup evaluasi performa, akurasi gerakan, keandalan komunikasi, dan responsivitas sistem terhadap berbagai skenario operasional. Apabila ditemukan kesalahan atau ketidaksesuaian selama pengujian, maka dilakukan revisi pada perangkat keras atau perangkat lunak hingga sistem bekerja sesuai spesifikasi. Pengujian menyeluruh sangat penting untuk memastikan bahwa semua elemen sistem berfungsi secara optimal dalam kondisi nyata.

Tahap akhir dari penelitian ini adalah evaluasi, di mana seluruh hasil pengujian dianalisis untuk menilai kinerja sistem secara keseluruhan. Evaluasi dilakukan dengan mempertimbangkan berbagai parameter seperti efisiensi waktu, presisi gerakan, dan stabilitas sistem dalam menjalankan tugas otomatisasi. Jika hasil evaluasi memenuhi kriteria yang telah ditetapkan, penelitian dinyatakan berhasil dan proses pengembangan sistem CNC terintegrasi dengan robot SCARA berbasis *Python* selesai. Evaluasi ini tidak hanya berfungsi untuk memastikan keberhasilan implementasi, tetapi juga memberikan umpan balik yang berguna untuk pengembangan lebih lanjut di masa depan.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Identifikasi dan Pemberian Nama Controller

*Firmware* GRBL merupakan perangkat lunak open-source berbasis mikrokontroler yang berfungsi sebagai interpreter *G-code* dan pengendali gerakan mesin CNC secara presisi. Firmware ini banyak digunakan dalam sistem manufaktur skala kecil maupun sistem robotik karena kemampuannya mengontrol motor stepper melalui sinyal STEP/DIR serta kemudahan integrasinya dengan mikrokontroler seperti Arduino berbasis AVR [13].

Langkah awal dalam proses integrasi adalah mengidentifikasi dan memberikan nama pada masing-masing controller (CNC dan SCARA). Setelah dilakukan modifikasi pada file `grbl.h` dan pengunggahan ulang firmware ke masing-masing mikrokontroler, dapat dilihat pada Gambar 2 dan Gambar 3. Sistem mampu mengenali dan membedakan antara controller CNC dan robot SCARA secara otomatis. Proses ini penting untuk memastikan bahwa instruksi yang dikirim ke perangkat sesuai dengan jenis controller yang dituju, menghindari tumpang tindih atau kesalahan eksekusi perintah.

```
24 // Grbl versioning system
25 #define GRBL_VERSION "CNC1.1h"
26 #define GRBL_VERSION_BUILD "20190825"
27
```

Gambar 2. Modifikasi File `grbl.h` menjadi CNC

```
24 // Grbl versioning system
25 #define GRBL_VERSION "SCARA1.1h"
26 #define GRBL_VERSION_BUILD "20190825"
27
```

Gambar 3. Modifikasi file `grbl.h` menjadi SCARA

Pada Gambar 2 dan Gambar 3, modifikasi dilakukan dengan mengubah alamat identifikasi (ID) di dalam file `grbl.h` yang terdapat di masing-masing perangkat. Pada tahap ini, kode sumber GRBL diubah untuk memberikan respons berbeda terhadap perintah "\$I", yang berfungsi sebagai instruksi untuk meminta informasi identitas controller.

#### 3.2 Pendeteksian port dan Pengenalan kontroler

Proses pendeteksian port dan pengenalan kontroler bertujuan untuk memastikan bahwa sistem dapat mengenali dan membedakan antara kontroler CNC dan robot SCARA secara otomatis. Pada tahap ini, masing-masing kontroler dihubungkan ke komputer melalui port serial menggunakan Arduino Uno yang telah dimodifikasi pada file `grbl.h`. Modifikasi ini memungkinkan sistem memberikan identitas unik kepada masing-masing kontroler, seperti "CNC" untuk mesin CNC laser dioda dan "SCARA" untuk robot SCARA.

Ketika sistem dijalankan, program *Python* melakukan pemindaian (*scanning*) terhadap semua port serial yang tersedia di komputer. Setelah port terdeteksi, sistem mengirimkan perintah "\$I" ke masing-masing port. Respons dari perintah ini berisi informasi identitas perangkat, yang memungkinkan sistem membedakan dan memetakan perangkat dengan benar. Jika perangkat merespons dengan nama "CNC" atau "SCARA," maka sistem berhasil mengidentifikasi dan menghubungkan port yang sesuai. Dengan metode ini, kesalahan dalam komunikasi antara perangkat dapat diminimalkan dan memastikan setiap perangkat menerima instruksi yang sesuai.

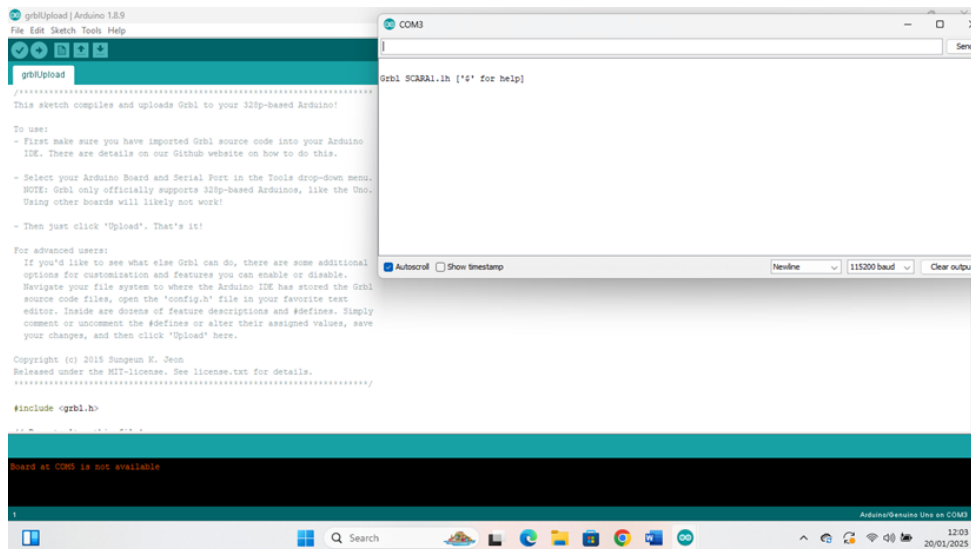
### 3.3 Sinkronisasi Pengiriman NC Program

Sinkronisasi pengiriman file NC bertujuan untuk memastikan bahwa mesin CNC laser dioda dan robot SCARA bekerja secara bersamaan dan terkoordinasi sesuai jalur gerakan yang telah ditentukan. File NC (Numerical Control) berisi serangkaian perintah berbasis *G-code* yang digunakan untuk mengatur gerakan kedua perangkat. Sistem berbasis *Python* berfungsi sebagai pengendali utama yang mengatur pengiriman perintah secara bergantian atau bersamaan ke masing-masing kontroler melalui port serial [14].

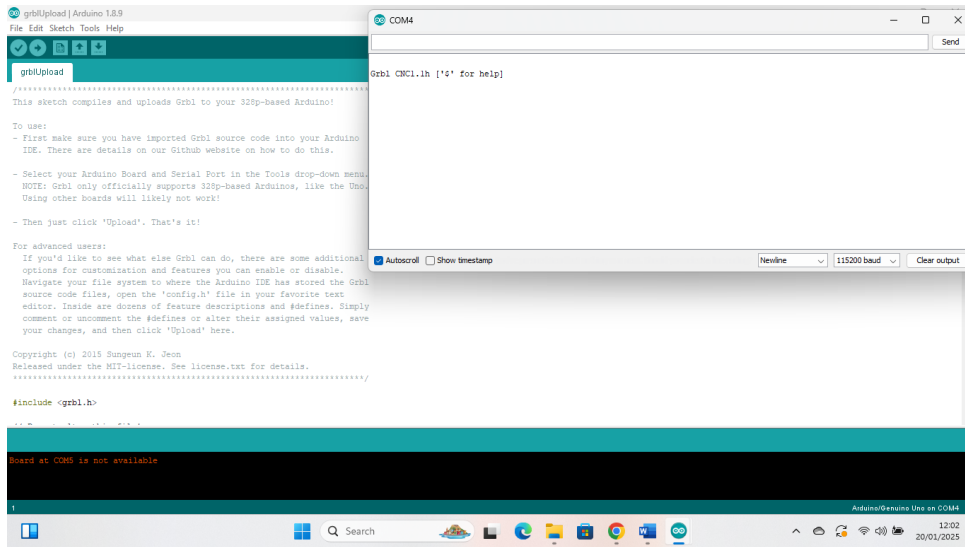
Proses sinkronisasi dimulai dengan membagi perintah NC berdasarkan tugas masing-masing perangkat. Perintah yang mengatur gerakan sumbu X dan Y dikirim ke controller CNC, sementara perintah untuk manipulasi dan pengambilan benda kerja dikirim ke controller robot SCARA. Program Python memantau status eksekusi dari kedua perangkat menggunakan umpan balik (*feedback*) berupa kode "ok" atau kode kesalahan dari masing-masing kontroler.

### 3.4 Pengujian Koneksi dan Respons Perangkat

Untuk memastikan bahwa sistem dapat mengenali dan membedakan antara controller CNC dan robot SCARA, dilakukan pengujian menggunakan *Serial Monitor* di Arduino IDE. Pengujian ini bertujuan untuk memverifikasi respons perangkat setelah dilakukan modifikasi pada file `grbl.h`. Gambar 4 dan Gambar 5 adalah informasi keluaran dari Serial Monitor yang menunjukkan identifikasi jenis kontroler.



Gambar 4. Informasi Keluaran Serial Monitor Pada Controller SCARA



Gambar 5. Informasi Keluaran Serial Monitor Pada Controller CNC

Gambar 4 menunjukkan bahwa perangkat yang terhubung di port COM3 telah dikenali sebagai robot SCARA yang menggunakan *firmware* GRBL versi 1.1h. Instruksi "[!'\$' for help]" menandakan bahwa pengguna dapat memasukkan perintah '\$' untuk melihat daftar perintah yang didukung oleh *firmware* ini. Respons ini membuktikan bahwa proses modifikasi dan pengunggahan *firmware* berhasil dilakukan, serta sistem dapat membedakan controller SCARA dari controller lainnya. Sedangkan Gambar 5 menunjukkan hasil deteksi perangkat yang terhubung melalui port COM4 menggunakan software Arduino IDE. Berdasarkan tampilan pada Serial Monitor, perangkat teridentifikasi sebagai "Grbl CNC1.1h", yang menunjukkan bahwa perangkat ini merupakan controller CNC yang telah diunggah dengan *firmware* GRBL versi 1.1h [15].

Dengan hasil keluaran tersebut, dapat disimpulkan bahwa sistem mampu mengenali controller SCARA secara otomatis melalui port serial yang tersedia. Proses ini memastikan bahwa perangkat dapat menerima dan mengeksekusi perintah dengan benar, yang menjadi langkah penting dalam integrasi dan sinkronisasi kedua controller.

### 3.5 Program Pengembangan Integrasi CNC dan SCARA

Dalam pengembangan sistem ini, beberapa file utama dibutuhkan untuk mengatur jalannya sistem dan memfasilitasi komunikasi antara komponen. Berikut adalah penjelasan mengenai fungsi dari masing-masing file ditunjukkan pada Tabel 1.

Tabel 1. Kebutuhan File Integrasi Sistem

Nama File	Fungsi
main.py	File utama yang menangani koneksi serial dan pengiriman perintah.
gcode_parser.py	(Opsional) Parser tambahan jika perlu memproses G-code yang kompleks.
arduino_cnc.ino	Kode yang diunggah ke Arduino untuk mengontrol mesin CNC laser.
arduino_scara.ino	Kode yang diunggah ke Arduino untuk mengontrol gerakan Robot SCARA.
cnc_laser.txt	File berisi perintah G-code untuk CNC Laser.
robot_scara.txt	File berisi program gerakan untuk Robot SCARA.

Tabel 2 menampilkan program *Python* yang digunakan untuk mengelola komunikasi dan pengendalian antara CNC laser dan robot SCARA. Program ini mencakup proses identifikasi port, pengiriman instruksi berbasis G-code ke CNC, serta perintah khusus untuk robot SCARA. Dengan adanya tabel ini, alur komunikasi antar perangkat dapat dipahami secara lebih jelas dan terstruktur.

**Tabel 2.** Program Python Integrasi Mesin CNC dan Robot Scara

Program	Penjelasan
<pre>def sendGCodeFile(file_path, com):     try:         with open(file_path, 'r') as file:             lines = file.readlines()             for line in lines:                 sendNCCode(line.strip(), com) # Mengirimkan setiap baris G-code ke CNC                 sleep(0.1) # Memberikan jeda antara setiap perintah     except Exception as e:         print(f"Error saat membaca file G-code: {e}")</pre>	<p>Fungsi <i>sendGCodeFile</i> digunakan untuk mengirim perintah G-code dari file ke mesin CNC melalui komunikasi serial. Fungsi ini membaca setiap baris dari file G-code, lalu mengirimkannya ke mesin menggunakan <i>sendNCCode</i> dengan jeda 0,1 detik di antara setiap perintah. Jika terjadi kesalahan saat membaca file, pesan error akan ditampilkan.</p>
<pre>def sendRobotProgram(file_path, com):     try:         with open(file_path, 'r') as file:             lines = file.readlines()             for line in lines:                 sendNCCode(line.strip(), com) # Mengirimkan setiap baris program ke robot SCARA                 sleep(0.1) # Memberikan jeda antara setiap perintah     except Exception as e:         print(f"Error saat membaca file program robot: {e}")</pre>	<p>Fungsi <i>sendRobotProgram</i> digunakan untuk mengirimkan program dari file ke robot SCARA melalui komunikasi serial. Fungsi ini membaca setiap baris dari file program, lalu mengirimkannya ke robot menggunakan <i>sendNCCode</i> dengan jeda 0,1 detik di antara setiap perintah. Jika terjadi kesalahan saat membaca file, pesan error akan ditampilkan.</p>
<pre>def readAndSendFile(file_path, com, type_program):     """ Fungsi untuk membaca file dan mengirimkan perintah ke perangkat """     print(f"Reading {type_program} program from {file_path}")     if type_program == "CNC":         sendGCodeFile(file_path, com)     elif type_program == "SCARA":         sendRobotProgram(file_path, com)     else:         print("Unknown program type")</pre>	<p>Fungsi <i>readAndSendFile</i> bertugas membaca file dan mengirimkan perintah ke perangkat sesuai jenis program. Jika parameter <i>type_program</i> bernilai "CNC", maka fungsi akan memanggil <i>sendGCodeFile</i> untuk mengirimkan G-code ke mesin CNC. Jika bernilai "SCARA", maka akan memanggil <i>sendRobotProgram</i> untuk mengirimkan perintah ke robot SCARA. Jika jenis program tidak dikenali, fungsi menampilkan pesan "Unknown program type".</p>
<pre>def sendNCCode(cd, com):     print ("sendNCCode()", cd)      feedback = com.readline()      cd = cd + '\n'     code2 = cd.encode('utf-8') #+ '\n' #Ini penting agar Karakter bisa dikirim dan diterima GRBL</pre>	<p>fungsi <i>sendNCCode</i> digunakan untuk mengirimkan perintah NC (Numerical Control) ke perangkat melalui komunikasi serial. Fungsi ini mencetak perintah yang dikirim, membaca umpan balik awal dari perangkat, kemudian mengonversi perintah menjadi format UTF-8 yang kompatibel dengan controller GRBL sebelum dikirim. Proses ini memungkinkan controller menerima dan menjalankan instruksi yang diberikan.</p>
<pre>if com.isOpen():     com.flushInput()     com.flushOutput()     com.write(code2) if com.isOpen():     com.flushInput()     com.flushOutput()     com.write(code2)     com.flush() feedback = com.readline() print ("Execute_line : ", cd , "&gt;&gt;&gt;Balasan GRBL:", feedback) # if feedback == b'[MSG:Pgm End]\r\n': #     return if feedback == b'[VER:CNC1.1h.20190825:]\r\n' :     return "CNC" if feedback == b'[VER:SCARA1.1h.20190825:]\r\n' :</pre>	<p>Kode tersebut berfungsi untuk mengirimkan perintah ke controller (CNC atau SCARA) melalui komunikasi serial. Jika port serial terbuka (<i>com.isOpen()</i>), sistem akan membersihkan buffer input dan output menggunakan <i>flushInput()</i> dan <i>flushOutput()</i>, kemudian mengirimkan data yang telah dikodekan (<i>code2</i>) ke controller. Setelah pengiriman, buffer dikosongkan lagi menggunakan <i>flush()</i> untuk memastikan tidak ada data yang tertinggal. Sistem kemudian membaca respons dari controller menggunakan <i>readline()</i>, dan mencetak perintah yang dieksekusi beserta balasan dari controller. Jika balasan menunjukkan versi perangkat ([VER:CNC1.1h.20190825:] atau [VER:SCARA1.1h.20190825:]), fungsi akan</p>

<pre>return "SCARA"</pre>	<p>mengembalikan jenis perangkat, yaitu "CNC" atau "SCARA".</p>
<pre>com.write(b'?) feedBack1= com.readline() print ("Status Kontroler:",feedBack) #if feedBack == b'[MSG:Pgm End]\r\n': #return com.write(b'?) feedBack2= com.readline() print ("Status Kontroler2:",feedBack) if cd == "M30" or cd == "M02" : while feedBack1 != b'[MSG:Pgm End]\r\n' or feedBack2 != b'[MSG:Pgm End]\r\n': feedBack1 = com.readline() print ("waiting to Finish &gt;&gt;" , cntrSkr, " jalan, ", feedBack , "    " , feedBack[1:6]) com.write(b'?) return feedBack</pre>	<p>Kode di atas memantau status controller (CNC atau SCARA) menggunakan perintah ?, yang mengirimkan permintaan status melalui port serial. Sistem membaca dan mencetak respons dari controller. Jika perintah yang dikirim adalah "M30" atau "M02" (penanda akhir program di CNC), maka program akan menunggu hingga menerima balasan "[MSG:Pgm End]\r\n", yang menandakan eksekusi program telah selesai. Looping akan terus berjalan hingga proses selesai, memastikan sinkronisasi antara pengiriman perintah dan status eksekusi di controller.</p>
<pre>COMSCARA = 0 COMCNC = 0</pre>	<p>Variabel COMSCARA dan COMCNC digunakan sebagai penanda atau penyimpan nomor port komunikasi serial yang terhubung ke robot SCARA dan mesin CNC. Nilai awal 0 menunjukkan bahwa belum ada port yang diidentifikasi atau dihubungkan. Nantinya, sistem akan memperbarui variabel ini sesuai dengan port yang terdeteksi secara otomatis.</p>
<pre>ports = serial.tools.list_ports.comports(include_links=False) info = "" nport = 0 for port in ports : print(port.device) nport = nport + 1 if nport == 1 : conSerial1 = serial.Serial(port.device, 115200,timeout=5) port1 = port.device print("Port 1:" + str(nport), port1) sendNCCode("\$X",conSerial1) controler1 =sendNCCode( "\$I",conSerial1) #infoCon1 = infoControler(port1,conSerial1) print ("Controller 1 connected ",controler1) if controler1 == "CNC": COMCNC = conSerial1 print ("PORTCNC") if controler1 == "SCARA" : COMSCARA = conSerial1 print ("PORTSCARA")</pre>	<p>Kode tersebut berfungsi untuk mendeteksi dan mengenali port serial yang terhubung ke mesin CNC atau robot SCARA. Program memindai semua port yang tersedia menggunakan fungsi serial.tools.list_ports.comports(). Jika port pertama ditemukan, koneksi dibuka dengan baud rate 115200 dan timeout selama 5 detik. Selanjutnya, program mengirimkan perintah "\$X" untuk membebaskan alarm dan "\$I" untuk mendapatkan informasi jenis kontroler. Jika respon yang diterima adalah "CNC", port disimpan dalam variabel COMCNC, sedangkan jika responnya "SCARA", port disimpan di COMSCARA. Dengan proses ini, sistem dapat secara otomatis mengidentifikasi dan membedakan perangkat CNC dan robot SCARA.</p>
<pre>if nport == 2 : conSerial2 = serial.Serial(port.device, 115200,timeout=5) port2 = port.device sendNCCode("\$X",conSerial2) controler2 =sendNCCode( "\$I",conSerial2) #info = info + " - " + port2 + " : " + controler2 controler2 =sendNCCode( "\$I",conSerial2) #infoCon = infoControler(port1,conSerial2) print ("Controller 2 connected") if controler2 == "CNC": COMCNC = conSerial2 print ("PORTCNC") if controler2 == "SCARA" : COMSCARA = conSerial2 print ("PORTSCARA connected")</pre>	<p>Kode tersebut berfungsi untuk mendeteksi dan mengenali perangkat kedua (port kedua) yang terhubung ke komputer, seperti mesin CNC atau robot SCARA. Jika port kedua ditemukan (nport == 2), maka koneksi serial dibuka menggunakan serial.Serial() dengan baud rate 115200 dan timeout 5 detik. Program mengirimkan perintah "\$X" untuk membebaskan alarm dan "\$I" untuk meminta informasi jenis kontroler. Jika respon yang diterima adalah "CNC", port tersebut disimpan di variabel COMCNC. Jika responnya "SCARA", maka disimpan di variabel COMSCARA. Dengan mekanisme ini, sistem dapat secara otomatis mengidentifikasi dan membedakan antara port CNC dan robot SCARA.</p>
<pre>if nport &lt; 2 : rint("Need 2 COM PORT, available : ", nport ) else : sendNCCode("\$X",conSerial1)</pre>	<p>Kode tersebut memeriksa jumlah port yang tersedia. Jika kurang dari dua, akan menampilkan pesan bahwa dua port diperlukan. Jika dua port tersedia, perintah "\$X" dikirim ke kedua port untuk</p>

<pre>sendNCCode("\$X",conSerial2) print("Selesai Koneksi Serial")</pre>	<p>membebaskan alarm pada perangkat. Setelah selesai, ditampilkan pesan "Selesai Koneksi Serial".</p>
<pre>gcode_file_path = "D:\cnc_laser.txt" robot_program_path = "D:\robot_scara.txt"  print ("Mengirim Program ke CNC laser") readAndSendFile(gcode_file_path, COMCNC, "CNC")  print ("Mengirim program ke Robot Scara") readAndSendFile(robot_program_path, COMSCARA, "SCARA")</pre>	<p>Kode tersebut mengirimkan program ke CNC laser dan robot SCARA. Pertama, program G-code dari file "D:\cnc_laser.txt" dikirim ke CNC menggunakan fungsi readAndSendFile. Setelah itu, program robot dari file "D:\robot_scara.txt" dikirim ke robot SCARA. Kode ini memastikan masing-masing perangkat menerima dan mengeksekusi instruksi yang sesuai.</p>

#### 4. KESIMPULAN

Berdasarkan hasil pengujian integrasi mesin \*Computer Numerical Control\* (CNC) laser Dioda dan robot *Selective Compliance Assembly Robot Arm* (SCARA) yang dikendalikan melalui aplikasi berbasis Python, dapat disimpulkan bahwa sistem kendali terintegrasi telah berhasil dikembangkan dan diimplementasikan. Aplikasi yang dibangun memanfaatkan protokol komunikasi serial sebagai antarmuka utama antara perangkat keras dan perangkat lunak. Pendekatan ini memungkinkan tercapainya sinkronisasi gerak serta koordinasi yang stabil antara mesin CNC dan robot SCARA, tanpa ditemukannya kendala signifikan selama proses pengujian berlangsung. Penggunaan bahasa Python dalam pengembangan sistem terbukti memberikan fleksibilitas yang tinggi dalam pengelolaan komunikasi perangkat, sekaligus mendukung integrasi berbagai fitur fungsional yang berkontribusi terhadap keandalan operasional sistem.

Lebih lanjut, keunggulan utama dari aplikasi yang dikembangkan terletak pada kemampuannya dalam melakukan deteksi otomatis terhadap port serial. Fitur ini memungkinkan sistem untuk mengidentifikasi dan terhubung dengan perangkat secara langsung tanpa memerlukan konfigurasi manual, sehingga mempercepat waktu inialisasi serta mengurangi potensi kesalahan akibat masukan pengguna yang tidak akurat. Dalam konteks lingkungan produksi, kemampuan deteksi otomatis ini menjadi aspek krusial karena memastikan konsistensi konektivitas perangkat CNC dan robot SCARA, bahkan ketika terjadi perubahan alokasi port atau proses pemasangan ulang koneksi perangkat keras.

#### DAFTAR PUSTAKA

- [1] I. Trinchieri, A. Tridello, and M. E. E, "Politecnico di Torino Politecnico di Torino," *Thesis*, no. October, p. 87316161, 2023.
- [2] and H. A. V. Akhmetov, Tolegen, ""An augmented reality-based warning system for enhanced safety in industrial settings.,," 2022.
- [3] D. Stimpson, "Process Automation and Robotics Engineering for Industrial Processing Systems.,," 2023.
- [4] D. Sivkov, "Entwicklung und Simulation der Steuerung ei- nes dreiachsigen Positioniertisches für den Ge- nauigkeitsnachweis von Laserscannern im dy- namischen Betrieb," 2014.
- [5] A. Stevens, "A Robotic System for Photopatterning of Freeform Surfaces," pp. 1–88, 2014.
- [6] M. K. Salah, "Wire Direct Energy Deposition of Cobalt Iron Alloy Wire Direct Energy Deposition of Cobalt Iron Alloy," 2025.
- [7] "Escuela Superior Politécnica De Chimborazo Elaboración De Figuras En Relieve Enfocado a," 2018.
- [8] L. C. M. Yip, "Development of a Near-Full-View Angle Coverage Photoacoustic Tomography System and Its Application Towards Optical Fluence Distribution Imaging.,," 2022.
- [9] H. Li *et al.*, "Squid: Simplifying Quantitative Imaging Platform Development and

- Deployment,” *bioRxiv*, 2020, doi: 10.1101/2020.12.28.424613.
- [10] R. Jabri, M., Sukhon, A., & Qadi, “CNC Plotter,” 2022.
- [11] A. Giri, A., Akib, A. S. M. A., Hasib, A., & Acharya, “Design and Development of a Cost-Effective Modular CNC Plotter for Educational Applications.,” 2025.
- [12] N. Miljković, Z., & Slavković, “Development of a Domestic 4-Axis SCARA Robot.,” 2023.
- [13] I. Fayez, M. Qahtan, and S. Riyadh, “Automatic Laser Engraving Machine for Different Materials based on Microcontroller,” vol. 19, pp. 323–331, 2023.
- [14] B. Harahap, H. A., & Satria, “Design and Construction of CNC Machines Based on Microcontrollers.,” 2024.
- [15] P. Appunu, “Control System for Three-Axis CNC Milling Machine with Automatic Tool Change.,” 2023.