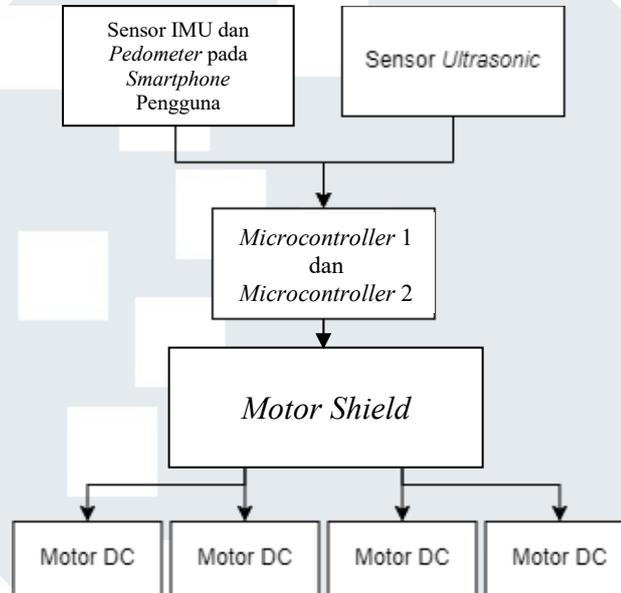


BAB III

PERANCANGAN DAN IMPLEMENTASI SISTEM

3.1 Tinjauan Desain Subsistem



Gambar 3.1 – DFD Level 2 Subsistem *User Follower*

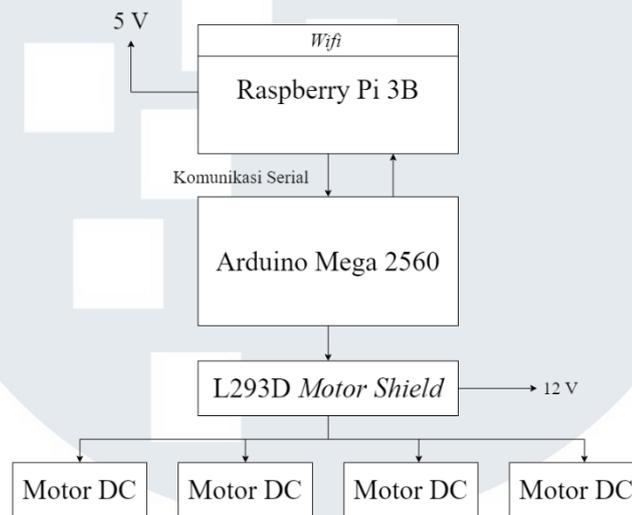
Tabel 3.1 – Penjelasan DFD Level 2 Subsistem *User Follower*

| Parameter | Keterangan |
|-----------|--|
| Input | <ul style="list-style-type: none">• Data yang didapatkan dari sensor IMU dan <i>pedometer</i> <i>smartphone</i>• Halangan di depan <i>Smart Shopping Cart</i> |
| Output | <ul style="list-style-type: none">• Pergerakan dari motor untuk mengikuti pengguna• Pergerakan dari motor untuk menghindari dari halangan |
| Fungsi | <ul style="list-style-type: none">• Memperkirakan lokasi <i>Smart Shopping Cart</i> dengan mengolah data kecepatan roda dan waktu tempuh• Mengolah jarak dan arah yang harus ditempuh oleh <i>Smart Shopping Cart</i> untuk mengikuti pengguna• Mengendalikan motor dalam pergerakan mengikuti pengguna maupun menghindari halangan yang ada |

Berdasarkan Gambar 3.1 dan Tabel 3.1 yang menunjukkan DFD Level 2 dari subsistem *user follower* yang akan diimplementasikan pada *Smart Shopping Cart*, terlihat bahwa subsistem *user follower* ini bertujuan untuk membuat *cart* dapat mengikuti pengguna secara otomatis setelah dihubungkan ke *smartphone* pengguna melalui *wifi*. Setelah *smartphone* dan *cart* berhasil dihubungkan, maka

smartphone akan mengirimkan data sensor IMU dan *pedometer* secara berkala melalui *wifi* yang sama. Sensor IMU ini bertujuan untuk memperkirakan *heading* dari *smartphone* pengguna, sedangkan sensor *pedometer* bertujuan untuk memperkirakan jarak yang telah dilalui pengguna.

Setelah *cart* berhasil memperkirakan arah dan jarak yang dilewati oleh *smartphone* pengguna, maka *cart* akan melakukan hal yang sama dengan kecepatan konstan. Selain itu, *cart* juga akan dilengkapi oleh sensor ultrasonic untuk mendeteksi adanya halangan.



Gambar 3.2 – *Wiring Diagram Subsystem User Follower*

Seperti yang terlihat pada Gambar 3.2, rangkaian untuk subsistem *user follower* adalah dengan menghubungkan Raspberry Pi sebagai penerima data dari *smartphone* melalui Wifi. Kemudian Raspberry Pi dihubungkan secara serial dengan Arduino Mega karena Arduino Mega berperan sebagai pengendali keempat motor DC melalui *motor shield*.

3.2 Implementasi Sistem

3.2.1 Hasil Implementasi

Dalam mengimplementasikan subsistem *user follower* ini, *smartphone* pengguna memiliki peran yang penting. Karena untuk mengetahui arah dan jarak yang dilalui pengguna memerlukan data-data dari sensor IMU dan *pedometer* pada *smartphone* pengguna selama pengguna berbelanja. Data-data tersebut diakses menggunakan aplikasi dan kemudian harus dikirimkan ke *microcontroller* Raspberry Pi untuk diolah menjadi arah dan jarak yang perlu dilalui oleh *cart* untuk mengikuti pengguna. Hasil olahan data pada Raspberry Pi kemudian dikirimkan ke Arduino Mega sebagai perintah untuk menjalankan

motor sehingga *cart* dapat mengikuti manusia. Tampilan aplikasi Smart Shopping Cart dapat dilihat pada Gambar 3.3.

| | | |
|--|--|--|
| Username: <input type="text" value="insert username"/> Password: <input type="password" value="insert password"/> <input type="button" value="Login"/> | IP ADDRESS <input type="text" value="insert Cart's IP Address"/> <input type="button" value="SUBMIT"/> | <input type="button" value="CONNECT"/> <input type="button" value="DISCONNECT"/> MAGNETOMETER X 22.86 Y 5.76 Z 25.74 PEDOMETER 6 |
|--|--|--|

Gambar 3.3 – Aplikasi *Smart Shopping Cart*

Dalam proses implementasi subsistem *user follower* pada *Smart Shopping Cart* ini, *smartphone* pengguna dan Raspberry Pi harus dikoneksikan ke *wifi* yang sama, dan Raspberry Pi perlu terkoneksi ke Arduino Mega secara serial. Arduino Mega merupakan *microcontroller* yang berperan sebagai pengendali motor sehingga selain terhubung ke Raspberry Pi, juga dihubungkan ke *motor shield*. *Motor shield* yang digunakan harus terhubung dengan keempat motor DC sebagai aktuator untuk menjalankan *cart*. Sumber tegangan yang digunakan adalah *accu* yang akan dihubungkan ke *step down* untuk kemudian dihubungkan ke Raspberry Pi agar dapat menyalakan Raspberry Pi. Gambar 3.4 memperlihatkan letak dari subsistem *user follower* ini dimana Arduino Mega, Raspberry Pi, *step down*, *accu*, serta *motor shield* dan motor DC diletakkan di bagian bawah *cart*. Sedangkan di sisi depan atas dan bawah terpasang sensor ultrasonik untuk mendeteksi halangan selama *cart* bergerak mengikuti pengguna.



Gambar 3.4 – Letak Subsistem *User Follower*

3.2.2 Hambatan Implementasi

Awal mulanya subsistem *user follower* ini menghubungkan *smartphone* dengan Raspberry Pi serta mengirimkan data-data sensor *smartphone* melalui *Bluetooth*. Namun pada saat proses implementasi subsistem *user follower*, terdapat masalah bahwa aplikasi yang dibuat melalui MIT Inventor App untuk mengakses sensor dan mengirimkan

data sensor pada *smartphone* tidak dapat mendeteksi perangkat *Bluetooth* yang ada di sekitar *smartphone*. Setelah membaca berbagai sumber dan forum dari MIT Inventor App sendiri, dapat diasumsikan bahwa masalah ini disebabkan oleh versi terbaru MIT Inventor App baru saja diperbaharui untuk Android versi 12, sedangkan OS Android terbaru adalah versi 13.

Dalam proses pengolahan sensor IMU dan *pedometer* di dalam Raspberry Pi, ditemukan masalah karena sensor *accelerometer* dan *gyroscope* dari *smartphone* nilainya terlalu berbeda jauh dan tidak stabil, menghasilkan hasil perhitungan yang berbeda-beda dengan selisih yang berbeda jauh. Bantuan filter dari Kalman Filter juga tidak membantu banyak dalam menstabilkan nilai *accelerometer* dan *gyroscope* tersebut. Sensor IMU ini berperan penting karena digunakan untuk mengestimasi *heading* atau arah dari *smartphone* pengguna.

Selama proses *cart* mengikuti pengguna, dibutuhkan *input* terus-menerus secara berkala oleh sensor ultrasonik. Namun sensor ultrasonik tidak dapat dihubungkan ke Arduino Mega karena terdapat subsistem lain yang menggunakan sensor inframerah lebih dari satu sehingga sinyal yang diterima sensor ultrasonik tidak beraturan dan menjadi tidak *valid*. Sensor ultrasonik juga tidak dapat dihubungkan ke Raspberry Pi karena *Apache Server* tidak memiliki ijin akses seluruh library Python di dalam Raspberry Pi sehingga tidak dapat mengakses modul *time*, yang mana merupakan modul yang paling penting dalam perhitungan jarak melalui sensor ultrasonik. *Apache server* ini merupakan sebuah *local server* yang digunakan untuk membuat Raspberry Pi menjadi *server* yang kemudian akan diakses oleh *smartphone* untuk mengirimkan data-datanya.

Selain tidak dapat mengakses modul *time* yang merupakan modul bawaan Python di Raspberry Pi, *Apache Server* juga tidak memiliki akses untuk menjalankan *script* Python untuk mengirim data ke Arduino Mega melalui komunikasi serial. Akses ini hanya dimiliki oleh *root* dan *user*. Sehingga diperlukan alternatif agar *Apache Server* atau lebih sering dikenal sebagai *www-data* di dalam Raspberry Pi dapat memiliki akses ke serial port yang terhubung ke Arduino Mega.

3.2.3 Solusi yang Diterapkan

Solusi yang ditemukan dan diterapkan untuk permasalahan *Bluetooth* ini adalah dengan mencari alternatif lain untuk menghubungkan *smartphone* dengan *cart* melalui aplikasi. Alternatif lain yang ditemukan adalah melalui *wifi*. Dengan memanfaatkan *Apache Server*, Raspberry Pi dibuat menjadi *local server* yang dapat diakses

dengan menghubungkan *smartphone* ke *wifi* yang sama dengan yang digunakan oleh Raspberry Pi. Sehingga *smartphone* tetap dapat mengirimkan data-data ke Raspberry Pi.

Solusi untuk proses pengolahan data sensor IMU dan *pedometer* adalah mencari cara estimasi *heading* lainnya tanpa menggunakan sensor *accelerometer* dan *gyroscope*. Sehingga sensor IMU yang digunakan hanyalah sensor *magnetometer*. Sensor ini sendiri menghasilkan *raw heading* sehingga hanya dengan menggunakan sensor *magnetometer*, Raspberry Pi dapat mengestimasi *heading* dari *smartphone* pengguna namun tanpa proses filter dan nilai *heading* tidak diperhalus dengan sensor lain.

Solusi yang ditemukan dan diterapkan untuk permasalahan sensor ultrasonik adalah dengan menghilangkan sensor ultrasonik dan menggantinya dengan alternatif lain. Namun dengan ketiadaan sensor ultrasonik ini maka fungsi *obstacle detection* pada *Smart Shopping Cart* ini tidak dapat dijalankan. Selain itu, tetap diperlukan alternatif lain agar *cart* dapat berhenti ketika *cart* mencapai jarak yang sama dengan pengguna.

Solusi untuk permasalahan akses *www-data* ke *serial port* adalah dengan memberikan akses *serial port* *ttyACM0* ke *www-data* melalui perintah *sudo thunar* di terminal Raspberry Pi. *ttyACM0* merupakan nama untuk *serial port* yang terhubung ke Arduino Mega di dalam Raspberry Pi. *Sudo thunar* merupakan sebuah perintah yang dijalankan di dalam terminal Raspberry Pi untuk mengakses seluruh *folder* dan *file* yang ada di dalam Raspberry Pi sebagai *root*. Melalui *sudo thunar*, *permission* untuk *ttyACM0* diganti menjadi *www-data*. Namun setiap kali Raspberry Pi dinyalakan atau *reboot*, perlu memberikan lagi akses *serial port* tersebut ke *www-data* atau *Apache Server* melalui *sudo thunar*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A