

BAB IV

HASIL DAN PEMBAHASAN

4.1 Deskripsi Alat Penghitung Pengunjung

Proyek ini mencakup rancangan dan pembuatan sebuah alat penghitung pasien otomatis berbasis mikrokontroler yang diimplementasikan di dalam bus unit Palang Merah Indonesia Kabupaten Labuhanbatu. Alat ini dirancang dengan tujuan khusus untuk melakukan pemantauan dan perhitungan jumlah pasien yang ada di dalam bus unit donor darah, mengoptimalkan proses operasional serta efisiensi dalam pelayanan medis.

Dalam implementasinya, alat ini dilengkapi dengan berbagai komponen yang mendukung fungsinya. Dua lampu sinyal, satu berwarna merah dan satu lagi berwarna hijau, dipasang secara strategis untuk memberikan petunjuk visual kepada petugas dan masyarakat sekitar. Lampu merah menunjukkan indikasi bus telah mencapai kapasitas maksimal dan tidak dapat menerima lebih banyak pasien. Sebaliknya, lampu hijau menandakan bahwa bus masih dapat menerima pasien lebih lanjut.

Mekanisme kerja alat ini didasarkan pada teknologi mikrokontroler yang memiliki kemampuan untuk menghitung secara akurat jumlah pasien yang masuk dan keluar dari bus. Sensor yang terpasang di pintu masuk dan keluar bus akan mendeteksi setiap pergerakan, dan informasi ini akan diolah oleh mikrokontroler.

Data perhitungan kemudian ditampilkan pada layar LCD untuk memberikan informasi yang jelas kepada petugas.

Selain itu, alat ini juga mencakup sistem komunikasi yang memungkinkan pengiriman data real-time kepada petugas di luar bus. Hal ini memungkinkan pengambilan keputusan yang lebih cepat dan akurat dalam mengatur lalu lintas pasien di dalam bus. Selain memberikan kenyamanan bagi pasien, alat ini juga membantu meningkatkan efisiensi dan kecepatan pelayanan medis di lapangan.

4.1.1 Konsep Dasar

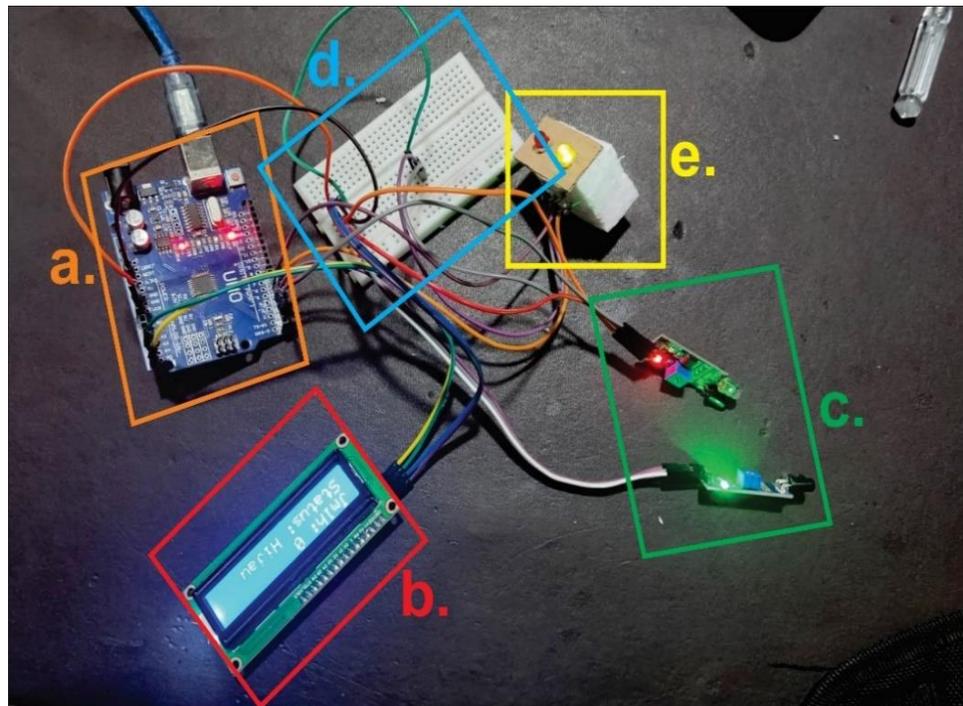
Proyek rancang bangun alat penghitung pasien otomatis berbasis mikrokontroler di bus unit Palang Merah Indonesia Kabupaten Labuhanbatu didasarkan pada pendekatan yang mengaplikasikan teknologi sensor guna mengidentifikasi dan melacak keberadaan individu di dalam bus. Alat ini mengadopsi prinsip penggunaan dua sensor Infrared yang ditempatkan dengan strategis, masing-masing terletak pada pintu masuk dan keluar bus, untuk melacak pergerakan orang masuk dan keluar dari ruang bus.

Kehadiran teknologi sensor Infrared ini berfungsi sebagai mata elektronik yang sensitif terhadap perubahan suhu yang dihasilkan oleh benda atau individu yang melewati daerah deteksinya. Ketika ada pergerakan, sensor akan mengirimkan sinyal elektronik ke unit pemrosesan data yang terintegrasi dalam mikrokontroler. Unit ini kemudian melakukan analisis atas sinyal yang diterima, mengenali pola perubahan, dan menghitung jumlah pasien yang memasuki serta keluar dari bus.

Hasil perhitungan yang dihasilkan oleh mikrokontroler kemudian ditampilkan dengan jelas pada layar LCD. Ini memungkinkan petugas dan staf medis di dalam bus untuk secara langsung mengamati jumlah pasien yang sedang berada dalam bus, memberikan panduan berharga terkait kapasitas serta kebutuhan perawatan medis.

4.1.2 Komponen Alat

RANCANG BANGUN ALAT PENGHITUNG PASIEN OTOMATIS BERBASIS MIKROKONTROLER DI BUS UNIT PALANG MERAH INDONESIA KABUPATEN LABUHANBATU terdiri dari beberapa komponen utama, antara lain:



Gambar 3. 1 Komponen Alat

a. Unit Pemrosesan Data memiliki peran utama dalam mengelola dan menganalisis data yang diterima dari sensor, dengan fokus pada perhitungan jumlah pengunjung yang memasuki atau meninggalkan ruangan. Dalam kerangka proyek ini, alat penghitung memanfaatkan sebuah unit pemrosesan yang menjadi inti sistem, dan unit yang digunakan untuk tujuan ini adalah Arduino Uno. Unit pemrosesan ini memainkan peran sentral dalam mengekstrak informasi relevan dari data sensorik, melakukan analisis yang akurat, dan menghasilkan perhitungan yang akurat mengenai kehadiran pengunjung di dalam ruangan.

b. Tampilan: Alat ini dilengkapi dengan tampilan yang menampilkan informasi jumlah pengunjung secara real-time. Tampilan dapat berupa LCD, LED, atau tampilan lain yang mudah dibaca. Alat yang digunakan untuk menampilkan informasi pada alat ini adalah LCD 16X2 yang sudah dilengkapi dengan modul I2C.

c. Alat bantu fisik: Dalam kerangka memahami peran alat bantu fisik dalam mendukung pengembangan dan pemanfaatan mikrokontroler, kita dapat melihat bahwa istilah "alat bantu fisik" mengacu pada beragam perangkat atau komponen yang secara khusus dirancang untuk memfasilitasi berbagai aspek yang terkait dengan penggunaan serta pengembangan mikrokontroler. Dengan demikian, alat bantu fisik tidak hanya sekadar komponen tambahan, tetapi juga menjadi elemen penting yang turut berperan dalam keberhasilan suatu proyek atau aplikasi yang melibatkan mikrokontroler.

Sebagai contoh konkret, dalam konteks proyek-proyek yang melibatkan mikrokontroler, kita dapat menemukan bahwa alat dukung yang menjadi andalan adalah breadboard. Breadboard, yang secara harfiah dapat diterjemahkan sebagai

"papan roti", muncul sebagai salah satu alat bantu fisik yang sangat relevan dalam menghadapi kerumitan pengembangan elektronik. Ini adalah platform fisik yang dirancang untuk memungkinkan pengembang atau ahli elektronik untuk dengan mudah menyusun dan menghubungkan komponen elektronik yang berbeda-beda dalam suatu sirkuit prototipe.

Melalui desain yang inovatif, breadboard memungkinkan para pengguna untuk menciptakan sirkuit yang kompleks tanpa memerlukan soldering atau penghubung permanen lainnya. Ini berarti bahwa pengembang dapat dengan bebas mengatur dan menghubungkan komponen sesuai kebutuhan proyek mereka, memungkinkan eksperimen dan iterasi yang cepat.

Dengan kata lain, peran breadboard sebagai alat bantu fisik dalam proyek-proyek yang melibatkan mikrokontroler adalah sangat penting. Ini memberikan kemampuan untuk menguji konsep, mengembangkan prototipe, dan merancang solusi yang lebih efektif sebelum menjalankan ke tahap produksi yang lebih lanjut. Oleh karena itu, alat dukung semacam breadboard bukan hanya sekadar komponen fisik, tetapi merupakan salah satu fondasi yang memungkinkan inovasi dan pengembangan teknologi berdasarkan mikrokontroler.

d. Alat peringatan : perangkat yang dirancang untuk memberikan tanda atau sinyal yang dapat memberikan peringatan atau peringatan kepada pengguna terhadap situasi atau kondisi tertentu. Alat peringatan yang digunakan dalam proyek ini adalah lampu LED yang warna merah dan hijau. Merah berfungsi untuk memberikan informasi atau peringatan bahwa ruangan telah penuh, sedangkan

hijau berfungsi untuk memberikan peringatan bahwa ruangan dalam kondisi masih bisa di masuki oleh pasien.

4.1.3 *Software*

Untuk software yang digunakan adalah Arduino IDE. Software ini sangat berperan penting terhadap pengolahan uji coba alat dikarenakan software ini berguna untuk mengisikan inputan data atau perintah kedalam modul mikrokontroler agar alat yang telah di rakit sedemikian rupa dapat beroperasi sesuai yang diinginkan.



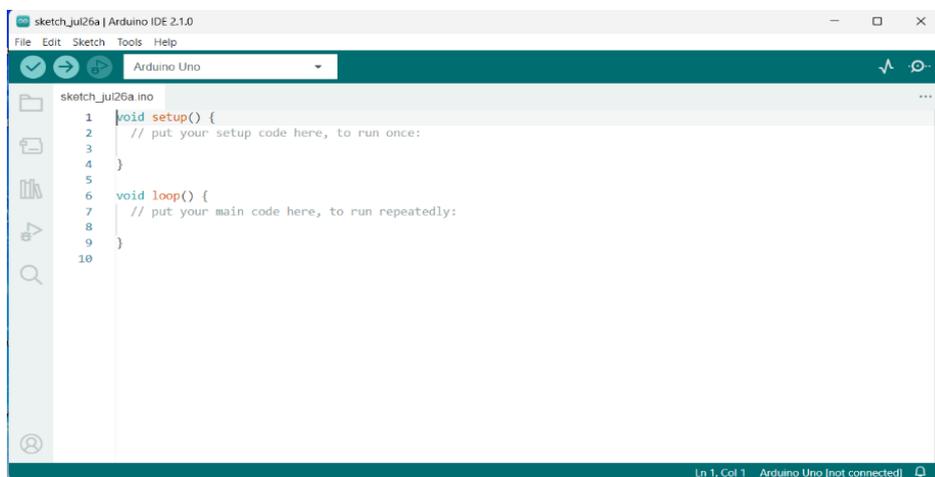
Gambar 3. 2 *Arduino IDE*

Berikut langkah-langkah untuk memasukkan data atau perintah kedalam modul arduino.

1. Instalasi dan buka aplikasi

Tahap awal yang perlu dilakukan untuk memulai proses pemasangan perintah adalah mengunduh dan menginstal perangkat lunak (software)

yang relevan pada perangkat komputer atau laptop, terutama jika perangkat tersebut belum memiliki instalasi Arduino IDE. Apabila Arduino IDE belum terpasang, langkah pertama ini mencakup tindakan mengunduh dan menginstal software tersebut guna menciptakan lingkungan kerja yang optimal untuk pengembangan perangkat keras. Setelah memastikan kehadiran atau instalasi Arduino IDE, langkah selanjutnya adalah membuka aplikasi Arduino IDE itu sendiri. Pada titik ini, pengguna akan dihadapkan pada antarmuka grafis yang khas dari perangkat lunak tersebut. Visualisasi antarmuka ini mencakup elemen-elemen penting seperti menu, toolbar, panel kode, dan lain-lain. Tampilan yang dihasilkan oleh software Arduino IDE menghadirkan kesan yang beragam, menampilkan elemen-elemen pengaturan dan kontrol yang memberikan akses langsung untuk memulai berbagai jenis proyek, termasuk pengembangan dan pengujian perangkat keras berbasis mikrokontroler.

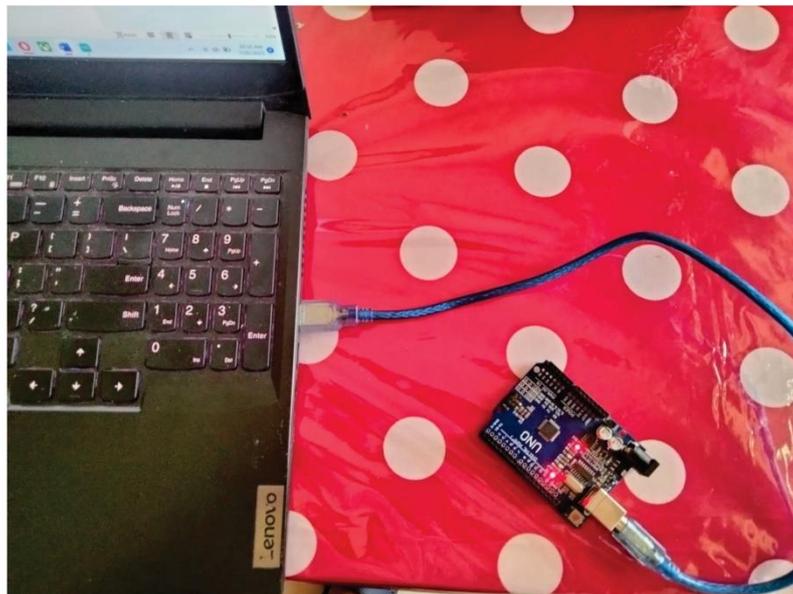


Gambar 3. 3 Tampilan Arduino IDE

2. Langkah kedua dalam proses ini adalah melakukan penghubungan antara papan mikrokontroler Arduino dengan perangkat komputer atau laptop. Hal pertama yang perlu dilakukan sebelum memulai penginputan source code adalah memastikan modul mikrokontroler terhubung secara fisik dengan perangkat komputer atau laptop menggunakan kabel USB yang disediakan oleh Modul Mikrokontroler. Langkah ini memungkinkan papan mikrokontroler Arduino untuk menjadi "jembatan" antara dunia digital dan perangkat komputer. Kabel USB bawaan dari Modul Mikrokontroler digunakan untuk menghubungkan port USB pada papan mikrokontroler dengan port USB pada perangkat komputer atau laptop. Ini adalah tahap awal yang krusial sebelum source code dapat dimasukkan dan dijalankan pada modul mikrokontroler.

Saat modul mikrokontroler berhasil terhubung dengan perangkat komputer, langkah selanjutnya adalah mengimpor source code ke dalam perangkat tersebut. Source code merupakan kumpulan instruksi pemrograman yang akan memberi tahu mikrokontroler apa yang harus dilakukan. Dalam fase ini, Anda dapat menggunakan berbagai perangkat lunak pengembangan Arduino yang memungkinkan Anda menulis, mengedit, dan mengimpor source code ke modul mikrokontroler.

Dengan demikian, tahap penghubungan dan pengimputan source code menjadi fondasi yang esensial dalam proses pengembangan dengan mikrokontroler Arduino. Kualitas dan keberhasilan proyek Anda dalam mengendalikan mikrokontroler secara efektif sangat bergantung pada langkah-langkah akurat ini, yang mulai dari menghubungkan modul mikrokontroler ke perangkat komputer hingga mengimpor source code yang sesuai.



Gambar 3. 4 Menghubungkan *Arduino* dengan Laptop/PC

3. Setelah langkah penghubungan selesai dilakukan dengan sukses, langkah selanjutnya yang perlu diambil adalah melakukan proses pengimputan source code yang sesuai dengan kebutuhan alat yang akan dioperasikan. Tahapan ini memiliki peranan penting dalam

memastikan bahwa alat dapat berfungsi sesuai dengan harapan dan tujuan yang telah ditentukan sebelumnya.

Penginputan source code ini merupakan proses di mana Anda akan menuliskan serangkaian instruksi pemrograman yang akan diberikan kepada alat atau perangkat mikrokontroler. Instruksi-instruksi ini akan mengatur bagaimana alat akan berinteraksi dengan lingkungannya, melakukan tugas-tugas tertentu, dan menghasilkan output yang diinginkan.

Proses ini sangat tergantung pada jenis dan tujuan alat yang sedang dikembangkan. Jika Anda, misalnya, sedang membangun alat untuk mengukur suhu, maka source code yang diperlukan akan berisi instruksi-instruksi untuk membaca sensor suhu dan menampilkan hasilnya dalam bentuk yang dapat dimengerti oleh pengguna. Begitu juga jika Anda mengembangkan alat untuk mengendalikan lampu, source code akan berisi instruksi-instruksi untuk mengontrol daya lampu berdasarkan input yang diterima.

Penting untuk menjaga kejelasan dan keakuratan source code. Kode yang baik harus mudah dibaca, dimengerti, dan dikelola. Selain itu, dokumentasi yang baik juga sangat dianjurkan, agar jika suatu saat diperlukan perubahan atau pemeliharaan, Anda atau orang lain dapat dengan mudah memahami dan memodifikasi kode tersebut.

Dengan begitu, source code yang diperlukan dalam tahap ini berperan sebagai "otak" dari alat yang Anda kembangkan. Semakin baik dan

sesuai dengan kebutuhan source code yang Anda buat, semakin optimal pula performa dan fungsionalitas alat yang akan dihasilkan. Oleh karena itu, langkah penginputan source code menjadi langkah kunci dalam menghasilkan alat yang efektif dan sesuai dengan tujuan yang telah ditetapkan sebelumnya.

Berikut Sourcode yang akan diinput ke arduino uno :

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2); // Alamat I2C
dan ukuran LCD
int pinSensorMasuk = 2;
int pinSensorKeluar = 3;
int pinLedHijau = 4;
int pinLedMerah = 5;
int jumlahPengunjung = 0;
bool pengunjungMasuk = false;
bool pengunjungKeluar = false;
const int kapasitasRuangan = 4; // Kapasitas
maksimum ruangan

void setup() {
  lcd.begin(16, 2); // Inisialisasi LCD
  lcd.setBacklight(255); // Atur kecerahan LCD
  menjadi maksimum
  lcd.print("Jmlh: ");
  lcd.print(jumlahPengunjung);
  lcd.setCursor(0, 1);
  lcd.print("Status: Hijau");

  pinMode(pinSensorMasuk, INPUT);
  pinMode(pinSensorKeluar, INPUT);
  pinMode(pinLedHijau, OUTPUT);
  pinMode(pinLedMerah, OUTPUT);
}
```

```

    digitalWrite(pinLedHijau, HIGH); // Hidupkan LED
hijau pada awalnya
    digitalWrite(pinLedMerah, LOW); // Matikan LED
merah pada awalnya

    Serial.begin(9600);
}

void loop() {
    int statusSensorMasuk =
digitalRead(pinSensorMasuk);
    int statusSensorKeluar =
digitalRead(pinSensorKeluar);

    if (statusSensorMasuk == HIGH &&
!pengunjungMasuk) {
        delay(50); // Debounce

        // Cek apakah pengunjung sudah masuk sebelumnya
        if (statusSensorMasuk == HIGH &&
!pengunjungMasuk) {
            pengunjungMasuk = true;
            if (jumlahPengunjung < kapasitasRuangan) {
                jumlahPengunjung++;
                lcd.clear();
                lcd.setBacklight(255); // Atur kecerahan
LCD menjadi maksimum
                lcd.print("Jmlh: ");
                lcd.print(jumlahPengunjung);
                lcd.setCursor(0, 1);
                lcd.print("Status: Hijau");
                digitalWrite(pinLedHijau, HIGH);
                digitalWrite(pinLedMerah, LOW);
                delay(500);
            } else {
                lcd.clear();
                lcd.setBacklight(255); // Atur kecerahan
LCD menjadi maksimum
                lcd.print("Ruangan Penuh!");
                lcd.setCursor(0, 1);
                lcd.print("  Status: Merah");
                digitalWrite(pinLedHijau, LOW);
                digitalWrite(pinLedMerah, HIGH);
            }
        }
    }
}

```

```

    } else if (statusSensorMasuk == LOW) {
        pengunjungMasuk = false;
    }

    if (statusSensorKeluar == HIGH &&
!pengunjungKeluar) {
        delay(50); // Debounce

        // Cek apakah pengunjung sudah keluar
        sebelumnya
        if (statusSensorKeluar == HIGH &&
!pengunjungKeluar && jumlahPengunjung > 0) {
            pengunjungKeluar = true;
            jumlahPengunjung--;
            lcd.clear();
            lcd.setBacklight(255); // Atur kecerahan LCD
menjadi maksimum
            lcd.print("Jmlh: ");
            lcd.print(jumlahPengunjung);
            lcd.setCursor(0, 1);
            lcd.print("Status: Hijau");
            digitalWrite(pinLedHijau, HIGH);
            digitalWrite(pinLedMerah, LOW);
            delay(500);
        }
    } else if (statusSensorKeluar == LOW) {
        pengunjungKeluar = false;
    }

    if (jumlahPengunjung >= kapasitasRuangan) {
        lcd.setCursor(0, 1);
        lcd.print("  SUDAH PENUH!!");
        digitalWrite(pinLedHijau, LOW);
        digitalWrite(pinLedMerah, HIGH);
    }
}

```

4. Setelah Anda telah memasukkan source code yang telah disiapkan di atas ke dalam perangkat lunak Arduino IDE, langkah selanjutnya adalah memastikan bahwa pengaturan port yang dituju telah diatur

dengan benar. Hal ini sangat penting agar source code yang telah Anda buat dapat dengan sukses dimuat ke dalam mikrokontroler yang sedang digunakan.

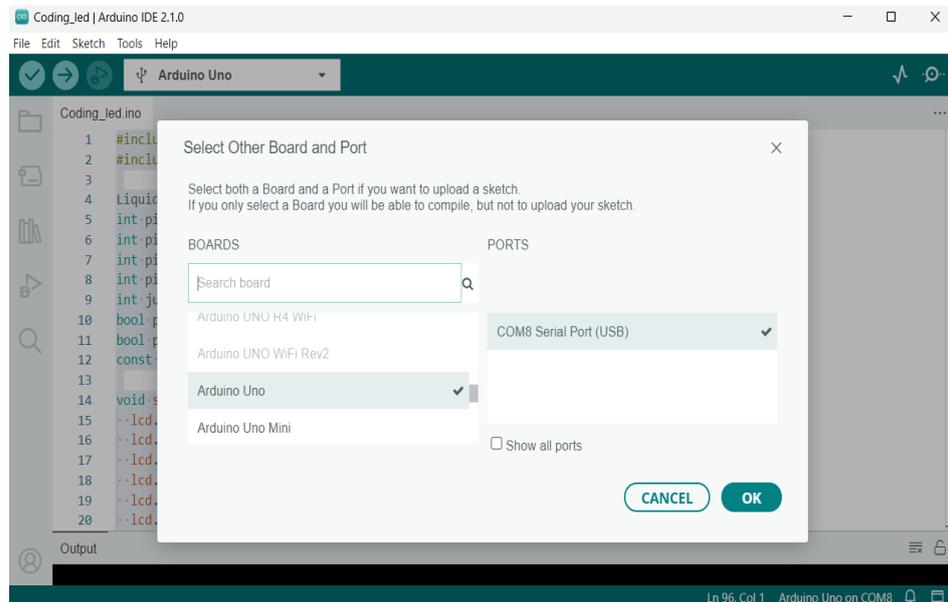
Proses memasukkan source code ke dalam mikrokontroler melibatkan komunikasi antara komputer Anda dan modul mikrokontroler. Ini dilakukan melalui koneksi port USB atau port komunikasi lainnya. Pengaturan port ini menginformasikan Arduino IDE tentang perangkat yang akan dihubungkan dan ditemui di port mana.

Jika pengaturan port tidak benar, Arduino IDE mungkin tidak dapat menemukan dan berkomunikasi dengan mikrokontroler. Ini akan mengakibatkan source code tidak dapat dimuat atau dieksekusi dengan baik oleh mikrokontroler. Oleh karena itu, pastikan untuk memeriksa pengaturan port dengan cermat sebelum melanjutkan proses selanjutnya.

Anda dapat menentukan port yang benar melalui menu "Tools" (Alat) di Arduino IDE. Pilih "Port" (Port) dan pilih port yang sesuai dengan modul mikrokontroler yang Anda gunakan. Jika Anda tidak yakin port mana yang harus dipilih, Anda dapat mencoba dengan mencabut dan menyambungkan kembali kabel USB dari komputer ke modul mikrokontroler, kemudian periksa port yang muncul dalam menu tersebut.

Dengan mengatur port dengan benar, Anda memastikan bahwa source code yang telah Anda susun dengan teliti akan berhasil dimuat ke dalam

mikrokontroler, sehingga alat yang Anda kembangkan dapat berfungsi sesuai dengan harapan. Dengan demikian, tahap pengaturan port menjadi langkah krusial dalam proses pengembangan dan pemanfaatan mikrokontroler melalui source code yang telah Anda buat.



Gambar 3. 5 *Setting port* tujuan

5. Apabila Anda merasa yakin bahwa semuanya telah benar, saatnya mengunggah source code tersebut ke dalam mikrokontroler yang saat ini terhubung.

Langkah selanjutnya adalah menjalankan proses unggah (upload) source code yang telah Anda buat ke dalam mikrokontroler yang sedang dihubungkan. Proses ini memungkinkan instruksi-instruksi yang telah Anda tuliskan dalam source code untuk dimuat ke dalam mikrokontroler, sehingga mikrokontroler memiliki pemahaman tentang tugas dan fungsinya sesuai dengan source code yang telah Anda susun.

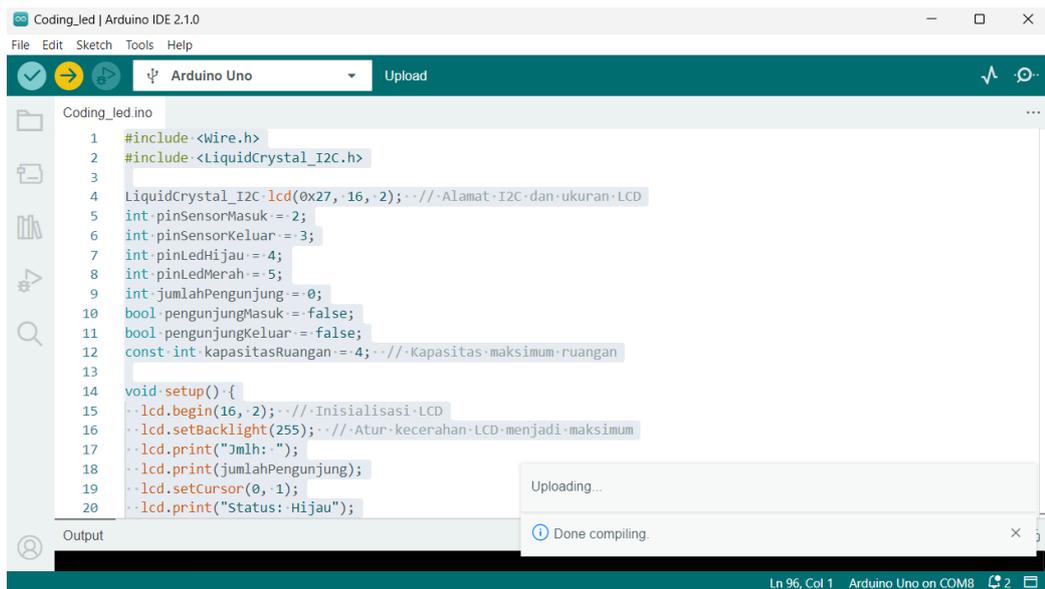
Langkah ini melibatkan interaksi antara perangkat komputer dan mikrokontroler melalui kabel USB atau koneksi komunikasi lainnya. Perangkat lunak Arduino IDE akan membantu mengelola proses pengunggahan ini. Pastikan mikrokontroler yang dihubungkan terdeteksi dengan benar oleh perangkat lunak dan port yang digunakan telah diatur sesuai.

Setelah langkah-langkah ini dilakukan, proses unggah dapat dimulai dengan mengklik tombol "Upload" (Unggah) yang tersedia di perangkat lunak Arduino IDE. Perangkat lunak akan mengompilasi source code, mengonversinya ke dalam format yang dapat dimengerti oleh mikrokontroler, dan mengirimkannya ke mikrokontroler melalui koneksi yang telah ditentukan.

Setelah proses unggah selesai, source code Anda akan menjadi bagian dari mikrokontroler, dan mikrokontroler akan mulai menjalankan instruksi sesuai dengan yang telah Anda tuliskan. Ini adalah tahap penting dalam mengaktifkan fungsionalitas alat atau sistem yang sedang Anda kembangkan.

Dengan mengikuti langkah-langkah ini dengan cermat, Anda akan memastikan bahwa source code yang telah Anda buat dapat dijalankan dengan sukses pada mikrokontroler, dan alat yang Anda kembangkan akan berfungsi sesuai dengan tujuan dan harapan yang telah Anda tetapkan sebelumnya. Setelah pemberitahuan "Done Compiling" telah keluar, berarti sourcecode telah berhasil di input ke mikrokontroler.

Setelah sourcecodenya diinputkan ke mikrokontroler, langkah selanjutnya memasang perangkat penunjang yang di perlukan ke mikrokontroler sesuai dengan aturan dari sourcecode yang diinputkan.



Gambar 3. 6 Pemberitahuan selesai *upload*

4.2 Uji Coba Alat

Ujicoba fungsionalitas alat ini dilakukan dengan mengimplementasikannya pada bus PMI Labuhanbatu yang tengah menjalankan operasi sebagai unit donor dalam proses pengumpulan dan distribusi sumbangan darah. Dalam konteks ini, alat yang telah dikembangkan akan diuji dalam situasi operasional nyata, di mana bus PMI Labuhanbatu berperan sebagai sumber donor bagi sumbangan darah.

Melalui pengujian ini, alat akan diintegrasikan dengan infrastruktur bus PMI, dan kehandalannya dalam mendeteksi parameter yang relevan, mengolah data, serta memberikan informasi yang akurat dan tepat waktu akan diuji secara menyeluruh. Hasil dari uji coba ini akan memberikan gambaran tentang kemampuan alat dalam meningkatkan efisiensi dan efektivitas proses pengumpulan darah dari unit donor, dan oleh karena itu, memiliki dampak yang potensial dalam memperkuat sistem pendonoran darah yang ada.

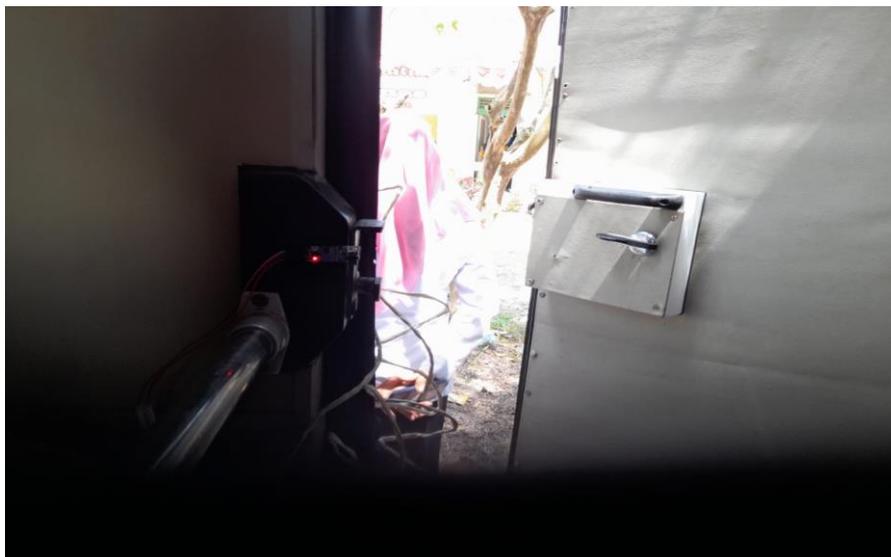


Gambar 3. 7 Kegiatan Unit Donor Darah PMI Labuhanbatu

Dalam rangka menjalankan proses pendonoran darah keliling yang mencakup berbagai tempat destinasi, alat ini dipasang dengan strategis di kedua pintu masuk dan keluar bus yang beroperasi. Dengan demikian, alat ini akan berfungsi sebagai pendukung integral dalam seluruh perjalanan bus yang melibatkan unit donor. Pemasangan di kedua pintu tersebut memungkinkan alat untuk secara efektif memantau masuk dan keluarnya setiap donor potensial.

Proses pemasangan ini memastikan bahwa alat dapat mendeteksi parameter darah yang diperlukan dengan tepat, menghasilkan data yang akurat, dan mengkomunikasikan informasi penting tentang setiap donasi yang terjadi. Dengan demikian, penggunaan alat ini secara merata di seluruh perjalanan bus mendukung efisiensi dan transparansi dalam proses pengumpulan darah, yang pada gilirannya dapat berkontribusi terhadap peningkatan jumlah donasi dan perbaikan sistem pendonoran.

Ujicoba alat pada bus PMI Labuhanbatu yang beroperasi sebagai unit donor di berbagai tempat destinasi akan membuktikan kemampuan dan adaptabilitas alat dalam kondisi praktis yang sebenarnya. Hasil dari uji coba ini akan memberikan wawasan berharga tentang efektivitas alat dalam mendukung pendonoran darah keliling, serta berpotensi menciptakan perubahan positif dalam cara pendonoran darah dilaksanakan.



Gambar 3. 8 Pemasangan sensor IR di pintu masuk



Gambar 3. 9 Pemasangan sensor di pintu keluar

Dalam Gambar 3.8, terlihat dengan jelas bahwa sensor inframerah (IR) telah diposisikan dengan cerdas pada bagian dalam pintu depan bus. Sensor ini memiliki peran vital dalam melakukan pemindaian terhadap individu yang memasuki bus, mengirimkan data yang diperlukan kepada modul Arduino untuk diproses lebih lanjut. Proses ini bertujuan untuk mengatur sistem dan menampilkan informasi terkait jumlah penumpang di dalam bus. Keterlibatan sensor IR dalam pengaturan ini membantu dalam mengoptimalkan operasi pengumpulan data secara real-time.

Pada Gambar 3.9, pendekatan serupa diterapkan dengan pemasangan sensor pada pintu keluar bus. Sensor ini juga memiliki peran penting dalam melakukan pemindaian terhadap individu yang meninggalkan bus, serta

mengidentifikasi jumlah orang yang meninggalkan bus atau mengalami kepergian. Data dari sensor ini kemudian diolah oleh modul Arduino untuk memperoleh informasi yang akurat.

Untuk menghindari tampilan yang berantakan dan memastikan kenyamanan di dalam bus, komponen Arduino dan breadboard disusun dengan rapi dan efisien di dalam bus. Ini mempertimbangkan ketersediaan kabel yang menghubungkan berbagai perangkat, yang pada kenyataannya dapat mencapai jumlah yang cukup signifikan. Oleh karena itu, pendekatan ini memungkinkan komponen yang lebih teknis untuk ditempatkan di bagian ujung samping mobil, sesuai yang diperlihatkan pada Gambar 3.10.

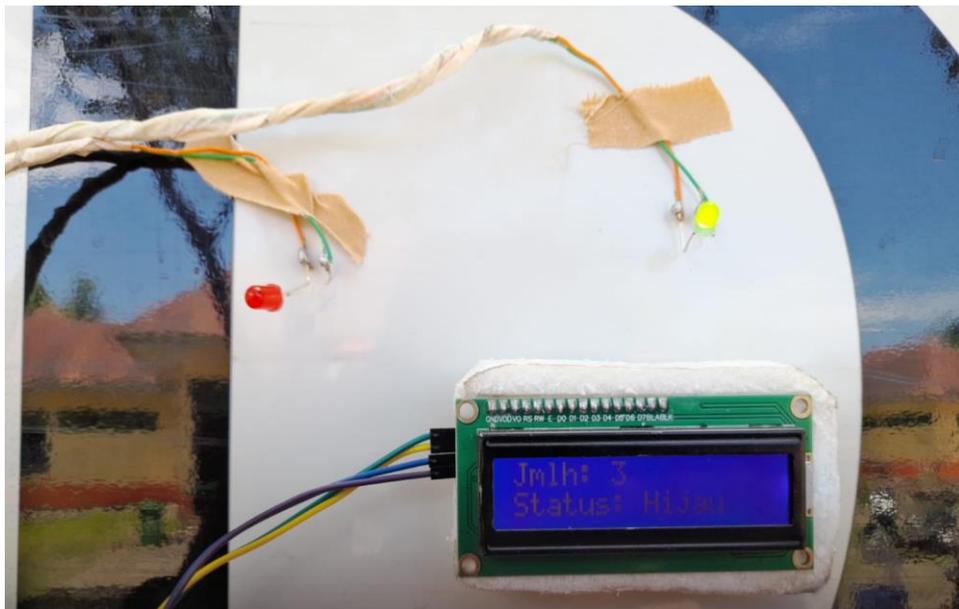


Gambar 3. 10 peletakan arduino dan breadboard

Sedangkan untuk LCD 16X2 dan Lampu LED nya di pasang di bagian samping pintu masuk. LCD 16X2 berfungsi untuk menampilkan jumlah pengunjung sudah berada di dalam bus. Sedangkan untuk lampu LED berfungsi

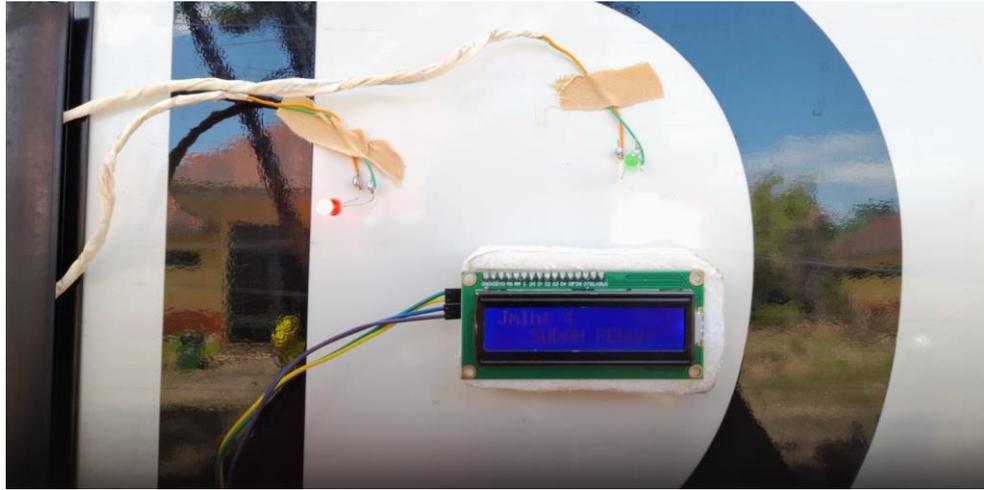
untuk memberikan sebuah peringatan bahwa bus telah penuh atau bus masih bisa di masuki oleh pasien.

Gambar 3.11 menunjukkan LCD menampilkan jumlah orang yang didalam bus, sedangkan LED yang menyala berwarna hijau menandakan bus masih bisa untuk di masukkan oleh pasien.



Gambar 3. 11 Lampu LED Hijau menyala

Sedangkan jika lampu LED berwarna hijaunya mati, di gantikan oleh lampu berwarna merah yang menyala dan tampilan LCD menampilkan “RUANGAN PENUH” berarti ruagan sudah tidak bisa di masuki lagi.



Gambar 3. 12 lampu LED Merah Menyala

4.3 Hasil Kesimpulan

Hasil Kesimpulan dari rangkaian alat di atas mendapatkan hasil yang sangat memuaskan dan sesuai yang diharapkan. Alat dapat menghitung pengunjung dengan akurasi yang tepat. Penampilan LCD dan peringatan dari Lampu LED berfungsi dengan normal. Bisa dilihat dari beberapa foto foto yang di atas. Foto foto diatas merupakan implementasi alat yang diterapkan langsung di bus PMI Labuhanbatu.