Tugas #8 Sistem Pemantau Toren Air

Tugas Kuliah EL4121 Perancangan Sistem Embedded

Nama : Mukhtar Amin

NIM : 13214123

Contents

[1 Spesifikasi 1](#_Toc494200881)

[2 Desain 1](#_Toc494200882)

[3 Implementasi Perangkat Keras 2](#_Toc494200883)

[4 Implementasi Perangkat Lunak 2](#_Toc494200884)

[5 Pengujian 2](#_Toc494200885)

[6 Analisa 2](#_Toc494200886)

[7 Kesimpulan 2](#_Toc494200887)

[8 Referensi / Pustaka 2](#_Toc494200888)

# Requirement

Requirement untuk project Sistem Pemantau Toren Air ini adalah sebagai berikut :

1. Sistem dapat mengukur tinggi air dalam toren atau tank dengan rate pengukuran setidaknya 1 pengukuran per detik
2. Sistem dapat menampilkan waktu dan tanggal saat ini
3. Nilai waktu dan tanggal dapat diadjust oleh pengguna
4. Sistem dapat mengirimkan nilai tinggi air yang terukur ke komputer dan ditampilkan oleh sofware data logger di komputer
5. Kecepatan pengiriman data nilai tinggi air yang terukur ke komputer dapat diubah oleh pengguna
6. Sistem menyimpan nilai tinggi toren dan tinggi maksimum air dan nilai-nilai dapat diubah oleh pengguna
7. Sistem memiliki relay yang terhubung dengan pompa air sehingga berfungsi sebagai switch yang menyalakan atau mematikan pompa pemasok air ke toren atau tanks
8. Sistem memiliki mode automatic dan manual yang mana pada mode automatic sistem akan memutus relay ketika tinggi air mencapai atau lebih dari nilai tinggi air maksimum sedangkan pada mode manual pengguna dapat memutus atau menghubungkan relay secara manual
9. Terdapat pilihan kalibrasi ulang ketinggian air untuk mereset nilai ketinggian air

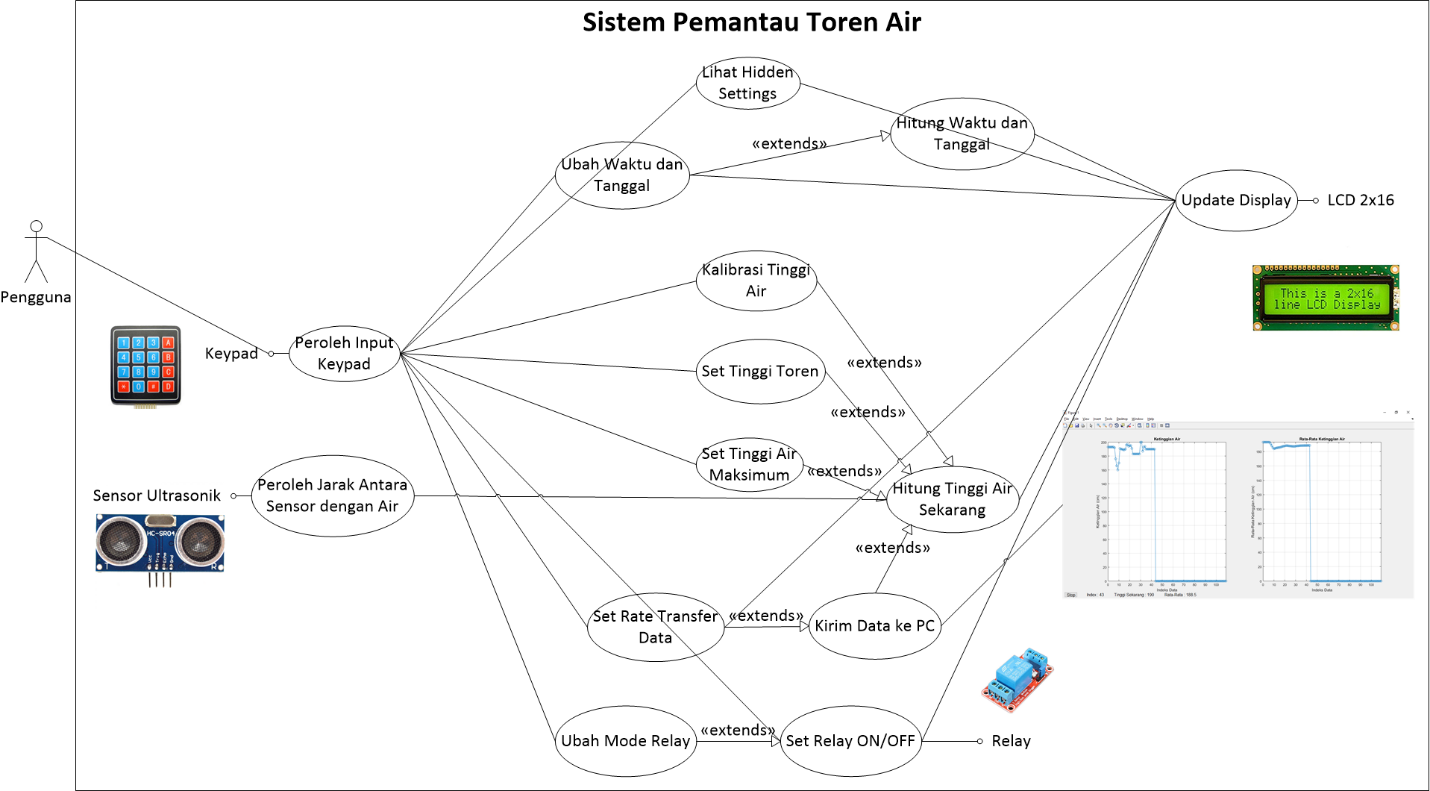
# Spesifikasi

* Mikrokontroller dan Lingkungan Pemrograman
  + MCU menggunakan Arduino Uno dan lingkungan pemrograman adalah AVR Dude dan gcc disertai MSYS untuk memprogram chip ATmega328P secara langsung
* Pengukur Ketinggian Air
  + Pengukuran ketinggian air dilakukan oleh sensor ultrasonik atau sonar modul HC-SR04
  + Sensor Ultrasonik mengirimkan nilai jarak sensor dengan air ke mikrokontroller minimal setiap perioda 1 detik
  + Saat inisiasi awal sistem menyimpan nilai tinggi tank default sebesar 200 cm dan nilai tinggi air maksimum sebesar 0 cm, kedua nilai dapat diset maksimum sebesar 9999 cm
  + Saat dilakukan kalibrasi tinggi air, sistem mereset nilai tinggi air menjadi nol lalu sistem menghitung tinggi air. Hal ini diperlukan bila tinggi air stuck di nilai yang besar karena sistem hanya memperbolehkan perubahan penurunan ketinggian yang dibatasi oleh nilai SONAR\_DELTA\_LIMIT dalam kode yang defaultnya adalah 50 cm, sedangkan sistem tidak membatasi perubahan kenaikan ketinggian
* Waktu dan Tanggal
  + Waktu yang ditampilkan adalah jam, hari, dan detik dalam format 24 jam (hh:mm:ss)
  + Tanggal ditampilkan dalam seperti atau mirip dengan format YYYY:MM:DD
* Transmisi Data dan Data Logging
  + Transmisi data dilakukan oleh MCU ke komputer lewat jalur komunikasi serial (UART)
  + Kecepatan transmisi data ketinggian air dari MCU ke komputer default saat inisiasi awal adalah satu data per 1 detik. Nilai dapat diset maksimum hingga 1 data per 9999 detik
  + Data logging pada komputer dilakukan oleh MATLAB yang menampilkan grafik data ketinggian air dan ketinggian air rata-rata dengan jumlah sampel yang diambil 110 data
* Display
  + Display menggunakan LCD 2x16
  + Pada saat idle atau default, display LCD menampilkan tanggal, waktu, kecepatan data logging, ketinggian air, dan status relay
  + Terdapat setting tersembunyi yang menampilkan tahun, tinggi maksimum air, dan tinggi toren
* Input dari Pengguna
  + Keypad 4x4 dapat digunakan untuk
    - Menampilkan setting tersembunyi
    - Mengatur tanggal dan waktu
    - Mengatur ketinggian air maksimum
    - Mengatur kecepatan pengiriman data tinggi air
    - Melakukan kalibrasi ulang tinggi air
    - Mengubah nilai tinggi tank atau toren
    - Mengatur mode relay
    - Pada mode manual untuk mengatur on/off relay secara manual
* Relay
  + Pada mode manual perubahan mode langsung ke mode automatic
  + Pada mode automatic perubahan mode ke mode manual dilanjutkan pemilihan state relay terhubung (ON) atau terputus (OFF)
  + Mode relay diindikasikan pada LCD dengan karakter ‘A’ untuk mode automatic dan karakter ‘M’ untuk mode manual

# Desain Sistem

## Use Case Diagram

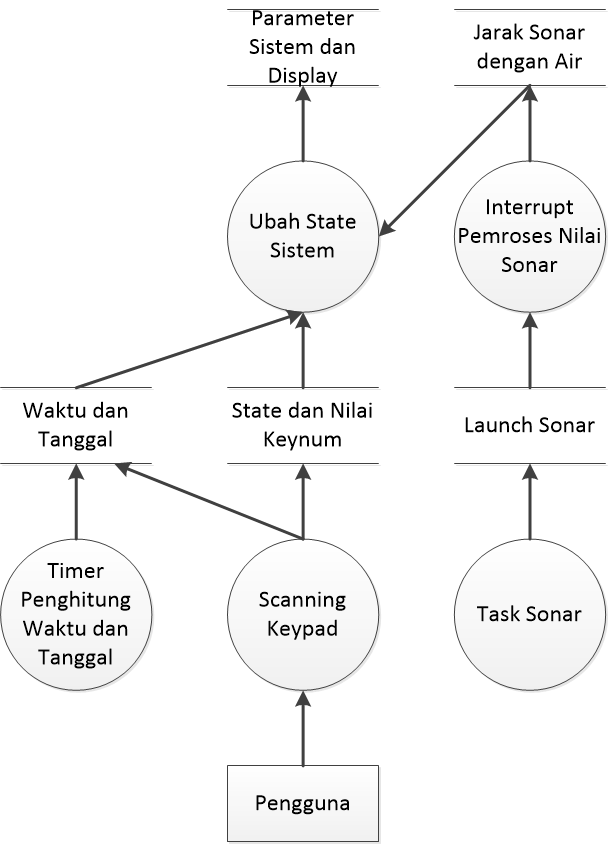
Use case diagram dapat digunakan untuk menggambarkan skenario dan menganalisis sistem dari segi fungsi-fungsi dan interaksi yang dapat dilakukan dengannya. Gambar 3.1.1 di bawah memperlihatkan use case diagram dari sistem.

  
Gambar 3.1.1 Use Case Diagram Sistem Pemantau Toren Air

Pengguna berinteraksi dengan sistem melalui keypad sedangkan sistem menerima input dari keypad dan sensor ultrasonik. Sistem lalu menjalankan proses misalnya memperlihatkan setting hidden layer atau mengubah waktu dan tanggal dan menampilkan responnya pada display LCD 2x16, mempengaruhi tampilan data yang diterima di PC, atau mempengaruhi status relay (ON atau OFF). Seluruh proses yang mungkin dilakukan oleh sistem dituliskan sebagai use case yang berbentuk oval di atas.

## Data Flow Diagram (DFD)

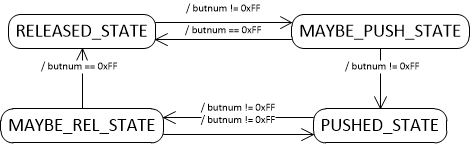
Gambar 3.2.1 di bawah memperlihatkan data flow diagram atau DFD dari sistem.

  
Gambar 3.2.1 Data Flow Diagram Sistem

Terdapat setidaknya tiga proses yang berjalan secara concurrent yaitu proses timer yang menghitung waktu dan tanggal, proses scanning keypad dan proses yang menghitung nilai sonar. Pengguna hanya berinteraksi dengan sistem melalui keypad yang nilainya diperoleh sistem dengan proses scanning keypad. Masing-masing proses menyimpan nilainya pada datanya masing-masing dan pada akhirnya digunakan oleh proses pengubah state sistem untuk mengubah parameter sistem seperti ketinggian air, ketinggian toren, rate transfer data, tanggal, waktu, status relay dan lain-lain, juga untuk mengubah tampilan display.

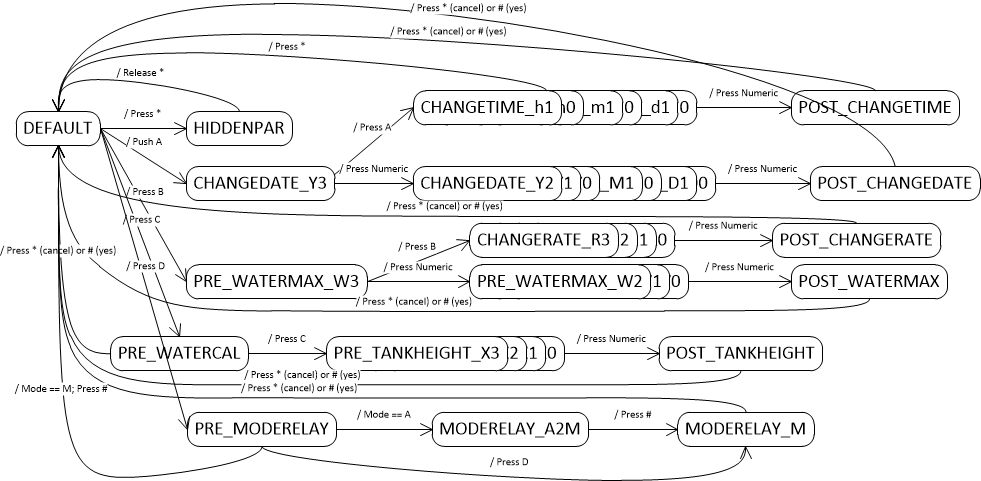
## Perancangan State Chart

Dalam sistem ini terdapat dua state chart yang diimplementasikan yaitu state chart untuk sistem utama yang digunakan oleh proses pengubah state sistem yang mana berisi ke mana state sistem setelah pengguna menekan tombol-tombol pada keypad, dan state chart untuk debouncing keypad itu sendiri sehingga sistem dapat membedakan saat pengguna baru saja menekan keypad, saat keypad ditekan terus menerus atau saat keypad dilepas. Pertama dijelaskan state chart debouncing keypad lalu dijelaskan state chart sistem utama.

  
Gambar 3.3.1 State Chart Debouncing Keypad

State awal saat keypad dibiarkan atau keypad tidak ditekan adalah RELEASED\_STATE dengan nilai butnum = 0xFF. Sistem secara terus menerus mengupdate nilai butnum dengan fungsi read\_keypad() yang mengembalikan nilai keypad saat ini dengan keypad scanning. Ketika pengguna memencet salah satu tombol keypad, nilai butnum akan terdeteksi bukan 0xFF sehingga sistem dapat pindah state ke MAYBE\_PUSH\_STATE. Bila pada scanning selanjutnya butnum masih bernilai 0xFF, maka state berpindah ke PUSHED\_STATE. Kemudian state sistem bertransisi dulu ke MAYBE\_REL\_STATE dan hanya ketika berada di state dan terdeteksi butnum bernilai 0xFF maka state kembali ke RELEASED\_STATE. State machine untuk debouncing keypad ini berguna supaya sistem dapat membedakan pengguna baru saja menekan keypad atau keypad sedang ditekan terus menerus. Untuk itu dalam setiap state diset parameter button\_state dan button\_flag yang sesuai.

Selanjutnya dijelaskan state chart sistem utama yang diperlihatkan oleh Gambar 3.3.2 di bawah.

  
Gambar 3.3.2 State Chart Sistem Utama

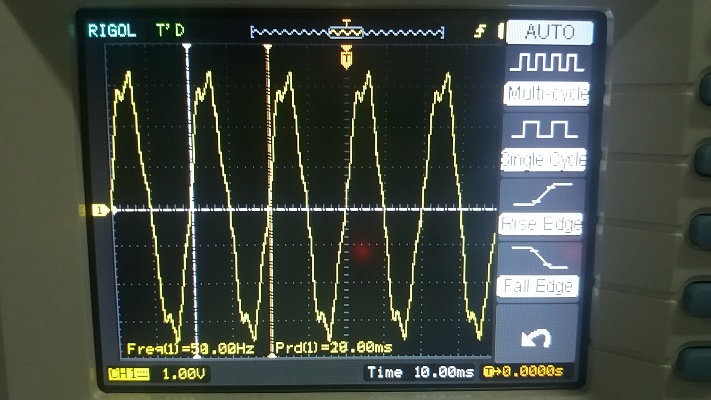
Terdapat total 36 state untuk sistem utama. Halaman home adalah state DEFAULT yang menampilkan bulan, hari, jam, rate pengiriman data, tinggi air, status relay dan mode relay. Memencet tombol \* di state DEFAULT memindahkan state ke HIDDENPAR yang menampilkan parameter tahun, tinggi toren dan tinggi air maksimum. Memencet tombol A, B, C atau D memindahakan state sistem ke state-state yang berfungsi mengubah parameter-parameter sistem seperti yang dapat dipelajari pada gambar 3.3.2 State Chart di atas.

## Task dan Frekuensi

Implementasi sistem pemantau air ini juga menggunakan FreeRTOS supaya sistem dapat beroperasi secara real time. Karena MCU yang digunakan adalah Arduino UNO dengan chip ATmega328P make task yang dapat dibuat maksimum adalah tiga dan diputuskan untuk dibuat dua buah task yaitu TaskMain() dan TaskSonar().

TaskMain() menjalankan proses debouncing keypad, mengubah state sistem, mengupdate display, mengirimkan data ke PC via UART, dan mengubah state relay (ON atau OFF) sedangkan TaskSonar() menjalankan proses triggering sensor ultrasonik yang kemudian mengirimkan jarak antara sensor dengan air untuk diproses menjadi tinggi air.

Setelah dilakukan pengukuran execution time diperoleh execution time untuk masing-masing task adalah tepat 20ms. Berikut gambar osiloskop hasil pengukuran execution time.

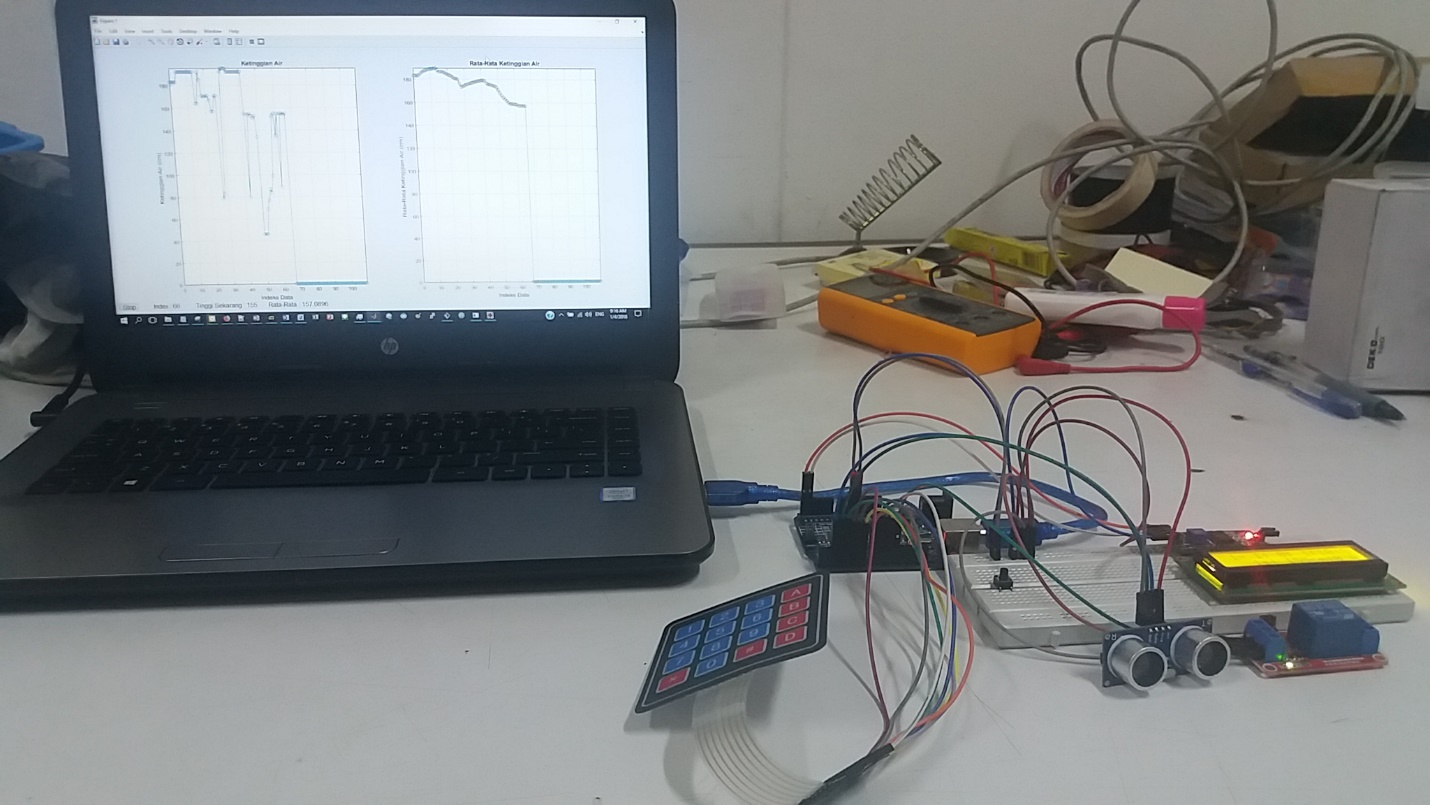
Gambar 3.4.1 Gambar di kiri adalah hasil pengukuran execution time TaskMain() sedangkan Gambar di kanan adalah hasil pengukuran execution time TaskSonar()

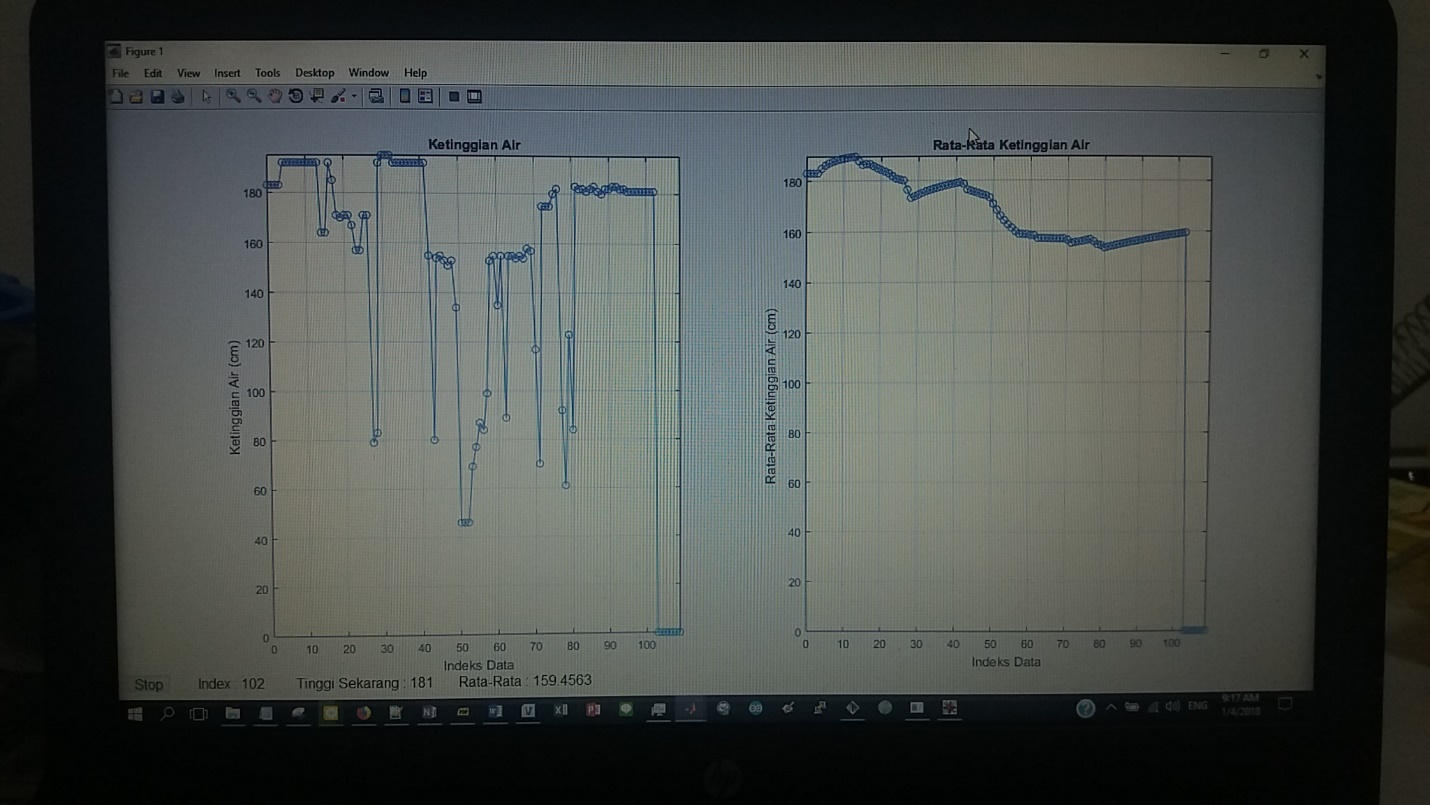
Dengan execution time masing-masing task 20ms, maka kedua task dapat dijalankan secara periodik dalam 40ms. Artinya proses dapat berjalan dalam frekuensi 25 Hz. Frekuensi ini cukup tinggi mengingat dalam spesifikasi diinginkan pengambilan data ketinggian air minimal satu pengukuran setiap 100 ms. Dengan frekuensi 25 Hz, rate pengukuran dan proses pengiriman data dapat berlangsung hingga 25 pengukuran tiap detik atau satu pengukuran dalam setiap 40 ms.

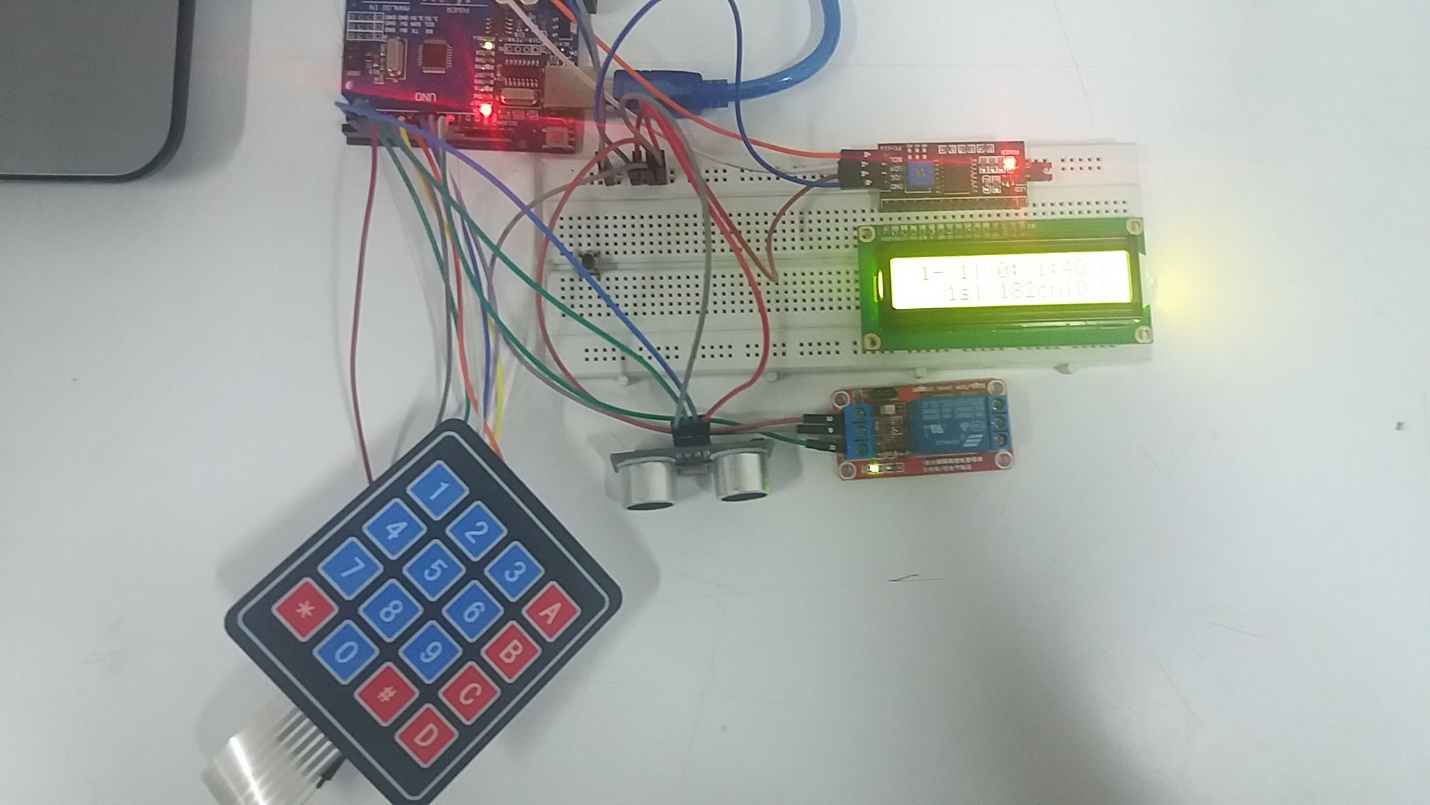
Untuk memperjelas sikap real time sistem, maka utilisasi dari sistem dapat dihitung sebagai berikut :

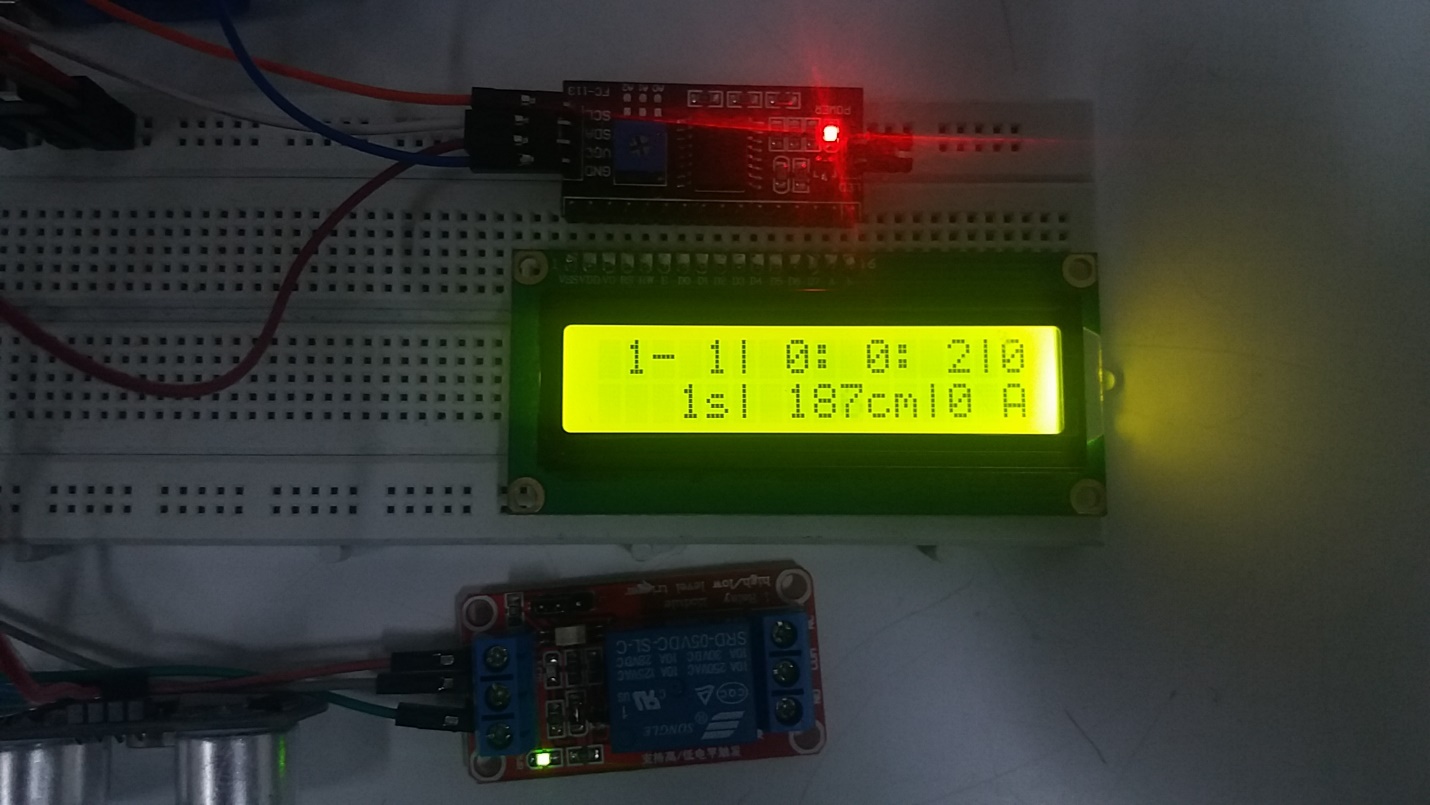
Dengan Ci merupakan waktu eksekusi task ke-i dan Pi adalah periode. Utilisasi sistem memberikan nilai 0.4 yang mana lebih kecil dari 1 (untuk satu prosesor) sehingga mengisyaratkan bahwa sistem dapat berjalan secara real time.

## Hasil Implementasi Sistem

  
Gambar 3.5.1 Hasil Implementasi Data Logger (laptop di kiri) dan Sistem Pemantau Toren Air (rangkaian di kanan)

  
Gambar 3.5.2 Hasil Implementasi Data Logger, menampilkan Ketinggian Air (kiri) dan Ketinggian Air Rata-rata, data yang diterima sekarang ditamplikan oleh nilai-nilai di pojok kanan bawah,  
contoh mengambil 110 sampel data dan mereset grafik setelah data ke 110

  
Gambar 3.5.3 Hasil Implementasi Rangkaian Sistem Pemantau Toren Air

  
Gambar 3.5.4 Menu Home setelah reset (menu default) sistem

  
Gambar 3.5.5 Menu Hidden Parameter (default, setelah reset)

  
Gambar 3.5.6 Menu Pengubah Tanggal, contoh mengubah tanggal menjadi 04 Januari 2018

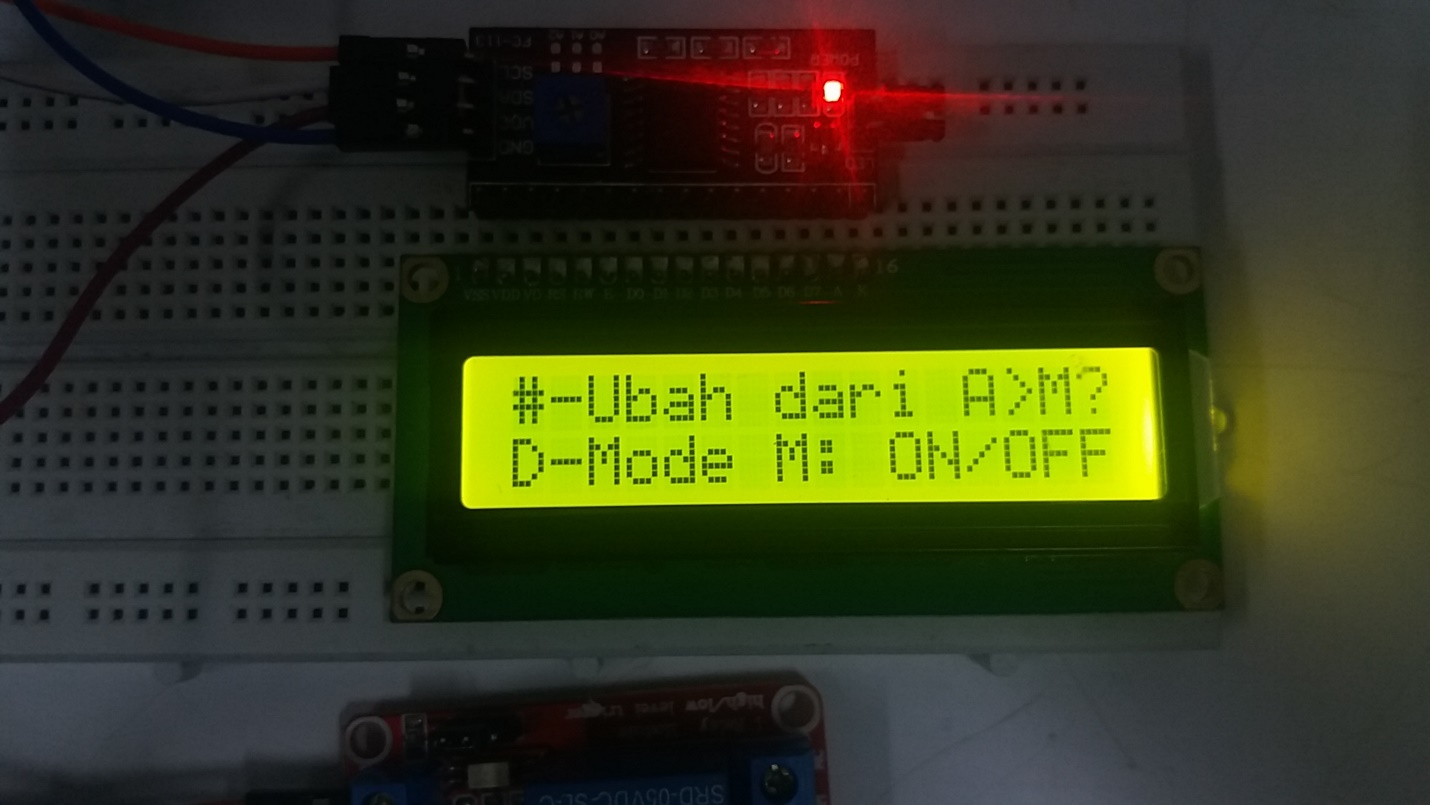
  
Gambar 3.5.7 Menu Pengubah Waktu, contoh mengubah waktu menjadi  
Jam 09 pagi, menit ke 18, detik ke 55

  
Gambar 3.5.8 Menu Pengeset Tinggi Maksimum Air,  
contoh mengeset tinggi maksimum air menjadi 170 cm

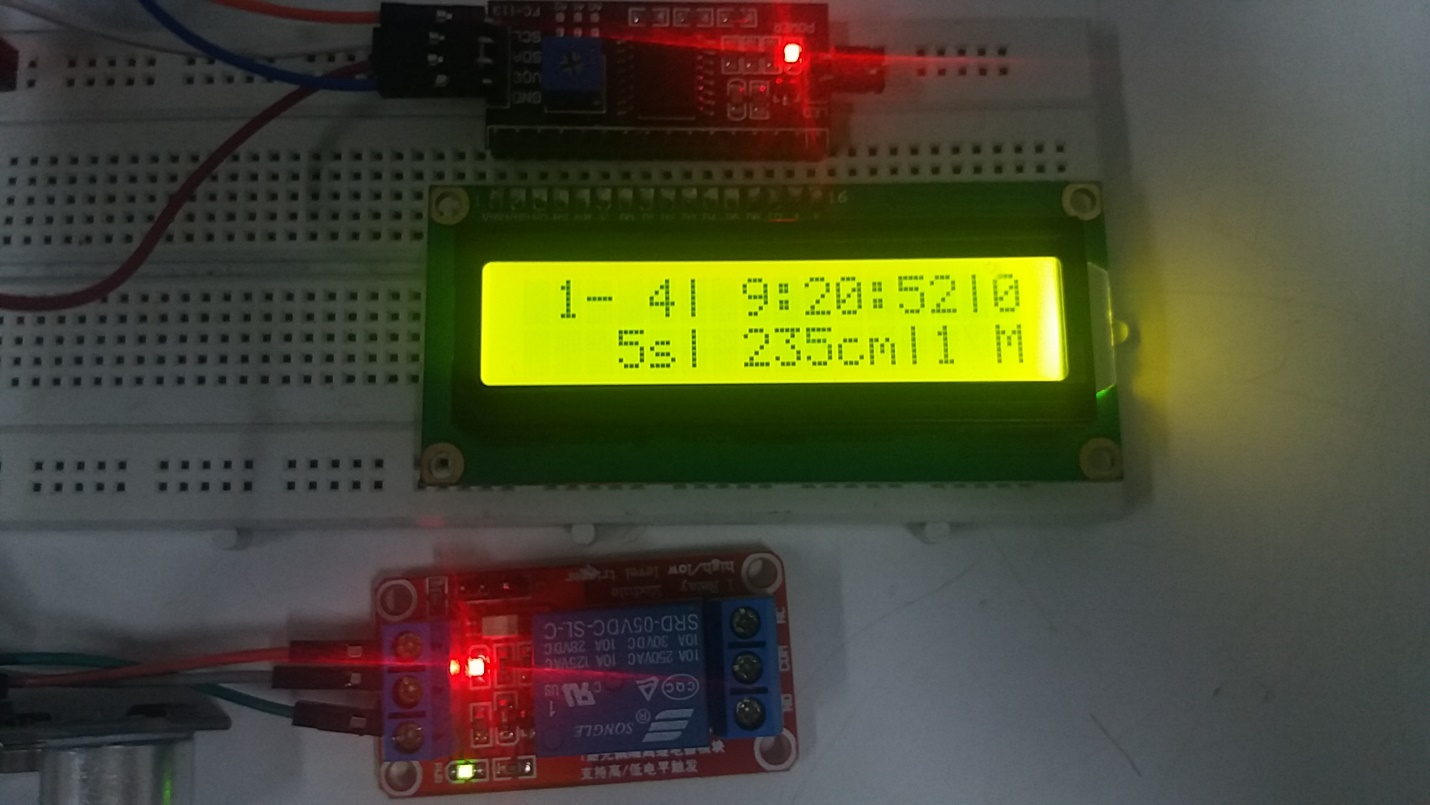
  
Gambar 3.5.9 Menu pengubah kecepatan transfer data ke PC,  
contoh mengubah kecepatan transfer data menjadi satu data per 5 detik

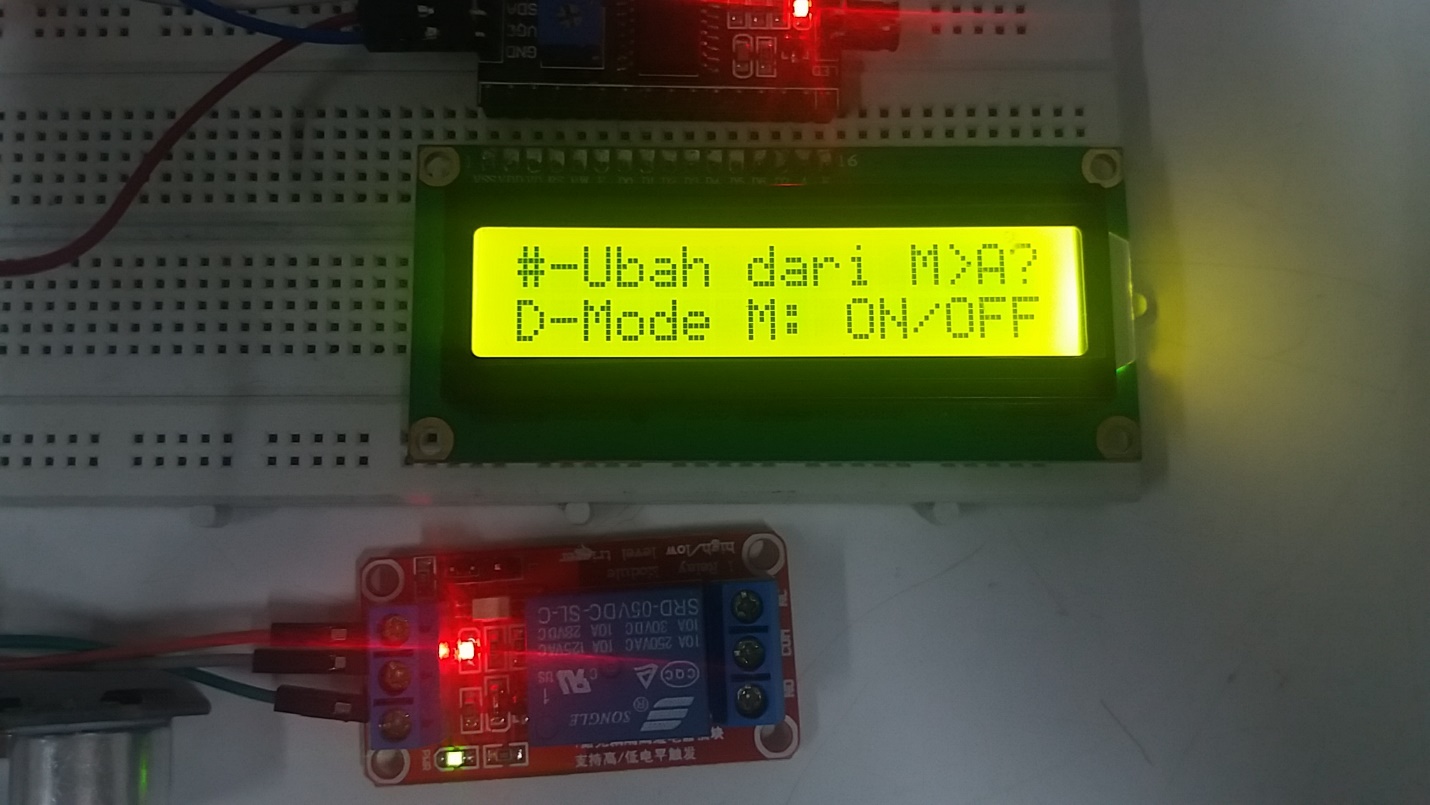
  
Gambar 3.5.10 Menu prompt kalibrasi ulang tinggi air

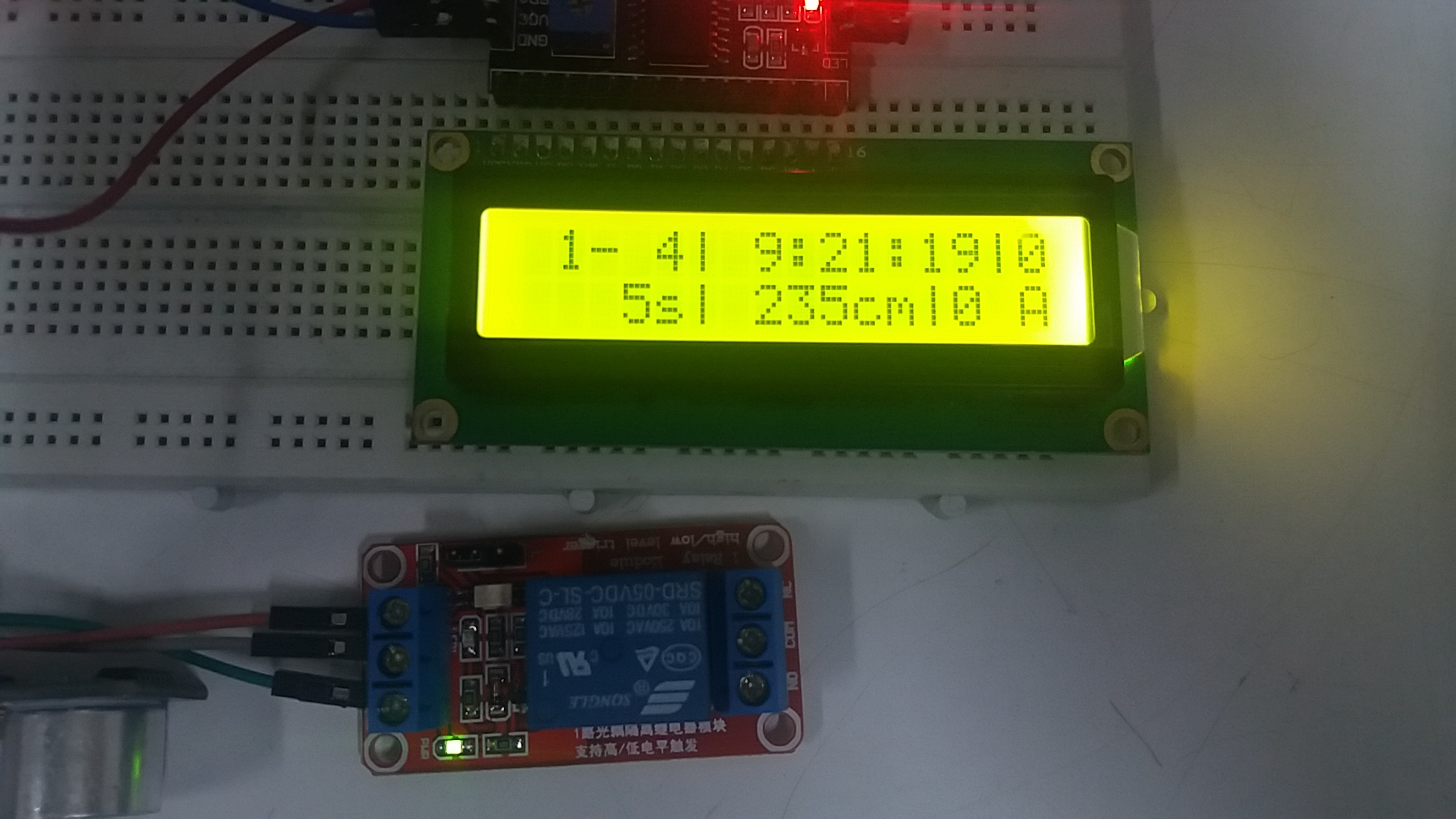
  
Gambar 3.5.11 Menu pengubah tinggi tank atau toren,  
contoh mengubah tinggi tank menjadi 250 cm

  
Gambar 3.5.12 Menu pengubah mode relay,  
contoh memperlihatkan perubahan dari Automatic ke Manual

  
Gambar 3.5.13 Menu setting relay manual, selanjutnya pencet tombol (1) untuk menghubungkan relay,  
atau pencet tombol (0) untuk memutus relay,  
atau pencet (\*) untuk membatalkan

  
Gambar 3.5.14 Hasil Setting contoh-contoh sebelumnya, Tanggal 04 Januari 2018, Waktu 9:20:52,  
rate transfer data per 5 detik, terukur tinggi air 235 cm, relay menyala dengan mode manual

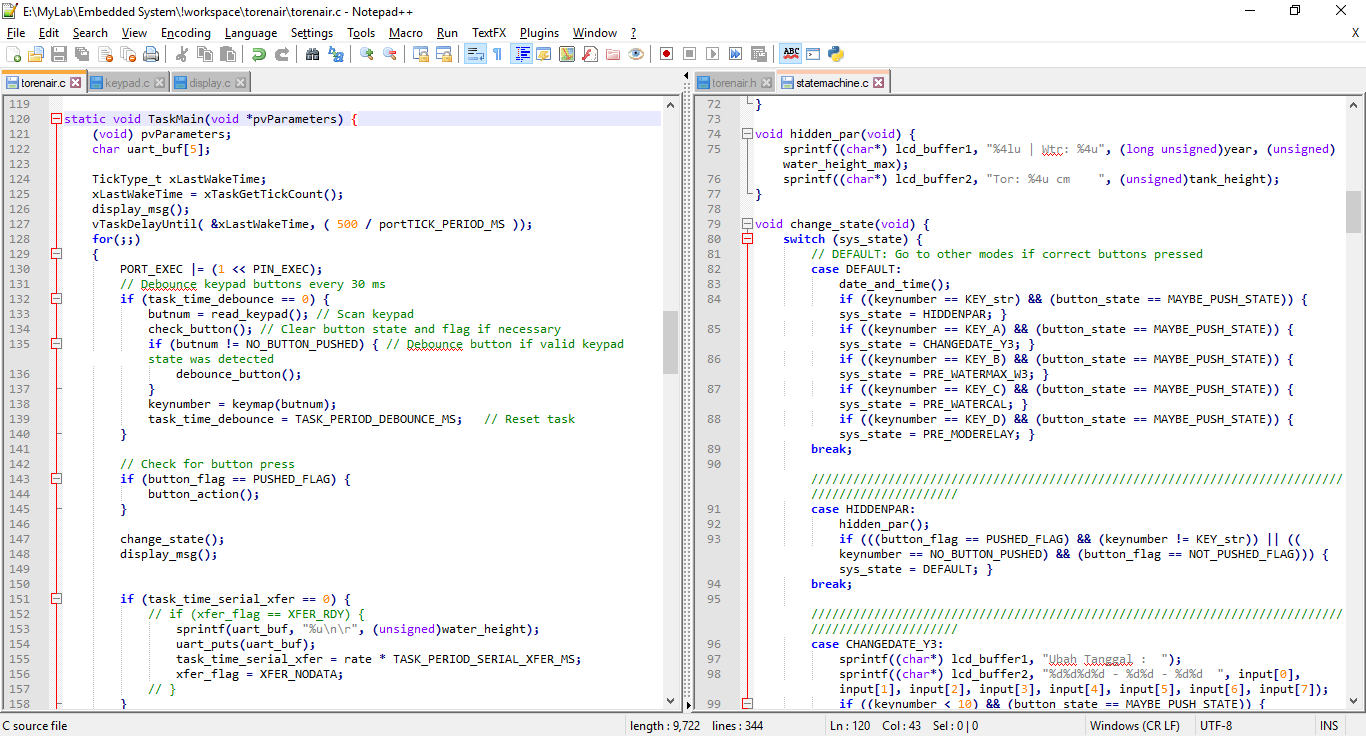
  
Gambar 3.5.15 Prompt Mengubah Mode relay dari Manual ke Automatic (selanjutnya menekan #), atau dapat memilih mengubah kembali state relay dengan menekan (D)

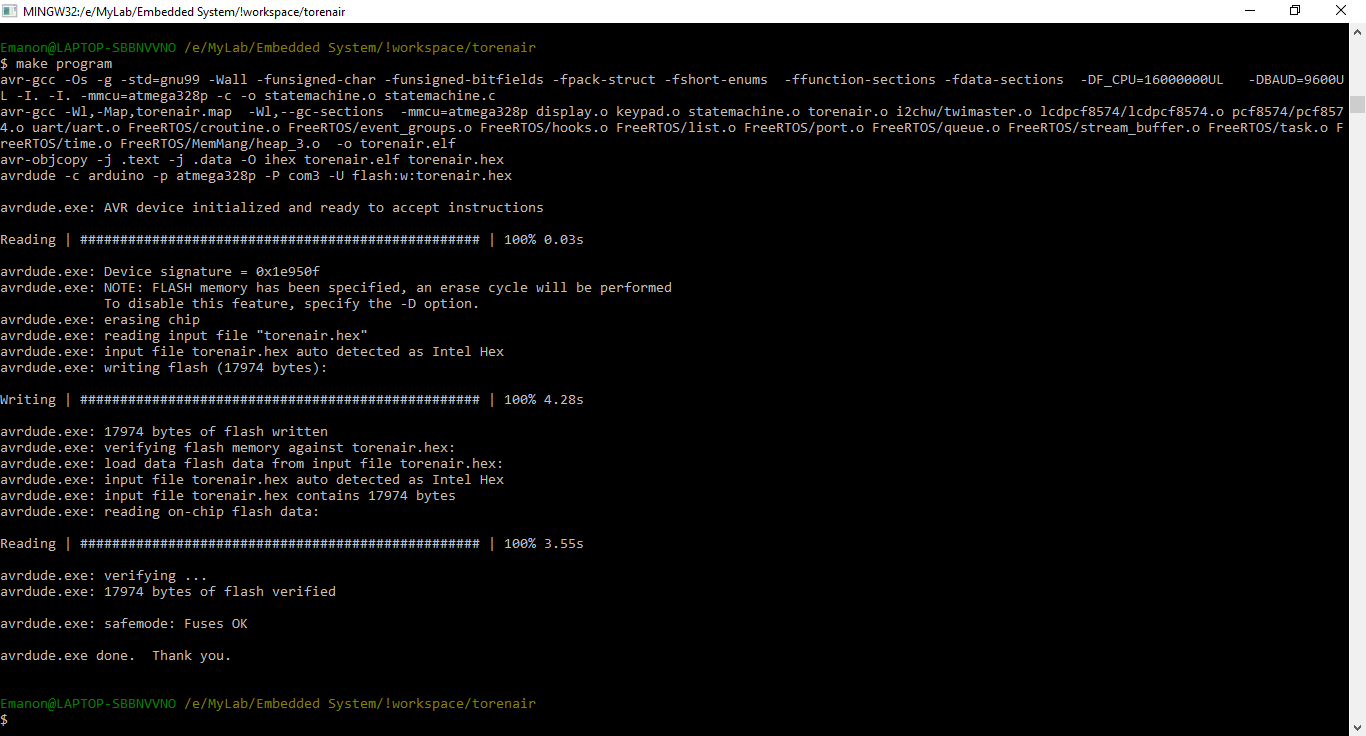
  
Gambar 3.5.15 Hasil Mengubah State Relay, mode menjadi Automatic, relay putus

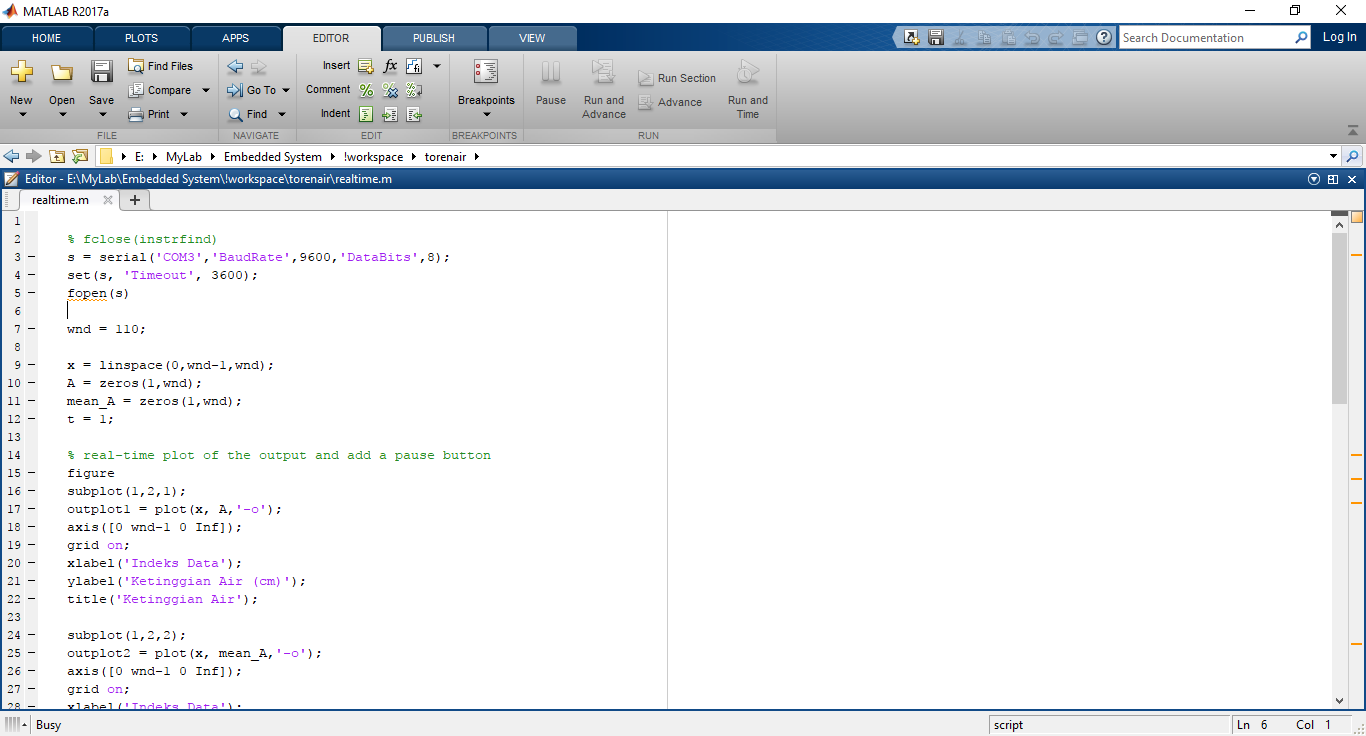
  
Gambar 3.5.16 Hidden Parameter menampilkan Tahun 2018, Tinggi Maksimum Air 170 cm dan tinggi toren atau tank 250 cm

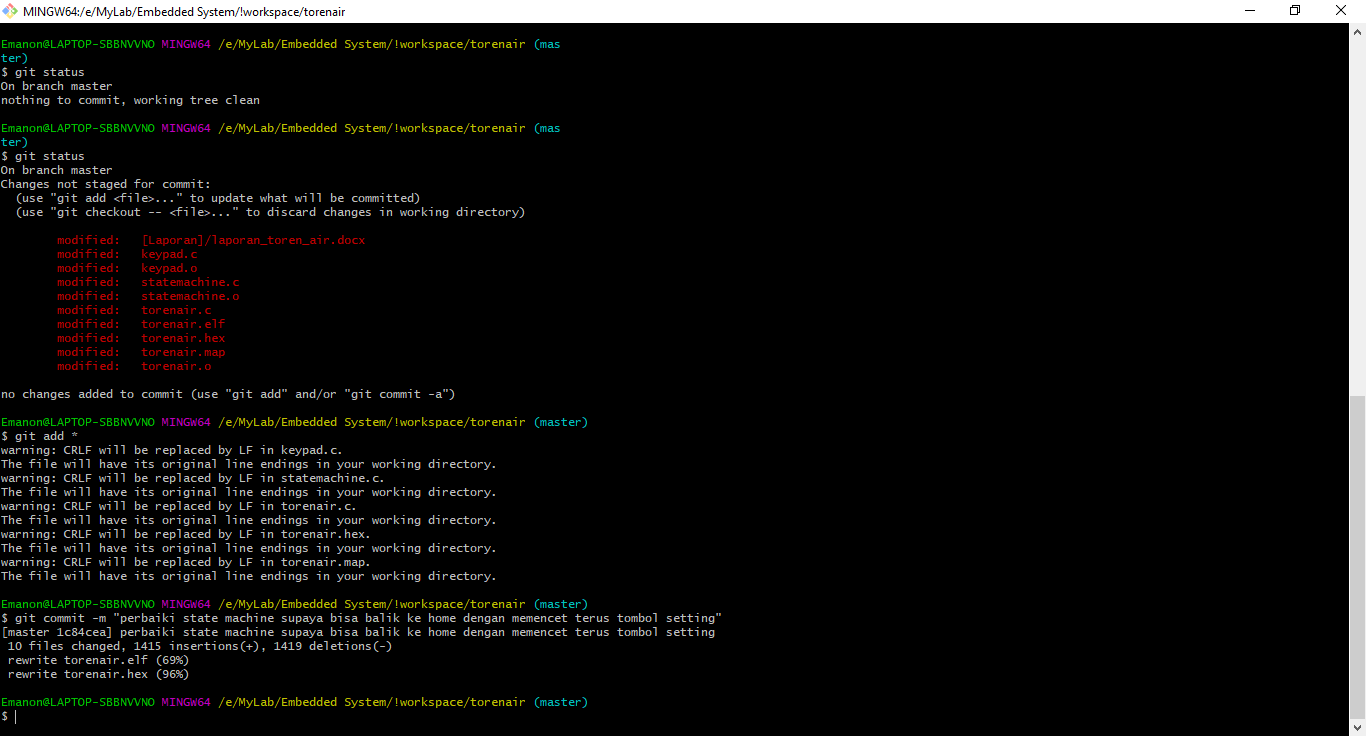
## Hasil Implementasi Software

Implementasi software atau pemrograman MCU dilakukan menggunakan AVR Dude dan gcc dengan dibantu MSYS, Notepad++, dan Git untuk version control. Berikut screenshot software-software yang digunakan.

  
Gambar 3.6.1 Editing dengan Notepad++

  
Gambar 3.6.1 Compiling dengan AVR Dude, gcc, dan MSYS

  
Gambar 3.6.2 Development Data Logger dengan MATLAB

  
Gambar 3.6.3 Version Control dengan Git

Development program dilakukan secara lokal sehingga tidak dilakukan push ke git repository hingga project selesai dilakukan.

# Kesimpulan

Asdf

# Dokumentasi

Asdf

# Implementasi State Chart

## Skema Rangkaian

(bikin di fritzing)

## Pembagian File

Untuk sementara pakai arduino IDE?

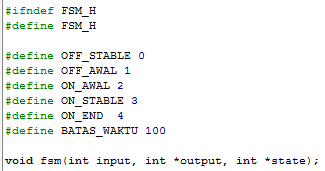
Daftar file

/atmega-100Hz/hasil.c : main(), ISR (platform dependent)

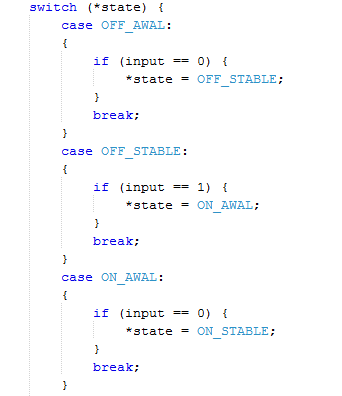
/fsm/fsm.c : implementasi FSM (platform independent)

/fsm/fsm.h: definisi-definisi FSM (platform independent)

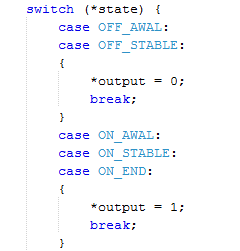
## Definisi Konstanta dan Variabel dan Fungsi



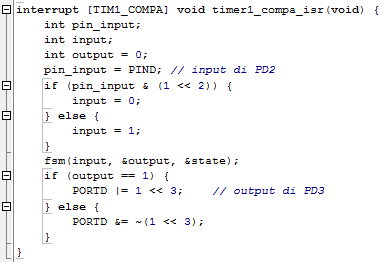
## Implementasi FSM



## Perhitungan Output



## Implementasi ISR



# Pengujian Sistem

Termasuk foto-foto pengujian

# Analisa

Perbandingan antara spesifikasi yang diinginkan dan hasil pengujian

# Kesimpulan

# Referensi / Pustaka

# Skenario

## Narasi Skenario

1. Power bank mati : ditekan 2x pendek (kurang dari 1 detik) maka power bank menyala
2. Power bank menyala : ditekan 2x pendek (kurang dari 1 detik) maka power bank mati
3. Senter mati + Baterai masih : ditekan 1x panjang (lebih dari 1 detik) maka senter menyala
4. Senter menyala + Baterai masih : ditekan 1x panjang (lebih dari 1 detik) maka senter mati
5. Baterai habis : senter mati, powerbank mati
6. Arus ada : power bank menyala

## Timing Diagram

### Diagram Skenario 1

Power bank mati: ditekan 2x pendek (kurang dari 1 detik) maka power bank menyala.

### Diagram Skenario 2

Power bank menyala : ditekan 2x pendek (kurang dari 1 detik) maka power bank mati.

### Diagram Skenario 3

Senter mati + Baterai masih : ditekan 1x panjang (lebih dari 1 detik) maka senter menyala.

### Diagram Skenario 4

Senter menyala + Baterai masih : ditekan 1x panjang (lebih dari 1 detik) maka senter mati.

### Diagram Skenario 5

Baterai habis : senter mati, power bank mati.

### Diagram Skenario 6

Arus ada : power bank menyala.