BUKU PANDUAN ALAT SIMULASI TAMAN PINTAR (ASITAP) DENGAN PENGONTROL BLUETOOTH HC-05 BERBASIS MIKROKONTROLER

Tim Pembuat : AGUS SUNANDAR ILHAM WAHYUDI MAISYAROH MUHAMMAD FAHMI BUDI SANTOSO HENNY DESTIANA YANA IQBAL MAULANA AGUS SALIM RACHMAN KOMARUDIN NURUL KHASANAH

JAKARTA

2021

KATA PENGANTAR

Alhamdulillah, dengan mengucapkan puji syukur kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya kepada kami semua, sehingga kami dapat menyelesaikan pembuatan "ALAT SIMULASI TAMAN PINTAR DENGAN PENGONTROL *BLUETOOTH HC-05* BERBASIS *MIKROKONTROLER*. Alat pengontrol ini dibuat untuk memudahkan pekerjaan pengurus taman dalam mengontrol dan merawat taman. Sehingga petugas taman dapat dengan mudah dan cepat mengontrol taman hanya dengan sebuah pengontrol *Bluetooth HC-05* berbasis *mikrokontroler* dan dengan sebuah perangkat *handphone* yang sudah terinstal aplikasi *Bluetooth terminal HC-05*.

Jakarta, 20 Januari 2021

Tim Pembuat

DAFTAR ISI

KATA PENGANTAR	2
DAFTAR ISI	3
A. PENDAHULUAN	4
B. SUMBER DAYA YANG DIBUTUHKAN	4
C. MENU DAN CARA PENGGUNAAN	5
D. SOURCE CODE	13

A. PENDAHULUAN

1. TUJUAN PEMBUATAN BUKU PANDUAN

Dokumen user manual alat simulasi taman pintar dengan pengontrol *bluetooth hc-*05 berbasis *mikrokontroler* bertujuan untuk menggambarkan dan menjelaskan penggunaan alat simulasi taman pintar dengan pengontrol Bluetooth hc-05 berbasis mikrokontroler untuk pengguna (user) dan administrator.

1.1. DESKRIPSI UMUM SISTEM

1. Deskripsi Umum Aplikasi

Deskripsi umum aplikasi meliputi deskripsi umum alat simulasi taman pintar dengan pengontrol *bluetooth hc-05* berbasis *mikrokontroler*, fungsi utama yang akan diberikan kepada pengguna yang meliputi penjaga taman.

2. Deskripsi Umum Kebutuhan Aplikasi Yang Akan Diimplementasikan

Deskripsi umum kebutuhan aplikasi yang akan diimplementasikan meliputi semua informasi yang bersifat teknis yang menjadi acuan dalam pengembangan aplikasi

1.2. DESKRIPSI DOKUMEN (IKHTISAR)

Dokumen ini dibuat untuk memberikan panduan penggunaan alat simulasi taman pintar dengan pengontrol *bluetooth hc-05* berbasis *mikrokontroler*. Dokumen ini berisi informasi sebagai berikut:

- 1. Berisi informasi umum yang merupakan bagian pendahuluan, yang meliputi tujuan pembuatan dokumen, deskripsi umum *system* serta deskripsi dokumen
- 2. Berisi spesifikasi perangkat yang dibutuhkan untuk menginstall aplikasi pengontrol alat.
- 3. Berisi user manual penggunaan aplikasi *Bluetooth terminal HC-05* untuk menjalankan *prototype* alat simulasi taman pintar

B. SUMBER DAYA YANG DIBUTUHKAN

1. PERANGKAT LUNAK

Perangkat lunak yang digunakan dalam pengujian aplikasi adalah:

- 1. Android OS 5 or high
- 2. Terinstal Aplikasi *Bluetooth terminal HC-05*

2. PERANGKAT KERAS

Perangkat keras yang dibutuhkan dalam pengujian aplikasi ini adalah:

- 1. Smartphone Android
- 2. Prototype Taman Pintar
- 3. Socket Listrik
- 4. Battery AA (2 Buah)

C. MENU DAN CARA PENGGUNAAN

1. STRUKTUR MENU DI APLIKASI BLUETOOTH TERMINAL HC-05

Adapun struktur menu pada aplikasi *Bluetooth Terminal HC-05* adalah sebagai berikut :

1. Menu SCAN

Adalah menu untuk men-scan *Bluetooth* yang tersedia di sekitar *handphone* kita.

2. Menu Titik Tiga

Adalah menu untuk melihat Setting aplikasi dan View More

3. Menu Paired Devices

Berisi informasi devices yang sudah terkoneksi dengan *Bluetooth handphone* kita

4. Menu Setting

Adalah menu untuk mengatur aplikasi Bluetooth Terminal HC-05

2. PENGGUNA (USER)

Pada bagian ini akan dijelaskan mengenai tata cara penggunaan aplikasi Bluetooth Terminal HC-05.

2.1. Cara Membuka Aplikasi

Untuk membuka aplikasi :

- 1. Install aplikasi *Bluetooth Terminal HC-05* di smartphone *android* melalui aplikasi *playstore.*
- 2. Lalu buka aplikasi yang telah terinstall
- 3. Akan muncul tampilan Home Page aplikasi

- 2.2. Tampilan Menu Aplikasi Bluetooth Terminal HC-05
- 1. Tampilan Menu Utama

Tampilan menu utama pada saat Aplikasi Bluetooth Terminal HC-05 dibuka.

15	5:10 🕥 🖵		8 al all 😤 🔲 84	ka -
Bl	uetooth Te	rminal HC-05	SCAN	
<u>ר</u> ך] Paired Devic	ces		
	Redmi 9 E0:1F:88:7D:0	C:FD	í	
	Bluetooth_Key 20:16:01:21:02	/board 2:3F	í	
	SonicGear BT2 B4:B2:08:8E:7	2100 C:E3	í	
	WS-858 41:42:26:8C:2	A:ED	í	
	TP-600BT 41:42:8C:00:C	F:E6	í	
	JESBOD-J10 61:00:00:00:12	2:19	í	
	S530 41:42:FE:DE:E	B:FF	í	
	tokopedia	BEBAS ONGKIR Sepuasnya!		
		۲	•	

Gambar 1. Tampilan Menu Utama

Dari Gambar 1, bisa dijelaskan bahwa menu utama dari Aplikasi *Bluetooth Terminal HC-05* terdiri dari menu *scan*, paired devices, dan tanda titik tiga,

^{2.} Tampilan Menu SCAN

Ketika kita memilih menu *scan* di menu utama maka kita akan di arahkan ke Bluetooth yang tersedia disekitar kita, bisa di contohkan seperti Gambar dibawah ini :



Gambar 2. Tampilan Menu SCAN

Pada menu ini kita bisa mencari Bluetooth yang tersedia disekitar kita. Pastikan Bluetooth yang terkoneksi adalah Bluetooth HC-05

3. Tampilan Paired Devices



Gambar 3. Tampilan Paired Devices

Menu ini tampil ketika aplikasi sudah terkoneksi dengan paired devices

(tersambung ke Bluetooth HC-05).

4. Tampilan Cara Menyalakan Lampu Taman

Ketika penjaga taman hendak menyalakan lampu taman, maka penjaga taman harus mengetikkan perintah angka yang sudah terinput sebelumnya di alat taman pintar tersebut lalu klik Send ASCII, berikut tampilannya :



Gambar 4. Tampilan Inputan Menyalakan Lampu

5. Tampilan Cara Mematikan Lampu Taman

Sama dengan cara menyalakan lampu, kita harus input angka yang sudah kita

masukkan di alat lalu kita klik Send ASCII. Berikut tampilannya :

Bluetooth Terminal HC-05 Connected to HC-05	ASCII	8
Auto Scroll		
	AS	CIII: 2
Enter ASCII Command	Send A	SCII

Gambar 5. Tampilan Cara Mematikan Lampu Taman

6. Tampilan Cara Membuka Pintu Taman

Cara membuka pintu taman yang menggunakan *servo* adalah menginput angka 7 lalu klik Send ASCII maka pintu akan terbuka.



7. Tampilan Cara Menutup Pintu Taman

Cara menutup pintu taman yang menggunakan *servo* adalah menginput angka 8 lalu klik Send ASCII maka pintu akan terbuka.



Gambar 7. Tampilan Cara Menutup Pintu Taman

8. Tampilan Menyalakan Pompa Air

Input angka 9 pada aplikasi *Bluetooth Terminal HC-05* lalu klik *Send ASCII* maka Pompa Air akan menyala dan menyalurkan air ke tanaman.



Gambar 8. Tampilan Menyalakan Pompa Air

9. Tampilan Mematikan Pompa Air



Gambar 9. Tampilan Mematikan Pompa Air

Input angka 0 pada aplikasi *Bluetooth Terminal HC-05* lalu klik *Send ASCII* maka Pompa Air akan mati secara otomatis.

3. PENJELASAN KOMPONEN

Pada rangkaian sistem minimum mikrokontroler Atmega 16 sebagai pusat pemproses data; *Bluetoot HC-05* sebagai perintah sensor, Dan komponen rangkaian elektronik lainnya sebagai pendukung agar sistem bekerja dan aktif untuk *output*nya *Relay,Motor servo* & *Led* Dan semua tegangan *input* nya di ambil dari sistem minimum. sistem minimun sendiri membutuhkan tegangan untuk tegangan nya di ambil dari supplay *adaptor*, Lalu untuk memerintahkan sistem aktif sistem pada aplikasi yang dihubungkan dengan bluetooth smartphone maka *Bluetooth HC-05* pada alat akan menerima perintah yang dijalankan di aplikasi *smartphone.*

4. CARA KERJA ALAT

Untuk menjalankan alat simulasi taman pintar, hal yang pertama kali dilakukan adalah memasang adaptor ke sistem minimum kemudian memasang *Bluetooth* pada ponsel *android* dan *Bluetooth hc-05*. Dengan menjalankan aplikasi "*Bluetooth terminal HC-05*" yang sudah terpasang pada ponsel *android*, setelah menjalankan aplikasi tersebut pilihlah nama *Bluetooth HC-05* di dalam daftar yang terdapat pada aplikasi tersebut.

Setelah terpasang atau terhubung antar perangkat Bluetooth maka masukan kode yang telah di tentukan untuk menyalakan/mematikan lampu LED, membuka/menutup pintu gerbang taman dan menyalakan/mematikan pompa penyiraman taman.

D. SOURCE CODE ALAT SIMULASI TAMAN PINTAR

```
#include <mega16.h>
// Alphanumeric LCD functions
#include <alcd.h>
// Declare your global variables here
// Standard Input/Output functions
#include <stdio.h>
#define RELAY PORTD.7
#define OFF 0
#define OFF 0
#define servo1 PORTB.7
#include <delay.h>
signed int i=0,data1=0;
// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
```

1. Input

```
if(val = 1')
PORTA.0=1;
}
else if (val=='2'){
PORTA.0=0;
}
else if (val = = '3'){
PORTA.1=1;
;
}
else if (val=='4'){
PORTA.1=0;
}
else if (val=='5'){
PORTA.2=1;
;
}
else if (val=='6'){
PORTA.2=0;
}
//PINTU
else if (val=='7'){
data1=25; //0 buka
}
else if (val=='8'){
```

```
data1=50; // tutup
;
}
else if (val=='9'){
PORTD.7=ON;
delay_ms(1200);
ł
else if (val=='0'){
PORTD.7=OFF;
delay_ms(1200);
;
}
2. Main program
void main(void)
{/
/ Declare your local variables here
// Input/Output Ports initialization
// Port A initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In
Bit0=In
PORTA=0X00;
DDRA=0XFF;
// Port B initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In
Bit0=In
DDRB=(0<<DDB7) | (0<<DDB6) | (0<<DDB5) | (0<<DDB4) | (0<<DDB3) |
(0<<DDB2) | (0<<DDB1) | (0<<DDB0);
// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) |
(0<<PORTB3) | (0<<PORTB2) | (0<<PORTB1) | (0<<PORTB0);
// Port C initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In
Bit0=In
PORTC=0x00;
DDRC=0xFF;
// Port D initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In
Bit0=In
PORTD=0xFF;
DDRD=0xFF;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=0x01;
```

TCNT0=0x00; OC

// Timer/Counter 1 initialization // Clock source: System Clock // Clock value: Timer1 Stopped // Mode: Normal top=0xFFFF // OC1A output: Disconnected // OC1B output: Disconnected // Noise Canceler: Off // Input Capture on Falling Edge // Timer1 Overflow Interrupt: Off // Input Capture Interrupt: Off // Compare A Match Interrupt: Off // Compare B Match Interrupt: Off TCCR1A=0x00; TCCR1B=0x00; TCNT1H=0x00; TCNT1L=0x00; ICR1H=0x00; ICR1L=0x00; OCR1AH=0x00; OCR1AL=0x00; OCR1BH=0x00; OCR1BL=0x00; // Timer/Counter 2 initialization // Clock source: System Clock // Clock value: Timer2 Stopped // Mode: Normal top=0xFF // OC2 output: Disconnected ASSR=0<<AS2: TCCR2=(0<<PWM2) | (0<<COM21) | (0<<COM20) | (0<<CTC2) | (0<<CS22) | (0<<CS21) | (0<<CS20); TCNT2=0x00; OCR2=0x00; // Timer(s)/Counter(s) Interrupt(s) initialization TIMSK=0x01; // External Interrupt(s) initialization // INT0: Off // INT1: Off // INT2: Off MCUCR=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00); MCUCSR=(0<<ISC2); // USART initialization // Communication Parameters: 8 Data, 1 Stop, No Parity // USART Receiver: On // USART Transmitter: On // USART Mode: Asynchronous

```
// USART Baud Rate: 9600
```

UCSRA=0x00; UCSRB=0x18; UCSRC=0x86: UBRRH=0x00; UBRRL=0x4D; // Analog Comparator initialization // Analog Comparator: Off // The Analog Comparator's positive input is // connected to the AIN0 pin // The Analog Comparator's negative input is // connected to the AIN1 pin ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) | (0<<ACIS0); SFIOR=(0<<ACME): // ADC initialization // ADC disabled ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) | (0<<ADPS0); // SPI initialization // SPI disabled SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) | (0<<SPR0); // TWI initialization // TWI disabled TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE); // Alphanumeric LCD initialization // Connections are specified in the // Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu: // RS - PORTB Bit 0 // RD - PORTB Bit 1 // EN - PORTB Bit 2 // D4 - PORTB Bit 4 // D5 - PORTB Bit 5 // D6 - PORTB Bit 6 // D7 - PORTB Bit 7 // Characters/line: 16 lcd init(16); // Global enable interrupts #asm("sei") PORTD.7=0;

3. Output

#include <mega16.h> // Declare your global variables here
#include <stdio.h> // Standard Input/Output functions
#define RELAY PORTD.7 // Relay ada di PORTD.7
#define OFF 0 // Angka 0 berarti mati/off

```
#define ON 1 // Angka 1 berarti hidup/on
#define servo1 PORTB.7 // Servo berada pada PORTB.7
#include <delay.h>
signed int i=0,data1=0;
// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
// Place your code here
if (i>864) //Algoritma standart untuk menentukan buka tutup pintu
{i=0;
}
else
{i++;}
if
(i<data1)
{servo1=1;}
else
{servo1=0;}
}
```