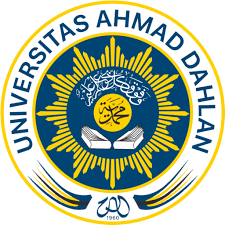
**Tugas Akhir**

**Program Aplikasi Stopwatch Digital**

Disusun untuk memenuhi tugas akhir Dasar Sistem Komputer

Dosen Pengampu : Ali Tarmuji, S.T., M.Cs.

****

**Disusun oleh:**

**NAMA : Anggun Septiani**

**NIM : 2400018147**

**KELAS : C**

**Link Github**

[**https://github.com/Anggunseptiani/Program---Aplikasi---Stopwatch-Digital**](https://github.com/Anggunseptiani/Program---Aplikasi---Stopwatch-Digital)

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS AHMAD DAHLAN**

**2024/2025**

**BAB I: PENDAHULUAN**

**1.1 Latar Belakang**

Stopwatch digital merupakan aplikasi sederhana yang digunakan untuk menghitung waktu yang berlalu. Aplikasi ini sering digunakan dalam kegiatan sehari-hari, seperti olahraga, eksperimen, atau kegiatan yang memerlukan pengukuran waktu. Dalam dunia pemrograman, pembuatan stopwatch digital menjadi salah satu contoh penerapan penghitungan waktu berbasis pemrograman assembly.

Penggunaan bahasa assembly dalam pembuatan stopwatch bertujuan untuk memahami cara kerja dasar sistem komputer, termasuk register, interrupt, dan timer. Selain itu, melalui proyek ini, mahasiswa dapat mempelajari konsep dasar pemrograman waktu nyata (real-time programming) di lingkungan prosesor Intel 8086**.**

**1.2 Tujuan**

1. Membuat aplikasi stopwatch digital menggunakan bahasa assembly.

2. Memahami mekanisme kerja interrupt dan penghitungan waktu pada prosesor 8086.

3. Mengimplementasikan aplikasi berbasis lingkungan EMU8086.

**BAB II: DASAR TEORI**

**2.1 Ruang Lingkup Aplikasi**

Aplikasi Stopwatch Digital adalah aplikasi berbasis teks yang dirancang menggunakan bahasa Assembly dan berjalan pada emulator EMU8086. Aplikasi ini berfungsi untuk:

Menghitung waktu dengan fitur **Start**.

Menghentikan waktu menggunakan fitur **Stop.**

Mengatur ulang waktu dengan fitur **Reset.**

Menghentikan program dengan fitur **Exit**.

Aplikasi ini bertujuan untuk memberikan pengalaman dasar dalam pembuatan aplikasi stopwatch sederhana dengan memanfaatkan fungsi pemrograman tingkat rendah.

**2.2 Desain / Rancangan Aplikasi**

**Sketsa Visual Aplikasi**

**--------------------------------**

**Stopwatch Digital**

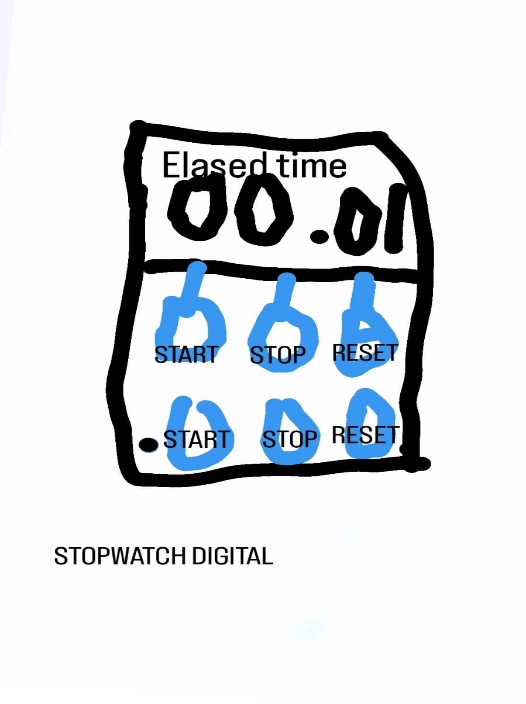
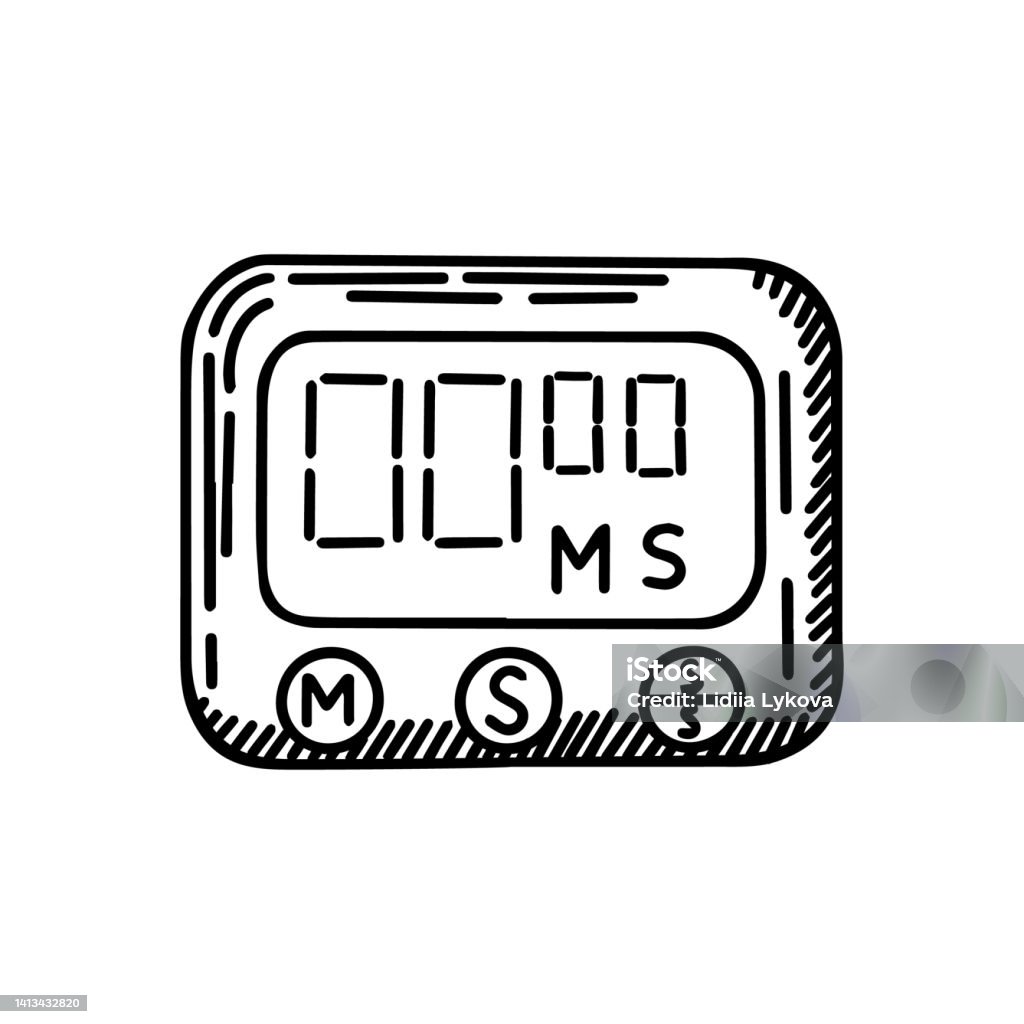
**1. Start**

**2. Stop**

**3. Reset**

**4. Exit**

**--------------------------------**

****

**Input pilihan:**

Selama proses berjalan, waktu akan ditampilkan dengan format berikut:

**Waktu saat ini: 05 detik**

Stopwatch digital adalah alat pengukur waktu yang memiliki fungsi dasar:

1. Start: Memulai perhitungan waktu.

2. Stop: Menghentikan perhitungan waktu.

3. Reset: Mengatur ulang waktu ke nol.

4. Display: Menampilkan waktu yang telah berlalu dalam format detik, menit, atau jam.

**2.3 EMU8086**

**EMU8086** adalah emulator yang dirancang untuk mempelajari dan menjalankan program assembly berbasis prosesor Intel 8086. Fitur utama EMU8086 meliputi:

1. Debugger untuk melacak eksekusi instruksi.
2. Lingkungan simulasi DOS.
3. Dukungan interrupt berbasis real mode.

**Diagram Proses**

1. Program memulai dengan menampilkan menu utama.

2. Pengguna memilih menu:

A. Start: Stopwatch mulai menghitung waktu.

B. Stop: Stopwatch berhenti menghitung waktu.

C. Reset: Stopwatch mengatur ulang waktu ke nol.

D. Exit: Program keluar.

**BAB III: IMPLEMENTASI**

**3.1 Alat dan Bahan**

**1. Perangkat Lunak**:

EMU8086

Text Editor bawaan EMU8086

**2. Perangkat Keras:**

Laptop/PC

**3.2 Langkah-Langkah Pembuatan**

**1. Desain Program:**

Menentukan struktur utama stopwatch: Start, Stop, dan Reset.

Menggunakan loop atau interrupt timer untuk perhitungan waktu.

**2. Penulisan Kode Assembly:**

Memanfaatkan fungsi INT 21H untuk input/output.

Menggunakan register untuk menyimpan nilai waktu.

**3. Debugging dan Uji Coba:**

Menggunakan debugger EMU8086 untuk menguji jalannya program.

**3.3 Kode Program**

Berikut adalah contoh kode sederhana untuk stopwatch digital

org 100h ; Mengatur titik awal program untuk COM file

.data

msg\_start db 'Stopwatch Started...', 0Dh, 0Ah, '$'

msg\_stop db 'Stopwatch Stopped...', 0Dh, 0Ah, '$'

msg\_reset db 'Stopwatch Reset.', 0Dh, 0Ah, '$'

msg\_menu db '1. Start 2. Stop 3. Reset 4. Exit', 0Dh, 0Ah, '$'

msg\_time db 'Elapsed Time (seconds): ', '$'

elapsed\_time dw 0 ; Variabel untuk menyimpan waktu yang berlalu

.code

start:

mov ah, 09h ; Tampilkan menu utama

lea dx, msg\_menu

int 21h

mov ah, 01h ; Ambil input dari pengguna

int 21h

sub al, '0' ; Ubah input ASCII ke angka

cmp al, 1 ; Pilihan 1: Start Stopwatch

je start\_stopwatch

cmp al, 2 ; Pilihan 2: Stop Stopwatch

je stop\_stopwatch

cmp al, 3 ; Pilihan 3: Reset Stopwatch

je reset\_stopwatch

cmp al, 4 ; Pilihan 4: Exit

je exit\_program

jmp start ; Kembali ke menu jika input tidak valid

start\_stopwatch:

lea dx, msg\_start ; Tampilkan pesan start

mov ah, 09h

int 21h

simulate:

inc word ptr elapsed\_time ; Tambahkan waktu

lea dx, msg\_time ; Tampilkan waktu yang berjalan

mov ah, 09h

int 21h

; Konversi waktu menjadi angka dan cetak

mov ax, elapsed\_time

call print\_number

; Tunggu beberapa saat (simulasi delay)

mov cx, 5000

delay\_loop:

loop delay\_loop

; Kembali ke simulasi jika tidak dihentikan

jmp simulate

stop\_stopwatch:

lea dx, msg\_stop ; Tampilkan pesan stop

mov ah, 09h

int 21h

jmp start

reset\_stopwatch:

lea dx, msg\_reset ; Tampilkan pesan reset

mov ah, 09h

int 21h

mov word ptr elapsed\_time, 0 ; Reset waktu ke nol

jmp start

exit\_program:

mov ah, 4Ch ; Keluar dari program

int 21h

print\_number:

; Cetak angka desimal

push ax

push bx

push cx

xor cx, cx

mov bx, 10

convert\_loop:

xor dx, dx

div bx

push dx

inc cx

cmp ax, 0

jne convert\_loop

print\_digits:

pop dx

add dl, '0'

mov ah, 02h

int 21h

loop print\_digits

pop cx

pop bx

pop ax

ret

end start

**BAB IV: HASIL DAN PEMBAHASAN**

**4.1 Hasil Implementasi**

Stopwatch berhasil dijalankan di EMU8086. Fungsi Start, Stop, dan Reset dapat berjalan dengan baik. Perhitungan waktu menggunakan loop memberikan simulasi yang mendekati waktu nyata.

**4.2 Analisis**

**1. Keuntungan:**

Program sederhana, cocok untuk belajar dasar assembly.

Menggunakan fitur interrupt dan register untuk memahami cara kerja prosesor.

**2. Kelemahan:**

Akurasi waktu tidak sempurna karena menggunakan delay berbasis loop.

Penghitungan waktu real-time membutuhkan timer hardware.

**BAB V: PENUTUP**

**5.1 Kesimpulan**

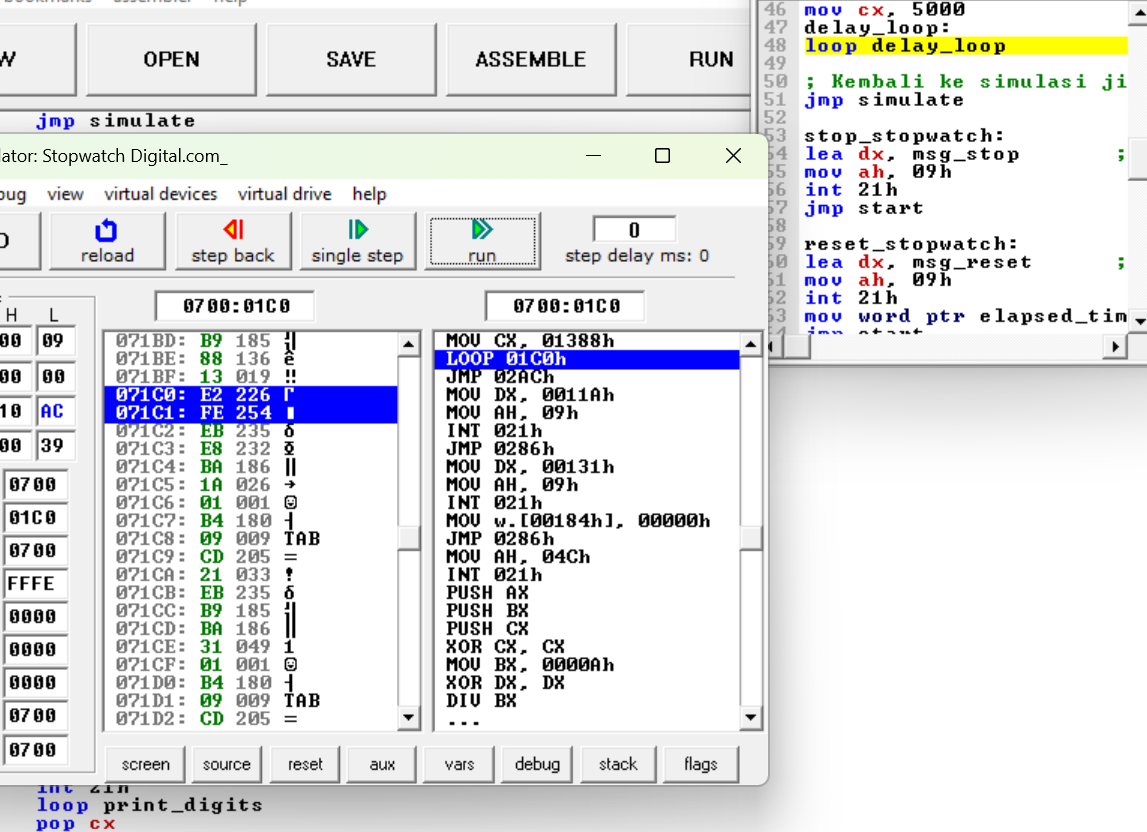
Proyek stopwatch digital memberikan pemahaman dasar tentang pemrograman waktu nyata menggunakan bahasa assembly. Dengan memanfaatkan EMU8086, program dapat dijalankan dalam lingkungan simulasi tanpa memerlukan perangkat keras asli.

**5.2 Saran**

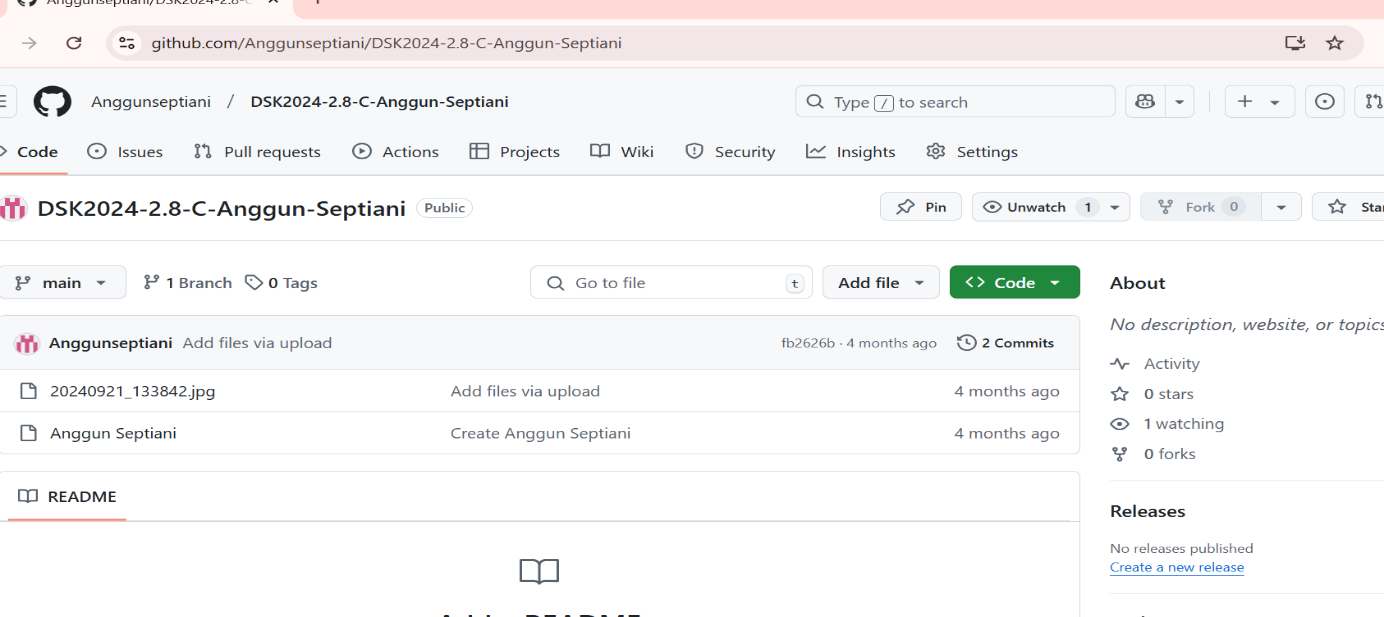
1. Untuk pengembangan lebih lanjut, program dapat diintegrasikan dengan fungsi timer hardware untuk meningkatkan akurasi.

2. Penggunaan algoritma optimasi loop dapat mempercepat eksekusi program.

**Lampiran Di Emu8086**



**Lampiran di Github**



**LINK PPT CANVA**

[**https://www.canva.com/design/DAGbVmSwC-k/ZKp4h6qXS4Wmczojfwd\_7w/edit?utm\_content=DAGbVmSwC-k&utm\_campaign=designshare&utm\_medium=link2&utm\_source=sharebutton**](https://www.canva.com/design/DAGbVmSwC-k/ZKp4h6qXS4Wmczojfwd_7w/edit?utm_content=DAGbVmSwC-k&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

**LINK YOUTUBE**

[**https://youtu.be/sJjBQ3MYeaU?si=6yWtc9\_l-Z92PQDH**](https://youtu.be/sJjBQ3MYeaU?si=6yWtc9_l-Z92PQDH)