

## TUTORIAL LENGKAP SIMULASI SISTEM KONTROL LINGKUNGAN

### BAGIAN 1: SIMULASI HDL (VERILOG) dengan ICARUS VERILOG

#### STEP 1: Download dan Install Software

##### Untuk Windows:

###### 1. Download Icarus Verilog:

- Buka browser, kunjungi: <http://bleyer.org/icarus/>
- Klik link "iverilog-v12-xxxx-setup.exe" (versi terbaru)
- File sekitar 50-80 MB

###### 2. Install Icarus Verilog:

- Double-click file installer
- Klik "Next" → "I Agree" → "Next"
- **PENTING:** Centang "Add to PATH" agar bisa dijalankan dari Command Prompt
- Klik "Install" → tunggu selesai → "Finish"
- GTKWave sudah include dalam installer

###### 3. Verifikasi Instalasi:

- Buka **Command Prompt** (tekan Win+R, ketik cmd, Enter)
- Ketik: iverilog -v
- Jika muncul versi number, instalasi berhasil

##### Untuk Linux (Ubuntu/Debian):

# Update repository

```
sudo apt-get update
```

# Install Icarus Verilog dan GTKWave

```
sudo apt-get install iverilog gtkwave
```

# Cek instalasi

```
iverilog -v
```

```
gtkwave --version
```

### Untuk MacOS:

```
# Install Homebrew jika belum ada
```

```
/bin/bash -c "$(curl -fsSL
```

```
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

```
# Install tools
```

```
brew install icarus-verilog gtkwave
```

```
# Cek instalasi
```

```
iverilog -v
```



## STEP 2: Persiapkan File Verilog

### 1. Buat Folder Project:

- Buat folder baru, misalnya: D:\Environmental\_Control\_HDL
- Atau di Linux/Mac: ~/Environmental\_Control\_HDL

### 2. Buat File Verilog:

- Buka **Notepad++** (Windows) atau **Text Editor** apapun
- Copy kode Verilog yang sudah saya buat di artifact sebelumnya
- Save As: environmental\_control.v
- **PENTING:** Pastikan extension .v (bukan .txt)
- Save di folder yang sudah dibuat



## STEP 3: Compile dan Simulasi

### 1. Buka Command Prompt / Terminal

- Windows: Win+R → ketik cmd → Enter
- Linux/Mac: Buka Terminal

## 2. Navigasi ke Folder Project:

3. cd D:\Environmental\_Control\_HDL

4. # atau di Linux/Mac:

5. cd ~/Environmental\_Control\_HDL

## 6. Compile Kode Verilog:

7. iverilog -o simulation environmental\_control.v

### Penjelasan:

- iverilog = compiler
- -o simulation = output file bernama "simulation"
- environmental\_control.v = source code

**Jika sukses:** Tidak ada error, akan muncul file simulation (atau simulation.exe di Windows)

## 8. Jalankan Simulasi:

9. vvp simulation

### Output yang diharapkan:

Time State S1 S2 S3 S4 S5 S6 A1 A2 A3 A4 A5 A6

```
-----
40  S0  1 1 1 1 1 1 0 0 0 0 0 0
80  S1  0 1 1 1 1 1 0 0 0 0 1 0
120 S2  0 0 1 1 1 1 0 0 1 0 1 0
160 S3  0 0 0 0 1 1 1 0 1 0 1 0
200 S4  0 0 0 0 0 0 1 0 1 1 1 1
240 S0  1 1 1 1 1 1 0 0 0 0 0 0
```

**File baru terbentuk:** environmental\_control.vcd (waveform data)

---

## STEP 4: Visualisasi dengan GTKWave

### 1. Buka GTKWave:

2. gtkwave environmental\_control.vcd

Atau di Windows: double-click file environmental\_control.vcd

### 3. Tampilkan Signals:

- **Panel kiri atas:** Anda akan melihat hierarchy tb\_environmental\_control
- **Klik** pada tb\_environmental\_control untuk expand
- **Klik** pada uut (unit under test)

### 4. Drag Signals ke Waveform:

- Panel tengah atas akan menampilkan daftar signals
- **Pilih signals** yang ingin ditampilkan:
  - clk (clock)
  - reset
  - S1, S2, S3, S4, S5, S6 (sensors)
  - current\_state (state FSM)
  - A1, A2, A3, A4, A5, A6 (actuators)
- **Drag** (klik kiri tahan) ke panel bawah (waveform window)

### 5. Atur Tampilan:

- **Zoom In:** Klik tombol + atau scroll mouse
- **Zoom Out:** Klik tombol -
- **Zoom Fit:** Klik tombol dengan icon "fit"
- **Zoom to Full:** Menu → Time → Zoom → Zoom Best Fit

### 6. Analisis Waveform:

- Lihat perubahan current\_state dari S0 → S1 → S2 → S3 → S4 → S0
- Perhatikan aktivasi actuator sesuai dengan state
- **Screenshot** untuk laporan!

## 7. Tips GTKWave:

- Klik kanan pada signal → Format → Binary/Hex/Decimal
  - Untuk current\_state: pilih format **Decimal** agar mudah dibaca
  - Gunakan **Markers** (klik Insert → Comment) untuk menandai event penting
- 

## TROUBLESHOOTING HDL:

### Problem 1: "iverilog not recognized"

- **Solusi:** PATH belum teraset. Restart Command Prompt atau reboot PC
- Manual add PATH: Control Panel → System → Advanced → Environment Variables → Edit PATH → tambahkan C:\iverilog\bin

### Problem 2: Error saat compile

- **Solusi:** Cek syntax, pastikan tidak ada typo
- Lihat baris error yang ditunjuk oleh compiler

### Problem 3: GTKWave tidak muncul signals

- **Solusi:** Pastikan file .vcd sudah terbentuk
  - Cek testbench sudah ada \$dumpfile dan \$dumpvars
- 

## BAGIAN 2: SIMULASI C# dengan VISUAL STUDIO

### STEP 1: Install Visual Studio 2022 Community

#### 1. Download Visual Studio:

- Kunjungi: <https://visualstudio.microsoft.com/downloads/>
- Klik "**Free download**" pada Visual Studio 2022 Community
- File installer sekitar 3-4 MB (downloader kecil)

#### 2. Install Visual Studio:

- Double-click VisualStudioSetup.exe
- Tunggu installer mempersiapkan

- **Pilih Workload:**
  - ☒ Centang **".NET desktop development"**
  - (Optional) ☒ "Universal Windows Platform development" jika butuh
- Klik **"Install"** (akan download ~5-10 GB, butuh waktu)
- Tunggu sampai selesai (15-30 menit tergantung internet)
- **Restart PC** jika diminta

### 3. Verifikasi Instalasi:

- Buka Visual Studio 2022
- Jika muncul sign-in, bisa skip (klik "Not now")
- Pilih tema warna (Dark/Blue/Light)



## STEP 2: Buat Project Baru

### 1. Launch Visual Studio 2022

### 2. Create New Project:

- Klik **"Create a new project"**
- Atau: Menu File → New → Project

### 3. Pilih Template:

- Di search box, ketik: **"console"**
- Pilih **"Console App"** (dengan icon C#)
- **PENTING:** Pastikan yang dipilih adalah C# (bukan VB.NET atau F#)
- Klik **"Next"**

### 4. Configure Project:

- **Project name:** EnvironmentalControlSystem
- **Location:** Pilih folder, misalnya D:\Projects\
  - ☒ Centang **"Place solution and project in the same directory"**
- Klik **"Next"**

## 5. Framework:

- Pilih **.NET 6.0** atau **.NET 7.0** atau **.NET 8.0** (yang tersedia)
- Klik "**Create**"

## 6. Project Terbuka:

- Akan muncul file Program.cs dengan template code
  - **DELETE semua isi file Program.cs**
- 

## STEP 3: Masukkan Kode dan Jalankan

### 1. Copy Kode:

- Buka artifact C# yang sudah saya buat sebelumnya
- **Select All** (Ctrl+A) → **Copy** (Ctrl+C)

### 2. Paste ke Visual Studio:

- Klik di editor Program.cs
- **Paste** (Ctrl+V)
- Kode akan otomatis ter-format

### 3. Save File:

- Ctrl+S atau File → Save All

### 4. Build Project:

- Menu: **Build → Build Solution**
- Atau tekan: **Ctrl+Shift+B**
- Lihat output window di bawah, pastikan "Build succeeded"

### 5. Run Program:

- Klik tombol ► **Start** (hijau) di toolbar atas
- Atau tekan: **F5** (Run with Debugging)
- Atau tekan: **Ctrl+F5** (Run without Debugging) ← **RECOMMENDED**

### 6. Lihat Output:

- Console window akan muncul
  - Program akan menampilkan 6 test cases
  - Output sesuai dengan desain FSM Anda
- 

#### **STEP 4: Screenshot untuk Laporan**

##### **1. Console Output:**

- Ketika program running, tekan **Alt+Print Screen** untuk screenshot console
- Paste ke Paint/Word untuk crop dan save

##### **2. Kode Editor:**

- Screenshot tampilan kode di Visual Studio
- Zoom in agar kode terbaca jelas

##### **3. Build Success:**

- Screenshot output window yang menunjukkan "Build succeeded"
- 

#### **STEP 5: Modifikasi untuk Testing Tambahan (Opsional)**

Tambahkan test case custom di dalam Main():

```
// Test Case Custom: VOC + Dust abnormal
```

```
Console.WriteLine("\nTEST CUSTOM: VOC + Dust Abnormal");
```

```
sensors.S1_Temperature = true;
```

```
sensors.S2_Humidity = true;
```

```
sensors.S3_VOC = false;    // Abnormal
```

```
sensors.S4_Dust = false;   // Abnormal
```

```
sensors.S5_Airflow = true;
```

```
sensors.S6_Light = true;
```

```
fsm.UpdateSensors(sensors);
```

```
fsm.Execute();
```



```
Console.WriteLine(fsm.GetStatus());  
  
Thread.Sleep(1000);
```

---

### **BAGIAN 3: ALTERNATIF - VISUAL STUDIO CODE (Lebih Ringan)**

#### **STEP 1: Install VS Code dan .NET SDK**

##### **1. Install .NET SDK:**

- Kunjungi: <https://dotnet.microsoft.com/download>
- Download **.NET 8.0 SDK** (atau versi terbaru)
- Install dengan Next-Next-Finish

##### **2. Install Visual Studio Code:**

- Kunjungi: <https://code.visualstudio.com/>
- Download dan install

##### **3. Install C# Extension:**

- Buka VS Code
  - Klik icon Extensions (Ctrl+Shift+X)
  - Search: "**C#**"
  - Install extension "**C# Dev Kit**" by Microsoft
- 

#### **STEP 2: Buat Project di VS Code**

##### **1. Buka Terminal di VS Code:**

- Menu: Terminal → New Terminal
- Atau tekan: **Ctrl+`** (backtick)

##### **2. Navigate ke Folder:**

3. `cd D:\Projects`

4. `# atau`

5. `mkdir EnvironmentalControl`

6. `cd EnvironmentalControl`
  7. **Create New Console Project:**
  8. `dotnet new console -n EnvironmentalControlSystem`
  9. `cd EnvironmentalControlSystem`
  10. **Buka Project di VS Code:**
  11. `code .`
  12. **Edit Program.cs:**
    - Klik file Program.cs di sidebar kiri
    - Delete semua isi
    - Paste kode C# dari artifact saya
    - Save (Ctrl+S)
- 

### **STEP 3: Run Program di VS Code**

#### **Cara 1: Via Terminal**




`dotnet run`



#### **Cara 2: Via Debugger**

- Tekan **F5**
  - Pilih "C#" jika ditanya
  - Program akan running
- 






### **CHECKLIST UNTUK LAPORAN**

#### **Untuk HDL (Verilog):**

-  Screenshot Command Prompt menunjukkan compile berhasil
-  Screenshot output text simulasi (dari vvp simulation)
-  Screenshot GTKWave menampilkan waveform lengkap

-  Screenshot detail transisi state  $S0 \rightarrow S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S0$
-  Screenshot aktivasi actuator saat state tertentu

#### Untuk C# (Mikrocontroller):

-  Screenshot Visual Studio dengan kode terbuka
  -  Screenshot Build succeeded
  -  Screenshot console output lengkap (semua test cases)
  -  Screenshot detail output menunjukkan transisi state
  -  Screenshot struktur project di Solution Explorer
- 

#### TIPS PENTING:

##### 1. Untuk HDL:

- Selalu compile ulang jika ubah kode: `iverilog -o simulation environmental_control.v`
- Waveform tidak update? Hapus file .vcd lama, run ulang simulasi
- Gunakan marker di GTKWave untuk menandai transisi penting

##### 2. Untuk C#:

- Gunakan `Console.ReadKey()` di akhir agar console tidak langsung tertutup
- Tambahkan `Thread.Sleep(1000)` agar output lebih mudah dibaca
- Build ulang jika ada perubahan code

##### 3. Dokumentasi:

- Screenshot harus jelas dan terbaca
  - Beri label/caption pada setiap screenshot
  - Bandingkan hasil simulasi dengan Truth Table di jurnal
- 

#### FAQ (Frequently Asked Questions)

**Q: File .vcd tidak terbentuk?** A: Cek testbench ada \$dumpfile dan \$dumpvars. Run ulang vvp simulation.

**Q: GTKWave tidak bisa dibuka?** A: Install ulang Icarus Verilog, pastikan GTKWave tercentang saat install.

**Q: Visual Studio terlalu berat?** A: Gunakan VS Code + .NET SDK (lebih ringan, ~2 GB total).





**Q: Error "System.Threading not found"?** A: Sudah included di .NET default, pastikan using statement ada di atas.

**Q: Bagaimana cara export waveform dari GTKWave?** A: File → Print → Print to File → Save as PDF atau PNG.

---

## KESIMPULAN

Setelah mengikuti tutorial ini, Anda akan memiliki:

1.  **Simulasi HDL** yang berjalan di Icarus Verilog dengan visualisasi GTKWave
2.  **Simulasi C#** yang berjalan di Visual Studio/VS Code
3.  **Screenshot lengkap** untuk dokumentasi jurnal
4.  **Pemahaman praktis** implementasi FSM dalam hardware dan software

**Selamat mengerjakan! Semoga jurnalnya mendapat nilai bagus!** 