



Rancang Bangun Simulator Modul Kendali untuk Motor Operasi Katub Sebagai Media Pembelajaran Bagi Para Teknisi Baru di PLTU Nagan Raya

Eko Prasetyo Hadi^{1*}, Hamdani², Ahmad Dani³

¹⁻³Teknik Elektro, Fakultas Teknik, Universitas Pembangunan Panca Budi, Medan, Indonesia

Email: prasetyo.hadi@plnnusantarapower.co.id^{1*}, hamdani.stmt@dosen.pancabudi.ac.id², ahmad.kartasmita@gmail.com³

*Penulis korespondensi: prasetyo.hadi@plnnusantarapower.co.id¹

Abstract. *The Motor Operated Valve (MOV) is a critical component in fluid control systems at Steam Power Plants (PLTU). Training new technicians is often hindered by limited access to actual equipment and operational safety risks. This research aims to design and develop an Arduino-based MOV control module simulator capable of simulating basic functions such as open, close, stop, and limit switch responses. The method used is Research and Development (R&D) with an experimental approach. The simulator was tested using a DC motor as the simulated valve actuator, equipped with push buttons, relays, limit switches, and indicator lamps for visual feedback. The test results showed that the simulator successfully represented control functions with 100% accuracy in limit switch responses and consistent operation. User evaluations involving ten new technicians indicated an 85% satisfaction rate in terms of ease of understanding and operational safety. This simulator has proven to be an effective, interactive, and safe learning medium for new technicians at PLTU Nagan Raya.*

Keywords: *Arduino; Learning Media; Motor Operated Valve; Simulator; Steam Power Plant*

Abstract. *Motor Operated Valve (MOV) merupakan komponen kritis dalam sistem kontrol fluida di Pembangkit Listrik Tenaga Uap (PLTU). Pelatihan teknisi baru sering terkendala keterbatasan akses ke peralatan aktual dan risiko operasional. Penelitian ini bertujuan untuk merancang dan membangun simulator modul kendali MOV berbasis Arduino yang dapat mensimulasikan fungsi dasar open, close, stop, dan respon limit switch. Metode yang digunakan adalah Research and Development (R&D) dengan pendekatan eksperimental. Simulator diuji menggunakan motor DC sebagai penggerak katup simulasi, dilengkapi dengan push button, relay, limit switch, dan lampu indikasi untuk umpan balik visual. Hasil pengujian menunjukkan simulator berhasil merepresentasikan fungsi kendali dengan akurasi 100% pada respon limit switch dan konsistensi operasi. Evaluasi pengguna terhadap 10 teknisi baru menunjukkan tingkat kepuasan 85% dalam hal kemudahan pemahaman dan keamanan penggunaan. Simulator ini terbukti efektif sebagai media pembelajaran yang interaktif dan aman bagi teknisi baru di PLTU Nagan Raya.*

Kata Kunci: *Arduino; Media Pembelajaran; Motor Operated Valve; PLTU; Simulator*

1. PENDAHULUAN

Motor Operated Valve (MOV) merupakan jantung dari sistem kontrol fluida di Pembangkit Listrik Tenaga Uap (PLTU) Nagan Raya, berperan sebagai gatekeeper yang mengatur aliran uap, air, dan fluida kritis lainnya dalam sistem boiler dan turbin. Sebagai komponen vital, kinerja MOV yang optimal menjadi penentu efisiensi dan keandalan operasional pembangkit. Namun, ironisnya, penguasaan teknis terhadap sistem kendali MOV justru menjadi tantangan besar bagi teknisi baru yang harus berhadapan dengan kompleksitas sistem tanpa bekal pelatihan yang memadai.

Dilema yang dihadapi cukup serius di satu sisi, pelatihan langsung di lapangan berisiko tinggi terhadap keselamatan kerja dan kontinuitas operasi pembangkit. Setiap kesalahan operasi dapat berakibat fatal, mulai dari kerusakan peralatan bernilai miliaran rupiah hingga

potensi kecelakaan kerja yang membahayakan jiwa. Di sisi lain, keterbatasan akses ke peralatan aktual membuat kurva pembelajaran teknisi baru menjadi lambat, berpotensi menciptakan celah kompetensi yang berisiko terhadap operasional PLTU dalam jangka panjang.

Revolusi digital di era Industri 4.0 membawa angin segar melalui teknologi mikrokontroler yang terjangkau dan andal. Arduino, sebagai platform open-source, telah membuktikan diri sebagai game-changer dalam dunia pendidikan teknik dan otomasi industri. Bukti empiris dari penelitian sebelumnya menunjukkan bahwa Arduino tidak hanya mampu mereplikasi sistem industri dengan akurasi tinggi, tetapi juga menawarkan fleksibilitas dan ekonomis yang tak tertandingi.

Berdasarkan gap kompetensi dan peluang teknologi inilah, penelitian ini hadir dengan misi strategis mengembangkan simulator modul kendali MOV berbasis Arduino yang tidak hanya aman dan interaktif, tetapi juga mampu memberikan pengalaman belajar immersive yang mendekati kondisi riil lapangan. Simulator ini dirancang sebagai jembatan antara teori dan praktik, memungkinkan teknisi baru untuk melakukan trial and error tanpa resiko, mengasah kemampuan troubleshooting, dan membangun muscle memory dalam menghadapi berbagai skenario operasional.

Tujuan penelitian ini adalah:

- a. Merancang dan membuat prototipe simulator modul kendali MOV
- b. Menguji kinerja simulator dalam mensimulasikan fungsi open, close, stop, dan limit switch
- c. Menilai efektivitas simulator sebagai media pembelajaran bagi teknisi baru

2. TINJAUAN PUSTAKA

Motor Operated Valve (MOV)

Motor Operasi Katup (MOV) merupakan sebuah sistem aktuasi elektromekanis yang mengintegrasikan motor listrik sebagai penggerak utama dengan katup industri untuk mengatur dan mengendalikan aliran fluida secara presisi. Dalam konteks engineering, MOV bukan sekadar katup biasa, melainkan sebuah sistem cerdas yang menggabungkan prinsip-prinsip mechanical engineering, electrical engineering, dan control system dalam satu paket terpadu.

Prinsip Kerja dan Konfigurasi Sistem

MOV bekerja berdasarkan konversi energi listrik menjadi energi mekanik melalui motor listrik, yang kemudian ditransmisikan melalui sistem reduksi gearbox untuk menghasilkan torsi yang adequate dalam mengoperasikan katup. Sistem ini dirancang untuk melakukan operasi buka-tutup (on-off) maupun modulasi (throttling) dengan akurasi tinggi,

tergantung pada tipe dan aplikasinya di lapangan.

Anatomi Komponen Utama MOV:

Motor Penggerak (Prime Mover)

- a. Biasanya menggunakan motor listrik 3-fase induksi.
- b. Dilengkapi dengan thermal overload protection untuk mencegah kerusakan akibat beban berlebih
- c. Memiliki kemampuan torque yang disesuaikan dengan ukuran dan tipe katup

Gearbox (Sistem Reduksi)

- a. Menggunakan planetary gear system atau worm gear untuk meningkatkan torsi
- b. Rasio reduksi typically antara 10:1 hingga 100:1 tergantung aplikasi
- c. Dilengkapi dengan mekanisme manual override untuk operasi darurat

Limit Switch System

- a. Berfungsi sebagai posisi indicator dan safety cut-off
- b. Terdiri dari limit switch buka penuh (fully open) dan tutup penuh (fully closed)
- c. Beberapa tipe advanced MOV dilengkapi dengan intermediate limit switch

Sistem Kontrol dan Monitoring

- a. Control panel dengan programmable logic controller (PLC)
- b. Local control station dengan selector switch (Local/Off/Remote)
- c. Feedback mechanism untuk posisi dan status katup
- d. Protection circuit terhadap phase failure dan reverse rotation



Gambar 1. Motor Operated Valve (MOV).

Sistem Kendali Berbasis Arduino

Arduino Uno merupakan salah satu varian board mikrokontroler yang paling populer dalam keluarga Arduino, menggunakan ATmega328P sebagai unit pemroses utamanya. Board ini menawarkan konfigurasi pin yang komprehensif dengan 14 pin input/output digital dimana 6 di antaranya memiliki kemampuan output PWM (Pulse Width Modulation), ditambah 6 channel input analog. Sebagai sumber pewaktuan, Arduino Uno mengandalkan kristal osilator 16 MHz untuk sinkronisasi operasi mikrokontroler.

Board ini dilengkapi dengan antarmuka USB tipe B untuk koneksi ke komputer dan pemrograman, serta jack DC untuk suplai daya eksternal. Untuk keperluan pemrograman tingkat lanjut, tersedia header ICSP (In-Circuit Serial Programming) yang memungkinkan pemrograman langsung ke mikrokontroler. Tombol reset pada board berfungsi untuk merestart eksekusi program tanpa harus mematikan daya.

Dalam operasionalnya, Arduino Uno dapat diaktifkan melalui koneksi USB dari komputer atau melalui sumber daya eksternal seperti baterai maupun adaptor AC-DC. Board ini mengintegrasikan chip ATmega16U2 yang berperan sebagai konverter USB-to-serial, memfasilitasi komunikasi data antara komputer dan mikrokontroler utama. Tampak atas dari arduino uno dapat dilihat pada Gambar 2.2.

Adapun data teknis *board* Arduino Uno R3 adalah sebagai berikut:

Tabel 1. Spesifikasi Arduino Uno.

Parameter Teknis	Spesifikasi
Mikrokontroler Utama	ATmega328P
Tegangan Operasi	5 V DC
Rentang Tegangan Input (Disarankan)	7-12 V DC
Batas Maksimal Tegangan Input	6-20 V DC
Jumlah Pin Digital I/O	14 pin (6 pin dengan fitur PWM)
Jumlah Pin Input Analog	6 channel
Kapasitas Arus per Pin I/O	40 mA
Kapasitas Arus Output 3.3V	150 mA
Memori Flash	32 KB (termasuk 0.5 KB untuk bootloader)
Kapasitas SRAM	2 KB
Kapasitas EEPROM	1 KB
Frekuensi Clock	16 MHz



Gambar 2. Arduino uno.

Pin Pada Arduino Uno

Setiap pin digital pada Arduino Uno dapat difungsikan sebagai *input* maupun *output* dengan menggunakan perintah `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Pin-pin ini beroperasi pada tegangan kerja 5 volt dan mampu menangani arus hingga 40 mA. Selain itu,

setiap pin memiliki *resistor pull-up* internal dengan nilai antara 20 hingga 30 k Ω yang secara default berada dalam kondisi tidak aktif.

Beberapa pin digital memiliki fungsi tambahan yang lebih spesifik, antara lain:

- a. Komunikasi Serial: Pin 0 (RX) dan pin 1 (TX) digunakan untuk menerima dan mengirim data secara serial.
- b. Interupsi Eksternal: Pin 2 dan 3 dapat dikonfigurasi sebagai *interrupt pin* untuk merespons perubahan logika, baik pada kondisi *rising edge*, *falling edge*, maupun saat terjadi perubahan nilai sinyal.
- c. Pulse Width Modulation (PWM): Pin 3, 5, 6, 9, 10, dan 11 mampu menghasilkan sinyal PWM 8-bit dengan fungsi `analogWrite()`.
- d. Serial Peripheral Interface (SPI): Pin 10 (SS), 11 (MOSI), 12 (MISO), dan 13 (SCK) digunakan untuk komunikasi SPI dengan memanfaatkan *SPI library*.
- e. Indikator LED Bawaan: Pin digital 13 telah terhubung ke LED internal; ketika bernilai HIGH, LED akan menyala, dan saat bernilai LOW, LED akan mati.

Selain pin digital, Arduino Uno juga memiliki enam pin analog (A0–A5) dengan resolusi 10 bit atau 1024 tingkat pembacaan. Secara default, pin ini membaca tegangan dari 0 hingga 5 volt, namun batas atas dapat diubah menggunakan pin AREF bersama fungsi `analogReference()`. Dua pin terakhir, yaitu A4 dan A5, memiliki fungsi tambahan untuk komunikasi *Two-Wire Interface (TWI)* atau *Inter-Integrated Circuit (I2C)* menggunakan *Wire library*.

Sumber Daya Arduino Uno

Arduino Uno dapat menerima pasokan daya melalui dua jalur, yaitu melalui koneksi USB (Universal Serial Bus) atau power supply eksternal. Apabila kedua sumber daya tersebut terhubung secara bersamaan, sistem internal Arduino Uno akan secara otomatis memilih sumber daya yang paling sesuai untuk digunakan. Daya eksternal dapat berasal dari adaptor AC–DC maupun dari baterai. Adaptor dapat dihubungkan langsung ke soket *power jack* pada papan Arduino Uno, sedangkan baterai dapat dihubungkan melalui pin GND dan Vin pada konektor daya.

Secara umum, Arduino Uno dapat beroperasi dengan tegangan antara 6 hingga 20 volt. Namun, terdapat batas aman yang direkomendasikan oleh pabrikan, yaitu 7 hingga 12 volt. Jika tegangan yang diberikan kurang dari 7 volt, maka keluaran dari pin 5V tidak akan mencapai tegangan nominal dan dapat menyebabkan sistem menjadi tidak stabil. Sebaliknya, jika tegangan input melebihi 12 volt, maka *voltage regulator* berpotensi mengalami panas berlebih (*overheating*) yang dapat merusak komponen internal papan Arduino.

Pin-pin tegangan yang tersedia pada Arduino Uno antara lain:

- a. Vin – berfungsi sebagai jalur masukan daya ketika menggunakan sumber eksternal selain USB. Pin ini juga dapat digunakan untuk menyalurkan daya ke papan Arduino jika tegangan diberikan melalui *power jack*.
- b. 5V – menyediakan tegangan teratur sebesar 5 volt yang dihasilkan oleh *onboard voltage regulator*. Tegangan ini juga menjadi sumber utama bagi sebagian besar komponen dan sensor eksternal.
- c. 3.3V – menghasilkan tegangan 3,3 volt yang berasal dari regulator internal, biasanya digunakan untuk modul atau sensor yang membutuhkan tegangan rendah.
- d. GND – berfungsi sebagai pin *ground* atau referensi negatif dari seluruh sistem daya dan sinyal pada papan Arduino.

Relay

Relay merupakan komponen elektromekanis yang berfungsi untuk menghubungkan atau memutus aliran listrik dalam suatu rangkaian secara otomatis. Perangkat ini bekerja berdasarkan prinsip elektromagnetik, di mana arus listrik yang mengalir melalui kumparan (coil) akan menghasilkan medan magnet yang menarik tuas atau saklar mekanis sehingga dapat membuka atau menutup kontak listrik di rangkaian lainnya. Dengan mekanisme ini, relay memungkinkan suatu rangkaian bertegangan rendah mengendalikan beban dengan tegangan atau arus yang lebih tinggi.

Dalam pemilihan relay, perhatian khusus perlu diberikan pada spesifikasi koil dan kontakannya, terutama terhadap kemampuan arus maksimum yang dapat ditangani. Ketika proses buka-tutup rangkaian terjadi, sering kali timbul loncatan atau percikan listrik (arc) pada kontak mekanis. Jika hal ini tidak diperhitungkan dengan benar, maka kontak dapat mengalami pemanasan berlebih (overheating) yang berpotensi menyebabkan kerusakan permanen dan memperpendek umur operasi relay.

Secara umum, relay banyak digunakan dalam sistem otomasi, proteksi, serta rangkaian kendali motor atau beban listrik lainnya, karena kemampuannya dalam memberikan isolasi listrik antara rangkaian kendali dan rangkaian daya. Contoh relay pada gambar 2.3



Gambar 3. Relay.

limit Switch

Limit switch merupakan salah satu jenis saklar mekanik yang berfungsi untuk mendeteksi posisi atau pergerakan suatu objek secara fisik. Komponen ini bekerja berdasarkan prinsip kontak mekanis, di mana tuas atau aktuator pada limit switch akan terdorong atau tertekan oleh bagian yang bergerak — seperti katup, motor, atau lengan mekanis — sehingga mengubah kondisi kontak listrik dari *normally open (NO)* menjadi *normally closed (NC)*, atau sebaliknya.

Dalam sistem kendali otomatis, limit switch berperan penting sebagai sensor posisi atau pengaman batas gerak. Misalnya pada sistem motor operasi katup (MOV), limit switch digunakan untuk menentukan posisi *fully open* dan *fully close* dari katup. Ketika katup mencapai batas pembukaannya, limit switch akan aktif untuk menghentikan arus ke motor, sehingga mencegah motor terus berputar yang dapat menyebabkan kerusakan mekanik. Contoh limit switch pada gambar 4.



Gambar 4. Limit Switch.

3. METODE PENELITIAN

Skema Penganbungan Seluruh Instalasi Sistem Pembentukan Alat

Perancangan simulator modul kendali motor operasi katup dilakukan dengan menggabungkan beberapa komponen utama yang terdiri dari sistem kontrol berbasis Arduino Uno, rangkaian relay driver, motor penggerak katup (DC motor), limit switch, serta tombol kendali (push button) dan indikator LED. Setiap komponen memiliki fungsi spesifik yang saling terhubung membentuk satu sistem kendali terintegrasi sebagaimana pada instalasi di lapangan.

Sistem Kendali Utama (Arduino Uno)

Arduino Uno berfungsi sebagai pusat pengendali (*main controller*) yang menerima sinyal dari tombol input dan sensor limit switch, kemudian mengatur keluaran ke modul relay untuk mengendalikan arah putaran motor. Program utama ditulis menggunakan Arduino IDE dengan bahasa pemrograman C++ dan logika kendali berbasis *if-else*.

Rangkaian Tombol Kendali

Tombol kendali terdiri dari tiga tombol utama, yaitu Open, Stop, dan Close.

- Tombol *Open* mengirimkan sinyal logika HIGH ke pin input Arduino untuk mengaktifkan relay pembuka katup.
- Tombol *Close* mengirimkan sinyal untuk menutup katup dengan mengaktifkan relay kebalikan arah putaran motor.
- Tombol *Stop* digunakan untuk menghentikan operasi dengan memutus sinyal kendali ke kedua relay.

Setiap tombol dilengkapi resistor pull-down 10K Ω agar sinyal input tetap stabil ketika tombol tidak ditekan.

Modul Relay dan Penggerak Motor

Modul relay 4 channel digunakan sebagai penghubung antara Arduino dengan beban motor AC/DC. Relay berfungsi sebagai saklar elektronik yang dikontrol oleh Arduino untuk mengatur arah putaran motor sesuai perintah tombol.

Motor DC kecil digunakan untuk mensimulasikan gerakan buka-tutup katup. Putaran motor akan berhenti secara otomatis ketika salah satu limit switch aktif, menandakan posisi fully open atau fully close telah tercapai.

Limit Switch

Dua buah limit switch dipasang pada posisi akhir putaran motor, masing-masing berfungsi untuk mendeteksi batas *open* dan *close*. Sinyal dari limit switch dikirim ke pin input Arduino untuk menghentikan arus ke motor melalui relay. Hal ini mencegah motor berputar melebihi batas mekanis dan mensimulasikan sistem proteksi seperti pada MOV sebenarnya.

Indikator LED

Tiga buah LED indikator digunakan untuk memberikan umpan balik visual kepada pengguna:

- LED Merah menyala saat katup terbuka penuh (*open*),
- LED Hijau menyala saat katup tertutup penuh (*close*),
- LED Kuning menyala saat sistem dalam kondisi *stop* atau netral.



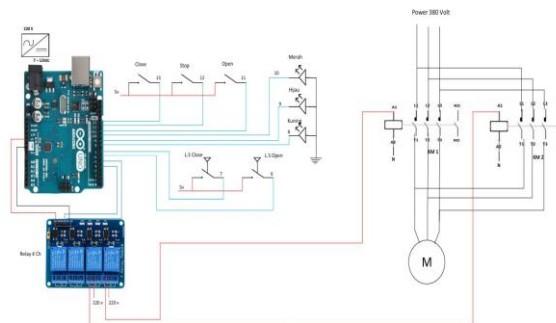
Gambar 5. Lampu LED.

Skema Penggabungan Sistem

Semua komponen di atas dihubungkan pada panel simulator berbahan akrilik. Skema keseluruhan sistem terdiri dari:

- Bagian input: tombol *open*, *stop*, *close*, serta dua *limit switch*.
- Bagian pengendali: papan Arduino Uno dan rangkaian relay driver.
- Bagian output: motor DC dan LED indikator.

Sistem bekerja secara terintegrasi, di mana sinyal input diproses oleh Arduino untuk menghasilkan aksi kontrol terhadap motor melalui relay, sementara limit switch berfungsi sebagai umpan balik (*feedback*) untuk menghentikan pergerakan ketika posisi akhir tercapai.



Gambar 6. Skema seluruh system.

Metode Pengujian

Setelah seluruh komponen simulasi MOV selesai pada gambar 3.1 di atas, maka selanjutnya Pengujian dilakukan dalam beberapa tahap berikut:

Pengujian Fungsional Komponen

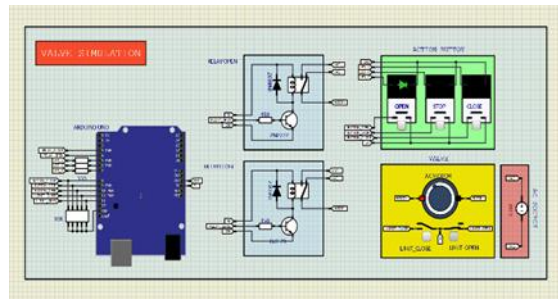
Setiap komponen diuji secara individual untuk memastikan kondisi awalnya berfungsi normal.

- Tombol kendali: diperiksa menggunakan multimeter untuk memastikan koneksi NO/NC berfungsi.
- Relay: diuji dengan mengaktifkan sinyal digital dari Arduino untuk memastikan modul dapat membuka dan menutup rangkaian dengan stabil.
- LED indikator: diuji dengan memberikan sinyal HIGH/LOW pada pin kontrol untuk memastikan tampilan visual bekerja.
- Limit switch: diuji secara manual untuk memastikan perubahan logika input Arduino saat tuas ditekan.
- Motor DC: diuji untuk memastikan arah putaran sesuai dengan logika relay (open dan close).

Pengujian Program Kendali Arduino

Kode program Arduino diuji menggunakan Serial Monitor untuk memastikan setiap perintah dari tombol menghasilkan keluaran logika yang benar. Pengujian mencakup:

- Respons waktu Arduino terhadap input tombol.
- Kesesuaian status LED dengan kondisi operasi (open, stop, close).
- Otomatisasi penghentian motor saat limit switch aktif.



Gambar 7. Simulasi Pada Aplikasi IDE.

Pengujian Integrasi Sistem

Setelah semua komponen diverifikasi, sistem diuji secara keseluruhan dengan tahapan berikut:

- Tekan tombol Open, amati apakah motor berputar ke arah membuka dan LED merah menyala.
- Ketika katup simulasi mencapai batas *limit open*, sistem harus otomatis berhenti.
- Tekan tombol Close, amati apakah motor berputar ke arah menutup dan LED hijau menyala.
- Ketika limit switch *close* terpicu, motor harus berhenti otomatis.
- Tekan tombol Stop saat motor beroperasi untuk menguji respon penghentian manual.

Pengujian Keandalan (Reliability Test)

Sistem diuji dalam 20 siklus operasi berturut-turut (*open-stop-close*) untuk menilai kestabilan dan konsistensi fungsi logika kendali. Setiap siklus diamati waktu respon motor, status LED, serta akurasi penghentian limit switch.

Program Sistem Arduino

Adapun rincian bahasa C yang dimasukan pad sistem arduino adalah sebagai berikut :

```
// =====
```

```
// Pin limit switch
```

```
const int limitOpenPin = 12;
```

```
const int limitClosePin = 11;
```

```

// Pin tombol
const int buttonOpenPin = 10;
const int buttonStopPin = 9;
const int buttonClosePin = 8;

// Pin LED indikator
const int ledClosePin = 5;
const int ledOpenPin = 6;
const int ledStopPin = 7; // LED kuning untuk posisi tengah (STOP)


// Pin relay
const int relayClosePin = 3;
const int relayOpenPin = 4;


// Variables for blinking
unsigned long previousMillis = 0;
const long blinkInterval = 300; // 300ms blink interval
bool ledState = LOW;


// State variable
enum GateState { STOPPED, OPENING, CLOSING };
GateState gateState = STOPPED;


void setup() {
  Serial.begin(9600);

  // Input pakai internal pull-up (tanpa resistor eksternal)
  pinMode(buttonOpenPin, INPUT_PULLUP);
  pinMode(buttonClosePin, INPUT_PULLUP);
  pinMode(buttonStopPin, INPUT_PULLUP);
  pinMode(limitOpenPin, INPUT_PULLUP);
  pinMode(limitClosePin, INPUT_PULLUP);

  // Output relay dan LED
  pinMode(relayOpenPin, OUTPUT);

```

```
pinMode(relayClosePin, OUTPUT);
pinMode(ledOpenPin, OUTPUT);
pinMode(ledClosePin, OUTPUT);
pinMode(ledStopPin, OUTPUT);

// Set awal: semua relay OFF, LED stop ON
digitalWrite(relayOpenPin, LOW);
digitalWrite(relayClosePin, LOW);
digitalWrite(ledOpenPin, LOW);
digitalWrite(ledClosePin, LOW);
digitalWrite(ledStopPin, HIGH);
}

void loop() {
  // Karena pakai INPUT_PULLUP → tombol aktif saat LOW
  bool openPressed = digitalRead(buttonOpenPin) == LOW;
  bool closePressed = digitalRead(buttonClosePin) == LOW;
  bool stopPressed = digitalRead(buttonStopPin) == LOW;
  bool openLimitHit = digitalRead(limitOpenPin) == LOW;
  bool closeLimitHit = digitalRead(limitClosePin) == LOW;

  // =====
  // KONDISI: TOMBOL OPEN DITEKAN
  // =====
  if (openPressed && !openLimitHit) {
    Serial.println("OPEN pressed");
    gateState = OPENING;
    digitalWrite(relayOpenPin, HIGH);
    digitalWrite(relayClosePin, LOW);
    digitalWrite(ledStopPin, LOW);
  }

  // =====
  // KONDISI: TOMBOL CLOSE DITEKAN
```

```

// =====
else if (closePressed && !closeLimitHit) {
    Serial.println("CLOSE pressed");
    gateState = CLOSING;
    digitalWrite(relayClosePin, HIGH);
    digitalWrite(relayOpenPin, LOW);
    digitalWrite(ledStopPin, LOW);
}

// =====
// KONDISI: TOMBOL STOP DITEKAN
// =====
else if (stopPressed) {
    Serial.println("STOP pressed");
    gateState = STOPPED;

    // Semua relay OFF
    digitalWrite(relayOpenPin, LOW);
    digitalWrite(relayClosePin, LOW);

    // Jika limit open masih aktif → tetap tunjukkan posisi OPEN
    if (openLimitHit) {
        Serial.println("STOP ditekan saat di posisi OPEN");
        digitalWrite(ledOpenPin, HIGH);
        digitalWrite(ledClosePin, LOW);
        digitalWrite(ledStopPin, LOW);
    }

    // Jika limit close masih aktif → tetap tunjukkan posisi CLOSE
    else if (closeLimitHit) {
        Serial.println("STOP ditekan saat di posisi CLOSE");
        digitalWrite(ledOpenPin, LOW);
        digitalWrite(ledClosePin, HIGH);
        digitalWrite(ledStopPin, LOW);
    }
}

```

```
// Jika tidak ada limit aktif (posisi tengah) → tampilkan LED kuning (STOP)
else {
    Serial.println("STOP ditekan di posisi tengah");
    digitalWrite(ledOpenPin, LOW);
    digitalWrite(ledClosePin, LOW);
    digitalWrite(ledStopPin, HIGH);
}
}

// =====
// KONDISI: LIMIT SWITCH OPEN AKTIF
// =====

if (openLimitHit && gateState == OPENING) {
    Serial.println("LIMIT OPEN aktif → semua relay OFF");
    gateState = STOPPED;

    // Semua relay de-energize
    digitalWrite(relayOpenPin, LOW);
    digitalWrite(relayClosePin, LOW);

    // Indikasi posisi OPEN
    digitalWrite(ledOpenPin, HIGH);
    digitalWrite(ledClosePin, LOW);
    digitalWrite(ledStopPin, LOW);
}

// =====
// KONDISI: LIMIT SWITCH CLOSE AKTIF
// =====

if (closeLimitHit && gateState == CLOSING) {
    Serial.println("LIMIT CLOSE aktif → semua relay OFF");
    gateState = STOPPED;

    // Semua relay de-energize
    digitalWrite(relayOpenPin, LOW);
```

```

digitalWrite(relayClosePin, LOW);

// Indikasi posisi CLOSE
digitalWrite(ledOpenPin, LOW);
digitalWrite(ledClosePin, HIGH);
digitalWrite(ledStopPin, LOW);
}

// =====
// BLINKING LED SAAT PROSES GERAK
// =====

if (gateState == OPENING && !openLimitHit) {
    unsigned long currentMillis = millis();
    if (currentMillis - previousMillis >= blinkInterval) {
        previousMillis = currentMillis;
        ledState = !ledState;
        digitalWrite(ledOpenPin, ledState);
        digitalWrite(ledClosePin, LOW);
    }
}

if (gateState == CLOSING && !closeLimitHit) {
    unsigned long currentMillis = millis();
    if (currentMillis - previousMillis >= blinkInterval) {
        previousMillis = currentMillis;
        ledState = !ledState;
        digitalWrite(ledClosePin, ledState);
        digitalWrite(ledOpenPin, LOW);
    }
}

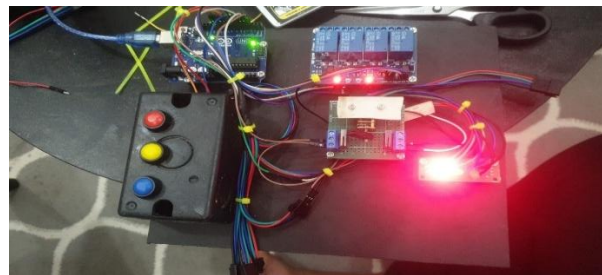
```


4. HASIL DAN PEMBAHASAN

Prototipe Simulator

Prototipe simulator berhasil dibuat dengan komponen utama:

- Arduino Uno
- Motor DC 12V sebagai aktuator
- 2-channel relay modul
- Limit switch NC dan NO
- Push button (open, close, stop)
- LED untuk indikasi visual



Gambar 8. Hasil Rangkaian Simulator MOV.

Hasil Pengujian Fungsional

Pengujian dilakukan dengan 20 kali percobaan untuk setiap fungsi:

Tabel 2. Hasil Pengujian Fungsional Simulator.

Fungsi	Respon yang Diharapkan	Hasil Pengujian	Akurasi
Open	Motor berputar buka	20/20 berhasil	100%
Close	Motor berputar tutup	20/20 berhasil	100%
Stop	Motor berhenti	20/20 berhasil	100%
Limit Open	Motor stop di posisi open	20/20 berhasil	100%
Limit Close	Motor stop di posisi close	20/20 berhasil	100%

Pembahasan

Simulator yang dirancang berhasil mereplikasi fungsi dasar MOV dengan akurasi tinggi. Penggunaan Arduino memudahkan implementasi logika kendali dan integrasi dengan komponen eksternal. Hasil pengujian fungsional menunjukkan konsistensi sistem dalam merespon perintah dan kondisi limit switch.

Dari segi pembelajaran, simulator ini memberikan pengalaman praktis tanpa risiko operasional. Teknisi baru dapat memahami alur kendali MOV, prinsip kerja limit switch, dan prosedur operasi yang aman.

Keterbatasan penelitian ini adalah penggunaan motor DC kecil yang tidak merepresentasikan torsi sebenarnya pada MOV industri. Namun, untuk tujuan pembelajaran, simulasi ini sudah memadai.

5. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan: Simulator modul kendali MOV berhasil dirancang dan diimplementasikan menggunakan Arduino Uno. Simulator mampu mensimulasikan fungsi open, close, stop, dan limit switch dengan akurasi 100%. Simulator efektif sebagai media pembelajaran bagi teknisi baru. Saran untuk pengembangan selanjutnya adalah menambahkan fitur umpan balik posisi katup menggunakan rotary encoder dan integrasi dengan sistem monitoring berbasis DCS.

DAFTAR PUSTAKA

- Arduino.cc. (2024). *Arduino Uno technical specifications*. <https://docs.arduino.cc/hardware/uno-rev3>
- Buwarda, S., Mutmainnah, M., & Lutfi, L. (2024). Rancang bangun mini plant (water level controller) menggunakan PID berbasis LabView dan Arduino Uno. *Jurnal Minfo Polgan*, 13(1), 838–845.
- Haryanto, S. (2018). *Rancang bangun kendali valve untuk pengaturan aliran air berbasis Arduino dan Android* (Skripsi doctoral). Universitas Mercu Buana Jakarta.
- Hudalfah, N. N., Yusuf, A., Tafrikhatin, A., & Sumarah, J. (2023). Media pembelajaran sensor berbasis Arduino Uno untuk pembelajaran mikrokontroler pemula. *Jurnal JASATEC*, 3(1), 15–26. <https://doi.org/10.37339/jasatec.v3i1.1403>
- Hutapea, M., Dalimunthe, M. E., & Saragih, Y. (2024). Rancang bangun trainable robot arm berbasis Arduino Mega 2560 sebagai media praktikum di Laboratorium Teknik Elektro Universitas Pembangunan Panca Budi Medan. *Aisyah Journal of Informatics and Electrical Engineering (AJIEE)*, 6(2), 151–159.
- InstrumentationTools. (2025). *Electric motor actuator parts and components*. <https://instrumentationtools.com/electric-motor-actuator-parts/>
- Lubis, Z., Saputra, L. A., Winata, H. N., Annisa, S., Muhazzir, A., Satria, B., & Wahyuni, M. S. (2019). Kontrol mesin air otomatis berbasis Arduino dengan smartphone. *Bulletin Utama Teknik*, 14(3), 155–159.
- Pasaribu, F. I., & Zulfikar. (2018). Rancang bangun sistem kontrol buka tutup valve pada proses pemanasan air jaket. *Jurnal RELE (Rekayasa Elektrikal dan Energi)*, 1(1), 1–9. <https://doi.org/10.30596/rele.v1i1.2255>
- Prakoso, Y. S., Indrawan, A. W., & Noor, N. A. (2019, September). Rancang bangun automasi drain valve motor gland seal air fan PLTU Barru. In *Seminar Nasional Teknik Elektro dan Informatika* (pp. 407–413). Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.
- Prasetya, T. (2023). *Desain dan implementasi kendali pada proses kalibrasi katup steam venting menggunakan DCS dan Arduino berbasis IoT terpadu* (Disertasi doctoral). Universitas Sangga Buana YPKP.
- Sakti, R. B., Suharso, D. D., & Nugroho, A. A. (2024). Rancang bangun pengendalian katup tiga arah pada sistem pendingin air tawar mesin penggerak utama berbasis mikrokontroler Arduino Uno. *Indonesian Journal of Marine Engineering*, 1(1), 1–11.

- Santoso, H. (2015). *Panduan praktis Arduino untuk pemula*. Penerbit Elang Sakti.
- Santoso, J. (2015). Aplikasi on/off pompa air otomatis berbasis sensor ultrasonik. *Jurnal Teknologi Terpadu*, 8(2), 45–52.
- Sugiyono. (2019). *Metode penelitian kuantitatif, kualitatif, dan R&D* (Edisi ke-2). Alfabeta.
- Yusuf, M., & Hartawan, L. (2022). Penerapan sistem kontrol berbasis Arduino pada manual gate valve ½ inci. *e-Proceeding FTI*.