

# **SISTEM AUTOMAT DE REGLARE AL TEMPERATURII**

**Schifirneț Petre-Justin**

Petroleum-Gas University of Ploiești  
Faculty of Mechanical and Electrical Engineering – Applied Informatics and Automation

## **Introducere**

Acest proiect este o simulare de bază a unui sistem de control pentru un aparat de aer condiționat, dezvoltat folosind pachetul software Rockwell Automation. În special, am lucrat cu Logix Designer pentru programarea logică a unui PLC și FactoryTalk View Studio pentru realizarea interfeței HMI. Deși nu am conectat fizic niciun echipament, proiectul a fost un exercițiu esențial pentru a înțelege nu doar logica de control, ci și structura, tipurile de date și modul de interacțiune între om și mașină. Miza a fost una clară: să învăț să folosesc aceste instrumente în mod corect și organizat, exact ca într-un proiect real.

Am pornit de la o idee simplă: vreau să controlez un aparat de aer condiționat, să îl pot porni/opri, să îi setez o temperatură și să am un sistem care decide când să răcească și când să încălzească. Această simulare a fost punctul de plecare pentru a mă obișnui cu mediile de dezvoltare Rockwell, cu noțiuni precum taguri, funcții CPT (compute), comparatoare logice și design grafic HMI.

## **Descrierea proiectului și logica implementată**

Logica programului a fost implementată în trei secvențe principale de tip *rung* în *ladder logic*. În primul rând, am tratat activarea generală a sistemului. Dacă utilizatorul apasă butonul START și nu este apăsat STOP, atunci sistemul se activează și AC\_Enable devine TRUE. Este o logică de bază, dar esențială pentru orice sistem automat.

A doua parte a logicii verifică dacă trebuie să pornească funcția de răcire. Am comparat temperatura actuală (Temp\_SV) cu limita superioară (Temp\_HLimit) și, dacă aceasta este

depășită, atunci se activează Cool\_ON. Limita superioară nu este fixă, ci este calculată din *setpoint*-ul de temperatură (Temp\_SetPoint) plus o valoare de histereză (Hyst).

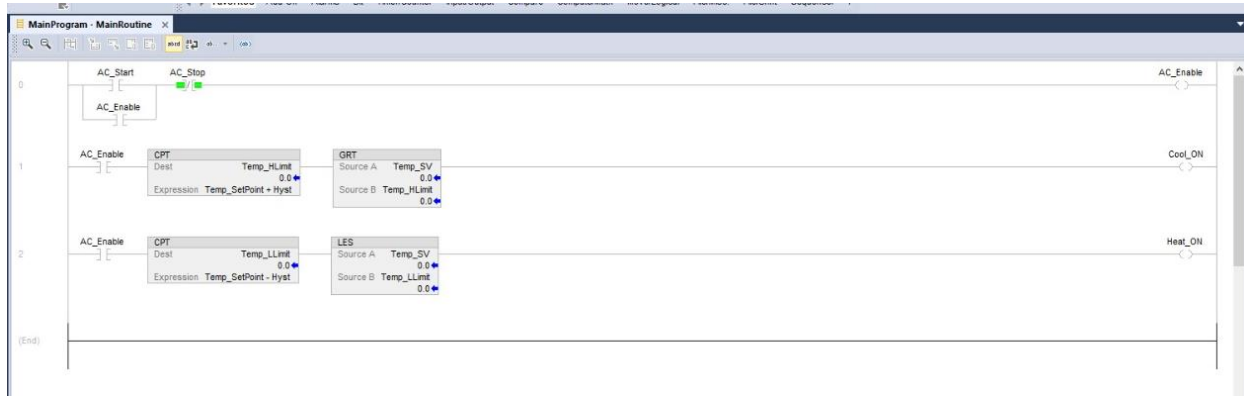


Fig. 1, Schema Ladder a sistemului de reglare al temperaturii

A treia parte este similară, dar pentru încălzire. Dacă temperatura actuală este sub limita inferioară (Temp\_LLimit), atunci Heat\_ON devine activ. Limita inferioară este calculată ca Temp\_SetPoint minus Hyst. Prin acest mecanism se creează o bandă de histereză pentru a preveni porniri și opriri dese ale sistemului.

- Calculul limitelor se face cu blocuri CPT:
- $\text{Temp\_HLimit} = \text{Temp\_SetPoint} + \text{Hyst}$
- $\text{Temp\_LLimit} = \text{Temp\_SetPoint} - \text{Hyst}$

Această abordare este standard în automatizări și previne oscilația excesivă a comenzilor.

## Organizarea tagurilor și tipurile de date

Tagurile utilizate în acest proiect au fost gândite simplu, dar clar. Am folosit:

- Tipul BOOL pentru semnale de comandă (ex: AC\_Start, AC\_Stop, AC\_Enable, Cool\_ON, Heat\_ON)
- Tipul REAL pentru variabilele numerice (ex: Temp\_SV, Temp\_SetPoint, Hyst, Temp\_HLimit, Temp\_LLimit)

Aceste variabile sunt ușor de urmărit și permit scalare ulterioară a proiectului, dacă se dorește adăugarea de alte funcții sau senzori. Vizualizarea acestor taguri în Studio 5000 ajută la urmărirea rapidă a valorilor, mai ales în faza de testare în simulare.

Name	Value	Force Mask	Style	Data Type	Description	Constant
AC_Enable	0		Decimal	BOOL		<input type="checkbox"/>
AC_Start	0		Decimal	BOOL		<input type="checkbox"/>
AC_Stop	0		Decimal	BOOL		<input type="checkbox"/>
Cool_ON	0		Decimal	BOOL		<input type="checkbox"/>
Heat_ON	0		Decimal	BOOL		<input type="checkbox"/>
Hyst	0.0		Float	REAL		<input type="checkbox"/>
Local1:C	{...}	{...}		AB:Embedded_Discre...		<input type="checkbox"/>
Local1:I	{...}	{...}		AB:Embedded_Discre...		<input type="checkbox"/>
Local1:O	{...}	{...}		AB:Embedded_Discre...		<input type="checkbox"/>
Temp_HLimit	0.0		Float	REAL		<input type="checkbox"/>
Temp_LLimit	0.0		Float	REAL		<input type="checkbox"/>
Temp_SetPoint	0.0		Float	REAL		<input type="checkbox"/>
Temp_SV	0.0		Float	REAL		<input type="checkbox"/>

Fig.2, Tagurile utilizate

## Interfața HMI și integrarea cu PLC

Interfața HMI a fost realizată în FactoryTalk View Studio. Aici, am creat o pagină simplă, dar funcțională, care conține:

- Un display numeric mare care afișează temperatura actuală (Temp\_SV)
- Butoane START și STOP colorate sugestiv (verde și roșu)
- Câmpuri de introducere pentru temperatura dorită și histereză

Totul a fost legat la tagurile din PLC prin conexiuni directe. Am folosit tipuri de obiecte momentary push button pentru comenzi și numeric input enable pentru valori. Deși pare simplu, a fost o parte esențială pentru a înțelege cum funcționează schimbul de date între HMI și controller.

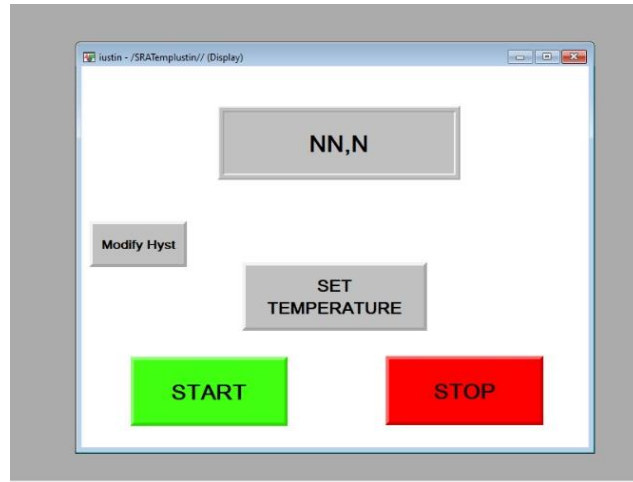


Fig.3, Interfața HMI

### **Ce am învățat din acest proiect**

Acest proiect m-a ajutat să înțeleg fundamentele lucrului cu un sistem de automatizare Rockwell. Nu doar partea de programare, ci și:

- Modul în care se structurează un proiect
- Importanța numirii corecte a tagurilor
- Rolul histerezei în controlul temperaturii
- Legătura dintre partea logică și partea grafică (HMI)

Am învățat să gândesc în termeni de funcționalități clare și să le structurez într-un mod intuitiv. De asemenea, am înțeles cum să testez logica în mod izolat, chiar dacă nu am un echipament fizic.