Exemplu gratuit, cu comentarii: licenta-disertatie-romania.com/domenii/automatica-mecatronica/

UNIVERSITATEA [NUMELE UNIVERSITĂȚII]
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
Programul de studii: Automatică și Informatică Aplicată

LUCRARE DE LICENȚĂ

Sistem de reglare a temperaturii cu regulator PID implementat pe Arduino

Identificare, acordare și validare experimentală pe prototip

EXEMPLU FICTIV – structura, tabelele și figurile unei lucrări de licență. Datele sunt simulate.

Coordonator științific:

Ș.l. dr. ing. Adrian Pârvu (nume fictiv)

Absolvent:

Vlad Munteanu (nume fictiv)

BRAȘOV, 2026

Exemplu fictiv cu date simulate · **Ai nevoie de ajutor cu lucrarea ta?** licenta-disertatie-romania.com/contact

DECLARAȚIE DE AUTENTICITATE

Subsemnatul(a) **Vlad Munteanu**, absolvent(ă) al(a) programului de studii Automatică și Informatică Aplicată, declar pe propria răspundere că lucrarea de licență cu titlul „Sistem de reglare a temperaturii cu regulator PID implementat pe Arduino” a fost elaborată de mine, sub îndrumarea coordonatorului științific, nu a mai fost prezentată la o altă facultate sau instituție de învățământ superior și că toate sursele folosite, inclusiv cele de pe internet, sunt citate în lucrare, cu respectarea regulilor privind drepturile de autor.

Data: [data]

Semnătura: [semnătura]

Majoritatea facultăților au un model oficial de declarație de autenticitate. Copiază-l exact din ghidul facultății și semnează-l; textul de aici este doar orientativ.
Sfat
20/05/2026 10:00

CUPRINS

Cuprinsul arată din prima pagină logica lucrării: introducere, capitole teoretice, cercetarea proprie, concluzii. În Word îl generezi automat din stilurile de titlu (Referințe → Cuprins), ca să se actualizeze singur.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Lista figurilor și a tabelelor.....4
Rezumat.....5
Introducere.....6
Capitolul 1. Stadiul actual și fundamente teoretice.....7
1.1. Metode de reglare a temperaturii.....7
1.2. Regulatorul PID.....7
Capitolul 2. Modelarea procesului și proiectarea regulatorului.....8
2.1. Etapele proiectului.....8
2.2. Identificarea procesului.....8
2.3. Cerințele de proiectare.....9
2.4. Acordarea regulatorului.....9
Capitolul 3. Realizarea prototipului și rezultate experimentale.....11
3.1. Structura hardware.....11
3.2. Implementarea software.....12
3.3. Răspunsul la treapta de referință.....12
3.4. Validarea modelului.....14
3.5. Rejecția perturbației.....15
3.6. Analiza autorului: alegerea configurației.....15
Concluzii.....16
Bibliografie.....17
Anexa 1. Codul sursă: declarații și inițializare.....18
Bonus: același paragraf, înainte și după corectare.....19

Lista figurilor și a tabelelor

Listele de figuri și tabele sunt cerute de multe facultăți. În Word le generezi automat dacă pui titlurile cu Referințe → Inserare legendă, apoi Referințe → Inserare tabel de figuri. **Ghid:** licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Lista figurilor

Figura 1. Etapele proiectului....8
Figura 2. Răspunsul în buclă deschisă la treapta de comandă 0 → 50% PWM....8
Figura 3. Schema bloc a sistemului de reglare automată....11
Figura 4. Ciclul de execuție al buclei de reglare (Ts = 1 s)....12
Figura 5. Răspunsul la treapta de referință 23 → 60 °C pentru cele patru configurații....13
Figura 6. Timpul de creștere și timpul de stabilizare, pe configurații (s)....14
Figura 7. Variația medie a comenzii în regim staționar (%)....14

Lista tabelelor

Tabelul 1. Metode de reglare a temperaturii în incinte mici....7
Tabelul 2. Identificarea modelului cu timp mort prin metoda celor două puncte....9
Tabelul 3. Configurațiile de regulator testate....10
Tabelul 4. Componentele prototipului....11
Tabelul 5. Indicatorii de performanță măsurați la treapta de referință 23 → 60 °C....13
Tabelul 6. Configurația D: simulare cu modelul identificat comparată cu experimentul....15
Tabelul 7. Rejecția perturbației: ventilator pornit la t = 1800 s, în regim staționar la 60 °C…15
Tabelul 8. Verificarea cerințelor de proiectare....15

Rezumat

Lucrarea prezintă proiectarea și realizarea unui sistem de reglare a temperaturii unei incinte mici, cu regulator PID implementat pe o placă Arduino Uno. Procesul termic a fost identificat experimental ca model de ordinul I cu timp mort (K = 0,9 °C/%, τ = 420 s, θ = 30 s). Pe baza modelului au fost acordate și comparate patru configurații: Ziegler–Nichols cu și fără anti-windup, un regulator PI acordat prin metoda SIMC și o acordare proprie, cu anti-windup și filtru pe derivată. Testele pe prototip arată că lipsa mecanismului anti-windup produce un suprareglaj de 73,3%, iar componenta derivativă nefiltrată amplifică zgomotul de cuantizare al senzorului. Configurația propusă atinge referința de 60 °C cu un suprareglaj de 0,1% și un timp de stabilizare de 300 s și este singura care îndeplinește toate cele cinci cerințe de proiectare.

Cuvinte-cheie: regulator PID, reglarea temperaturii, Arduino, identificare, anti-windup

Rezumatul are de obicei 150–300 de cuvinte și spune pe scurt tema, obiectivul, metoda, rezultatele și concluzia. Multe facultăți îl cer și în limba engleză, cu 3–5 cuvinte-cheie.
Rolul secțiunii
20/05/2026 10:00

Abstract

This thesis presents the design and construction of a temperature control system for a small enclosure, using a PID controller implemented on an Arduino Uno board. The thermal process was identified experimentally as a first-order plus dead-time model (K = 0.9 °C/%, τ = 420 s, θ = 30 s). Based on this model, four configurations were tuned and compared: Ziegler–Nichols with and without anti-windup, a PI controller tuned with the SIMC method and a custom tuning with anti-windup and a derivative filter. Tests on the prototype show that without anti-windup the overshoot reaches 73.3%, while an unfiltered derivative term amplifies the sensor quantisation noise. The proposed configuration reaches the 60 °C setpoint with 0.1% overshoot and a settling time of 300 s, and it is the only one that meets all five design requirements.

Keywords: PID controller, temperature control, Arduino, system identification, anti-windup

Introducere

La Automatică introducerea prezintă procesul reglat, obiectivele de control și contribuția proprie. Spune de la început ce ai construit și ce ai măsurat.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Reglarea temperaturii apare în aplicații de la incubatoare și imprimante 3D până la cuptoare industriale. Regulatorul PID rămâne soluția cea mai folosită, datorită structurii simple și a metodelor de acordare bine cunoscute [1].

Obiectivul lucrării este proiectarea și realizarea unui sistem de reglare a temperaturii unei incinte mici, cu regulator PID implementat pe Arduino, care să atingă referința fără suprareglaj semnificativ și să compenseze perturbațiile.

Contribuţiile personale sunt: realizarea prototipului, identificarea experimentală a modelului procesului, compararea a patru configuraţii de regulator, implementarea mecanismului anti-windup și a filtrului pe derivată și propunerea unei acordări care îndeplinește toate cerințele.

Capitolul 1. Stadiul actual și fundamente teoretice

Capitolul teoretic prezintă metodele de reglare și justifică alegerea făcută. Nu reproduce un curs de teoria sistemelor; păstrează doar ce folosești în proiect.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

1.1. Metode de reglare a temperaturii

Tabelul 1 compară metodele folosite pentru incinte mici [2]. Reglarea bipoziţională este cea mai simplă, dar menţine oscilaţii permanente; regulatorul PID oferă un compromis bun între precizie şi complexitate.

Tabelul 1. Metode de reglare a temperaturii în incinte mici

Metoda	Principiul	Avantaje	Dezavantaje
Bipoziţional (on/off)	Releu cu histerezis	Simplu, ieftin	Oscilaţii permanente în jurul referinţei
P	Comandă proporţională cu eroarea	Simplu de acordat	Eroare staţionară
PI	Se adaugă componenta integrală	Elimină eroarea staţionară	Suprareglaj; windup la saturarea comenzii
PID	Se adaugă componenta derivativă	Amortizare mai bună a răspunsului	Amplifică zgomotul de măsură
Fuzzy	Reguli lingvistice	Nu cere un model exact	Acordare empirică; stabilitate greu de demonstrat

Sursa: Sinteză proprie după [1], [2].

1.2. Regulatorul PID

Legea de reglare PID în forma standard este [1]:

$$u(t) = Kp \cdot \left[e(t) + (1/Ti) \cdot \int e(t)\,dt + Td \cdot de(t)/dt \right], \quad e(t) = r(t) - y(t)$$

Două probleme practice apar la implementare. Când comanda este saturată (PWM 100%), componenta integrală continuă să crească şi produce un suprareglaj mare (*windup*). Componenta derivativă amplifică zgomotul de măsură, mai ales când senzorul are rezoluţie finită. Lucrarea tratează ambele probleme în Capitolul 3.

La IEEE sursele se numerotează în ordinea primei citări: [1], [2], [3]. Fişele tehnice ale componentelor (datasheet) sunt surse acceptate şi trebuie citate.
Sfat
20/05/2026 10:00

Capitolul 2. Modelarea procesului și proiectarea regulatorului

Capitolul arată cum ai obținut modelul procesului și cum ai calculat parametrii regulatorului. Fiecare valoare din tabele trebuie să poată fi refăcută de comisie din formule.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

2.1. Etapele proiectului

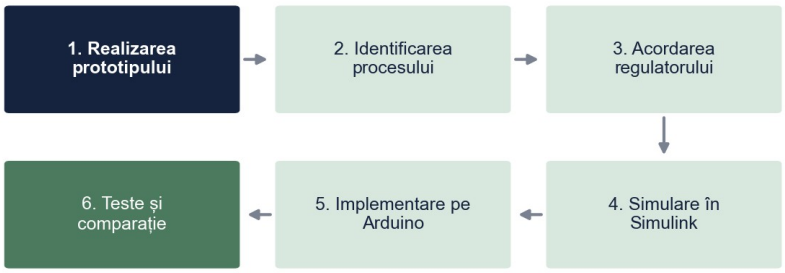


Figura 1. Etapele proiectului
Sursa: Prelucrare proprie.

Proiectul a urmat etapele din Figura 1: modelul a fost identificat pe prototip, regulatoarele au fost acordate și simulate în MATLAB/Simulink, apoi implementate și testate pe Arduino.

2.2. Identificarea procesului

S-a aplicat o treaptă de comandă de 50% în buclă deschisă și s-a înregistrat temperatura (Figura 2). Procesul a fost aproximat cu un model de ordinul I cu timp mort, ai cărui parametri au fost determinați prin metoda celor două puncte (Tabelul 2) [3]:

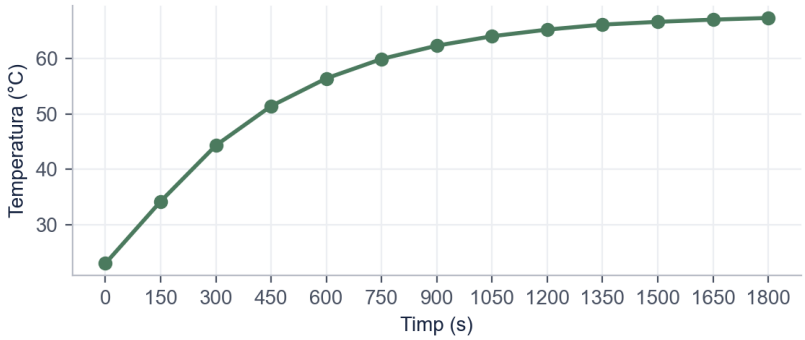


Figura 2. Răspunsul în buclă deschisă la treapta de comandă 0 → 50% PWM
Sursa: Măsurători pe prototip; valori afișate din 150 în 150 s (valori simulate pentru exemplu).

Tabelul 2. Identificarea modelului cu timp mort prin metoda celor două puncte

Mărimea	Valoarea	Cum a fost obținută
Treapta de comandă Δu	50%	PWM de la 0 la 50%, în buclă deschisă
Temperatura inițială / finală	23,0 / 68,0 °C	Regim staționar după 60 de minute
Factorul de amplificare K	0,9 °C/%	K = Δy / Δu = 45 / 50
t_{28} (y = 35,7 °C)	170 s	Momentul atingerii a 28,3% din Δy
t_{63} (y = 51,4 °C)	450 s	Momentul atingerii a 63,2% din Δy
Constanta de timp τ	420 s	$\tau = 1{,}5 \cdot (t_{63} - t_{28})$
Timpul mort θ	30 s	$\theta = t_{63} - \tau$

Sursa: Măsurători pe prototip (valori simulate pentru exemplu).

$$G(s) = K \cdot e^{-\theta s} / (\tau s + 1) = 0{,}9 \cdot e^{-30s} / (420s + 1)$$

Perioada de eșantionare aleasă, Ts = 1 s, este mult mai mică decât timpul mort (30 s) și constanta de timp (420 s) și acoperă timpul de conversie al senzorului (cel mult 750 ms la 12 biți).

2.3. Cerințele de proiectare

- suprareglaj de cel mult 5% la treapta de referință 23 → 60 °C;
- timp de stabilizare în banda ±0,5 °C de cel mult 600 s;
- abatere în regim staționar de cel mult ±0,25 °C;
- variația medie a comenzii de cel mult 5% pe eșantion (fără amplificarea zgomotului);
- abatere de cel mult 1,5 °C la pornirea ventilatorului.

2.4. Acordarea regulatorului

Pentru configurațiile A și B s-au folosit relațiile Ziegler–Nichols pentru curba de răspuns [4]: Kp = 1,2·τ/(K·θ) = 18,7 %/°C, Ti = 2θ = 60 s și Td = 0,5θ = 15 s. Configurația C este un regulator PI acordat prin metoda SIMC cu τc = θ [5]: Kp = τ/(K·(τc + θ)) = 7,8 %/°C și Ti = min(τ, 4(τc + θ)) = 240 s. Configurația D pornește de la valorile Ziegler–Nichols, cu Kp redus la aproximativ jumătate, Ti mărit și filtru pe derivată (Tabelul 3).

Tabelul 3. Configurațiile de regulator testate

Config.	Metoda de acordare	Kp (%/°C)	Ti (s)	Td (s)	Anti-windup	Filtru pe derivată
A	Ziegler–Nichols (curba de răspuns)	18,7	60	15	Nu	Nu
B	Ziegler–Nichols (curba de răspuns)	18,7	60	15	Da	Nu
C	SIMC, regulator PI (τc = θ)	7,8	240	–	Da	–
D	Ajustare proprie (propusă)	9,0	150	12	Da	Tf = 5 s

Sursa: Calcul propriu pe baza modelului din Tabelul 2.

Nu prezenta parametrii regulatorului fără calcul („am pus Kp = 20 pentru că merge bine”). Scrie metoda, formulele și modelul din care provin, apoi arată prin măsurători dacă au fost buni.
Greșeală frecventă
20/05/2026 10:00

Capitolul 3. Realizarea prototipului şi rezultate experimentale

Acesta este capitolul cu contribuția personală: prototipul, codul, măsurătorile și comparația lor. Încheie cu analiza ta: ce configurație alegi și de ce, pe baza cerințelor.
Rolul secțiunii
20/05/2026 10:00

3.1. Structura hardware

Schema bloc a sistemului este prezentată în Figura 3. Arduino-ul citeşte temperatura de la senzorul DS18B20 [6], calculează comanda şi o aplică rezistorului de încălzire prin PWM, cu ajutorul unui MOSFET.

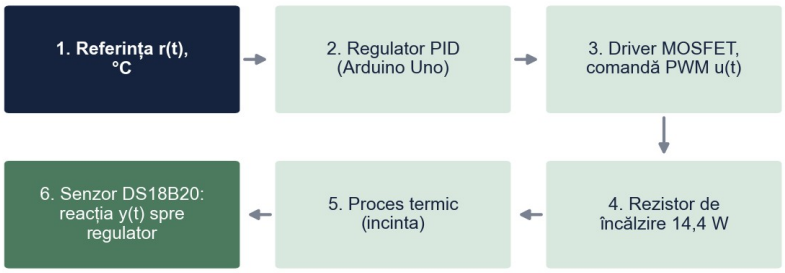


Figura 3. Schema bloc a sistemului de reglare automată
Sursa: Prelucrare proprie.

Componentele şi costul lor sunt prezentate în Tabelul 4. La 12 V, rezistorul de 10 Ω disipă 14,4 W şi absoarbe 1,2 A, sub limita sursei de 2 A.

Tabelul 4. Componentele prototipului

Componenta	Model şi caracteristici	Rolul în sistem	Cost (lei)
Placă de dezvoltare	Arduino Uno R3 (ATmega328P, 16 MHz)	Regulatorul PID, PWM, comunicație serială	60
Senzor	DS18B20, 1-Wire, rezoluție 0,0625 °C	Măsurarea temperaturii	12
Element de execuție	MOSFET IRLZ44N (comandă la nivel logic)	Comanda rezistenței prin PWM	10
Încălzitor	Rezistor de putere 10 Ω / 25 W	14,4 W la 12 V	15
Alimentare	Sursă 12 V / 2 A	Circuitul de putere	35
Interfață	LCD 16×2 cu modul I2C	Afișarea referinței și a temperaturii	25
Incintă	Cutie de aluminiu cu izolație	Procesul termic reglat	30
Perturbație	Ventilator 12 V	Testul de rejecție a perturbației	15
Diverse	Rezistor 4,7 kΩ (pull-up 1-Wire), cabluri, placă de test	–	20
Total			222

Sursa: Prețuri orientative (valori fictive pentru exemplu).

3.2. Implementarea software

Bucla de reglare rulează o dată pe secundă, după ciclul din Figura 4. Fragmentul de mai jos conține partea esențială a algoritmului; codul complet este în Anexa 1.

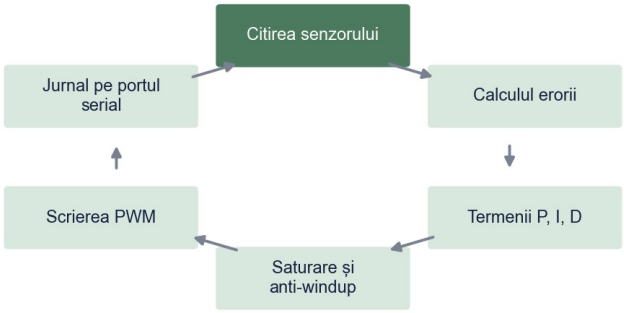


Figura 4. Ciclul de execuție al buclei de reglare (Ts = 1 s)
Sursa: Prelucrare proprie.

```
void loop() {
  if (millis() - tAnt < 1000) return;    // Ts = 1 s
  tAnt += 1000;
  senzor.requestTemperatures();          // max. 750 ms la 12 biti
  float y = senzor.getTempCByIndex(0);
  if (y == DEVICE_DISCONNECTED_C || y > 90.0) {
    analogWrite(PIN_PWM, 0);             // oprire de siguranta
    return;
  }
  float e = ref - y;
  // derivata pe masura (fara salt la schimbarea referintei)
  float dBrut = isnan(yAnt) ? 0 : -Kp * Td * (y - yAnt) / Ts;
  D = a * D + (1 - a) * dBrut;           // filtru de ordinul I
  yAnt = y;
  float uNes = Kp * e + I + D;
  float u = constrain(uNes, 0, 100);
  // anti-windup: integrare conditionata
  if (u == uNes || (uNes > 100 && e < 0) || (uNes < 0 && e > 0))
    I += Kp * Ts / Ti * e;
  analogWrite(PIN_PWM, (int)(u * 2.55)); // 0-100% -> 0-255
  Serial.print(y); Serial.print(';'); Serial.println(u);
}
```

Componenta integrală se actualizează doar când comanda nu este saturată sau când eroarea o scoate din saturație. Derivata se calculează pe mărimea măsurată și este filtrată cu un filtru de ordinul I, cu a = Tf / (Tf + Ts).

Pentru fiecare fragment de cod explică problema pe care o rezolvă (aici: windup și zgomotul derivatei). Comisia întreabă des de ce derivata se calculează pe măsură și nu pe eroare.
Sfat
20/05/2026 10:00

3.3. Răspunsul la treapta de referință

Toate configurațiile au fost testate la aceeași treaptă de referință, de la temperatura camerei (23 °C) la 60 °C (Figura 5, Tabelul 5).

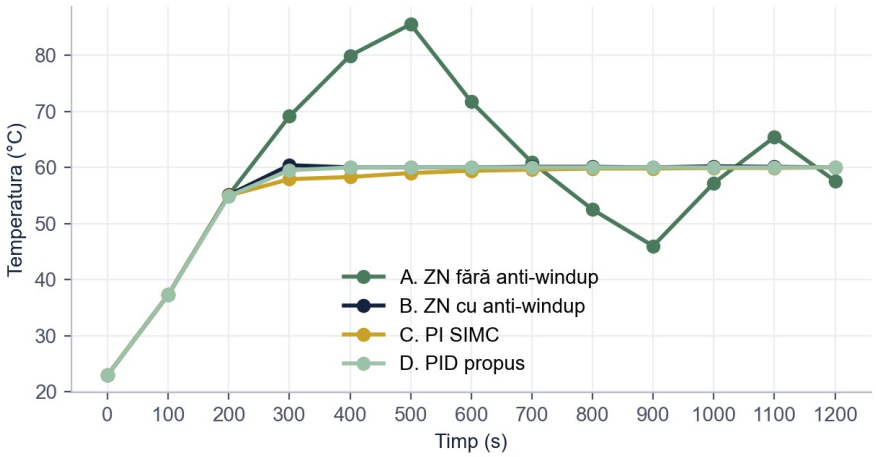


Figura 5. Răspunsul la treapta de referință 23 → 60 °C pentru cele patru configurații
Sursa: Măsurători pe prototip; în figură, valorile sunt afișate din 100 în 100 s (valori simulate pentru exemplu).

Tabelul 5. Indicatorii de performanță măsurați la treapta de referință 23 → 60 °C

Configurația	Suprareglaj (%)	Temp. maximă (°C)	Timp de creștere 10–90% (s)	Timp de stabilizare ±0,5 °C (s)	Variația medie a comenzii (%)
A. ZN fără anti-windup	73,3	87,1	157	1243	36,6
B. ZN cu anti-windup	2,8	61,0	157	302	38,3
C. PI SIMC	0,0	60,0	161	649	0,4
D. PID propus	0,1	60,0	161	300	1,5

Notă. Variația medie a comenzii = media lui $|u(k) - u(k-1)|$ în regim staționar (t = 1500–1800 s), în procente din PWM.
Sursa: Măsurători pe prototip, Ts = 1 s (valori simulate pentru exemplu).

Fără anti-windup (A), temperatura urcă până la 87,1 °C, cu un suprareglaj de 73,3%, și se stabilizează abia după 1243 s. Același regulator cu anti-windup (B) are un suprareglaj de doar 2,8%. Timpul de creștere este aproape același pentru toate configurațiile (157–161 s), pentru că în această fază comanda este saturată la 100%: viteza de încălzire este limitată de puterea rezistorului, nu de regulator (Figura 6).

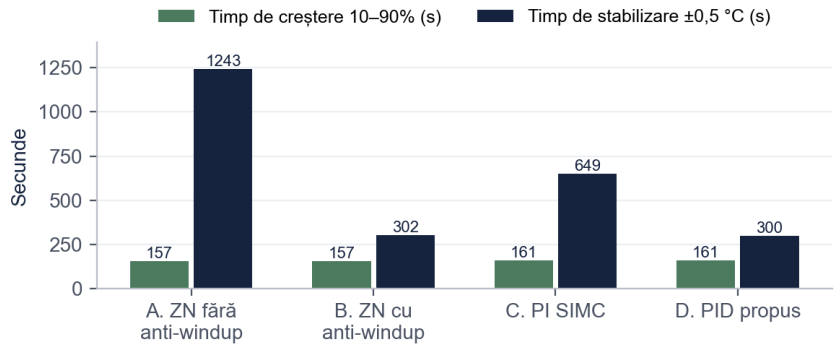


Figura 6. Timpul de creștere și timpul de stabilizare, pe configurații (s)

Sursa: Prelucrare proprie după Tabelul 5.

Configurația B are însă o variație medie a comenzii de 38,3% pe eșantion: un pas de cuantizare al senzorului (0,0625 °C), înmulțit cu Kp·Td = 280, produce salturi de aproximativ 17,5% în PWM. Filtrul pe derivată și valorile mai mici ale lui Kp și Td reduc variația la 1,5% în configurația D (Figura 7).

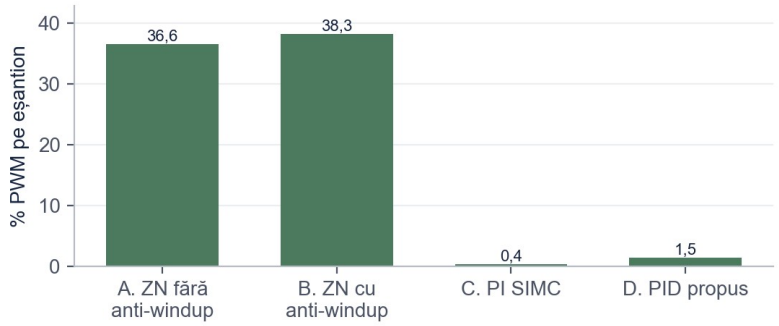


Figura 7. Variația medie a comenzii în regim staționar (%)

Sursa: Prelucrare proprie după Tabelul 5.

Graficele de răspuns se pun cu axele etichetate și cu unități (timp în secunde, temperatură în °C), nu ca o captură de ecran din Serial Plotter.
Greșeală frecventă
20/05/2026 10:00

3.4. Validarea modelului

Pentru configurația D, răspunsul măsurat este puțin mai rapid decât cel simulat, iar comanda în regim staționar este cu 2,1 puncte procentuale mai mică (Tabelul 6). Diferențele arată că amplificarea reală a procesului este ușor mai mare decât cea identificată, dar rămân suficient de mici pentru ca modelul să fie util la acordare.

Tabelul 6. Configurația D: simulare cu modelul identificat comparată cu experimentul

Indicatorul	Simulare (Simulink)	Experiment	Diferența
Suprareglaj (%)	0,6	0,1	−0,5 pp
Timp de creștere 10–90% (s)	186	161	−25 s
Timp de stabilizare ±0,5 °C (s)	314	300	−14 s
Comanda în regim staționar (%)	41,1	39,0	−2,1 pp

Sursa: Simulare în MATLAB/Simulink și măsurători pe prototip (valori simulate pentru exemplu).

3.5. Rejecția perturbației

După stabilizarea la 60 °C a fost pornit ventilatorul, care mărește pierderile de căldură (Tabelul 7). Configurația B revine cel mai repede (65 s), iar configurația D are o abatere maximă de 0,9 °C și revine în bandă în 139 s.

Tabelul 7. Rejecția perturbației: ventilator pornit la t = 1800 s, în regim staționar la 60 °C

Configurația	Abaterea maximă (°C)	Timp de revenire în ±0,5 °C (s)
B. ZN cu anti-windup	0,7	65
C. PI SIMC	1,1	192
D. PID propus	0,9	139

Sursa: Măsurători pe prototip (valori simulate pentru exemplu).

3.6. Analiza autorului: alegerea configurației

Tabelul 8 verifică cele cinci cerințe de proiectare. Configurația B rejectă cel mai bine perturbația, dar comanda ei variază prea mult; configurația C are o comandă foarte liniștită, dar se stabilizează în 649 s. Doar configurația D îndeplinește toate cerințele, de aceea o propunem ca soluție finală, cu prețul unei rejecții a perturbației puțin mai lente decât în cazul B.

Tabelul 8. Verificarea cerințelor de proiectare

Cerința	Valoare impusă	B. ZN + AW	C. PI SIMC	D. PID propus
Suprareglaj	≤ 5%	2,8%	0,0%	0,1%
Timp de stabilizare (±0,5 °C)	≤ 600 s	302 s	649 s ✗	300 s
Abatere în regim staționar	≤ ±0,25 °C	±0,24 °C	±0,03 °C	±0,03 °C
Variația medie a comenzii	≤ 5%	38,3% ✗	0,4%	1,5%
Abatere la perturbație	≤ 1,5 °C	0,7 °C	1,1 °C	0,9 °C
Cerințe îndeplinite	–	4 din 5	4 din 5	5 din 5

Notă. ✗ = cerință neîndeplinită. Configurația A nu este inclusă: suprareglajul de 73,3% depășește cu mult limita.
Sursa: Prelucrare proprie după Tabelele 5 și 7.

Concluzii

Concluziile evaluează performanțele obținute față de cerințe, spun limitele sistemului și propun îmbunătățiri. Nu repeta toate valorile din tabele, doar pe cele importante.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

- Modelul de ordinul I cu timp mort ($K = 0{,}9$ °C/%, $\tau = 420$ s, $\theta = 30$ s) descrie suficient de bine procesul pentru acordarea regulatorului.
- Mecanismul anti-windup este esențial: fără el, suprareglajul ajunge la 73,3%.
- Configurația propusă (D) îndeplinește toate cerințele: suprareglaj 0,1%, timp de stabilizare 300 s și o comandă stabilă în regim staționar.

Limite și dezvoltări ulterioare: sistemul poate doar să încălzească, deci nu poate coborî activ temperatura sub cea a mediului. Ca dezvoltări propunem răcirea activă cu o celulă Peltier, acordarea automată (autotuning prin metoda releului) și o interfață pentru modificarea referinței de pe calculator.

Bibliografie

La IEEE sursele sunt numerotate în ordinea citării, nu alfabetic. Lucrările [1], [4], [5] și fișa tehnică [6] sunt reale; [2] și [3] sunt surse generice, inventate pentru acest exemplu.
Rolul secțiunii
20/05/2026 10:00

[1] K. J. Åström și T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, ed. a 2-a. Research Triangle Park, NC, SUA: ISA, 1995.

[2] A. Dobre și C. Lupu, „Reglarea temperaturii în incinte mici cu regulatoare PID discrete”, *Revista de Automatică Aplicată*, vol. 7, nr. 1, pp. 12–20, 2022.

[3] I. Marin, *Identificarea proceselor și acordarea regulatoarelor. Aplicații cu microcontrolere*. Brașov: Editura Exemplu, 2021.

[4] J. G. Ziegler și N. B. Nichols, „Optimum settings for automatic controllers”, *Transactions of the ASME*, vol. 64, pp. 759–768, 1942.

[5] S. Skogestad, „Simple analytic rules for model reduction and PID controller tuning”, *Journal of Process Control*, vol. 13, nr. 4, pp. 291–309, 2003.

[6] Maxim Integrated, „DS18B20 Programmable Resolution 1-Wire Digital Thermometer”, fișă tehnică. [Online]. Disponibil: https://www.analog.com. [Accesat: 12 mai 2026].

Anexa 1. Codul sursă: declarații și inițializare

În anexe pui codul complet, schema electrică și lista conexiunilor. În text rămân doar fragmentele explicate. Verifică dacă facultatea cere codul pe un suport separat sau într-un repository.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

```
#include <OneWire.h>
#include <DallasTemperature.h>

const byte PIN_PWM = 9, PIN_1WIRE = 2;
const float Kp = 9.0, Ti = 150.0, Td = 12.0;  // configuratia D
const float Ts = 1.0, Tf = 5.0, a = Tf / (Tf + Ts);
float ref = 60.0, I = 0, D = 0, yAnt = NAN;
unsigned long tAnt;
OneWire ow(PIN_1WIRE);
DallasTemperature senzor(&ow);

void setup() {
  pinMode(PIN_PWM, OUTPUT);
  Serial.begin(115200);
  senzor.begin();
  senzor.setResolution(12);   // 0,0625 grade C
  tAnt = millis();
}
```

Conexiuni: DS18B20 pe pinul digital 2, cu rezistor de 4,7 kΩ între linia de date și 5 V; poarta MOSFET-ului pe pinul 9 (PWM); LCD-ul pe magistrala I2C (A4, A5). Codul complet: https://github.com/exemplu/pid-temperatura (adresă fictivă).

Bonus: același paragraf, înainte și după corectare

Această pagină nu face parte din lucrare. Arată cum se schimbă un paragraf la o corectare atentă: aceeași idee, scrisă clar, cu sursă și în stil academic.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Fragmentul de mai jos este scris intenționat cu greșelile pe care le întâlnim des în proiectele de licență la Automatică. Dedesubt este aceeași idee după corectare.

VARIANTA INIȚIALĂ

> Am folosit un PID si am pus Kp=20 pentru ca asa merge bine. Temperatura ajunge repede la valoare si sistemul e foarte stabil, cum se vede in poza de mai jos facuta din Serial Plotter. PID-ul este cel mai bun regulator.

VARIANTA CORECTATĂ

> Parametrii regulatorului au fost calculați cu relațiile Ziegler–Nichols pentru curba de răspuns [4], pe baza modelului identificat (Tabelul 2), apoi ajustați experimental (configurația D, Tabelul 3). La treapta de referință 23 → 60 °C, sistemul atinge un suprareglaj de 0,1% și se stabilizează în banda ±0,5 °C după 300 s (Figura 5). Regulatorul PID a fost ales pentru că elimină eroarea staționară și are o implementare simplă pe microcontroler.

CE AM CORECTAT

1. Am pus diacriticele (ș, ț cu virgulă), obligatorii în lucrare.
2. „Am pus Kp = 20 pentru că așa merge” a devenit o acordare justificată: metoda, modelul și sursa citată în stil IEEE.
3. „Ajunge repede” și „foarte stabil” au devenit indicatori măsurați, cu unități: suprareglaj, timp de stabilizare.
4. Captura din Serial Plotter a fost înlocuită cu o figură numerotată, cu axe etichetate.
5. „Cel mai bun regulator” a devenit o justificare a alegerii, potrivită pentru acest proces.

Vrei aceeași corectare pentru lucrarea ta? licenta-disertatie-romania.com/contact