

AUTOMATIZAREA UNUI DESHIDRATOR CU ALIMENTARE HIBRIDĂ

Lucian GRUȘAC, Svetlana BRANDUȘ

Universitatea Tehnică a Moldovei, Facultatea Energetică și Inginerie Electrică, Departamentul Inginerie Electrică, grupa IE-20M, IMC-20M, Chișinău, Republica Moldova

Rezumat: În lucrarea prezentată se expune materialul referitor la automatizarea unui deshidrator hibrid, explicarea procesului de lucru a acestuia. În continuare este descrisă schema bloc de comandă și schema electrică ale deshidratorului. Problema care am discutat-o este eficiența energetică și sistemul de control inclusiv și de comandă a deshidratorului. Am decis că adăugarea unui microcontroler, în cazul nostru o placă arduino R3 ne va ușura mult controlul și comanda asupra deshidratorului.

Cuvinte cheie: deshidrator, arduino, transfer de căldură, temperatură, automatizare, soft, limbaj de programare.

Introducere

Automatizarea deshidratoarelor constă din controlul și reglarea parametrilor de climă în interiorul camerei de uscare – temperatura, debitul aerului și umiditatea. Acești factori influențează calitatea de uscare.

Calitatea produsului uscat depinde de mai mulți factori, depinde atât de producătorul deshidratorului cât și de persoana care conduce procesul în momentul deshidratării unui produs. Producătorul își asumă răspunderea să confecționeze deshidratorul din materiale alese corect, trebuie să aleagă metale care la încălzire să nu elimine oxizi sau alte elemente otrăvitoare, să facă calculele corecte pentru curenții de aer să nu fie mari sau mici. În momentul când pornești un proces de deshidratare trebuie să respecti mai multe cerințe, cum ar fi alegerea corectă a condițiilor optime de generare a temperaturii. Eroarea admisibilă este de 2°C, indiferent de etapa de deshidratare în care se află. Unii, din cauza diferenței mai mari de temperatură pe toate tăvile, încearcă să furnizeze un debit de aer prea intens, ceea ce nu este corect. Prin aceasta comit o greșală, deoarece produsele devin mai mult uscate decât deshidratate. Pentru o deshidratare corectă a produselor, nici într-un caz nu trebuie grăbite. Cel mai important parametru pentru calitatea uscării este temperatura. Controlul temperaturii va fi realizat de un termostat, care diferă cel mai mult de un uscător de fructe și legume simplu. Acest termostat ne asigură realizarea clară a temperaturii dorite. Majoritatea magazinelor de electrocasnice la moment vând modele de uscătoare simple. Acest termostat ne permite nu doar stabilirea temperaturii exacte într-un deshidrator, dar și reglarea acesteia în dependență de umiditatea pe care o avem în interior. Temperatura și umiditatea din interior se măsoară cu senzori special instalați. Aceasta este nevoie deoarece fructele și legumele de aceeași specie pe care le uscăm nu au întotdeauna aceeași umiditate, pot fi mai verzi și mai coapte sau păstrate diferit. Totodată uscarea produselor la temperaturi de peste 38°C mărește riscul și viteza distrugerii substanțelor nutritive.

Metode de control/reglare PID, PI, P, control în histereză. Am ales să controlăm cu Arduino Uno, senzori de temperatură și umiditate iar controlul va fi efectuat prin histereză. Această metodă permite controlul pe baza diferenței de temperatură dintre temperatura de intrare în ciclu și temperatura de revenire la stadiu inițial. De exemplu dacă vom seta temperatura de 40°C și valoarea de histerezis va fi setată de 3°C, atunci elementul termic va fi deconectat la 37°C. Datorită inerției temperatura automat se va ridica până la aproximativ 40°C cu eroare de maxim 3°C. Controlerul ne asigură reglarea pragului de histerezis între 1°C și 10°C.

1. Sistemul de control al deshidratorului hibrid automatizat

Microcontrolerul **ATMega** dirijează senzorii, releul pentru alimentarea elementului termic și driverul pentru alimentarea ventilatorului și display-ul **OLED I2C**.

Elementul **switch1** are destinația de a oferi plăcii semnal de start în automat, butonul **switch** ne oferă posibilitatea de a ne folosi de mașină manual.

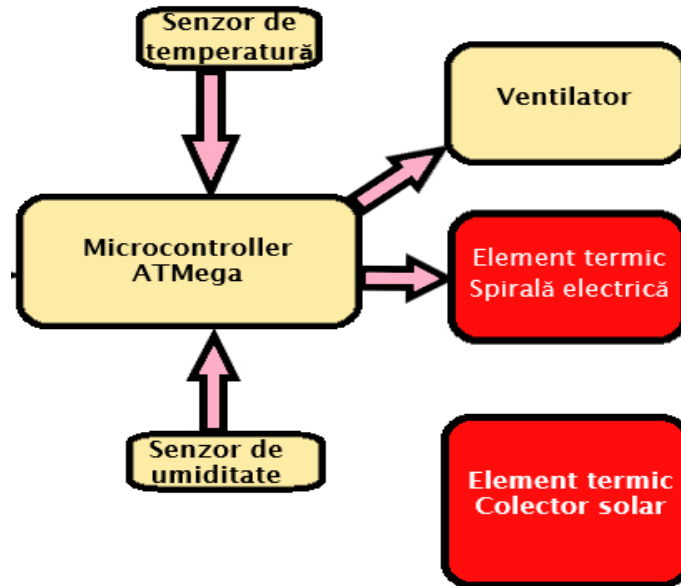


Figura 1. Schema bloc de comandă a deshidratorului hibrid automatizat

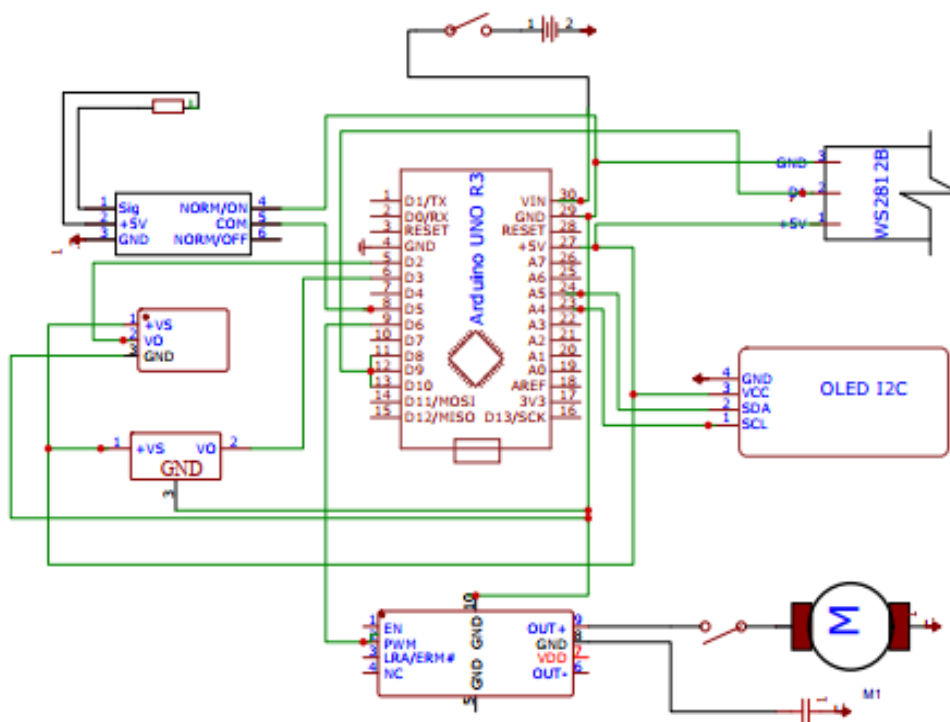


Figura 2. Schema electrică a unui Deshidrator Hibrid Automatizat

2. Limbajul de programare al controlerului arduino R3

Programele pentru Arduino se scriu în softul IDE Arduino în care limbajul de programare este asemănător cu limbajul de programare C++. Softul IDE Arduino îl putem descărca de pe site-ul oficial <http://www.arduino.cc/en/Main/Software>, care lucrează pe baza Windows, Mac OS sau Linux. Placa electronică Arduino are propriul procesor și memoria pe care noi instalăm programul cu ajutorul softului menționat. Placa poate realiza o mulțime de comenzi datorită zecilor de perechi de contacte pe care le conține. Începând cu conectarea unor led-uri simple până la conectarea și deconectarea diferitor lacăți magnetice, motoare, senzori ș.a. Prin urmare, înțelegem că acest microcontroler ne oferă posibilitatea să efectuăm o mulțime de operațiuni, experimente, proiecte și o gamă largă de gadget-uri.

Softul Arduino IDE este cel mai simplu. Sunt și alte softuri compatibile cu Arduino, dar ele sunt mai complicate, ca de exemplu: Eclipse, Visual Studio și necesită cunoașterea altor limbaje de programare ca: CSS, Java, HTML. Acest soft necesită conectarea plăcii Arduino la PC prin intermediul unui USB, alegerea portului și înscrierea programului printr-o metodă simplă. Softul conține multe biblioteci care le putem înscrie în codul programatorului sau microcontrolerului.

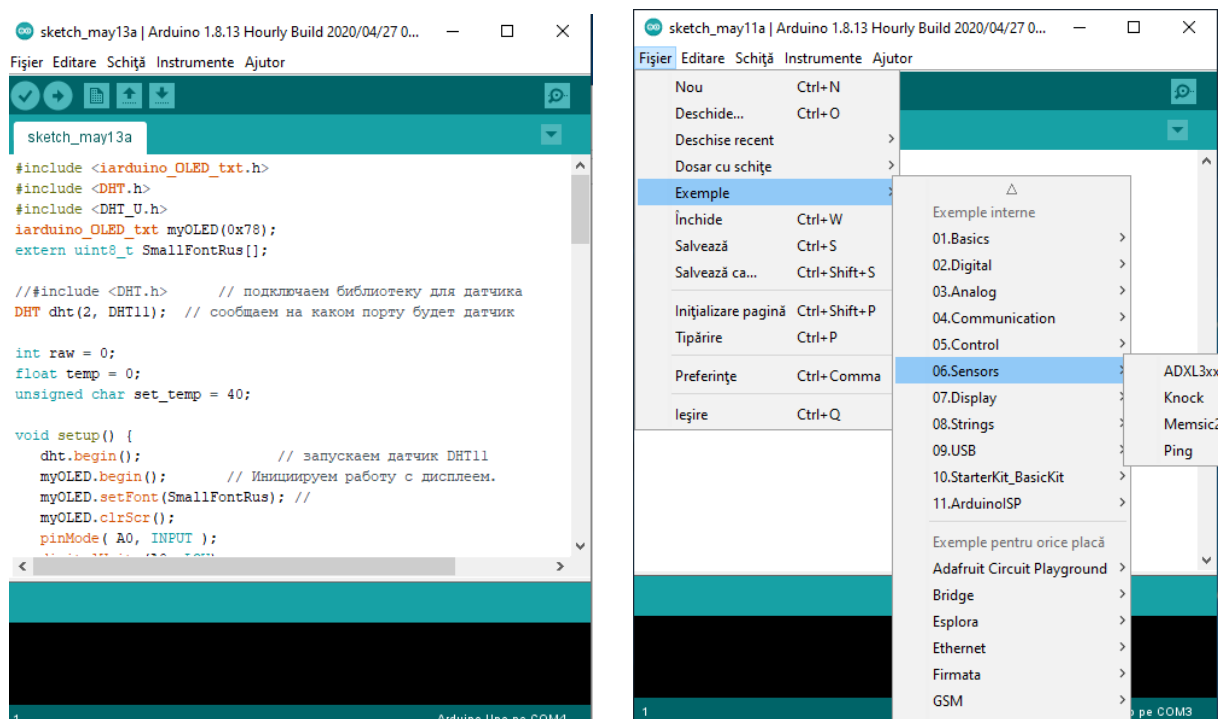


Figura 3 a) Interfața Softului Arduino IDE; b) Bibliotecile softului Arduino ID

Programul este împărțit în 3 componente.

Includerea bibliotecilor

În prima parte deschidem programul și începem cu adăugarea bibliotecilor de care vom avea nevoie pe parcursul scrierii programului nostru, apoi conectăm biblioteca `//#include <DHT.h>`, care este destinată lucrului cu senzorul și anunțăm la ce pin vom conecta senzorul nostru pentru ca mai târziu să-l putem conecta să cunoască microcontrolerul de unde să citească datele.

Inițializarea sistemului Arduino

În a doua parte este `void setup` care este configurarea sau pregătirea pentru a începe lucru cu programul. Pentru a putea da o oricare comandă aici trebuie să notăm fiecare element de care ne vom folosi, la ce pin individul va fi conectat. Deci am notat pornirea senzorului, pregătirea

display-ului pentru lucru și am notat pinii la care vor fi conectate părțile componente (releul, tranzistorul ventilatorului, butoanele cu ajutorul cărora vom regla temperatura manual).

În *void loop()* este scris însuși programul de execuție automatizată. Aici sunt următoarele funcții ale programului: citirea și compararea cu referință a temperaturii și a umidității, înscrierea pe LCD, controlul elementului termoelectric, controlul ventilatorului și controlul diodelor electro luminescente și închiderea ciclului loop sau închiderea programului.

După ce cunoaștem temperatura și umiditatea noi deja putem face careva concluzii că avem nevoie de o temperatură mai mare sau mai mică. Deci cunoaștem ce să întreprindem mai departe. Pentru a mări sau micșora temperatura avem conectate la controler 2 butoane. Cu ajutorul acestor butoane putem stabili o temperatură exactă de care avem nevoie pentru uscarea produsului. În cazul când temperatura primită de la colectorul solar nu este suficient de mare controlerul automat va conecta elementul termoelectric care este o spirală electrică conectată printr-un releu de comandă la controlerul Arduin până când va fi atinsă temperatura normală, adică pe care am dat-o noi cu ajutorul butoanelor sau la cea montată inițial în programă, care este de 40°C. Pentru uscarea corectă avem nevoie de mici curenți de aer, pentru aceasta este instalat un ventilator care are legătura cu Arduino făcut printr-un driver, care se va conecta la temperatura de 20°C și va funcționa permanent, atât timp cât temperatura este depășită. Când temperatura va scădea mai jos de 20°C ventilatorul automat se va deconecta. Pentru a putea observa dacă deshidratorul lucrează și dacă temperatura este cea pe care am dat-o noi, am conectat 3 led-uri RGB care se vor conecta în dependență de temperatură. Până la 25 °C va fi conectat led-ul de culoare albastră, de la 25 °C până la temperatura pe care am dat-o noi manual sau care este deja dată inițial de 40 °C va fi conectat led-ul de culoare verde, iar în caz de depășire a temperaturii date de noi se va conecta culoarea roșie, care este un semnal de alarmă pentru ca să nu coacem alimentele.

Concluzii

- Un sistem automatizat de reglare a elementului termic, de măsurare a temperaturii, umidității și de monitorizare a procesului de deshidratare.
- Eroarea de abatere a temperaturii de 3°C din motivul inerției și care nu este o eroare semnificativă.
- Mărirea eficienței energetice maxim posibilă pe timp de vară datorită colectorului.
- S-au exclus operațiile adăugătoare de control manual a producției și de control a temperaturii pe suprafețele tavelor care deja se verifică automat.

Referințe

Referințe Web:

1. https://www.researchgate.net/figure/ATmega48-88-168-Microcontroller_fig3_283732939
2. <https://www.researchgate.net/figure/Block-Diagram-of-ATmega16fig6303647719>

Teze:

3. GRUȘAC L. *Deshidrotor hibrid automatizat pe bază de arduino în Republica Moldova: teză de licență*. Chișinău: UTM, 2020.