**Lucrarea 12.**

Controlul unor procese complexe printr-o platforma de tip Arduino

**1. Obiectivul lucrarii**

Lucrarea isi propune sa prezinte modul de implementare a unor aplicatii complexe controlate prin intermediul unei placi Arduino:

* o instalatie industriala
* o casa inteligenta (realizata in lucrarea urmatoare)

In ambele cazuri se folosesc machete ale proceselor fizice reale.

**2. Consideratii teoretice**

O aplicatie de monitorizare si control este destinata pentru urmarirea si controlul automat al unui anumit proces, cum ar fi o instalatie industriala, o linie de fabricatie, o retea de distributie (energie electrica, gaze, etc.), un echipament complex (ex. robot, utilaj cu comanda numericǎ, etc.) sau o cladire. O astfel de aplicatie integreaza mai multe categorii de componente de automatizare cum ar fi:

a.**Senzori sau traductoare** – menite sa masoare starea/valoarea anumitor parametrii de proces cum ar fi:

- temperatura, presiune, umiditate

- pozitie, orientare, nivel, proximitate, viteza, acceleratie,

- curent, tensiune, putere, factor de putere

- compozitie chimica, PH, vascozitate

In functie de natura semnalelor generate senzorii pot fi de mai multe tipuri:

- senzori analogici – genereaza o tensiune sau un curent proportional cu valoarea parametrului masurat; plaja de variatie:0-5V, 010V, 4-20mA, etc.

- senzori digitali – genereaza semnale digitale simple (0 sau 1) sau complexe: modulate in frecventa de impulsuri, latime de impuls (PWM), mesaj in retea, etc.

b. **Elemente de executie** – care au rolul de a actiona asupra procesului controlat cu scopul de a modifica anumiti parametrii de proces si implicit evolutia procesului; exemple de elemente de executie (actionare):

- motoare electrice pas-cu-pas, de curent continuu (cc) si de curent alternativ (ca)

- electromagneti, motoare liniare, servo-motoare

- relee

- elemente de incalzire (ex. rezistente de incalzire)

- robinete, valve, pompe, etc.

c. **Elemente de control** – destinate pentru generarea controlului pe baza unei anumite legi prestabilite (legea de reglaj automat); astfel de elemente pot fi:

- microcontroloare, procesoare de semnal, platforme cu astfel de circuite (ex. PICDEM 4, EZ430)

- regulatoare, controloare logice programabile (PLC)

- platforme universale de monitorizare si control (ex. Arduino, Raspberry pi, Edison, WeMOS, etc.)

d. **Elemente de comunicatie** – destinate pentru interconectarea componentelor de mai sus, cum ar fi:

- extensii de retea: BlueTooth, WiFi, ZeegBee,

- interfete pentru retele industriale: CAN, ProfiNet, EtherCat

Programul de aplicatie, inscris de obicei intr-o memorie nevolatila a elementului de control, implementeaza urmatoarele functii de baza:

* culege date despre proces prin intermediul senzorilor
* proceseaza datele achizitionate (ex. prin proceduri de filtrare, agregare, calculul comenzii urmatoare, etc.)
* genereaza comenzi in scopul controlarii procesului; comenzile sunt transmise catre elementele de executie
* comunica cu un operator uman printr-o interfata de tip “panou frontal” ce contine: butoane, afisaj cu leduri, cu 7 segmente, afisaj alfanumeric, etc.)
* optional comunica cu un calculator ierarhic superior de obicei printr-o interfata seriala sau printr-o retea (ex. wifi, bluetooth sau chiar Internet)

Primele trei functii (achizitie, procesare si generare de comenzi) se executa intr-o ordine secventiala cu o anumita periodicitate Prioada de repetitie este data de dinamica procesului controlat, adica de viteza de variatie a parametrilor urmariti. Conform teoremei lui Shanon din teoria semnalelor frecventa de repetitie sau de esantionare trebuie sa fie mai mare decat dublul frecventei maxime prezente in semnalul achizitionat. Pentru o calitate mai buna a reglajului se alege de obicei o frecventa de 5-10 ori mai mare decat frecventa maxima a semnalelor de intrare.

Functiile de reglaj (mentionate anterior) vor fi executate in regim concurent cu celelalte functii ramase (comunicarea cu operatorul uman si cu un alt calculator). Executia in regim concurent poate ridica anumite probleme de implementare mai ales in cazul sistemelor uni-procesor si care nu dispun de un sistem de operare care sa ofere mecanisme de tip multi-tasking sau multi-fir. Cum se va vedea in exemplele din lucrare, acest regim de lucru concurent trebuie sa se implementeze in aplicatia propriu-zisa de control si sunt necesare masuri speciale care sa nu blocheze procesorul pe o anumita functie in detrimentul celorlalte functii. De exemplu de durata cat operatorul uman sau un alt calculator comunica cu prcesorul functiile de monitorizare si reglaj nu trebuie sa fie intrerupte sau sa se modifice periodicitatea acestora.

In continuare vor fi descrise doua aplicatii de monitorizare si control, relevante pentru domeniul industrial si respectiv cel al cladirilor inteligente.

1. **Sistem de monitorizare a unui proces indistrial**

In scopul demonstrarii facilitatilor de achizitie, procesare, control si comunicare a unei platforme universale de tip Intel Arduino Galileo s-a realizat o macheta a unei linii de dozare. In figura 1 se pot observa principalele subsisteme ale procesului:

* Sistemul de alimentare cu cutii mici si mari
* Sistemul (masa) de deplasare a cutiilor
* Statia de dozare
* Sistemul de sortare (debarasare)
* Sistemul optic pentru detectia pozitiei cutiilor
* Sistemul de detectie a pozitiei initiale a mesei
* Modulul de control format din placa Arduino Galileo si Placa de extensie (shield)
* Sursa externa de alimentare



Figura 1. Macheta unei linii industriale de dozare

Cele 6 etape ale procesului de fabricație sunt:

**- calibrarea poziției mesei de deplasare**

Această etapă este necesară la începutul unui proces întrucât masa dedeplasare a cutiior trebuie să se afle sub sistemul de încărcare. Din diverse motive, cum ar fi o pană de curent aceasta ar putea să nu fie în starea potrivită. Astfel, această etapă verifică dacă masa de deplasare e în poziția corespunzătoare sau nu , în continuare acționându-se în consecință.

- **etapa de încărcare**

Aceata etapă presupune punerea pe masa de deplasar a unui recipient (cutie în cadrul machetei de față).

- **etapa de detecție**

În cadrul acestei etape se detectează dacă pe masa de deplasare se află un recipient care urmează a fi încărcat cu bile. În cazul în care se remarcă absenţa acestuia masa se întoarce la etapa de încărcare, în caz contrar se trece la etapa următoare.

- **etapa de deplasare**

Presupune deplasarea mesei împreună cu cutia până la etapa de dozare.

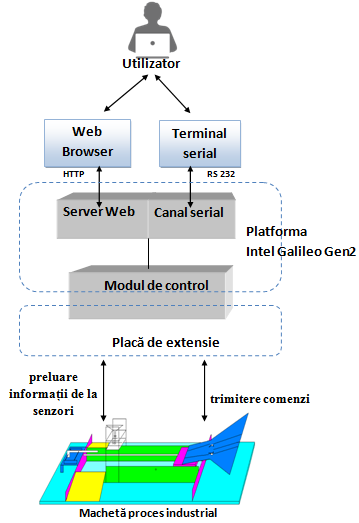
- **etapa de dozare**

În această etapă se realizează umplerea cu un anumit număr de bile (specificat de utilizator) a cutiei de pe masa de deplasare.

- **etapa de sortare**

Această etapă presupune sortarea celor două tipuri de recipiente : cutii mici şi mari. Astfel, cutiile mici sunt deplasate în partea dreaptă iar celemari în cealaltă parte.

Figura urmatoare prezinta modul de operare al sistemului. Se observa ca sistemul poate fi operat (control si monitorizare) atat local printr-un PC atasat pe un canal serial (implementat pe USB) cat si de pe Internet, printr-o interfata web.



Elementele de automatizare utilizate in cadrul sistemului au fost:

* + Motorul de antrenare a mesei – motor pas-cu-pas unipolar
  + Motorul de sortare – motor pas-cu-pas bipolar
  + Cate o pereche de electromagneti (stanga-dreapta) pentru alimentarea cu cutii mari si mici
  + Senzor pentru calibrarea pozitiei mesei – senzor mecanic bipozitional
  + Sistem optic pentru detectia pozitiei cutiilor pe masa – LED +fototranzistor
  + Electromagbet de dozare a bilelor – electromagnet bipozitional (normal destins)

Interfata dintre placa Arduino si elementele de automatizare mentionate mai sus s-a realizat pe o placa de extensie cuplata direct pe conectorii standard ai placii Arduino. Aceasta interfata contine urmatoarele circuite de adaptare:

* + Driver pentru motor pas-cu-pas bipolar – L293D
  + Driver pentru motor pas-cu-pas unipolar si controlul releelor – ULN2803A
  + 4 relee pentru comanda eliberarii cutiilor mari si mici
  + 1 releu + un tranzistor pentru comanda dozatorului de bile
  + Circuit invertor – pentru implementarea logicii de control
  + Rezistente de adaptare pentru senzorul optic si mecanic

Figura de mai jos indica modul de amplasare a conectorilor si destinatia acestora.

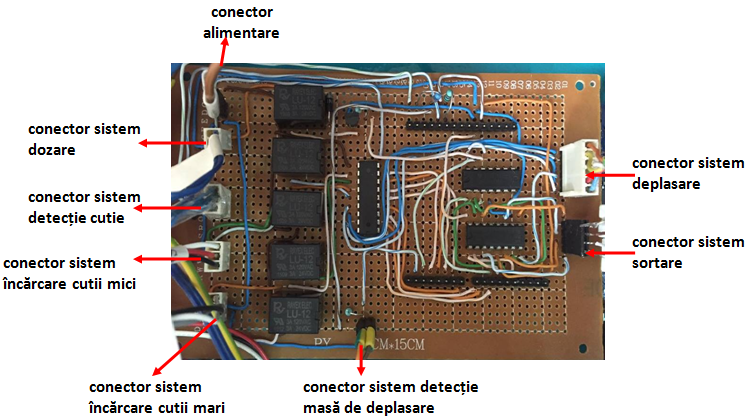


Figura 3 Placa de interfata

In tabelul de mai jos se prezinta modul de alocare a semnalelor placii Arduino

Tabel 1

|  |  |  |
| --- | --- | --- |
| Nume semnal | Tip | Destinatie |
| IO0 | nedefinit | Liber |
| IO1 | Intrare digitala | Senzor calibrare pozitie masa |
| IO2 | Iesire digitala | Releu dozator |
| IO3 | Iesire digitala | Selectie electromagnet stanga-dreapta cutii mari ? |
| IO4 | Iesire digitala | Alimentare releu cutii mari |
| IO5 | Iesire digitala | Selectie electromagnet stanga-dreapta cutii mici ? |
| IO6 | Iesire digitala | Alimentare releu cutii mici |
| IO7 | Iesire digitala | Motor masa Faza 1 |
| IO8 | Iesire digitala | Motor masa Faza 2 |
| IO9 | Iesire digitala | Motor masa Faza 3 |
| IO10 | Iesire digitala | Motor masa Faza 4 |
| IO11 | Iesire digitala | Motor sortare Faza 1 |
| IO12 | Iesire digitala | Motor sortare Faza 2 |
| IO13 | Iesire digitala | Validare motor sortare |
| GND | masa |  |
| +5V | alimentare |  |
| A0 | Intrare analogica | Senzor optic pentru detectia pozitie cutiilor |

Pentru a asigura o executie concurenta a functiilor implementate de aplicatie se impune mixarea pasilor care alcatuiesc aceste functii. De exemplu deplasarea mesei se descompune in mai multi pasi care pot fi mixati cu pasii motorului de sortare si respectiv cu pasii de deservire a unei cereri utilizator (venita fie pe canalul serial fie pe web). Figura 4 sugereaza acest mod de functionare.

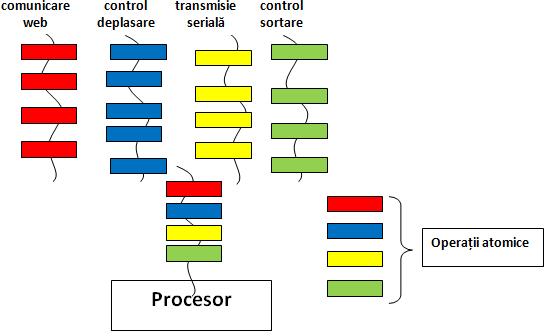


Figura 4 Executie in regim concurent

1. Monitorizarea si controlul unei case inteligente
2. Modul de desfasurare a lucrarii

* Se vor analiza componentele utilizate si schema electrica generala (anexa 1)
* Utilizandu-se mediul de programare Arduino Galileo IDE se vor scrie proceduri pentru operarea principalelor componente ale sistemului:
  + Selectie cutii
  + Deplasare masa (inclusiv cu detectie pozitie initiala)
  + Sortare,
  + Dozare bile
  + Detectie cutie (sistem optic)
  + Dispecerizarea comenzilor primite de la utilizator si vizualizarea pasilor executati
* Se va scrie un program care lucreaza in regim concurent
* Se va scrie un program ce permite controlul de la distanta prin Internet (aplicatie web-server)

Anexa 1 Schema Interfetei pentru procesul industrial

