

Capitolul 5: Comanda motoarelor cu microcontrollere

Foarte multe aplicații din domeniul microcontrollerelor au sarcina acționării unor motoare. De exemplu aplicațiile auto-motoarele de la geamurile electrice, motoarele indicatoarelor de bord etc. sau aplicațiile din domeniul echipamentelor periferice - imprimantă, hard disc etc. Sau aplicațiile industriale - acționări de vane, robinete etc. Motoarele pot fi:

- de curent continuu cu perii sau fără perii;
- motoare pas cu pas;
- de curent alternativ cu inducție;
- motoare cu reluctanță comutată.

Primele două categorii de motoare sunt utilizate mai des în aplicațiile cu microcontroller de aceea vor fi abordate în acest capitol. Un motor poate fi comandat de către un microcontroller atât direct, cât și prin intermediul unor interfețe specializate, programabile.

5.1. Comanda motoarelor de curent continuu

Schema bloc a sistemului de acționare cu motor de curent continuu este dată în figura 5.1.

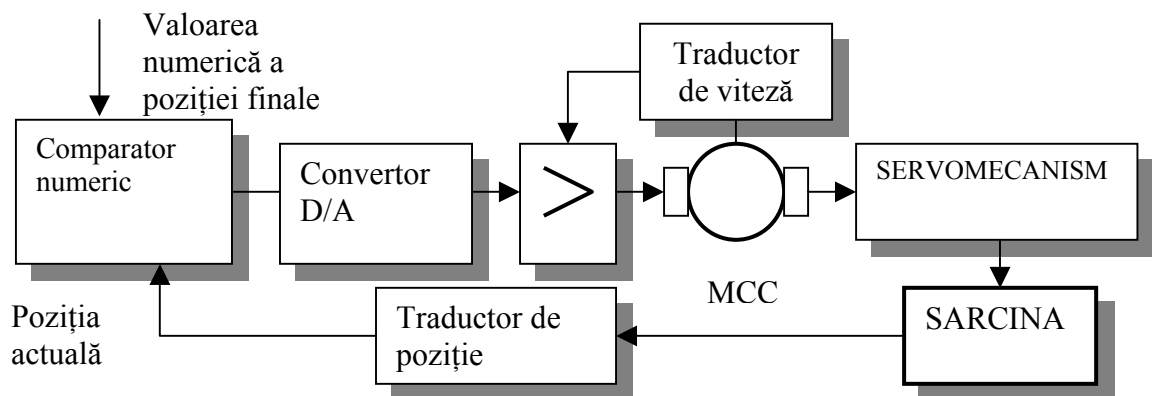


Figura 5.1: Schema bloc a unei acționări cu motor de curent continuu

Cel mai simplu de comandat, pentru a obține o viteză variabilă, sunt motoarele de c.c. deoarece cu cât tensiunea aplicată este mai mare cu atât viteza este mai mare. O mișcare de poziționare este compusă dintr-o accelerare, dintr-o deplasare cu viteză constantă și o frânare, conform unei traiectorii de viteză ca în figura 5.2.

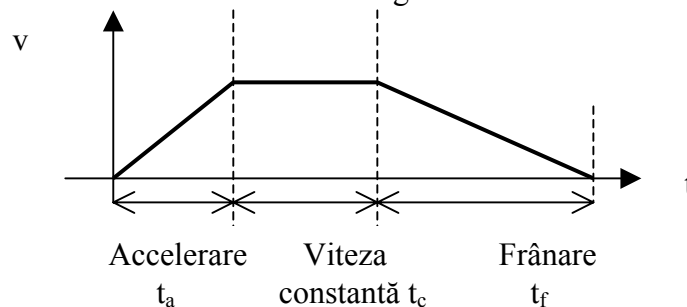


Figura 5.2: Traiectoria de viteză

Timpul de deplasare este $t_d = t_a + t_c + t_f$ iar timpul de poziționare este suma dintre timpul de deplasare și timpul de stabilire (de amortizare a oscilațiilor sistemului de poziționare după atingerea poziției finale), $t_p = t_d + t_s$

Pentru a obține viteza variabilă este suficient să aplicăm o tensiune variabilă. Tensiunea variabilă poate fi aplicată în mai multe feluri:

- informația numerică este convertită într-o informație analogică și este aplicată unui tranzistor (pentru comanda într-un sens) sau la doi tranzistori (pentru comanda în ambele sensuri). Tensiunea variabilă astfel obținută se aplică motorului de c.c. Un dezavantaj este folosirea unui convertor D/A și puterea pierdută în tranzistorii care lucrează în zona liniară.
- informația numerică creează un semnal PWM, cu frecvența destul de mare ca motorul, datorită inerției, să integreze impulsurile. Motorul va avea o viteză proporțională cu factorul de umplere. Acest mod de comandă este mult mai simplu și tranzistorul, fiind în regim de comutație nu disipă inutil.

Dificultatea acționărilor cu motor de curent continuu este partea de traductor de poziție care, mai ales la precizia cerută în aplicațiile noi sunt dificil de implementat.

În diagrama următoare (figura 5.3) se arată comanda PWM în ambele sensuri:

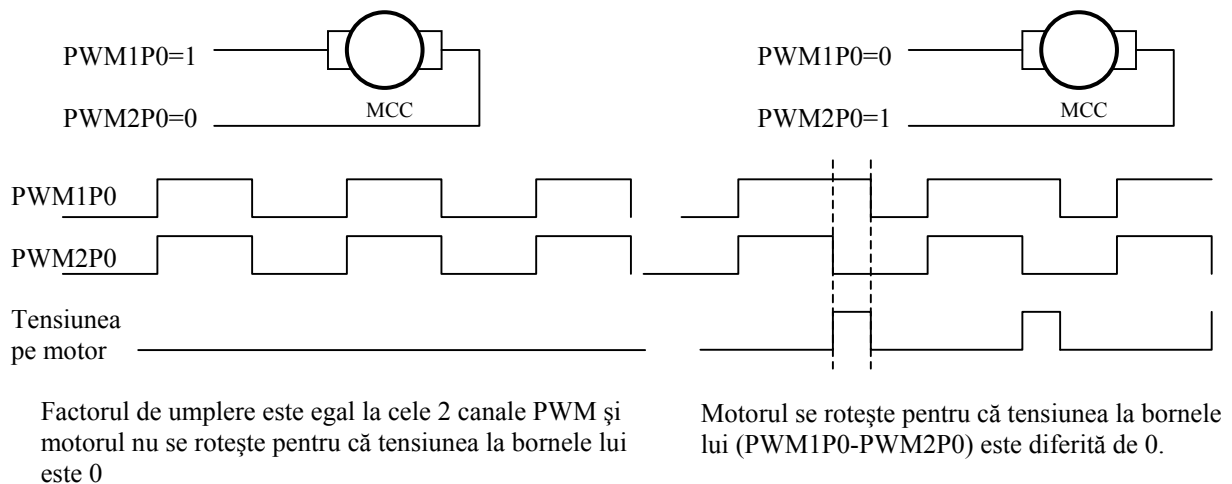


Figura 5.3: Comanda unui motor de curent continuu

Cuplând un motor de curent continuu între PWM1P0 și PWM2P0 se poate obține o rotire cu viteză variabilă în ambele sensuri ale motorului. Motorul de curent continuu se rotește cu o viteză proporțională cu tensiunea aplicată. Dacă tensiunea aplicată este sub formă de impulsuri motorul se rotește proporțional cu valoarea medie a tensiunii.

O simulare în SIMULINK dovedește valabilitatea acestei metode de control. S-a folosit un model din SIMULINK pentru motorul de curent continuu și cele 2 canale PWM au fost simulate cu generatoare de impulsuri cu lățime variabilă, figura 5.4.

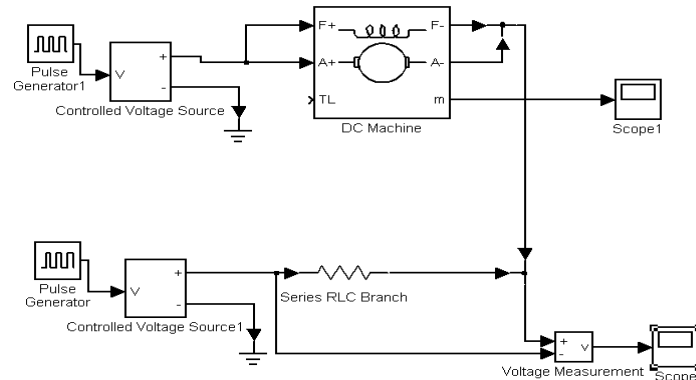


Figura 5.4: Model SIMULINK pentru simularea funcționării motorului de curent continuu
 Rezultatele simulării sunt reprezentate în figura 5.5. În stânga sus este reprezentată forma curentului prin motor și jos turația obținute pentru factor de umplere de 90% respectiv 10%.

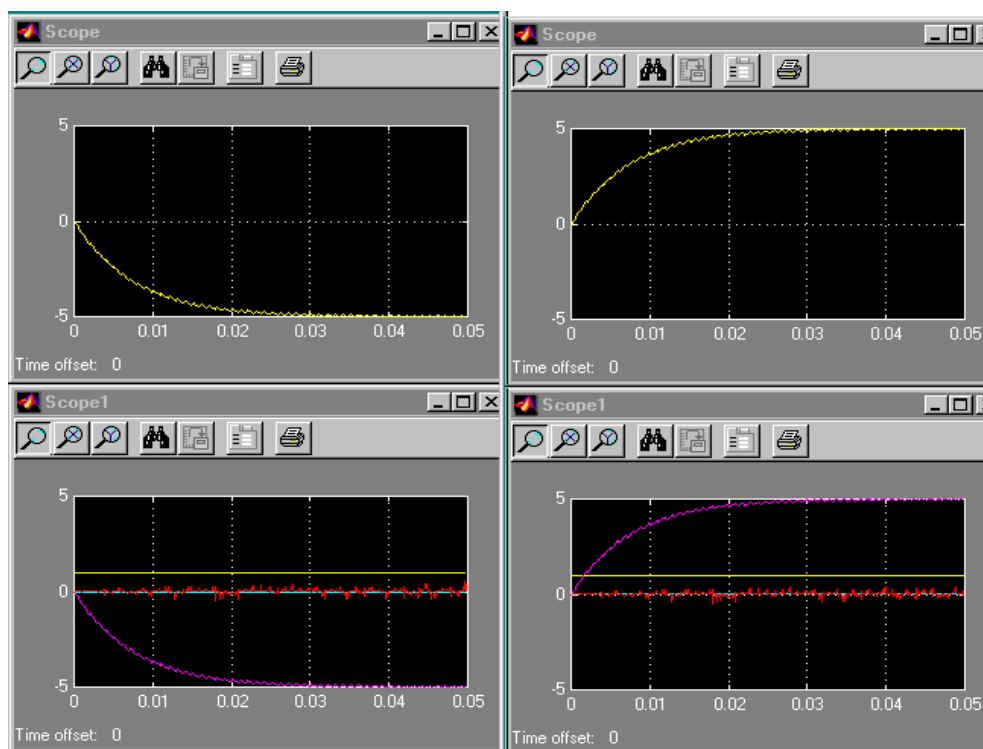


Figura 5.5: Rezultatele simulării SIMULINK

Se observă că acestea au aceeași formă de variație. În dreapta sunt reprezentate aceleași curbe, dar cu factor de umplere 10% respectiv 90%. Se vede că motorul se rotește în sens contrar.

Din punct de vedere constructiv motoarele de curent continuu pot fi cu perii la care rotirea câmpului magnetic este realizată prin comutarea curentului prin înfășurări cu un sistem colector și perii. Aceste motoare sunt asociate cu un nivel mare de perturbații electromagnetice generate. În figura 5.6¹ este reprezentat un motor de acționare a hârtiei la o imprimantă HP lângă o șurubelniță pentru a putea aprecia dimensiunea. Se poate observa tubul de ferită de pe cablu care are rolul de a micșora perturbațiile.



Figura 5.6: Motor de curent continuu

O altă variantă este motorul de c.c. la care câmpul învârtitor este realizat prin excitarea pe rând a unor bobine, funcționarea fiind asemănătoare motoarelor pas cu pas. Motoarele cu mișcare circulară pot acționa liniar printr-un sistem de conversie mecanic al mișcării, cele mai cunoscute fiind acționarea unui șurub fără sfârșit sau cu bandă elastică.

O categorie deosebită sunt motoarele de c.c. liniare electrodinamice. Motorul liniar este format dintr-un stator magnet permanent și o bobină mobilă. Aceasta este acționată de forța electrodinamică care rezultă din interacțiunea dintre câmpul produs de curentul care parcurge bobina și câmpul magnetic staționar produs în stator de magneții permanenți. Se utilizează două tipuri constructive: cu bobină lungă și întrefier scurt și cu bobină lungă și întrefier scurt.

Lungimea bobinei lungi și a întrefierului lung trebuie să fie suficient de mari pentru ca în timpul mișcărilor de poziționare forța să se mențină constantă, deci în întrefier să se afle aceeași lungime de bobină, figura 5.7.

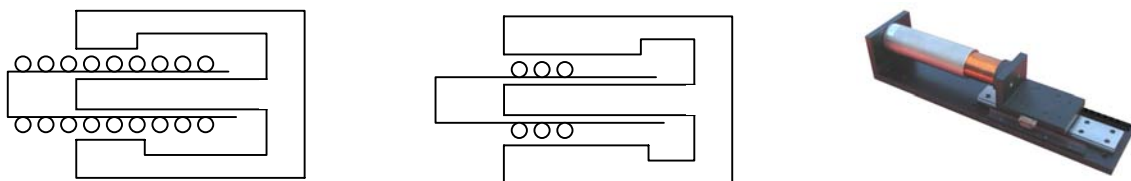


Figura 5.7: Motorul electrodinamic cu bobină lungă (stânga), cu bobină scurtă (mijloc) și fotografia unui motor

¹ În acest capitol majoritatea fotografiilor sunt ale unor subansamble din colecția autorilor

5.2. Comanda motoarelor pas cu pas

Motorul pas cu pas este un dispozitiv pentru conversia informațiilor numerice în lucru mecanic pe baza unui consum de energie de la o sursă. Motorul pas cu pas este un motor de curent continuu comandabil digital, cu deplasarea unghiulară a rotorului proporțională cu numărul de impulsuri primite. La fiecare impuls rotorul execută un pas unghiular apoi se oprește până la sosirea unui nou impuls. Motorul pas cu pas este capabil de reversarea sensului de mișcare. Dacă este comandat corect (cu o frecvență mai mică decât cea admisibilă) rămâne în sincronism cu impulsurile de comandă la accelerare, mers constant și încetinire. O schemă bloc de acționare este dată în figura 5.8.

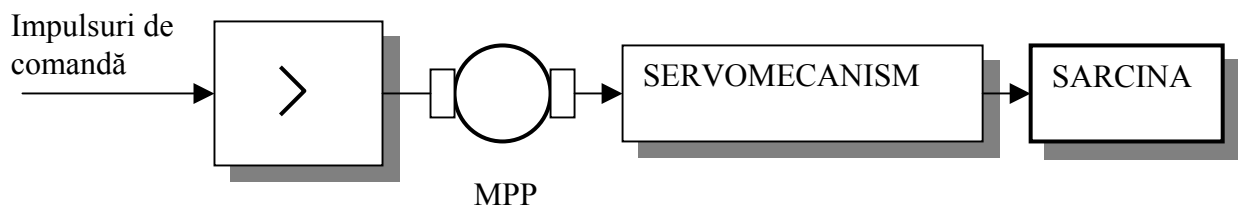
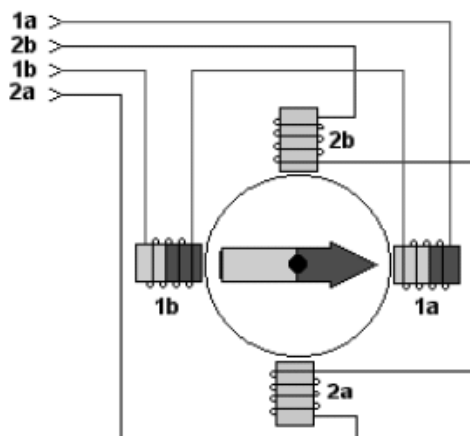


Figura 5.8: Schema bloc de acționare cu motor pas cu pas

Se poate observa din schema bloc că traductorul de poziție poate lipsi pentru că, în aplicații mai simple motorul va executa numărul de pași comandat și va ajunge în poziția dorită. Se câștigă astfel o simplificare a schemei dar se pierde la performanță. Comanda cu traiectorie de viteză este posibilă prin variația frecvenței impulsurilor dar schema de acționare se complică.

Motoarele pas cu pas pot fi motoare unipolare sau bipolare. La motoarele bipolare comanda pașilor se face prin inversarea curentului prin înfășurări. Principiul comenzii seamănă cu cel de la comanda motorului de curent continuu, cu diferența că în acest caz de regulă sunt 2 înfășurări, figura 5.9.



În acest caz controllerul trebuie să poată inversa polaritatea pentru o deplasare a curentului în ambele sensuri. Controllerul trebuie să alimenteze înfășurările succesiv cu o anumită secvență pentru un sens și secvența inversă pentru celălalt sens. În desenul alăturat o parcurgere a 4 faze înseamnă o rotație de 360°. Motoarele reale au mai multe înfășurări și un pas înseamnă o deplasare unghiulară mică.

Figura 5.9: Structura motorului pas cu pas bipolar

Conectarea unei înfășurări (de exemplu 1a-1b) la cele 2 canale PWM permite inversarea curentului ca în diagramele din figura 5.10.

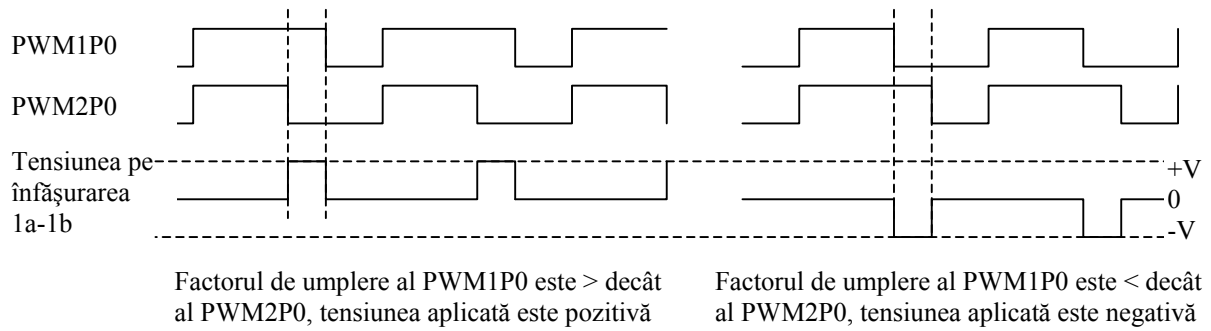


Figura 5.10: Inversarea curentului în înfășurări la motoarele bipolare

MPP unipolare folosesc o priză mediană legată la alimentare, inversarea curentului obținându-se prin legarea la masă succesivă a terminalelor extreme ale înfășurării, ca în figura 5.11.

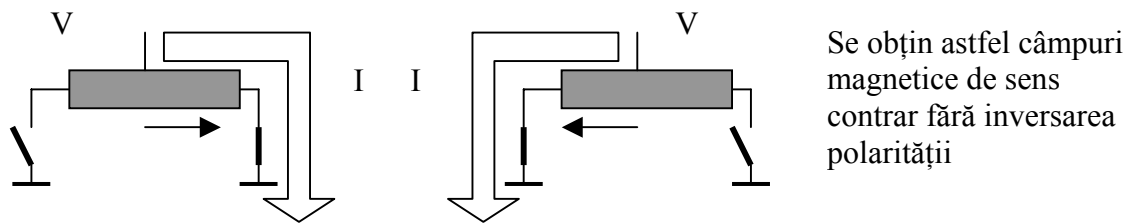


Figura 5.11: Inversarea curentului în motoarele unipolare

Înfășurările motorului sunt conectate ca în figura 5.12.

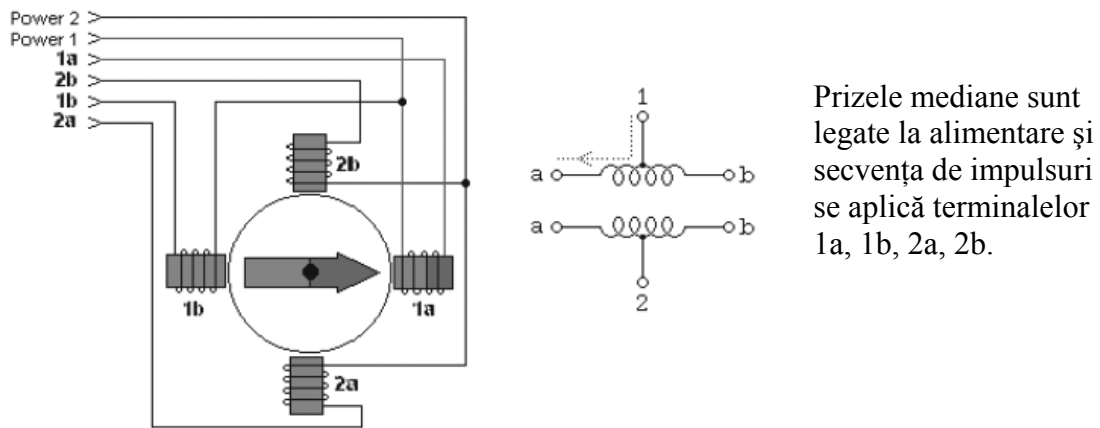


Figura 5.12: Structura și înfășurările motorului unipolar

Secvențele digitale pentru comanda unui motor pas cu pas unipolar cu 4 faze în varianta cea mai simplă sunt date în figura 5.13 [5].

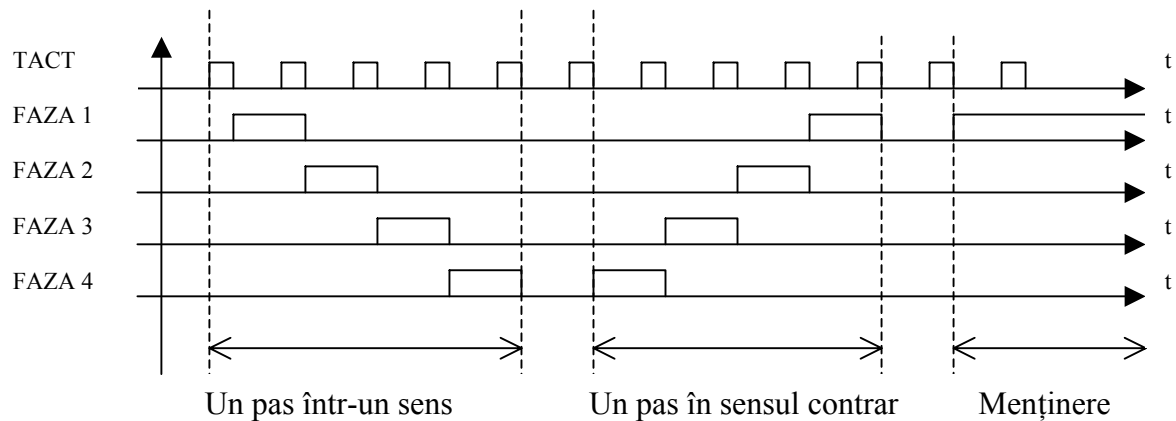


Figura 5.13: Secvențele digitale pentru comanda unui motor pas cu pas unipolar

Pentru exemplificare se alege un port paralel pentru microcontrollerelor compatibile x86, linia D0 este cel mai puțin semnificativ bit și un motor pas cu pas cu 3 faze. Un pas înseamnă parcurgerea tuturor fazelor. Schema de conectare și programul care generează un pas într-un sens sunt date în figura 5.14.

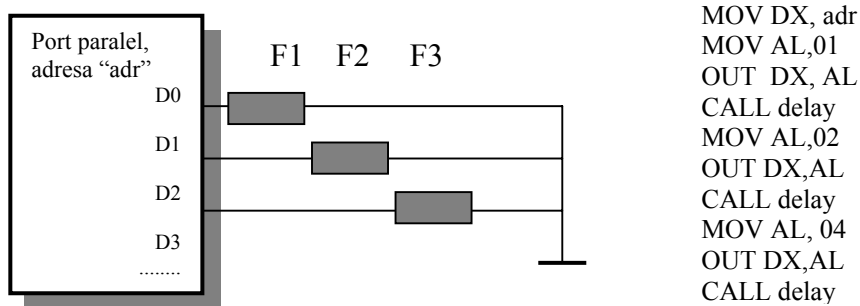
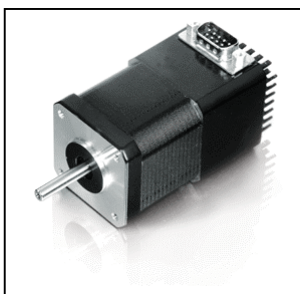


Figura 5.14: Schema de conectare a unui motor pas cu pas la un port paralel și programul

Programul din acest exemplu este dat pentru a arăta principiul de comandă. Este necesară introducerea unei întârzieri pentru menținerea alimentării fazei un timp suficient pentru acționarea electromecanică, timp care depinde de motor. Schema mai trebuie completată cu amplificatoare de curent și/sau tensiune pentru acționarea motorului. Un exemplu de motor pas cu pas modern, comandat cu două semnale (asigurarea semnalelor pentru faze sunt create intern) este dat în figura 5.15.



Motor bipolar pas cu pas

- Tensiunea de alimentare +12 - 24 VDC
- Curentul de pas pe fază 0.25 - 2.0 Amperi
- Pasul 1,8 grade
- Se poate comanda cu semnale: Pas, Directie, Validare, decuplate optic
- Protecție la depășirea temperaturii de lucru și la subtensiune

Figura 5.15: Motor pas cu pas

Observație

Un canal PWM poate fi folosit pentru a comanda o fază a motorului pas cu pas pentru a obține un curent variabil. Semnalul PWM nu asigură succesiunea fazelor ci valoarea medie a tensiunii de alimentare a unei faze. Conectarea ieșirilor PWM la fazele unui motor unipolar este dată în figura 5.16.

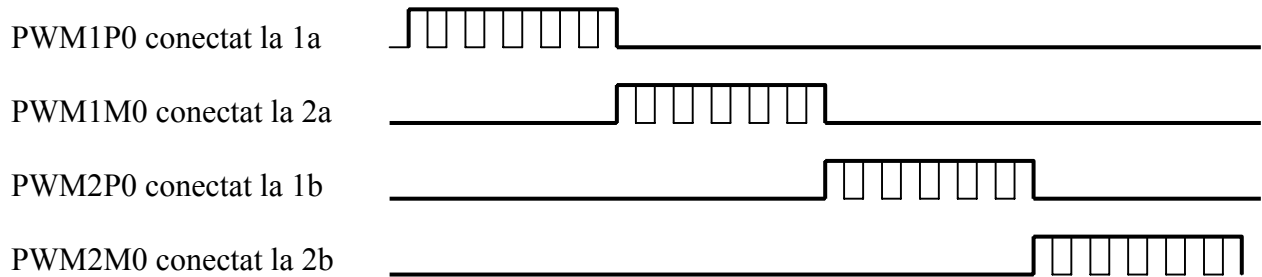


Figura 5.16: Comanda PWM a fazelor unui motor unipolar

Un modul de comandă poate controla un motor pas cu pas asigurând un curent bine definit prin înfășurări. Acest tip de comandă creează de exemplu posibilitatea de a comanda motorul cu un curent mai mare la pornire sau permite realizarea unor traiectorii optime de viteză (regim accelerat- frânat).

5.3.Traductori de poziție și circuite driver

Există o mare diversitate de traductoare de poziție pentru poziționarea motoarelor. Câteva exemple vor demonstra acest lucru. Senzorii cei mai utilizați sunt cei care generează două semnale defazate la deplasare (encoder), cu formă liniară sau circulară. Acești senzori pot fi inductivi, capacitivi sau optici. Senzorii optici echează imprimantele cu jet de cerneală la poziționarea capetelor de scriere. În funcție de modelul de imprimantă există senzori optici liniari sau circulari. Un senzor optic circular (encoder) și unul liniar sunt prezentați în figura 5.17, alături de o ruletă gradată în mm pentru comparație.

Se poate vedea precizia mai mare a traductorului circular (imprimantă de 600dpi) față de cea a traductorului liniar (300dpi). La echipamentele la care precizia unor asemenea senzori nu este suficientă, așa cum este poziționarea capetelor la hard disc sau unitatea optică, se folosesc pentru poziționare semnalele scrise pe suport.

La aplicații mai modeste pot fi folosiți senzori cum sunt cei prezentați în figura 5.18.

Senzorul de poziție M150 este un potențiomtru miniatură (18,8mm) la care axul este rotit prin tragerea unui fir de oțel acoperit cu plastic cu lungimea de 1,5” (38,1mm) care se retractează automat [1]. Acuratețea senzorului este mai bună de 1%. Rezistența maximă a potențiometrului este de 5k și tensiunea aplicată poate fi de maximum 20V. Extinderea firului poate să nu fie liniară, cu condiția respectării unor unghiuri limită. Senzorul SP1 este de

dimensiuni mai mari (50mm) dar firul poate atinge lungimea de 1270mm. Senzorul LX-EP [2] este un senzor cu fir de oțel dar principiul de detecție a poziției este optic iar ieșirea este digitală, oferind la ieșire două semnale defazate (encoder). Lungimea maximă a firului este de 1270mm.

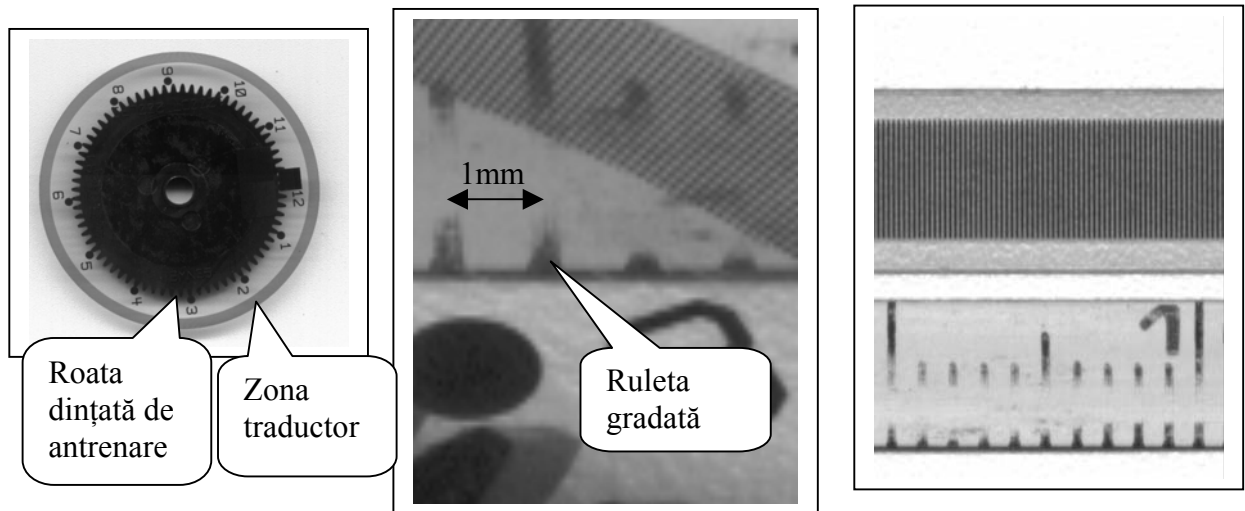


Figura 5.17: Traductor optic de poziție, circular (stânga), un detaliu (mijloc) și liniar (dreapta)



Figura 5.18: Senzor de poziție M150 (stânga) și SP1 (mijloc) și LX-EP (dreapta)

În funcționarea anumitor motoare se urmărește obținerea unei viteze constante și nu a poziționării precise (de exemplu la motoarele imprimantelor LASER). În aceste situații se pot utiliza traductori Hall sau spirale pe cablaj în care se induce o tensiune proporțională cu viteza de rotație, figura 5.19.

În figura 5.19 dreapta se poate remarca structura internă a unui motor de curent continuu fără perii în care câmpul învârtitor este realizat prin mai multe bobine. La acest motor rotorul este exterior și este un magnet permanent.

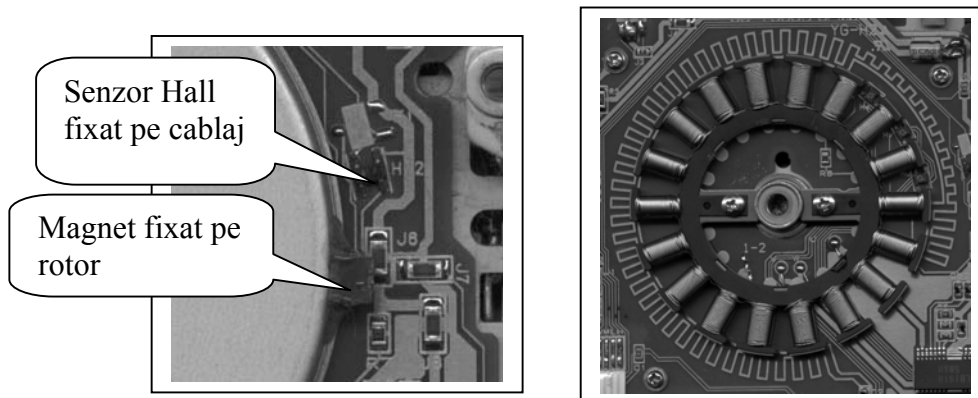


Figura 5.19: Traductor Hall (stânga) și spirală traductor de viteză (dreapta)

Un traductor Hall este de exemplu cel produs de Infineon, TLE4921. Traductorul constă în 2 senzori Hall la 2,5mm distanță formând un traductor diferențial utilizat pentru determinarea poziției și turației. Piesa mobilă poate fi un magnet permanent dar poate fi și din material feromagnetic, caz în care pe senzor trebuie montat un magnet permanent. Traductorul conține sisteme de protecție la supratensiune și alimentare inversă și poate fi alimentat în gama 0-30V. Ieșirea este digitală. Traductorul și o schemă de conectare sunt date în figura 5.20.

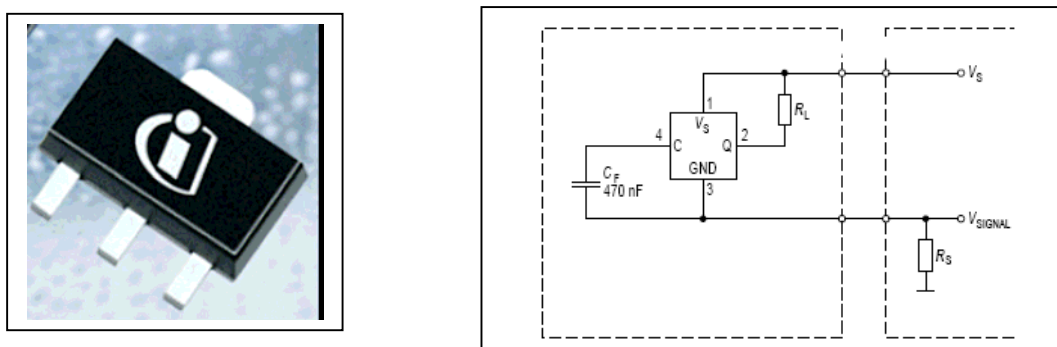


Figura 5.20: Traductorul Hall Infineon TLE4921 și schema electrică de conectare

Pentru acționarea motoarelor care absorb un curent de comandă mai mare decât cel generat de microcontroller se pot folosi amplificatoare de curent. Un circuit integrat preferat în proiectele noastre este ULN2003, figura 5.21. Circuitul conține șapte tranzistoare Darlington integrate care asigură un curent mediu de 500mA și un curent de vârf de 600mA la o tensiune de maxim 50V. Pentru mărirea curentului se pot folosi două canale în paralel. Intrările sunt comandabile TTL /CMOS. Intrările și ieșirile sunt pe părți diferite ale capsulei, ușurând astfel proiectarea cablajului. Câte o diodă pe fiecare canal asigură protecția în cazul sarcinilor inductive.

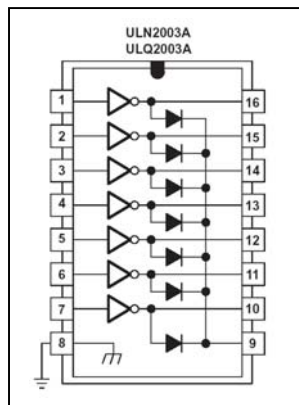


Figura 5.21: Circuit driver pentru motoare ULN2003

Un circuit driver mai sofisticat prin care se pot comanda motoare de curent continuu bidirecțional cu un singur canal PWM și 2 semnale linii de semnal ajutătoare este circuitul Allegro A3950. Acesta poate asigura curenți de până la 2,8A la o tensiune de 36V. Circuitul are protecție la scurtcircuitul motorului, la subtensiune și la supratemperatură. Circuitul conține o punte Full Bridge prin care curentul poate fi furnizat bidirecțional. Circuitul poate fi comandat într-o stare de economie de energie cu un semnal de SLEEP. Diagrama de timp pentru comanda bidirecțională este dată în figura 5.22.

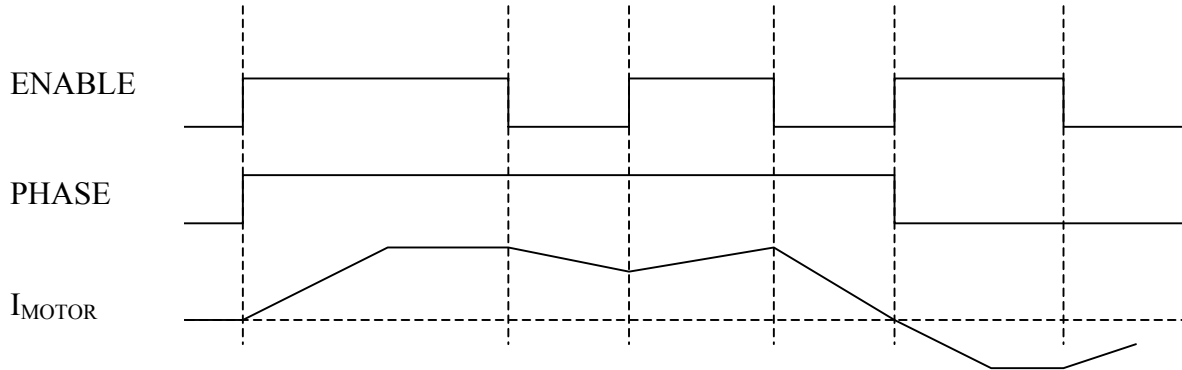


Figura 5.22: Diagrama de timp pentru comanda bidirecțională a unui motor de curent continuu cu Allegro A3950

Dacă semnalul PWM este aplicat la intrarea ENABLE motorul va fi comandat într-un sens dacă semnalul PHASE este 1 logic și în sensul contrar dacă semnalul PHASE este 0 logic.

Diversitatea driverelor este foarte mare, așa cum de fapt este diversitatea motoarelor. Pe pagina Sanyo (<http://semicon.sanyo.com/en/motor/index.php>) există prezentate multe modele de drivere grupate după tipul motoarelor la care se folosesc dar și după domeniul de aplicabilitate: uz casnic, auto, echipamente periferice, aparate foto etc.

5.4. Generarea semnalelor PWM

Generarea semnalelor PWM cu microcontrollere se poate realiza folosind un timer de uz general din structura microcontrollerului sau module specializate în cazul microcontrollerelor dedicate controlului motoarelor.

5.4.1. Timere I/O

Un timer de I/O tipic familiei de microcontrollere Fujitsu [3] constă în 2 numărătoare de 16 biți care au asociate 4 module de comparare și 4 module de captură. Modulele de comparare au 8 pini externi de ieșire iar modulele de captură au 8 pini externi de intrare.

Numărătorul numără liber un tact intern sau un tact extern care pot fi divizate de un circuit de prescalare. Se poate genera o întrerupere dacă numărătorul a ajuns la capăt (Overflow) sau la egalitatea ieșirilor numărătorului cu un registru de comparare. Numărătorul se poate inițializa la un Reset extern, la un Reset soft sau la egalitatea cu registrul de comparare.

Regiștrii de comparare sunt de 16 biți și sunt controlați de un registru de control. Când un numărător ajunge la valoarea stocată în registrul de comparare nivelul de ieșire al pinului extern corespunzător este modificat, se cere o întrerupere și se inițializează numărătorul (dacă această opțiune este validată). Pentru generarea PWM este folosit numărătorul și regiștrii de comparare.

Regiștrii de captură sunt de 16 biți și sunt controlați de un registru de control. La apariția unui front de la un pin extern valoarea numărătorului poate fi stocată într-un registru de captură și se generează o întrerupere. Frontul de declanșare poate fi programat: cel crescător, descrescător sau ambele fronturi.

Schema bloc a timerului este dată în figura 5.23. Liniile gri din spatele blocurilor semnifică faptul că fiecare numărător are asociate module de comparare OCU (Output Compare Unit) și module de captură ICU (Input Capture Unit). Numărătorul 0 este conectat cu OCU 0/1/2/3 și cu ICU 0/1/2/3 iar numărătorul 1 este conectat cu OCU 4/5/6/7 și cu ICU 4/5/6/7.

Se poate spune că timerul constă în 2 canale identice bazate pe câte un numărător, fiecare canal având asociate 2 module de comparare, 2 module de captură și 8 pini externi din care 4 de ieșire și 4 de intrare.

Numărătorul conține un registru de date TCDT0/1 (Data Register of Free Running Timer) de 16 biți din care se poate citi valoarea numărătorului sau se poate scrie (doar când numărătorul este oprit). Tactul numărătorului poate fi selectat tactul intern sau tactul de la un pin extern. Selecția se face cu un bit din registrul TCCSH0/1.

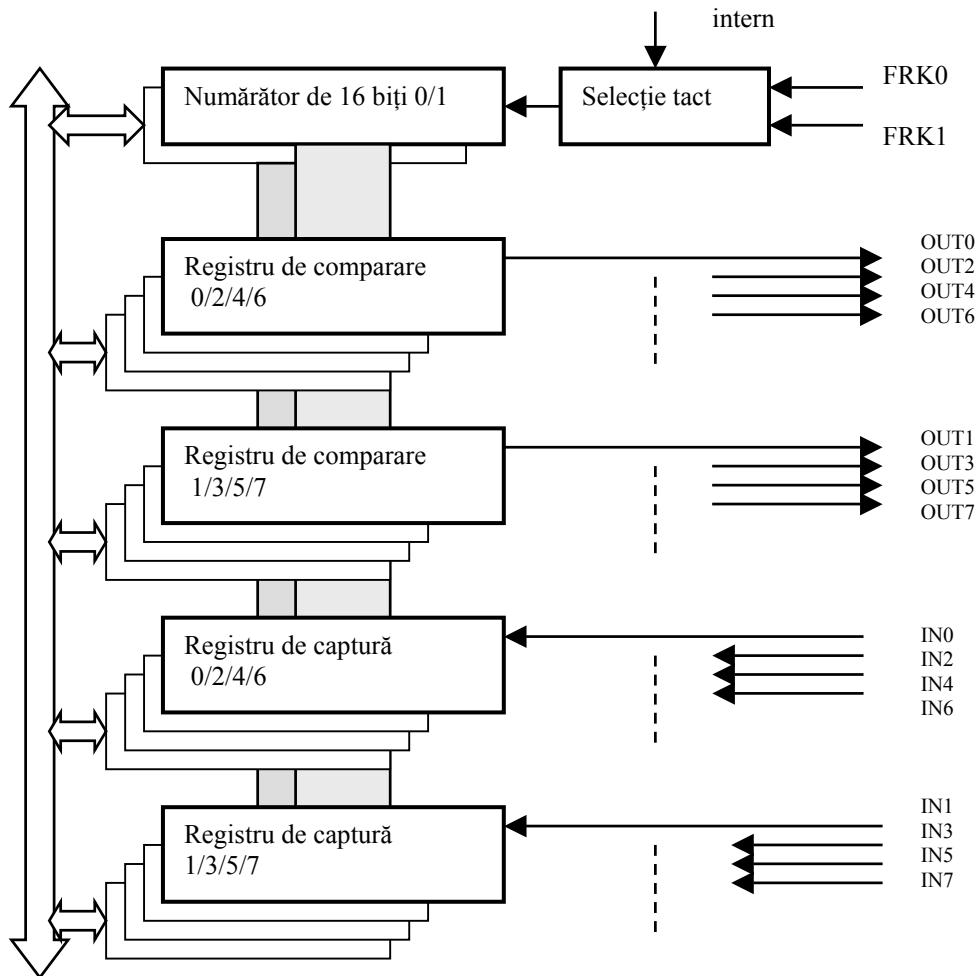


Figura 5.23: Schema bloc a timerului de I/O

Un registru de control TCCSL0/1 programează funcționarea numărătorului. Cererea de întrerupere IVF este validată de bitul IVFE. Schema bloc a numărătorului este dată în figura 5.24.

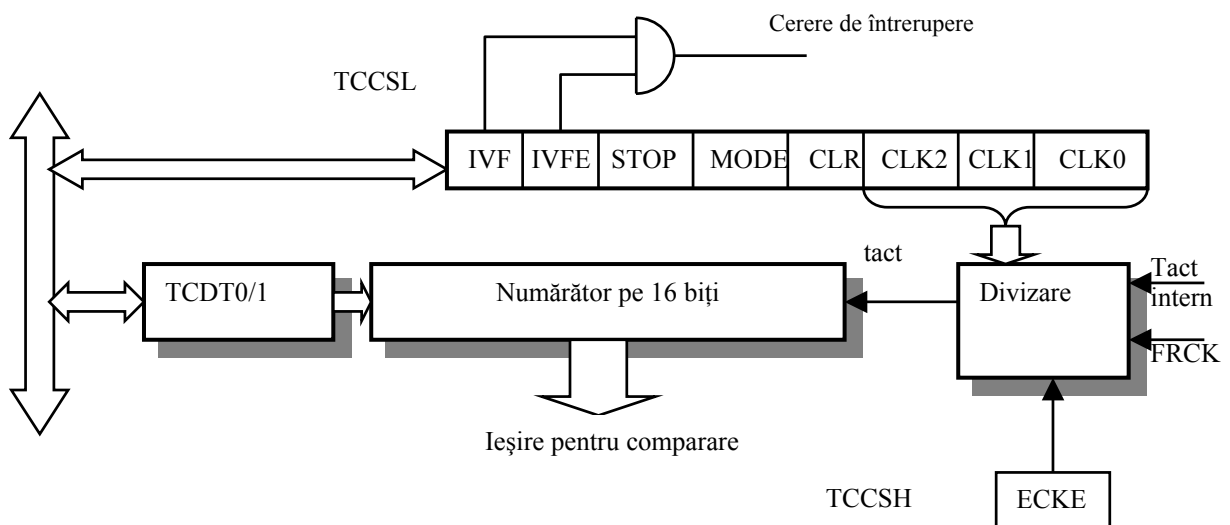


Figura 5.24: Schema bloc a numărătorului din componența timerului de I/O

TCCSL0/1 (Control Status Register of Free Running Timer) este un registru de 8 biți care conține: IVF este un flag de cerere de întrerupere, IVFE validează cererea de întrerupere, STOP oprește numărătorul, MODE stabilește modul de inițializare (cu nivel H numărătorul se inițializează prin comparare și egalitate cu regiștrii de comparare), CLR inițializează numărătorul cu 0000H, 3 biți CLK stabilesc factorul de divizare al tactului.

TCCSH0/1 (Control Status Register of Free Running Timer) este un registru de 8 biți din care este folosit un singur bit (ECKE) care selectează tactul intern sau tactul extern de la pinul FRCK.

O diagramă de timp arată evoluția numărătorului și inițializarea prin depășire (figura 5.25a) și inițializarea prin egalitate cu regiștrul de comparare (figura 5.25b).

Pentru analiza evoluției unui timer sunt utile diagramele de timp cu reprezentare “analogică” a valorii digitale din numărător. Modulul de comparare constă în 2 regiștrii de 16 biți, 2 pini de comparare de ieșire și un registru de control. În MB90350 sunt integrate 4 module de comparare separate (dar același numărător la 2 module).

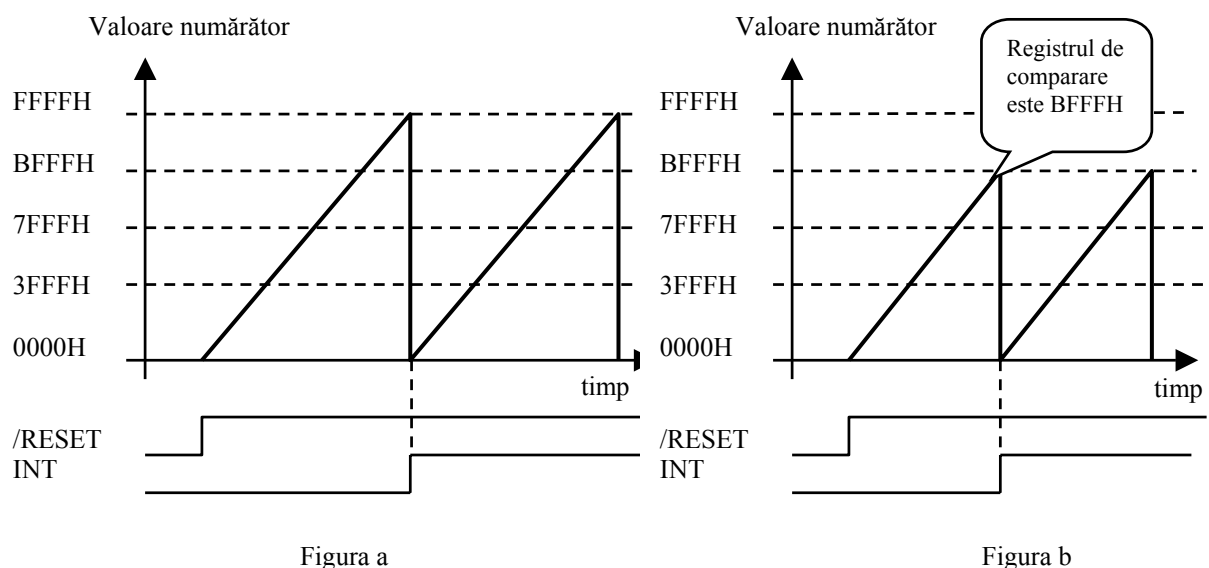


Figura 5.25: Inițializarea numărătorului prin depășire și prin egalitate cu regiștrul de comparare

Pentru fiecare modul regiștrii de comparare pot fi utilizați independent. În funcție de modul de operare regiștrii pot comanda pini de ieșire. Valoarea inițială a fiecărui pin poate fi programată separat. Fiecare modul poate genera un semnal PWM folosind ambii regiștrii de comparare. Schema bloc a modulului de comparare este dată în figura 5.26.

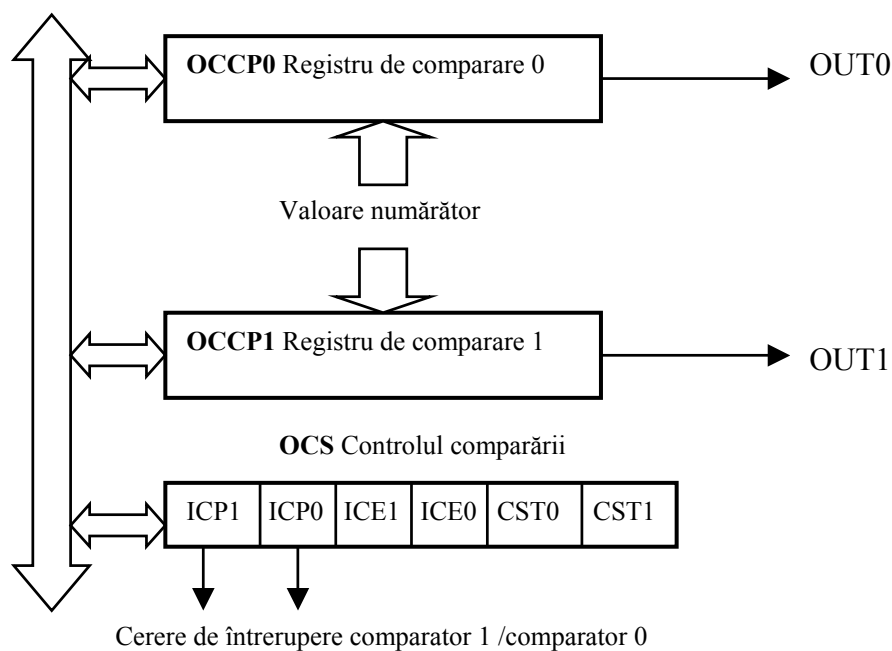


Figura 5.26: Schema bloc a modului de comparare

OCCP (Registru de comparare, Output Compare Register) de 16 biți conține valoarea cu care se compară valoarea numărătorului. Acest registru trebuie încărcat înainte de a valida operarea.

OCS0/2/4/6 (Control Status Register of Output Compare Low) sunt regiștrii de 8 biți în care ICP este o cerere de întrerupere la egalitate (câte una pentru fiecare registru de comparare), ICE sunt biți de validare a cererii de întrerupere iar 2 biți (CST) validează operația de comparare.

OCS1/3/5/7 (Control Status Register of Output Compare High) sunt regiștrii de 8 biți în care 2 biți (CMOD0 și CMOD1) validează modul de operare al pinilor de ieșire (nivelul lor logic se schimbă sau nu la egalitate), 2 biți validează pinii de ieșire iar 2 biți schimbă valoarea logică a ieșirilor.

Funcționarea poate fi fără schimbarea valorii logice a pinilor de ieșire la egalitate (CMOD0/1=00B) și funcționarea cu schimbarea valorii logice a ieșirii OUT0 și OUT1 la egalitatea OCCP0 cu OCCP1 (CMOD0/1=01B). De asemenea ieșirea poate să schimbe sau nu valoarea logică la inițializarea numărătorului. Generarea a 2 semnale PWM este exemplificată în figura 5.27.

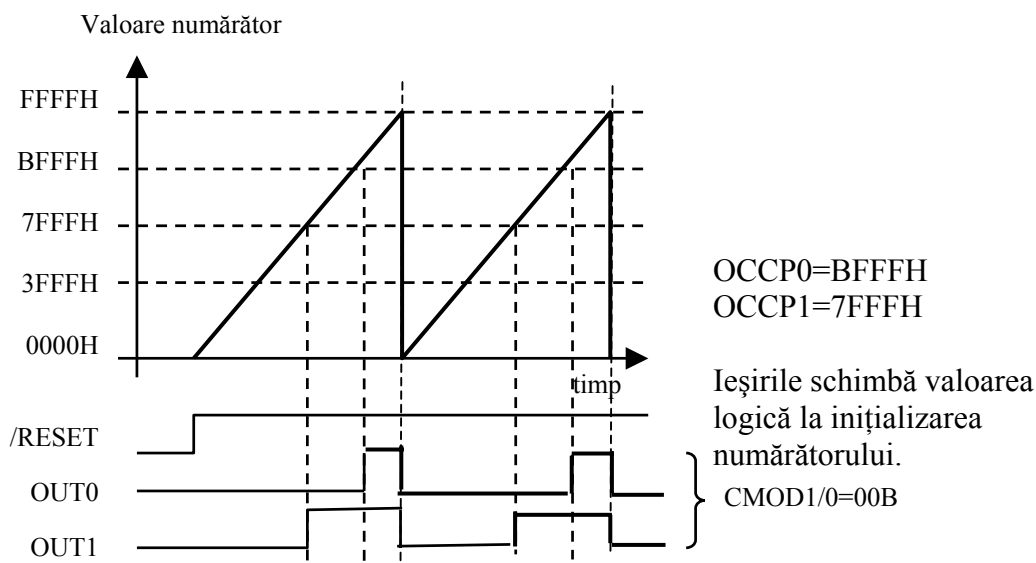


Figura 5.27: Două moduri de lucru ale modulului de comparare

5.4.2. Exemplu de modul pentru comanda motoarelor în microcontrollere specializate

MC Fujitsu sunt folosite mult în aplicații auto, de aceea se integrează un modul dedicat comenzii motoarelor pas cu pas, dar care poate comanda și motoare de curent continuu. Acest modul apare deja la MC pe 8 biți de tip vechi (MB89940), echipează multe MC pe 16 biți (MB90390, 90420, 90590, 90595) și MC pe 32 de biți (MB91360).

Modulul de comandă pentru comanda motoarelor conține 2 generatoare PWM, 4 etaje de ieșire de putere (drivere de motor) și logica de selecție. Driverul poate asigura un curent mare care permite conectarea directă a motoarelor de mică putere. Un mecanism de sincronizare asigură funcționarea sincronizată a celor 2 generatoare PWM. În general MC conțin mai multe asemenea module. Schema bloc a unui modul de control motoare care conține 2 canale PWM și 4 drivere de ieșire este dată în figura 5.28.

Numărătorul PWM pentru fiecare canal numără până la capăt, apoi se reîncarcă și numărarea reîncepe. La coincidența valorii din numărător cu cea stocată în registrul de comparare semnalul de ieșire își schimbă starea logică. La sfârșitul numărării semnalul de ieșire își schimbă din nou starea logică. Regiștrii de comandă sunt:

PWM Control Register (8 biți) controlează pornirea/ oprirea operării, pinii externi și întreruperile. 2 biți selectează funcționarea pinilor externi ca pini de control motor sau pini de I/O de uz general, 2 biți selectează tactul (tact divizat cu 1, 2, 4 sau 8), un bit pornește operarea (PWM2 pornește un tact după PWM1 pentru a micșora zgomotul produs în driver la o comutare simultană de curenți mari), un bit selectează operarea PWM pe 8 sau 10 biți.

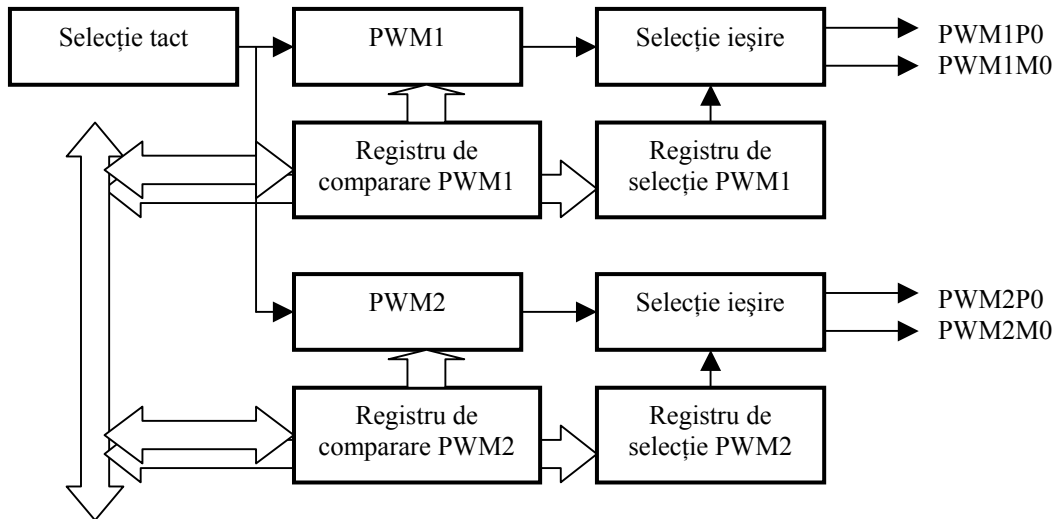


Figura 5.28: Schema bloc a modului de comandă a motoarelor

PWM1/2 Compare Register (16 biți) specifică lățimea impulsului PWM (formatul de scriere este pe 8 sau 10 biți). O valoare de 000H indică un factor de umplere de 0% iar o valoare de 3FFH indică un factor de umplere de 99,9% (99,6% pentru formatul pe 8 biți). Pentru a putea stoca valori pe 10 biți regiștrii de comparare sunt pe 16 biți. Trei variante de semnale cu factor de umplere diferit sunt arătate în figura 5.29.

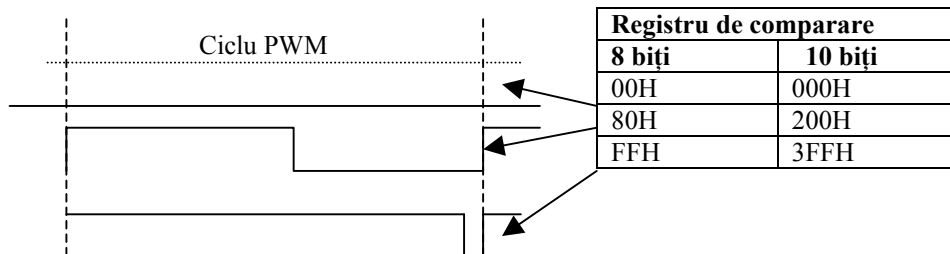


Figura 5.29: Trei diagrame de timp pentru 3 factori de umplere diferiți

PWM1/2 Selection Register controlează pinii de ieșire. Un bit asigură sincronizarea ieșirilor în sensul că orice modificare în configurarea canalelor PWM nu se reflectă la ieșire decât după ce acest bit este setat. Câte 3 biți selectează tipul de semnal de ieșire la ieșirile PWM1P0, PWM1M0, PWM2P0, PWM2M0- semnal în stare L, H, impedanță ridicată sau PWM.

5.5. Interfața pentru comanda motoarelor HCTL 1100 (Hewlett Packard)

Un circuit de interfață specializat în comanda motoarelor este circuitul HCTL 1100 [4]. Acesta poate comanda atât motoare pas cu pas cât și motoare de curent continuu cu traiectorie determinată de viteză. Această interfață poate fi conectată la un microcontroller de uz general obținând un sistem înglobat performant de control. Alegerea acestei variantei în locul celei cu microcontroller specializat în controlul motoarelor depinde de aplicație și de costurile din acel moment. Schema bloc a interfeței HCTL-1100 este dată în figura 5.30.

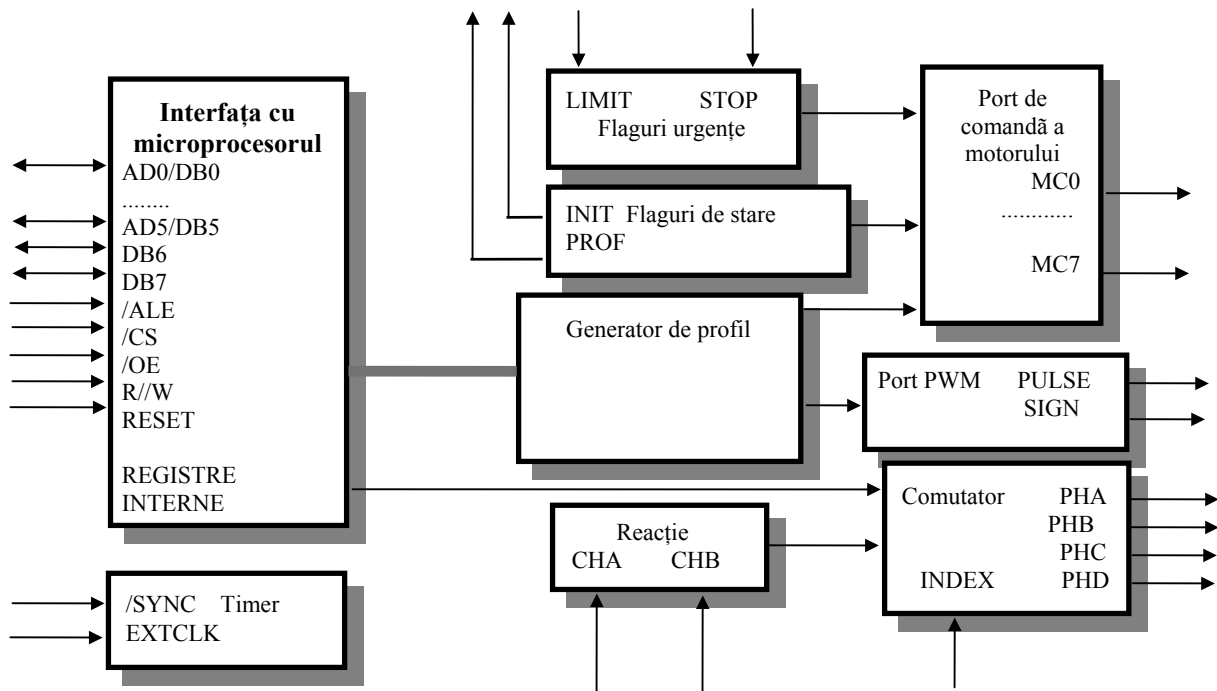


Figura 5.30: Schema bloc a interfeței HCTL-1100

AD0/DB0- AD5/DB5- sunt 6 linii de adrese multiplexate cu date.

Selecția adresei se face cu /ALE;

DB6, DB7- 2 linii de date nemultiplexate.

La început se stabilește adresa pe liniile de date, apoi cu /ALE, aceasta se stochează în registrele interne. Apoi se stabilesc datele și se validează cu R/W. Semnalul /OE stabilește momentul citirii datelor în registrul intern al lui HCTL1100.

Semnalele CHA, CHB și INDEX provin de la traductorul foto de turație. Aceste impulsuri incrementează un registru de poziție actuală de 24 de biți (se incrementează sau decrementează). Pentru evitarea impulsurilor eronate CHA și CHB, acestea trebuie să rămână active cel puțin 3 perioade EXTCLK pentru a fi luate în considerare.

Intrările LIMIT și STOP sunt semnale de urgență care comandă oprirea motorului independent de procesorul sau MC gazdă. Ieșirile PROF și INIT pot fi interogate de MC gazdă pentru a afla starea circuitului. Cu /SYNC se pot sincroniza între ele mai multe circuite HCTL 1100. Semnalul RESET aduce circuitul în starea inițială.

HCTL 1100 are 64 de registre interne pe 8 biți din care 35 utilizabile din exterior. Aceste registre sunt adresate pe liniile de adresă AD0-AD5.

Traductorul optic de poziție și de sens este format dintr-un disc cu orificii dreptunghiulare dispuse pe circumferință. Orificiile sunt sesizate de două sisteme optice decalate. Din formele de undă generate de cele două sisteme decalate se poate deduce sensul rotației, (figura 5.31).

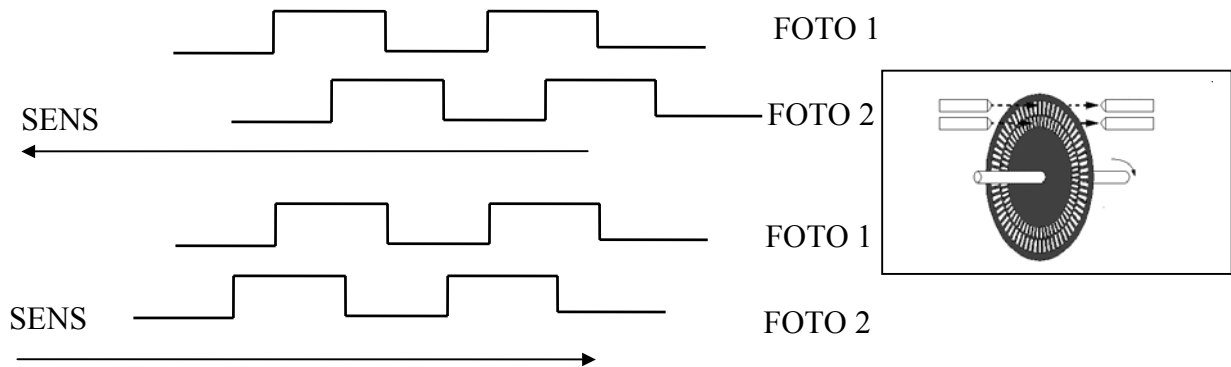


Figura 5.31: Formele de undă generate de traductorul de poziție și sens și un traductor optic cu disc

Pe disc mai este realizat un orificiu pentru impulsul de INDEX. La o rotație se generează un impuls INDEX. Impulsurile de la traductoare sunt sensibile la perturbații, de aceea se impun măsuri specifice EMC de protecție. Aceste semnale pot constitui atât semnale pentru stabilirea poziției (prin numărarea impulsurilor), a sensului prin verificarea defazajului cât și a vitezei, printr-un convertor tensiune/frecvență.

5.5.1. Comanda unui motor de curent continuu printr-un convertor D/A

La ieșirile MC0-MC7 se cuplează un convertor D/A, ca în figura 5.32.

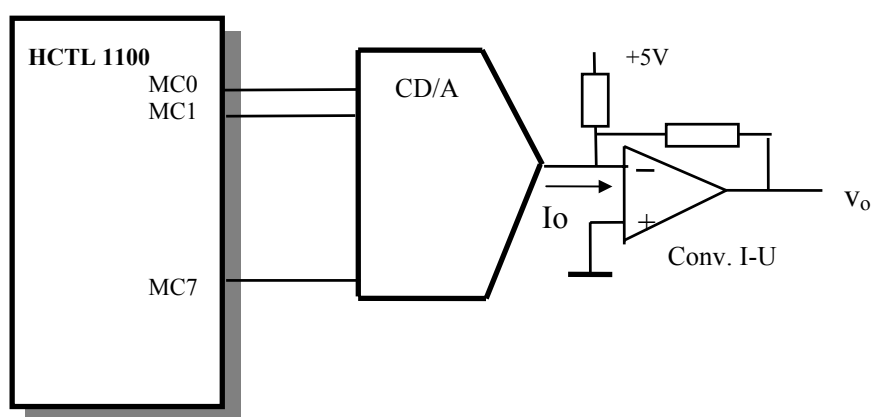


Figura 5.32: Conectarea unui motor de c.c. la HCTL 1100

Datele se stochează în registrul 08H al HCTL pe 8 biți. Pentru a putea comanda motorul în ambele sensuri, se consideră tensiuni negative în intervalul de comandă 00H-7FH și valori pozitive în intervalul 80H-FFH, figura 5.33.

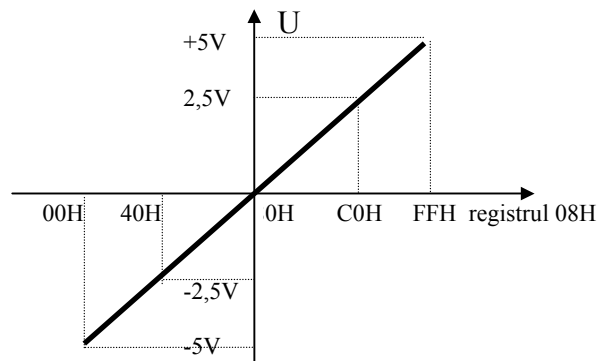


Figura 5.33: Generarea tensiunilor de comandă prin convertorul AD

5.5.2. Comanda PWM a unui motor de c.c. cu HCTL 1100

Comanda unui motor de c.c. se face cu semnalul PULSE. Frecvența acestuia este stabilită la $EXTCLK/100$. Semnalul SIGN comandă sensul de rotație. În registrul 09H al HCTL 1100 se încarcă lățimea impulsului, figura 5.34. Comanda se face cu valori zecimale, astfel 00H=0D înseamnă un factor de umplere de 0%, 32H=50D înseamnă 50% iar 64H=100D înseamnă 100%.

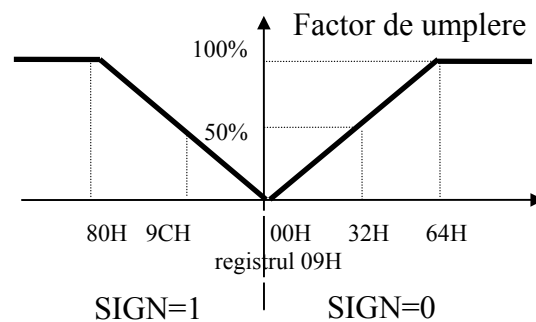


Figura 5.34: Programarea HCTL 1100 pentru comanda unui motor de c.c. cu modulație PWM

5.5.3. Comanda unui motor pas cu pas cu HCTL 1100

Se pot comanda motoare pas cu pas cu 2,3 sau 4 faze, cu diferite succesiuni de comandă a fazelor. Semnalul INDEX servește la stabilirea poziției inițiale a motorului. Succesiunea fazelor se poate programa prin registrul 07H. Pentru o programare în care toate fazele sunt active se generează o secvență ca în figura 5.35.


```

.device AT90S1200
    rjmp start

short_delay:                ;total 1+80*(1+1/2)-1+4=124
                            ;o întârziere de 124 perioade de tact
    ldi r26,80              ;1 tact
sd:    dec r26               ;1 tact
    brne sd                 ;1/2 tact
    ret                     ;4 tact
;#####
;PROCEDURA RESET HCTL-1100
;#####

rst:    cbi $12,4            ;bitul de reset din portul D
        rcall short_delay   ;se menține pinul de reset in 0 logic
        sbi $12,4           ;reset încheiat
        ret
;#####
;PROCEDURA CITIRE
;#####

cit:    clr r16
        out $17,r16         ;portul B intrare
        sbi $12,0           ;comandă read
        ldi r17,0b00000001  ;s-a selectat un registru din HCTL
        out $18,r17
        cbi $12,3           ;puls pe ALE
        sbi $12,3

        cbi $12,2           ;puls pe CS
        sbi $12,2

        rcall short_delay

        cbi $12,1           ;OE=0
        in r18,$18          ;citire propriu zisă din HCTL1100
        sbi $12,1           ;OE=1
        ret
;#####
;#####
;PROCEDURA SCRIRE IN HCTL1100
;#####

scr:    ser r16
        out $17,r16         ;portul B ieșire
        ldi r17,0b00000001  ;selecție latch din HCTL1100
        out $18,r17
        cbi $12,3
        sbi $12,3           ;puls ALE
        cbi $12,0           ;selecție R/W pe scriere
        ldi r18,0b1111111  ;date de trimis în HCTL1100

```



```

;#####
;PROCEDURA RESET
;#####
RS1100:    ORL   P2,#0FH      ;Setează liniile de R/W la citire, OE=1, CS=1, AE=1
           MOV   P0,#0FFH    ;Setează P1=HIGH
           CLR   P2.4        ;Setează RESET pe LOW
           NOP                    ;Întârziere corespunzătoare unui impuls de 5µs
           NOP
           NOP
           NOP
           SETB  P2.4         ;Reduce linia RESET în HIGH
           RET
;#####

;#####
;PROCEDURA CITIRE (Citește registrul lui HCTL)
;#####
RD1100     SETB  P2.0        ;Setează liniile R/W pentru citire
           MOV   P1,B        ;Adresa de LATCH
           CLR   P2.3        ;Se generează un puls ALE
           SETB  P2.3
           MOV   P1,#0FFH
           CLR   P2.2        ;Se generează un puls CS
           SETB  P2.2
           NOP                    ;Întârziere corespunzătoare unui impuls de 4µs
           NOP
           NOP
           CLR   P2.1        ;Setează OE=0
           MOV   A,P0        ;Se iau datele din HCTL 1100
           SETB  P2.1        ;Setează OE=1
           RET
;#####

;PROCEDURA SCRIERE (Scrie în registrul lui HCTL)
;#####
WR1100     MOV   P1,B        ;Adresa de LATCH
           CLR   P2.3        ;Se generează un puls ALE
           SETB  P2.3
           MOV   P1,#0FFH
           CLR   P2.0        ;Setează liniile R/W pentru scriere
           MOV   P1,A        ;Se emit datele
           CLR   P2.2        ;Se generează un puls CS
           SETB  P2.2
           SETB  P2.0        ;Întoarcere la modul de citire
           MOV   P1,#0FFH
           RET
;#####

```


5.6.Exemple de implementare

5.6.1. Comanda PWM a motoarelor folosind MC Motorola 6805

Schema bloc de control a unui motor de c.c. cu perii realizată de un MC68HC705 este dată în figura 5.38. Controlul vitezei poate fi realizat:

- cu un program intern pentru un sistem care nu are nevoie de feedback sau de schimbări dese ale programului;
- cu un program intern ghidat de câteva intrări, cu funcționalitățile sugerate în figură.

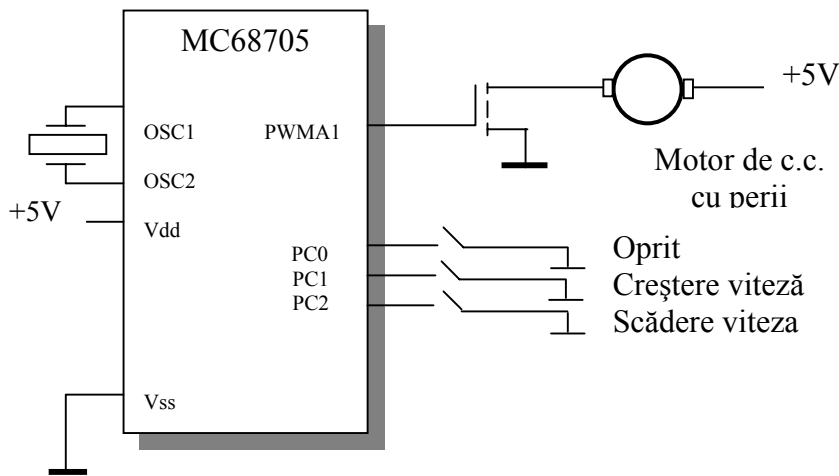


Figura 5.38: Schema bloc de control a unui motor de c.c. cu perii realizată de MC68HC705

Un program simplificat la maxim, care sugerează modul în care se comandă motorul este:

PORTA	EQU	\$00	Se aloacă denumiri sugestive porturilor A și C
PORTC	EQU	\$02	precum și registrelor pentru stabilirea sensului
DRDA	EQU	\$04	
DRDC	EQU	\$06	
PWMAD	EQU	\$10	portul de date PWM (pe liniile port A
CTLA	EQU	\$14	portul de control al PWM
Start	EQU	*	
	CLR	CTLA	registru de control al PWM este Resetat
	LDA	#\$20	se trimite 20h în portul de date PWM
	STA	PWMAD	(factor de umplere 50%)
	LDA	#\$00	registru C este definit de intrare
	STA	DDRC	(se putea și cu CLR PORTC)
adr	BRSET	0,PORTC,adr1	bitul 0 este setat (buton apăsat)?
	BSR	adr_off	dacă da, salt la subrutina motor oprit (off)
adr1	BRSET	1,PORTC,adr2	dacă nu, se verifică următorul buton

BRA adr

bucla se repetă continuu

Adr_off LDA #\$00

se trimite 00h în portul de date PWM

STA PWMAD

Dezavantajul acestei scheme de control este că permite mișcarea motorului doar într-un sens. Pentru a fi posibilă schimbarea sensului este nevoie de un bit suplimentar pentru comanda schimbării polarității tensiunii de alimentare a motorului.

Comanda PWM a unui motor pas cu pas folosind un MC Motorola 6805 poate fi realizată ca în figura 5.39. Bobinele trebuie alimentate într-o anumită succesiune pentru a obține sensul de rotație dorit. Majoritatea motoarelor sunt echipate cu traductoare de poziție (Hall sau optice). Un astfel de motor poate fi comandat cu viteză variabilă prin modificarea factorului de umplere (PWM), dar în acest caz trebuie modificată și viteza de comutație.

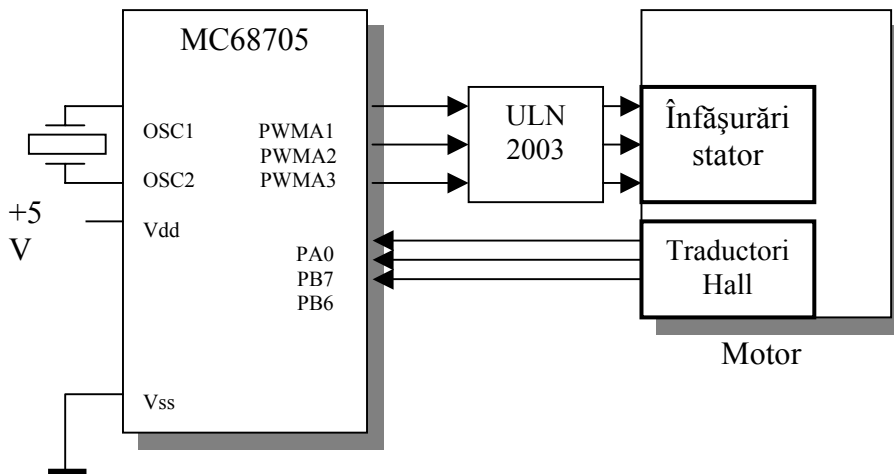


Figura 5.39: Comanda PWM a unui motor pas cu pas folosind un MC Motorola 6805

Acest sistem asigură performanțe de reglare superioare, având o complexitate mai mare. Secvența de acționare pentru comanda motoarelor este transmisă prin intermediul unui amplificator de putere. Semnalele digitale de la traductoare sunt citite pe 3 intrări digitale. Aplicația este detaliată în nota de aplicații Motorola AN1702/D, (<http://mcu.motps.com>).

5.6.2. Comanda PWM a motoarelor folosind MC 8051

În această aplicație se folosește un MC cu arie de numărătoare programabilă (PCA - 83C51FA) care are posibilitatea de a programa 2 canale PWM. O comandă bidirecțională a unui motor folosind 2 canale PWM și o buclă de reacție prevăzută pentru a menține o viteză constantă este arătată în figura 5.40.

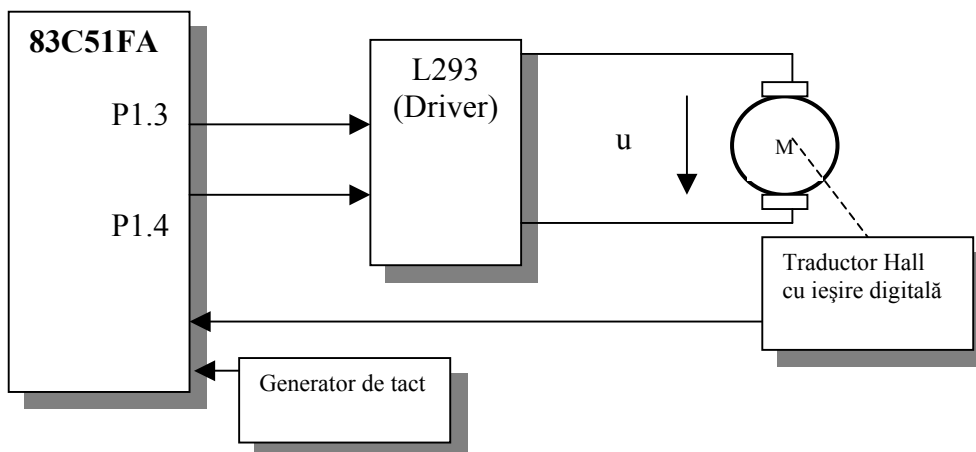


Figura 5.40: Comanda PWM a unui motor de c.c. folosind un MC 8051

MC comandă motorul folosind 2 canale PWM. Curentul necesar motorului nu poate fi furnizat de MC, de aceea se folosește un circuit driver L293 care conține 4 canale de amplificare, din care în acest caz sunt folosite 2 și un semnal de modificare a sensului. În acest montaj modificarea sensului se face prin modificarea factorului de umplere la cele 2 canale PWM.

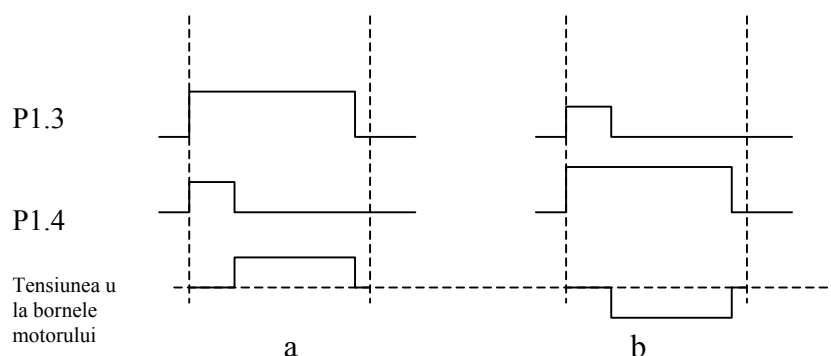


Figura 5.41: Diagrame de semnal pentru comanda PWM a motorului în ambele sensuri

Când P1.3 și P1.4 sunt egale diferență de potențial la bornele motorului este zero. În cazul (a), P1.3 este mai mult în 1 și motorul este alimentat cu +5V de la P1.3 la masă (P1.4) și se rotește într-un sens. În cazul B el este alimentat de la P1.4 și se rotește în celălalt sens. Viteza poate fi comandată prin variația factorului de umplere a celor 2 canale PWM, fiind proporțională cu diferența lor.

Bucula de feedback se poate închide montând pe rotorul motorului 2 piese magnetice, diametral opuse și formând 2 impulsuri, la fiecare rotire a motorului, cu un traductor Hall cu ieșire digitală. Semnalul este preluat de MC la un pin I/O de uz general, sau poate fi preluat la un canal al PCA în mod numărător (cum s-a realizat în acest caz), sau poate genera o întrerupere.

Pentru că motorul cu perii este foarte perturbator, se impun măsuri de protecție. Se recomandă conectarea unor condensatori de $6,8\mu\text{F}$ între bornele motorului și masă și de $0,33\mu\text{F}$ între bornele motorului. De asemenea se recomandă decuplarea tensiunii de alimentare cu condensatori de $50\mu\text{F}$, $6,8\mu\text{F}$ și 100nF în paralel, diode pentru eliminarea vârfurilor de tensiune de la fiecare bornă a motorului la masă și +5V.

Câteva elemente din programul de comandă sunt prezentate în continuare:

MOV CMOD,#06	:se stabilește intrarea pentru PWM de la pin exterior
MOV CCAPM0,#42H	:stabilește canalul 0 din arie în mod PWM
MOV CCAPM1,#42H	:stabilește canalul 1 din arie în mod PWM
MOV CCAP0L,#0H	:stabilește un factor de umplere 100% pentru canalul 0
MOV CCAP1L,#0H	:stabilește un factor de umplere 100% pentru canalul 1, ceea ce înseamnă că motorul stă.

După inițializarea acestor registre de comenzi trebuie pornit timerul, ceea ce se realizează prin poziționarea bitului 6 (CR) din registrul de control CCON.

SETB CR

Rotirea cu viteză maximă într-un sens se comandă cu secvența:

```
MOV CCAP0L,#0FFH
MOV CCAP1L,#0H
SETB CR
```

Rotirea cu viteză maximă în celălalt sens se comandă cu secvența:

```
MOV CCAP0L,#0H
MOV CCAP1L,#0FFH
ETB CR
```

Traductorul de viteză dă impulsuri canalului 4 programat în mod de captură. La primirea unui impuls pe P1.6 se încarcă conținutul timerului în registrele CCAP3H și CCAP3L, conținutul acestor registre fiind proporțional cu timpul scurs între 2 impulsuri ale traductorului.

MOV CMOD,#0	;se stabilește tactul timerului ca fosc/12
MOV CCAPM3	;canalul 4 în mod captură
SETB IP.6	;se setează întreruperea de la PCA la cea mai mare prioritate
MOV IE,0C0H	;se validează întreruperile de la PCA
SETB CR	;se pornește timerul

Când apare o întrerupere, în CCAP3L și CCAP3H apare conținutul timerului. O aplicație asemănătoare este descrisă de Jafar Modares în nota de aplicații AP-425, de la INTEL (www.questlink.com).

O altă aplicație a fost realizată la laboratorul de Interfațare. Cu un microcontroller 80C552 care dispune de un modul specializat PWM cu două ieșiri PWM, figura 5.42. Aceste două ieșiri pot fi folosite pentru comanda unui motor de curent continuu care poate fi comandat să se rotească în ambele sensuri cu viteză variabilă.. Frecvența de bază este asigurată de oscilatorul intern ($f_{osc}/2$) și de un prescaler de 8 biți, PWMP. Acesta poate fi încărcat prin

program cu o constantă de divizare, cu valori între 0 și FFh, corespunzătoare unui factor de divizare $1 \div 256$. Ieșirea prescalerului este conectată la intrarea unui numărator cu incrementare, PWMC, modulo 255 ($PWMC = 0 \div FEh$). Atât prescalerul cât și număratorul sunt utilizate în comun de către cele două canale PWM.

Conținutul număratorului este comparat continuu cu conținutul a două registre de 8 biți, PWM0 și PWM1, câte unul pentru fiecare canal ($PWMn, n=0,1$). Cât timp PWMC este mai mică decât valorile înscrise PWM0 și PWM1, ieșirile PWM0, respectiv PWM1 sunt forțate în "0" logic, în momentul în care PWMC devine egal cu una din valorile PWMn, n=0,1, ieșirea corespunzătoare este forțată în "1" logic, nivel care se menține atâta timp cât $PWMC \geq PWMn$.

Astfel, factorul de umplere pentru un ciclu complet de numărare (255 de perioade ale semnalului la ieșirea prescalerului) este controlabil prin conținutul registrelor PWM0 și PWM1, între 0 și 1, în incremenți de $1/255$. Frecvența semnalelor la ieșirile $\overline{PWM0}$ și $\overline{PWM1}$ este dată de relația:

$$f_{PWM} = \frac{f_{osc}}{2 \times (PWM0 + 1) \times 255}$$

Aceasta conduce la o frecvență cuprinsă între 92Hz și 23,5kHz, pentru $f_{osc}=12MHz$.

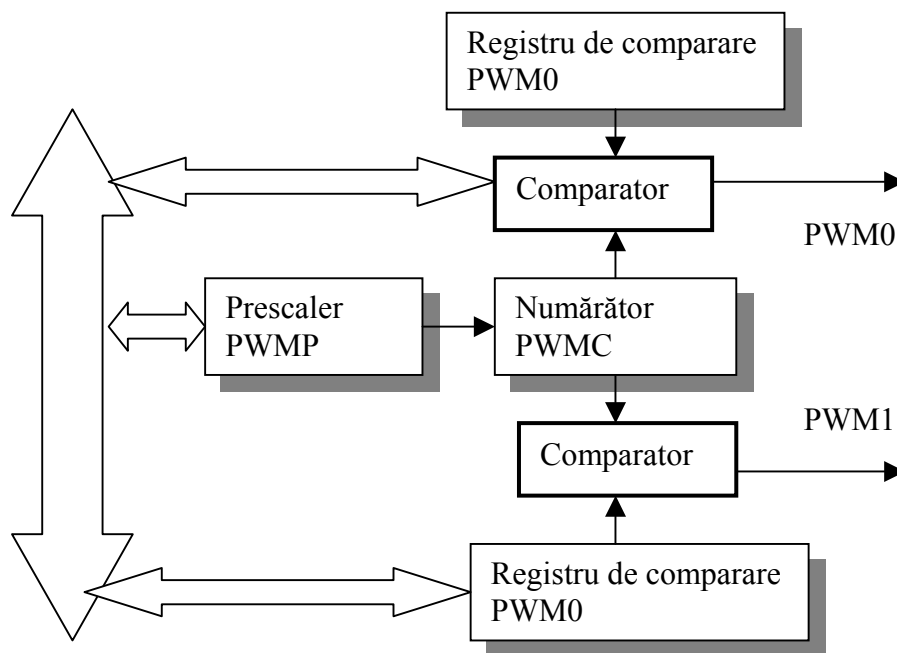


Figura 5.42: Blocul PWM

La modificarea conținutului registrelor de comparare PWMn, ieșirile corespunzătoare sunt actualizate imediat, nu după terminarea ciclului curent de numărare.

Cu acest tip de microcontroller a fost construit un dispozitiv de comandă a unui motor de c.c. Rotirea în ambele sensuri este realizată prin conectarea lui la un montaj în punte, figura 5.43.

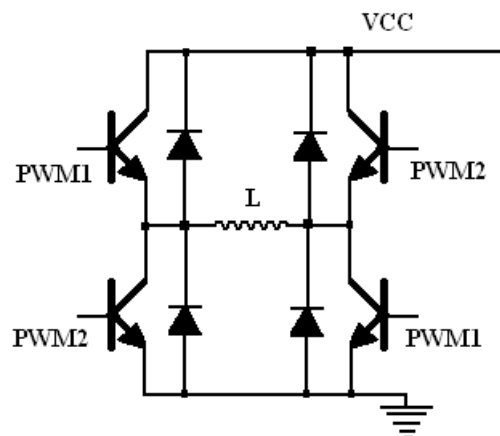


Figura 5.43: Schema amplificatorului de curent

Atunci când s-a testat această conectare și s-a folosit aceeași sursă de alimentare pentru partea de comandă și cea de putere s-a constatat că perturbațiile generate de motor influențează funcționarea plăcii.

Observație: ieșirile PWM sunt negațe, de aceea este nevoie ca valorile introduse în regiștrii PWMx trebuiesc complementate, pentru ca palierul activ sa fie proporțional cu ceea ce se dorește la ieșirile PWM.

```

Ex:      MOV A,#55          ; incarca in acumulator valoarea imediata 55H
          CPL A             ; se completeaza acumulatorul
                               ; deoarece iesirea PWM0 este negata
          MOV PWM0,A       ; încărcare registru de comparare
    
```

A fost realizat un program de test care comandă motorul astfel încât să se rotească până la viteza maximă posibilă și apoi să descrească turația până când se oprește.

; semnalul pornește ca fiind 100% din timp 1 logic, scade, apoi crește, operația
; se repetă.

```

PWMP      equ    0FEh
PWM0      equ    0FCh
PWM1      equ    0FDh
    
```

```

ORG 8000h
LJMP MAIN
ORG 8090h
MAIN:
    
```

```

      MOV A,#20          ; se incarca acumulatorul cu #XX pentru
                          ; frecventa semnalului de XkHz=11,059MHz/(2*(1+A)*255)

      MOV PWMP,A
      MOV A,#0FFh       ; in A este valoarea imediata 00h de start
      CLR SENS          ; pentru a stabili CRESTEREA/DESCRESTEREA turației
    
```

ROTESTE:

```
CPL A ; se complementeaza deoarece iesirea PWM este negata
MOV PWM0,A ; încărcare registru de comparare
MOV R2,#4
```

DELAY:

```
MOV R1,#0F0h
```

DELAY1:

```
MOV R0,#0FFh
DJNZ R0,$
DJNZ R1,DELAY1
DJNZ R2,DELAY
```

; verificare si apoi stabilirea noului sens daca este nevoie

```
CPL A
JB SENS,DIRECT
```

INVERS:

```
SUBB A,#10h
CJNE A,#0Fh,ROTESTE
SETB SENS ; se schimba sensul
SJMP ROTESTE
```

DIRECT:

```
CJNE A,#0FFh,CRESTE
CLR SENS ; se schimba sensul
SJMP ROTESTE
```

CRESTE:

```
ADD A,#10h ; factorul de umplere creste
SJMP ROTESTE
```

STOP:

```
SJMP $
```

BSEG AT 20H

SENS: DBIT 1 ; 1/0- factorul de umplere pozitiv creste/scade

END

5.6.3.Detectorul de stare zero

O aplicație de comandă a motoarelor poate să ofere surprize în faza de proiectare sau chiar de implementare. Un exemplu de astfel de surpriză este cea care a apărut la înlocuirea indicatoarelor de bord clasice auto cu indicatoare comandate cu motoare de curent continuu. Dacă cheia este decuplată în timpul mișcării mașinii sistemul electronic nu mai este alimentat și indicația de viteză, respectiv turație rămân la valorile din momentul scoaterii cheii. Pentru rezolvarea acestei probleme trebuie ca sistemul electronic să mai rămână alimentat un timp. Oricum, la mașinile noi după scoaterea cheii geamurile electrice se pot ridica, oglinzile se rabat etc. Pentru a ușura comanda motorului în acest caz au apărut microcontrollere care au un bloc intern de feedback.

Modulul de comandă al motoarelor din componența MC pe 32 de biți (MB91360) are integrat un bloc de detectare a stării rotorului¹. Schema bloc a blocului este dată în figura 5.44.

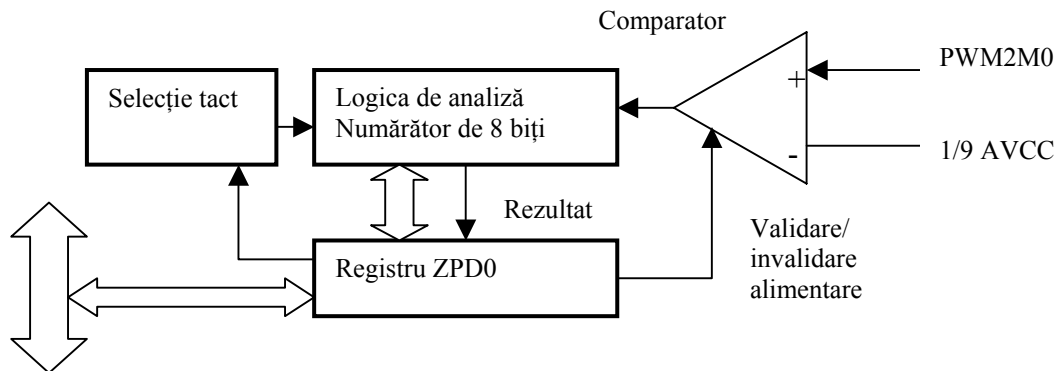


Figura 5.44: Schema bloc a detectorului de stare zero

Logica de analiză eșantionează ieșirea comparatorului cu tactul selectat în ZPD0. Comparatorul compară intrarea de la PWM2M0 cu tensiunea de referință și setează bitul de rezultat al comparării dacă tensiunea de intrare este mai mare decât tensiunea de referință. Rezultatul comparării devine 1 dacă toate eșantioanele (numărul lor este specificat în ZPD0) sunt 1. Ca urmare pinul PWM2M0 poate fi folosit pentru închiderea unei bucle de reacție de control a motorului.

Registrul ZPD0 (Zero Detect Register) conține: 3 biți selectează tactul (tact divizat cu 1,2,4,5,6 sau 8), un bit validează operarea detectorului de zero, 3 biți stabilesc numărul de eșantioane (1,2,3,4 sau 5), un bit validează/ invalidează alimentarea blocului analogic, un bit indică rezultatul comparării.

Bibliografie

- [1] http://www.celesco.com/?gclid=CLrzz_-A76ACFQO7ZwodlZrEHw, Traductori de poziție
- [2] <http://www.unimeasure.com/lx.htm>, Traductori de poziție
- [3] Ogrutan P., *Microcontrollere si controllere grafice Fujitsu*, Ed. Universitatii Transilvania Brasov, 2006, 182 pag, ISBN 973-635-621-3
- [4] Gerigan C., Ogrutan P., *Tehnici de interfațare*, Ed. Transilvania Brașov, 2000, 315p., ISBN 973-9474-94-2
- [5] Toacse Gh., Romanca M., Pană Gh., *Step transductor for PM hybrid stepping motor*, in Electric Power Components And Systems, Taylor & Francis INC, ISSN: 1532-5008 vol. 11/4, 1986, pp. 347-356

¹ Această funcție a controllerului este patent al Siemens VDO Automotive AG și poate fi folosită doar cu acordul proprietarului.