

6.UNELTE DE DEZVOLTARE

6.1. Mediul de dezvoltare Softune Workbench

Mediul de dezvoltare Softune Workbench este utilizat pentru toate familiile de MC pe 8, 16 sau 32 de biți și poate fi instalat sub sistemele de operare WINDOWS 9x, NT, 2000 și XP. În pachetul standard sunt incluse:

- Compilator ANSI C
- Macro asamblor
- Editor de legături
- Simulator
- Emulator
- Verificator C
- Analizor C

6.1.1.Crearea unui nou proiect

Există două moduri de a crea un nou proiect:

- 1) File->New->Workspace/ProjectFile
- 2) Se preferă însă varianta descrisă mai jos
 - a) Se copiază din directorul Examples subdirectorul Template care se redenumește cu numele proiectului pe care vrem să îl facem (de exemplu: Led). Se redenumesc fișierele următoare:

- Template.prj ->Led.prj
- Template.wsp->Led.wsp

- b) Se deschide cu un editor de texte fișierul Led.prj. Se caută linia

```
“[DirInfo]
```

```
PRJ=C:\Fujitsu\Sample\90340\template”
```

Această linie va fi completată cu calea în care se află proiectul. La fel se va completa și calea pentru WSP, după care se salvează și se închide fișierul.

```
[DirInfo]
```

```
WSP=C:\Fujitsu\Sample\90340\template\
```

```
to:
```

```
WSP=C:\Fujitsu\Sample\90340\led\
```

Fișierele conținute în directorul template/src

1.Start.asm

Se pot seta următoarele caracteristici în fișierul start.asm:

- **familia microcontrollerului;** În cazul nostru: #set FAMILY MB90300
- **modelul de memorie;** Se poate alege dintre tipurile small, medium, compact sau large. În funcție de setarea de memorie, funcția main() din fișierul main.c va fi apelată de instrucțiunea pe 16 biți CALL sau de instrucțiunea pe 24 biți CALLP. Este recomandat să se aleagă

pentru modelul de memorie automodelul care va folosi pentru apelul funcției main instrucțiunea CALLP(care funcționează întotdeauna).

- **manipularea constantelor;** se poate alege între manipularea RAM sau ROM a constantelor. Se recomandă alegerea manipulării AUTO a constantelor.
- **tipul și dimensiunea stivei;** este recomandat să se folosească system stack cu o dimensiune de 768 bytes.
- **bancul regiștrilor de uz general;** se recomandă să se folosească bancul 0
- **interfața pentru bibliotecile de nivel jos;** această opțiune trebuie să fie setată dacă se folosesc funcțiile stream-IO sau standard-IO(printf() sau fopen())
- **selecția clock-ului;** aici se setează factorul de ceas PLL. Implicit acesta este 4.
- **interfața magistralei externe și ROM mirror**
- **vectorii de reset;** așează adresa acestora la începutul codului
- **copierea codului RAM;** se alege între a copia sau nu în zona de RAM a unui segment special din cod(DTRANS – acesta poate fi folosit pentru software-ul de programare al flash-ului)

2.Vectors.c

- **nivelele de întrerupere;** sunt setate în interiorul funcției void InitIrqLevels(void), figura 6.1:

```
void InitIrqLevels(void)
{
/* ICRxx          shared IRQs for ICR */

    ICR00 = 7;      /* IRQ11
                    IRQ12 */
    ICR01 = 7;      /* IRQ13
                    IRQ14 */
    ICR02 = 7;      /* IRQ15
                    IRQ16 */
    ICR03 = 7;      /* IRQ17
                    IRQ18 */
    ICR04 = 7;      /* IRQ19
                    IRQ20 */
    ICR05 = 7;      /* IRQ21
                    IRQ22 */
    ICR06 = 7;      /* IRQ23
                    IRQ24 */
    ICR07 = 3;      /* IRQ25 timebased timer
                    IRQ26 */
    ICR08 = 7;      /* IRQ27
                    IRQ28 */
    ICR09 = 7;      /* IRQ29
                    IRQ30 */
    ICR10 = 7;      /* IRQ31
                    IRQ32 */
    ICR11 = 7;      /* IRQ33
                    IRQ34 */
    ICR12 = 7;      /* IRQ35
                    IRQ36 */
    ICR13 = 7;      /* IRQ37
                    IRQ38 */
    ICR14 = 7;      /* IRQ39
                    IRQ40 */
    ICR15 = 7;      /* IRQ41
                    IRQ42 */
}
```

Figura 6.1: setarea nivelelor de întrerupere

- **prototipurile funcțiilor de tratare a întreruperilor**, figura 6.2:

```

/*-----
  Prototypes
  Add your own prototypes here. Each vector definition needs is proto-
  type. Either do it here or include a header file containing them.
  -----*/

__interrupt void DefaultIRQHandler (void);
__interrupt void TimerInterrupt(void);

```

Figura 6.2: *prototipuri*

- **definirea vectorilor de întrerupere**; se face corespondența dintre numărul vectorului de întrerupere și funcția care tratează întreruperea, figura 6.3:

```

#pragma intvect DefaultIRQHandler 23 /* PPG 8/9/C/D */
#pragma intvect DefaultIRQHandler 24 /* PPG A/B/E/F */
#pragma intvect TimerInterrupt 25 /* Time Base Timer */
#pragma intvect DefaultIRQHandler 26 /* External interrupt 8-11 */
#pragma intvect DefaultIRQHandler 27 /* Watch Timer */
#pragma intvect DefaultIRQHandler 28 /* External interrupt 12-15 */
#pragma intvect DefaultIRQHandler 29 /* A/D-Converter */
#pragma intvect DefaultIRQHandler 30 /* I/O-Timer 0 / I/O-Timer 1 */

```

Figura 6.3: *definirea vectorilor de întrerupere*

- **funcția de tratare implicită a întreruperilor, DefaultIRQHandler**, figura 6.4:

```

/*-----
  DefaultIRQHandler()
  This function is a placeholder for all vector definitions. Either use
  your own placeholder or add necessary code here.
  -----*/

__interrupt
void DefaultIRQHandler (void)
{
    __DI(); /* disable interrupts */
    while(1)
        __wait_nop(); /* halt system */
}
//-----

```

Figura 6.4: *funcția de tratare a întreruperilor*

3.Main.c

Fișierul main.c conține funcția void main (void) care este executată imediat după fișierul start.asm. În fișierul main.c se scriu funcțiile care tratează întreruperile, care inițiază perifericele, etc., figura 6.5:

```

//-----main-----
void main(void)
{
    InitIrqLevels();
    __set_il(7); // allow all levels
    __EI();     // globally enable interrupts

    // initialize I/O-ports

    PDR0 = 0x00;
    DDR0 = 0xfc; // P00: SW2(INT8)  P01: SW3(INT9)

    PDR1 = 0x08; // SOT3 = 1
    DDR1 = 0xfb; // SIN3 = input

    PDR2 = 0x00;
    DDR2 = 0xff;

    PDR3 = 0xff;
    DDR3 = 0xff; // Set Port3 as output (7Segment Display)

    PDR4 = 0x08; // CAN TX = 1
    DDR4 = 0xfb; // CAN RX = input

    ADERS5 = 0; // Use Port 5 as I/O-Port
    PDR5 = 0x7f;
    DDR5 = 0xff; // Set Port5 as output (7Segment Display)

    PDR6 = 0x00;
    DDR6 = 0xff;

    initTimer(); // initialize time-based timer with the desired value
}

```

Figura 6.5: funcția void

4.MB90340.h, MB90340.asm

În aceste headere se definesc numele regiștrilor resurse ai microcontroller-ului. Numele acestor regiștrii sint aceleași cu numele folosite în Hardware Manual. Numele header-ului depinde de familia de microcontrollere folosită.

6.1.2.Simulatorul

Setarea simulatorului

Pentru setarea simulatorului se vor urma pașii descriși mai jos:
Click dreapta pe folderul debug din workspace, figura 6.6:

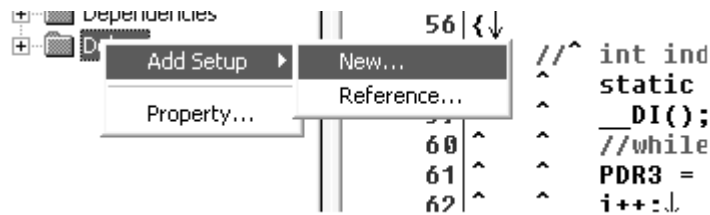


Figura 6.6: setarea simulatorului

Se dă un nume noii simulări (de exemplu simulare). Va apărea un wizard de setare a simulării. Se alege opțiunea “Simulator Debugger”, se apasă butonul “Next”, nu se alege nici un fișier Batch, se trece mai departe prin apăsarea butonului “Next”. Se bifează opțiunea “AutoLoad when starting Debug” iar în edit-box-ul “After” se va scrie “romconst.prc”. Se apasă butonul “Next”, se bifează opțiunea “Select All”, se apasă din nou butonul “Next” și apoi “Finish”.

Folosirea simulatorului

Pentru a începe simularea (pentru a intra în modul debug), se dă un dublu click pe numele simulării setate la pasul anterior sau se alege meniul Debug->Start Debug.

În fereastra de comandă se afișează constantele-mirror ale ROM-ului, figura 6.7:

```
>Set ROM-mirror memory map...
address      attribute
000000 .. 000137 read write
000138 .. 00018F code read write
000190 .. 000529 read write
00052A .. 00087F code read write
000880 .. 0008FF read write
000900 .. 0010FF code read write
001100 .. 003FFF undefined
004000 .. 00FFFF read
010000 .. FFFFFFFF undefined
FE0000 .. FFFFFFFF code read
>Copy ROMCONST for simulation...OK

-----
Use command "batch prc\romconst.prc" after each download
-----
```

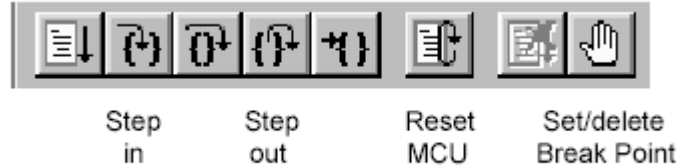
Figura 6.7: folosirea simulatorului

În figura 6.7 se arată că s-a copiat zona const din ROM începând cu bancul 0xFF în zona ROM mirror începând cu bancul 0x00.

Pentru a începe simularea se deschide fișierul main.c (dublu click pe numele acestuia în fereastra de workspace). Pentru a vedea atât codul sursă cât și codul assembler rezultat în urma compilării, se dă un click dreapta în fereastra main.c și se alege Mix display.

Pentru a rula programul în modul debugg, se folosesc următoarele butoane:

- Run Continously
- Step Over
- Execute until cursor position
- Stop Execution



Setările necesare simulării porturilor I/O

Deoarece simulatorul este un simulator pentru core, nu este posibilă simularea resursele I/O ca de exemplu UART. Este însă posibilă simularea întreruperilor și a porturilor I/O. Porturile I/O pot fi simulate folosind un fișier pentru scrie sau a citi date în momentul în care se accesează o adresă I/O. Pentru a simula întreruperile sau porturile I/O se folosește meniul Setup->Debug Environment->IOPort după ce s-a pornit debuggerul, figura 6.8:

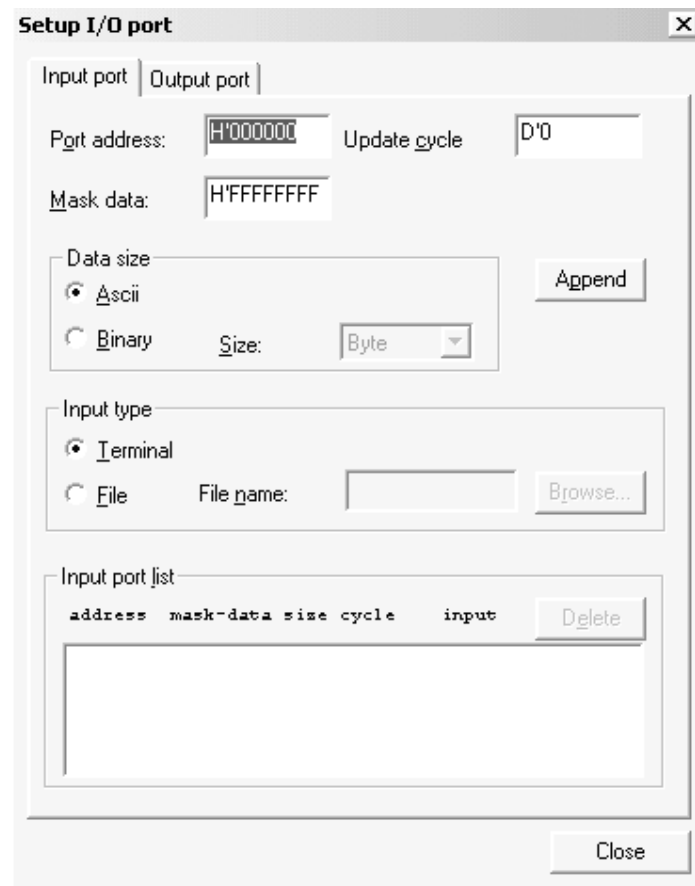


Figura 6.8: ecran pentru setarea porturilor I/O

Setările necesare simulării întreruperilor

Se alege meniul *Setup > Debug Environment > Interrupts* pentru a genera întreruperi la un interval stabilit de timp. Această facilitează simularea rutinelor de tratare a întreruperilor, figura 6.9:

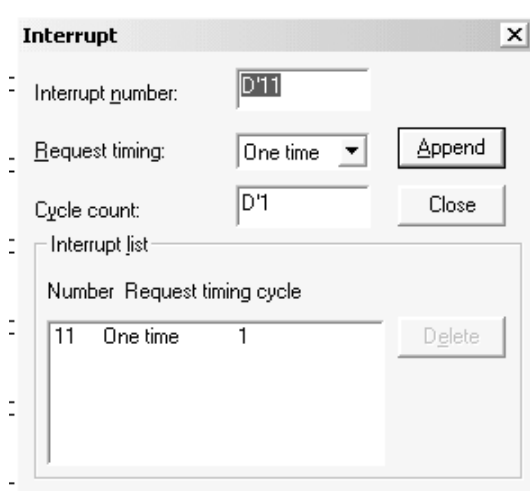


Figura 6.9: simularea întreruperilor periodice

6.2. Module de start și emulatoare

Laboratorul Fujitsu al Universității a fost echipat cu 2 tipuri de module de start, unul echipat cu MC pe 8 biți (Concerto) iar celălalt cu MC pe 16 biți (Dice Kit). Aceste module, așa cum le spune și numele sunt folosite pentru a încerca un program în scopul realizării unor aplicații simple. Modulele sunt echipate cu butoane pentru simularea întreruperilor, afișaje 7 segmente pentru semnalizare iar pinii MC sunt accesibili prin scoaterea lor la o matrice de pini. MC sunt lipite pe placă, ca urmare nu se pot încerca diferite variante de MC cu diferite resurse.

Modulul Concerto

Schema bloc a acestui modul este dată în figura 6.10:

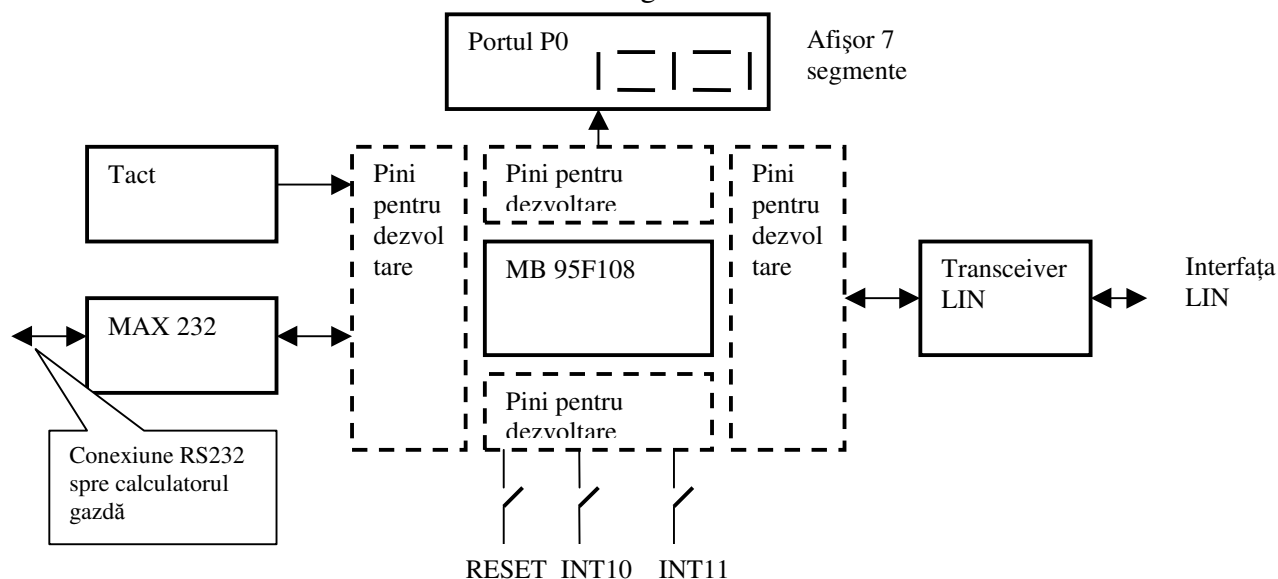


Figura 6.10: schema bloc a modulului CONCERTO

Interfața LIN se poate folosi direct, pe placă fiind montat un transceiver LIN. Celelalte resurse ale MC pot fi folosite prin conectarea circuitelor adiționale la pini pentru dezvoltare. Modulul Concerto arată ca în figura 6.11:



Figura 6.11: *modulul Concerto*

Modulul Dice Kit

Modulul Dice Kit este echipat cu un microcontroller pe 16 biți și are schema bloc dată în figura 6.13:

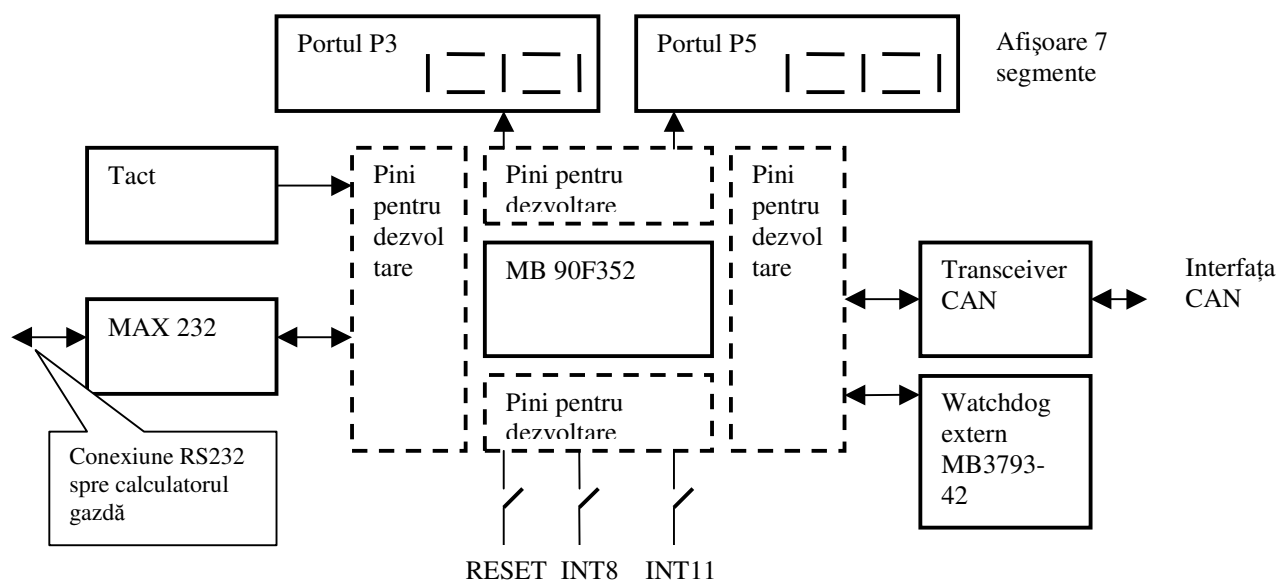


Figura 6.13: *modulul Dice Kit*

Ambele module conțin variante de MC cu Flash care pot fi programate prin RS232 de la calculatorul gazdă. Modulul Dice Kit conține un transceiver CAN și o interfață CAN precum și un watchdog extern care este descris în capitolul dedicat aplicațiilor auto. Aceste module sunt folosite atât la dezvoltarea de aplicații cât și la laborator pentru a demonstra ușurința cu care se pot realiza aplicațiile.

Modulul Dice Kit este prezentat în figura 6.14:



Figura 6.14: *modulul Dice Kit*

Emulatoare

Un pas înspre profesionalism în punerea la punct a aplicațiilor este folosirea emulatoarelor. În principiu se folosesc MC de evaluare care nu au memorie Flash proprie dar conțin de regulă toată periferia familiei și plăci de emulare prin care MC folosește memoria oferită de calculatorul gazdă. Fiecare familie are un MC de evaluare propriu, notat cu V, de exemplu:

Familia			MC de evaluare
MB90540G			MB90V540G
MB90440			MB90V440G
MB90385	MB90495G	MB90895	MB90V495G
MB90455			
MB90340	MB90860	MB90350	MB90V340
MB90360			
MB90330 MB90335			MB90V330
MB90470			MB90V470
MB90480			MB90V480
MB90580			MB90V580
MB90595G			MB90V595G
MB90390			MB90V390H
MB90800			MB90V800
MB95100			MB95V100
.....		

Un sistem de evaluare cu emulator pentru MC de tip 8FX conține:

- placă de evaluare pe care se pot monta MC de evaluare pe soclu
- Adaptor USB de conectare la calculatorul gazdă prin USB, calculatorul gazdă având în memoria proprie programul MC
- Software Softune Workbench

Un astfel de sistem este prezentat în figura 6.15:



Figura 6.15: emulator prin USB

Schema bloc de conectare a emulatorului este dată în figura 6.16:

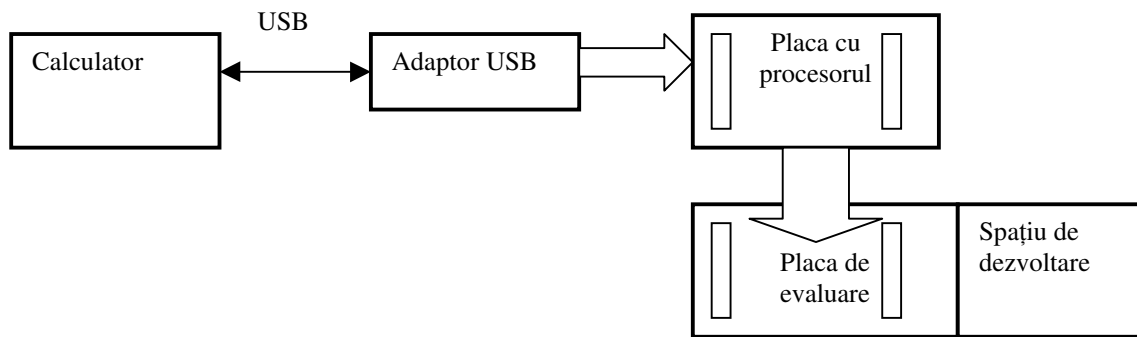


Figura 6.16: schema de conectare a emulatorului

Emulatorul prezentat în figurile 6.15 și 6.16 este modelul folosit la familia MB95100 iar MC de evaluare este modelul MB95V100-101 (alimentat la 3V) și MB95V100-103 (alimentat la 5V).

Emulatorul se folosește împreună cu mediul Softune Workbench, astfel:

- Se încarcă un program monitor în emulator care gestionează programul utilizator și răspunsul MC. Pentru aceasta se precizează legătura (USB), tactul (4MHz), tipul emulatorului, tipul MC.
- Se pot defini puncte de întreruperea programului prin:
 1. Specificarea valorii PC
 2. La execuția unei instrucțiuni care setează o adresă (max. 256 puncte de întrerupere)
 3. La încărcarea unei anumite valori la o adresă care se setează (max. 2)
- Execuția programului este urmărită și evoluția execuției este stocată într-un fișier.
- Programul se poate executa continuu sau pas cu pas.

Un emulator folosit pentru familia 16LX care suportă toate seriile de MC pe 16 biți la frecvența maximă de 25MHz este prezentat în figura 6.17. Tensiunea de alimentare poate fi între 2,7 și 5,5V, deci se poate folosi orice MC de evaluare. Ca software se poate folosi Softune Workbench. În plus față de emulatorul anterior se pot defini mai multe condiții pentru punctele de întrerupere, se pot măsura viteze de execuție între 2 puncte ale programului etc. Interfața cu calculatorul poate fi RS232, LAN (10Base-T, 100Base-TX) sau USB. Emulatorul este prevăzut cu un LCD.



Figura 6.17: emulator

Un emulator pentru familia 16LX cu preț mai mic este arătat în figura 6.18:

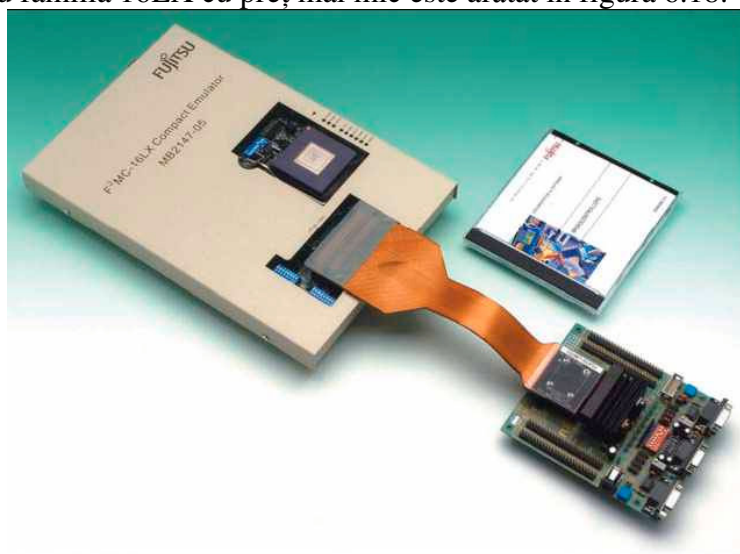


Figura 6.18: emulator low cost

Emulatorul dispune de mai puține variante de conectare la calculatorul gazdă (USB și RS232) și suportă o gamă mai redusă de MC.

Plăci de evaluare

Aceste plăci pot fi folosite în aplicații ca plăci independente, ca și modulele de start. Datorită faptului că sunt prevăzute cu un soclu de MC ele pot fi folosite și ca sisteme destinație pentru emulatoare. Emulatorul se conectează cu un cablu special și emulează diferitele tipuri de MC, oferind o imagine utilizatorului despre diferențele între diferitele tipuri de MC din familie. Placa de evaluare furnizează emulatorului tensiunea de alimentare, tactul și Reset-ul.

În figura 6.19 este prezentată placa de evaluare FLASH CAN 64P-M09-V2 pentru MC de tip MB90495, MB90560, MB90565, MB90460.



Figura 6.19: *placa de evaluare*

Există plăci de evaluare și pentru alte familii:

FLASH CAN 48P pentru MB90360, MB90385, MB90455, MB90890

FLASH CAN 64P-350 pentru MB90350

FLASH CAN 100P-340 pentru MB90340

FLASH CAN 120-390 pentru MB90390

Astfel de plăci de evaluare și emulatoare au fost construite și pentru MC pe 32 de biți. O placă de evaluare pentru MC din familia MB91360 este dată în figura 6.20:

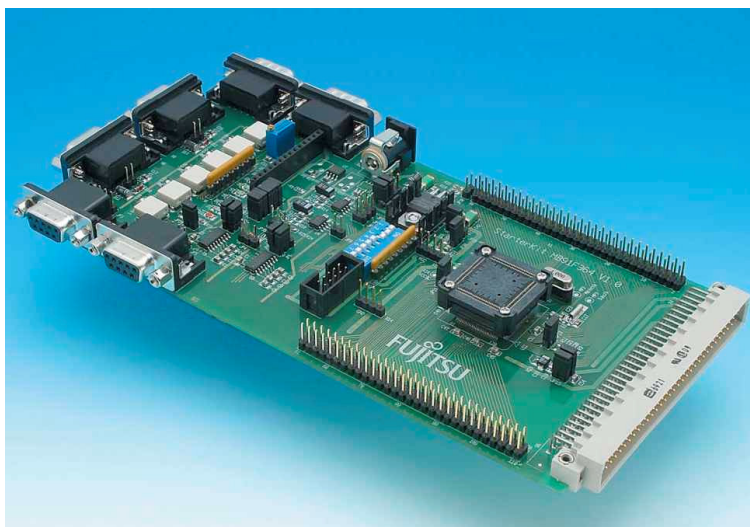


Figura 6.20: *placă de evaluare pentru MB91360*

Placa poate funcționa independent sau cu un emulator. Pe placă sunt montate transceivere pentru CAN și UART.

7.APLICAȚII AUTO

Aplicațiile auto au câteva particularități care trebuie considerate atunci când se alege un microcontroller care să gestioneze o astfel de aplicație. Câteva dintre aceste particularități sunt:

- Respectarea legislației specifice
- Posibilitatea de lucru cu nivele digitale auto
- Consum mic de energie
- Generare mică de perturbații electromagnetice
- Siguranță mare de funcționare
- Microcontrollerul care gestionează aplicațiile trebuie să aibă integrate modulele specifice aplicațiilor.
- Temperatura de lucru

Consumul mic de energie și generarea unui spectru redus de perturbații sunt legate de generarea și folosirea tactului.

7.1.Legislație

Directiva RoHS (Restriction of Hazardous Substances in Electrical and Electronic Equipment) a fost adoptată în Comunitatea Europeană (europa.eu.int). Această directivă cu numărul 2002/95/EC din 27 ian. 2003 interzice utilizarea anumitor substanțe în echipamentele electronice utilizate în industria auto. Aceste substanțe sunt Cadmiul (Cd), plumb (Pb), mercur (Hg), crom hexavalent (Cr), și doi compuși ai bromului, polibrominat bifenil (PBB) și polibrominat bifenil eter (PBDE). Procentul în care apar aceste substanțe trebuie să fie mai mic de 0,1%.

AEC-Q100 este un document realizat de Automotive Electronics Council care definește câteva condiții pe care trebuie să le îndeplinească circuitele integrate pentru a fi acceptate în industria auto (www.aecouncil.com). Se definesc teste în regim de stress (teste accelerate) care, dacă sunt trecute garantează circuitelor un anumit nivel de fiabilitate în aplicații.

Se definește gama de temperaturi de funcționare a circuitelor astfel:

Denumire categorie	Gama de temperaturi de lucru în grade Celsius
Grad 0	-40 până la +150
Grad 1	-40 până la +125
Grad 2	-40 până la +105
Grad 3	-40 până la +85
Grad 4	0 până la +70

Conform cu declarațiile producătorului de încadrare a temperaturii de lucru într-o anumită categorie, testele se execută în acele condiții de temperatură.

Câteva dintre testele la care sunt supuse circuitele:

- Variația temperaturii
- Variația umidității
- Stocarea la temperaturi mari

- Străpungerea izolației
- Caracteristici la lipire
- Descărcări electrostatice
- Compatibilitate electromagnetică
- Șocuri mecanice
- Accelerații
- Vibrații
- Torsiuni
- Timp de memorare și număr de programări pentru memoria FLASH
- Rata de defectare timpurie

Fiecare tehnologie (CMOS, NMOS, TTL etc.) se testează cu teste diferite. De asemenea la capsularea în plastic sau ceramică se folosesc teste diferite. Pentru anumite teste există standarde, de exemplu testarea la compatibilitate electromagnetică se face conform SAE J1752/3, alte teste folosesc norme militare, MIL-STD-883 iar pentru unele teste se definesc condiții specifice, descrise în AEC-Q100.

Rata de defectare timpurie se determină printr-o încercare accelerată, supunând probele la cicluri de suprațemperaturi urmate de reveniri la temperatura camerei, un timp de 48 de ore la temperaturile maxime admise conform gradului declarat de producător sau 24 de ore la temperaturi mai mari (cu 25 de grade pentru grad 0 și cu 20 de grade mai mari pentru grad 1, 2, 3 și 4).

7.2.Nivele logice auto

În plus față de majoritatea microcontrollerelor, cele produse de Fujitsu admit pe lângă nivelele logice obișnuite (TTL și CMOS) și nivele logice auto, la care V_{IL} este $0,5 \times V_{CC}$. Modul de lucru cu nivele auto mărește siguranța în funcționare deoarece la autovehicule nivelul masei de semnal este destul de variabil și are uzual un decalaj de nivel a 1V deoarece toate circuitele se închid prin același circuit de masă. Lucrul cu nivele auto poate fi selectat soft pentru anumiți pini de I/O. Datele de catalog pentru nivelele de intrare arată importanța nivelelor logice auto pentru mărirea imunității la perturbații. În ipoteza unei tensiuni de alimentare $V_{CC}=5V$ tabelul următor arată nivelele admise la intrare pentru 3 configurații ale pinilor de intrare:

Tip de nivel pentru nivel High la intrare	Plaja admisă în funcție de tensiunea de alimentare	Plaja admisă (Volt)
Nivel CMOS cu histerezis	între $0,8 V_{CC}$ și $V_{CC} + 0,3V$	4V-5,3V
Nivel TTL	între 2V și $V_{CC} + 0,3V$	2V-5,3V
Nivel auto	între $0,8 V_{CC}$ și $V_{CC} + 0,3V$	4V-5,3V

Tip de nivel pentru nivel Low la intrare	Plaja admisă în funcție de tensiunea de alimentare	Plaja admisă (Volt)
Nivel CMOS cu histerezis	între $V_{SS} - 0,3V$ și $0,2V_{CC}$	-0,3V până la 1V
Nivel TTL	între $V_{SS} - 0,3V$ și 0,8V	-0,3V până la 0,8V
Nivel auto	între $V_{SS} - 0,3V$ și $0,5V_{CC}$	-0,3V până la 2,5V

Dacă pentru nivel High la intrare nivelele auto au aceeași margine de imunitate ca și nivelele CMOS cu histerezis, la nivel Low de intrare marginea de imunitate este mult mai mare, nivelul maxim admis pentru 0 logic fiind 2,5V față de 1V la CMOS cu histerezis și 0,8V la TTL.

7.3.Tactul

Avantajele microcontrollerelor Fujitsu care sunt determinate de structura modului de generare și control al tactului sunt:

- multiplicarea internă a tactului cu buclă PLL
- existența tactului dual
- existența tactului de rezervă
- modulația tactului

Obținerea tactului intern prin **multiplicarea tactului extern cu buclă PLL** are 2 avantaje majore:

- Tactul extern poate fi de valoare mai mică, ceea ce înseamnă o generare de perturbații de valoare mai mică.
- Este posibilă alegerea factorului de multiplicare prin software, deci pentru aplicații care acceptă viteze mai mici se obține un consum mai mic de la sursa de alimentare.

Operarea cu tact dual duce mai departe al doilea avantaj. Astfel, pentru modelele care au subtact (un tact de valoare mult mai mică decât tactul principal), la aplicații de viteză redusă se poate obține un consum de curent de la baterie cu câteva ordine de mărime mai mic. De exemplu la operare normală curentul consumat este de 50mA cu tactul principal și 300μA cu subtact, iar în mod adormit (Sleep Mode) 25mA cu tact principal și 30μA cu subtact.

Pentru mărirea siguranței în funcționare, anumite modele de MC au integrat un **tact de rezervă** RC integrat și un supervisor de tact. Atunci când supervisorul de tact detectează dispariția tactului principal sau a subtactului comută pe tactul de rezervă (cu frecvența, precizia și stabilitatea mai mică decât tactul principal sau subtact) și MC execută un program de urgență (salvare date, comunicarea stării de defect etc.).

Modulația tactului are rolul de a reduce interferențele electromagnetice (EMI) prin împrăștierea spectrului semnalului de tact. Metoda este implementată la MC pe 32 de biți și la cele pe 16 biți. Modulația se aplică tactului generat de bucla PLL, astfel încât unele interfețe care generează intervale de timp precise (timer de bază, Watchdog sau CAN) nu pot avea ca tact tactul modulat. Frecvența dată de tactul PLL este F_0 iar spectrul are un vârf corespunzător acestei frecvențe. Modularea tactului înseamnă variația frecvenței /fazei între 2 limite, ceea ce micșorează vârful spectrului.

Tactul de la bucla PLL poate fi modulat în 2 feluri:

- modulație în fază, semnalul modulator fiind un semnal triunghiular
- modulație în frecvență, semnalul modulator fiind un semnal pseudo aleator.

Modulatorul poate fi validat sau invalidat cu un registru de comandă, se poate alege tipul de modulație și se pot programa limitele de variație ale frecvenței sau fazei. În general comportarea cea mai bună din punct de vedere EMI se obține la variația maximă a frecvenței /fazei tactului, dar aceasta nu este o regulă și în cele mai multe cazuri este nevoie de încercări experimentale.

MC pe 16 biți pot avea un modulator de tact prin modulație în fază, de exemplu MB90340 și MB90350 (descriși la MC pe 16 biți) sau un modulator cu modulație în fază sau frecvență la MB90390. MC pe 32 de biți au un modulator în frecvență (cel de la MB 91360 este descriși la MC pe 32 de biți). În general modulația în frecvență are performanțe mai bune din punctul de vedere al EMI.

Moduri de funcționare cu economie de energie

Reducerea curentului consumat este o prioritate pentru toate echipamentele electronice actuale. Reducerea puterii consumate de microcontrollere se poate realiza în două moduri:

- reducerea frecvenței tactului
- moduri de funcționare cu economie de energie

Modurile de lucru cu economie de energie ale MC sunt:

1. Mod de operare normal (cu tact principal, subtact sau tact PLL)

2. Mod de operare intermitent (tact intermitent, subtact intermitent sau tact PLL intermitent)(nu se aplică la MC pe 8 biți). În acest mod de lucru tactul se oprește după ce CPU execută o instrucțiune. Numărul de tacte lipsă poate fi programat și poate fi 0, 8, 16 sau 32. Tactul intermitent nu se aplică și interfețelor interne.

3. Mod Standby care poate fi:

- Mod adormit (Sleep Mode), se oprește tactul la CPU și timerul de gardă.
- Mod timer de bază, (Timebase Timer Mode) tactul este furnizat la timerul de bază. Rămân în funcțiune timerul și întreruperile externe.
- Mod de observare (Watch Mode), tactul principal este oprit. Este un mod de *standby* în care MC funcționează cu subtact.
- Modul STOP (Stop Mode) toate tactele sunt oprite. Funcționează doar întreruperile externe

În modurile cu economie de energie se poate programa ca pini de ieșire să treacă în înaltă impedanță sau să păstreze ultima valoare logică. Ieșirea din aceste moduri se poate face printr-un RESET sau o cerere de întrerupere.

7.4.Elemente de mărire a siguranței de funcționare. Circuit de monitorizare a tensiunii de alimentare și watchdog MB 3793-42

Aplicațiile auto solicită o siguranță de funcționare mărită datorită efectelor tragice pe care le poate avea un defect în partea electronică. Modulele integrate în microcontroller care contribuie la mărirea siguranței în funcționare sunt:

- Supervizorul de tact
- Protecția la scăderea tensiunii de alimentare (subtensiune)
- Ceasul de gardă (Watchdog)

Supervizorul de tact generează un RESET dacă oscilațiile se opresc în mod anormal. După RESET funcționarea MC se realizează cu un tact intern RC (opțiune valabilă doar la unele din modelele alimentate la 5V). Supervizorul comută tactul intern dacă nu este detectat un front al tactului principal timp de 4 cicluri RC sau nu este detectat un front al subtactului timp de 32 de cicluri RC.

Protecția la scăderea tensiunii de alimentare constă în monitorizarea tensiunii de alimentare și generarea unui RESET dacă tensiunea scade sub 2,6V (la variantele de 3V) sau sub 4V (la variantele alimentate cu 5V). Circuitul de detecție este realizat cu un comparator și o referință. Dacă tensiunea revine peste 2,7V (4V) RESET-ul nu mai este menținut. Detecția este operațională în orice mod de lucru dar partea analogică a detectorului se poate deconecta în modurile de lucru cu economie de energie. Deoarece semnalul de subtensiune de la partea analogică poate surveni în orice moment și pentru a evita stările instabile, RESET-ul devine activ după câteva impulsuri de tact.

Watchdog-ul supraveghează funcționarea corectă a programelor. Se activează scriind un bit într-un registru de comandă și se stabilește perioada de timp. După ce a fost activat watchdog-ul trebuie inițializat periodic prin program re-scriind bitul de activare. Dacă în perioada de timp programată nu apare un impuls de re-activare Watchdog-ul generează un RESET. Watchdog-ul mai poate fi inițializat și de Reset sau de tranzițiile în standby: mod adormit, mod stop, mod timer de bază, mod watch. Oprirea watchdog-ului se poate face numai prin Reset. Acest modul are rolul de a proteja aplicația în ipoteza că MC nu mai rulează programul util (de exemplu se află într-o buclă infinită) și în consecință nu mai controlează aplicația. Un RESET în această situație readuce MC la condițiile inițiale și se poate astfel relua controlul asupra aplicației.

În cazul în care MC se defectează nici unul dintre aceste module interne nu pot proteja aplicația. De aceea se pot folosi componente externe. Un exemplu de componentă externă folosită ca element de protecție este circuitul de monitorizare a tensiunii de alimentare care conține și un watchdog.

Circuit de monitorizare a tensiunii de alimentare și watchdog MB 3793-42

Circuitul de monitorizare a tensiunii de alimentare are schema bloc dată în figura 7.1:

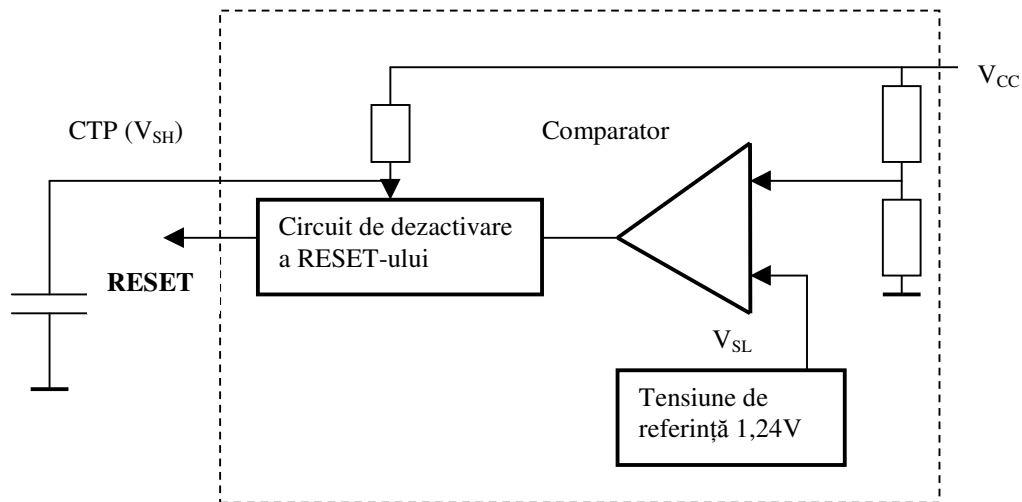


Figura 7.1: Schema bloc a monitorizării tensiunii de alimentare din MB 3793-42

Comparatorul compară o fracțiune a tensiunii de alimentare cu tensiunea de referință. La scăderea tensiunii sub pragul V_{SL} se generează un RESET dacă scăderea tensiunii este mai lungă de o microsecundă. Terminalul CTP controlează revenirea la starea inactivă a RESET-ului, astfel încât dacă tensiunea pe CTP revine peste pragul V_{SH} RESET-ul se inactivează.

Observație

Se pot comanda circuite cu alt prag de de tensiune, de la 4,9V la 2,4V în pași de 0,1V cu precizia de 2,5%.

Watchdog-ul poate monitoriza 2 impulsuri te tact CK1 și CK2, schema bloc fiind dată în figura 7.2:

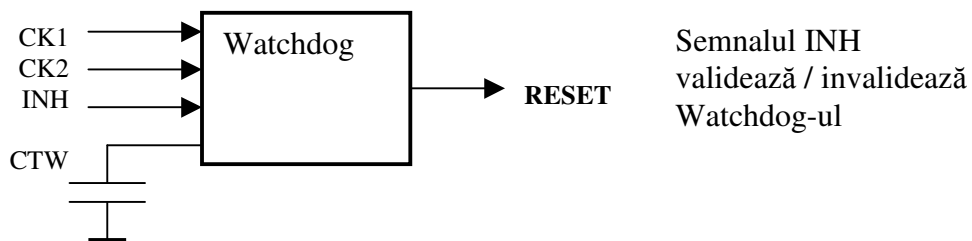
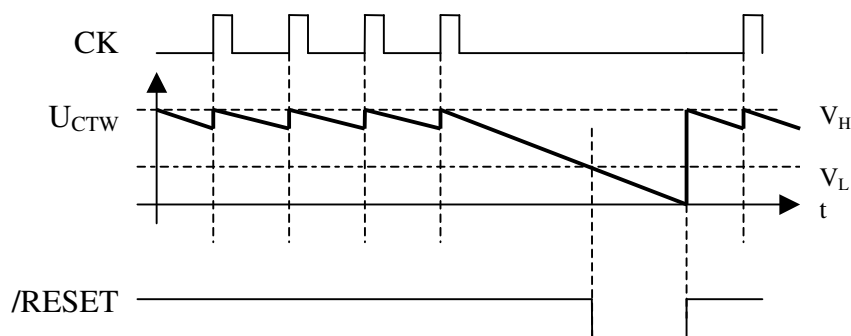


Figura 7.2: Schema bloc a Watchdog-ului din MB 3793-42

Funcționarea ca Watchdog extern este asemănătoare cu cea a Watchdog-ului intern. MC trebuie să trimită impulsuri cu o anumită periodicitate prin program, la un pin extern care se cuplează la MB 3793 la una din intrările CK. Dacă impulsurile nu vin cu periodicitatea așteptată se generează un RESET.

Funcționarea ca Watchdog poate fi urmărită în diagrama de timp din figura 7.3:



Timpul de întârziere poate fi stabilit cu condensatorul cuplat la CTW.

Figura 7.3: Diagrama de timp pentru funcționarea ca Watchdog a MB 3793-42

Funcționarea ca monitor al tensiunii de alimentare poate fi urmărită în diagrama de timp din figura 7.4:

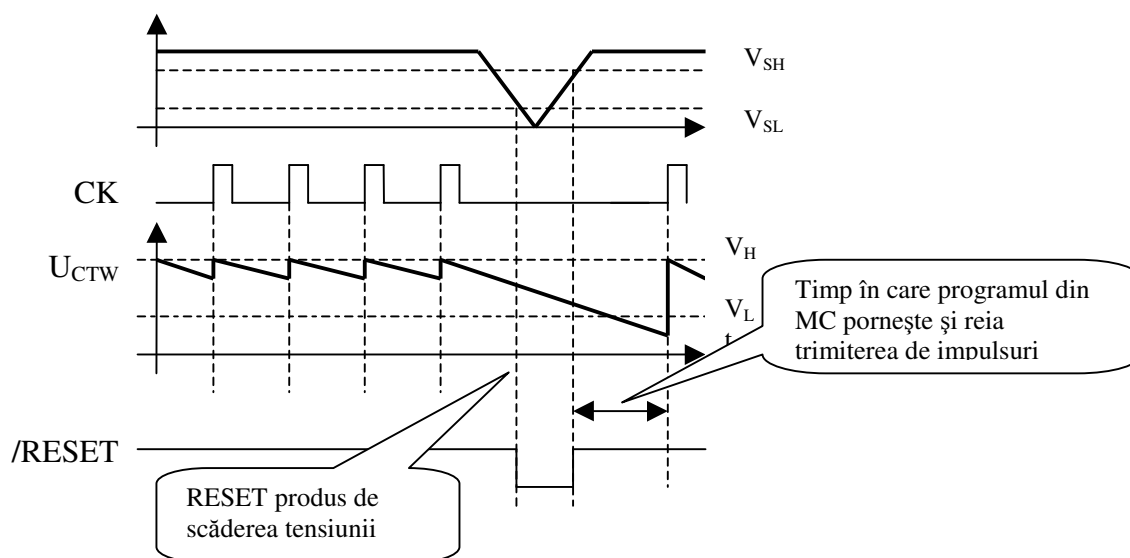


Figura 7.4: Diagrama de timp pentru funcționarea ca supervisor al tensiunii de alimentare

Scăderea tensiunii de alimentare sub prag dtermină generarea unui RESET. La revenirea tensiunii (depășirea pragului superior) RESET-ul se inactivează. Dacă pentru inactivarea RESET-ului s-ar aștepta și impulsurile pentru Watchdog ar apare o situație imposibilă: impulsurile nu sunt generate pentru că MC este în RESET și MC este în RESET pentru că nu vin impulsurile. De aceea RESET-ul se inactivează la trecerea peste prag a tensiunii de alimentare, indiferent de situația în care este Watchdog-ul.

7.5. Interfețele utilizate în domeniul auto integrate în MC

Modulele electronice care echipează un autovehicul au câteva sarcini care pot fi grupate în următoarele categorii:

- Preiau mărimi neelectrice prin intermediul senzorilor, prelucrează valorile electrice după conversia analog numerică, comandă elemente de execuție și eventual afișează rezultate.
- Comunică între ele pentru a se informa reciproc despre evenimentele din sistem.

Interfețele care echipează MC utilizate în auto trebuie să îndeplinească în principal câteva condiții importante:

- Siguranța în funcționare (pentru că la viteza de rulare o eroare poate duce la pierderea de viață omenești)
- Consum redus de energie (pentru că alimentarea se face de la baterie)
- Pentru comunicații între module trebuie folosite cât mai puține fire (cablajele sunt scumpe și nesigure).

Toate interfețele au fost tratate pe larg în capitolul 5. În acest paragraf vor fi amintite interfețele care sunt des folosite în industria auto.

Interfețe pentru prelucrarea mărimilor neelectrice

Toate MC pe 8, 16 și 32 de biți conțin în structură convertoare ADC, minimum 8 canale de conversie și maximum 24 de canale, cu aproximări succesive având rezoluția maximă de 10 biți. Toate aceste canale pot fi folosite și cu o rezoluție de 8 biți. Este de remarcat modul de funcționare care permite conversii continue și baleierea unui grup de canale de achiziție în mod continuu. Grupul de canale poate fi definit software. De asemenea achiziția poate fi declanșată software, printr-un semnal extern sau periodic, determinat de un timer. Timpul de conversie de câteva microsecunde face posibilă achiziția oricăror semnale din domeniul auto. Precizia și acuratețea rezultatelor este îmbunătățită de posibilitatea de programare a timpilor de eșantionare și de menținere. Transferul de date poate fi programat sau în întreruperi la familia pe 8 biți, programat, prin întreruperi sau EI²OS la familia pe 16 biți și programat, prin întreruperi sau DMA la familia pe 32 de biți.

Unele MC sunt echipate cu convertoare DAC, de regulă 2 canale de 10 biți pentru a putea comanda sisteme analogice.

Interfețe pentru comanda elementelor de execuție și afișare

Multe din elementele de execuție sunt construite pe baza motoarelor pas cu pas și mai puține pe baza motoarelor de curent continuu. Motoarele pas cu pas oferă precizie de poziționare și simplitate de comandă la costuri mici. Astfel, multe MC sunt echipate cu interfețe specializate pentru comanda motoarelor pas cu pas și a celor de curent continuu. Pot fi comandate atât motoarele pas cu pas unipolare cât și cele bipolare. Comanda fazelor cu un semnal PWM face posibil controlul precis al curentului prin înfășurare. Conectarea unei înfășurări între 2 module de control permite comanda cu curenți în ambele sensuri în înfășurare, deci o comandă bidirecțională pentru un motor de curent continuu. Driverele integrate permit comanda înfășurărilor cu curent mare, așa încât motoarele de mică putere pot fi conectate direct la MC. De exemplu indicatoarele de bord (viteză, turație) sunt realizate actual cu motoare pas cu pas.

Multe elemente care trebuie afișate sunt mai bine acceptate dacă sunt afișate pe LCD. Multe MC din familiile pe 8, 16 și 32 de biți conțin interfețe pentru comanda unui LCD în mod caracter. Interfețele conțin de regulă 4 ieșiri comune și 40 de ieșiri pentru segmente. Comanda poate fi realizată prin multiplexare și cu mai multe nivele de tensiune.

La MC care echipează bordul auto este importantă semnalizarea auditivă pe lângă cea vizuală. De aceea unele MC au integrată o interfață de generare sunete, cu posibilitatea reglării volumului și posibilitatea generării tonurilor cu volum descrescător.

Interfețe pentru comunicații

1.LIN (Local Interconnect Network) este o interfață care realizează o comunicație serială asincronă punct la punct pe un singur fir. Rata maximă de transfer este de 4 MBd și distanța de transmisie, în funcție de transceiverul folosit este de câțiva metri. Unul dintre circuitele conectate la LIN trebuie să fie master, iar celelalte slave. Masterul generează un nivel *space* pe perioada 13-16 biți urmat de un cuvânt 55H. Ca și slave, interfața LIN detectează trecerea liniei în *space* și calculează

viteza de transmisie din cuvântul 55H recepționat. Ca avantaje se poate menționa simplitatea interfeței, ușurința de programare, viteza mare de transfer, transferul pe un singur fir. Distanța de transfer este suficientă pentru aplicații în interiorul autovehiculului. Un dezavantaj major este lipsa siguranței în funcționare, nu se face nici măcar verificarea corectitudinii transferului cu bit de paritate, de aceea aplicațiile trebuie să se rezume la aplicații care nu afectează siguranța autovehiculului.

2.CAN (Controller Area Network) este o magistrală serială introdusă de Robert Bosch GmbH în 1986 pentru comunicația de date între 3 module electronice pentru automobilele Mercedes, deoarece interfața UART nu putea lucra decât punct la punct. Magistrala CAN a fost dedicată industriei auto dar s-a răspândit rapid și în alte domenii: automatizări, sisteme integrate etc. Transmisia se face pe 2 fire, half duplex, cu mare viteză (până la 1MBps) pe distanțe de ordinul zecilor de metri. Arbitrarea magistralei se face prin detectarea coliziunilor iar un sistem complex și eficient de detectare și tratare a erorilor face ca magistrala să fie extrem de sigură. Fiecare interfață CAN conține câte un registru de erori pentru recepție și transmisie care se incrementează cu 1 la o eroare minoră și cu 8 la o eroare gravă. Un număr de erori mare face ca interfața să fie exclusă din comunicație. Transmisia se realizează cu cadre și fiecare mesaj conține un identificator care arată prioritatea mesajului. Interfețele receptoare recepționează toate mesajele și le rețin pe acelea cu prioritate mare sau cele care le sunt destinate.

Avantajele nete pe care le are CAN față de LIN au dus la răspândirea CAN, cu toate că prețul unui MC cu CAN este mai mare decât al celui cu LIN. Prețul mai mare al CAN este justificat de suprafața mare de siliciu ocupată de această interfață, figura 7.5:

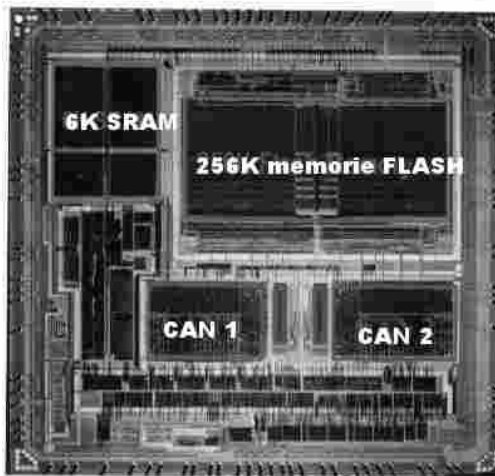


Figura 7.5: chip de MC echipat cu 2 interfețe CAN (sursa Fujitsu Microcontroller Seminar, 2003)

Se observă comparativ suprafața ocupată de cele 2 interfețe CAN față de memorie. Multe MC sunt echipate cu 2 canale CAN pentru a putea lucra în același timp cu CAN High Speed și Low Speed.

Au fost standardizate mai multe tipuri de niveluri fizice de transfer pentru CAN. Un MC cu interfață CAN poate lucra în oricare dintre aceste standarde fiind programat corespunzător și având adăugat în exterior transceiverul CAN potrivit. Standardele sunt:

ISO 11519 (CAN Low Speed) standardizează transmisia diferențială pe 2 fire. Cablul este torsadat, ecranat sau neecranat, cu impedanța caracteristică de $120\Omega \pm 10\%$. La capătul liniei nu este nevoie de rezistențe de adaptare pentru că viteza maximă de transfer de 125kbps nu dă naștere la reflexii pe linie.

ISO 11898 (CAN High Speed) se folosește același tip de cablu, dar linia se adaptează cu rezistențe de 120Ω pe fiecare fir. Lungimea de transmisie depinde de rata de transfer. Astfel la 40 metri se poate obține 1Mbps iar la 6km rata de transfer nu poate depăși 10kbps.

ISO 11898-3 (CAN High Speed tolerant la erori) este folosit îndeosebi în industria auto. Linia este scurtă de aceea nu este nevoie de adaptare, iar viteza maximă este 125kbps. Ca urmare topologia poate fi și alta decât cea liniară iar consumul de putere este mai redus. Este de asemenea posibilă transmiterea asimetrică a datelor (pe un singur fir) dacă o linie se defectează. Pentru acest standard se folosesc transceivere speciale cu comutatoare pentru transmisia simetrică sau asimetrică.

SAE J2411 standardizează transmisia CAN pe un singur fir și asigură o viteză de 83,3kbps (High Speed) și 33,3kbps (Low Speed). Transferul de date este dedicat pentru zone mici, cum ar fi interiorul motorului. Firul de transmisie poate fi necranat.

3.Flex Ray (www.flexray.com)

Evoluția automobilelor a dus la creșterea numărului de senzori, elemente de execuție și sisteme de control, care a dus mai departe la nevoia unui transfer de date mai rapid și cu toleranță la defecte. Aceste cerințe sunt îndeplinite de noua magistrală Flex Ray care urcă viteza de transfer la 10Mbps. Fujitsu este membru al consorțiului Flex Ray din 2002 și primul kit de evaluare a fost lansat de Fujitsu în 2005.

O schemă bloc a unui sistem complex de control auto este dată în figura 7.6, în care se observă că elementele de confort pot fi comandate printr-o magistrală mai simplă (LIN sau cel mult CAN) iar elementele mai complexe cum ar fi motorul sau frânele se comandă prin magistrale mai complexe (Flex Ray sau cel puțin CAN).

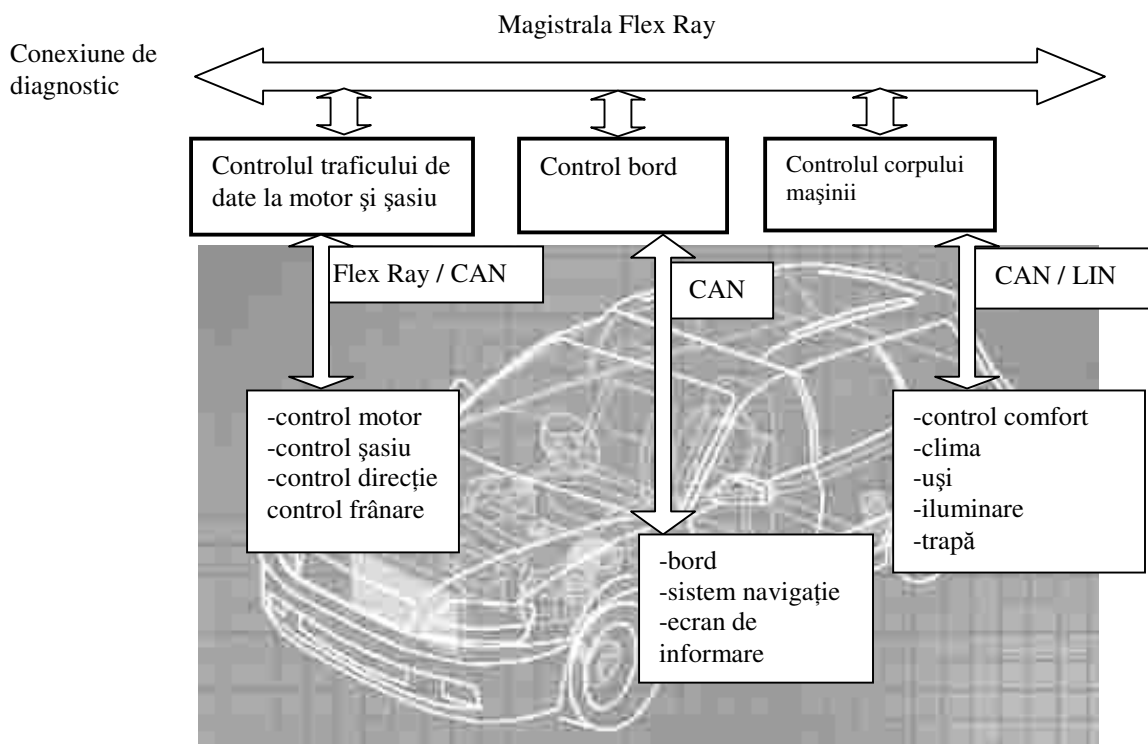


Figura 7.6: arhitectura unui sistem complex de control auto

7.6. Temperatura de lucru

La familiile dedicate domeniului auto gama de temperaturi este între -40°C și 105°C . Microcontrolerele pot fi folosite și în gama -40°C până la 150°C , dar se recomandă să se ia legătura cu Fujitsu pentru precizări. Temperatura de stocare este între -55°C și 150°C . Un grafic al temperaturii de lucru admise în funcție de frecvență (pentru modelul MB90350) este dat în figura 7.7:

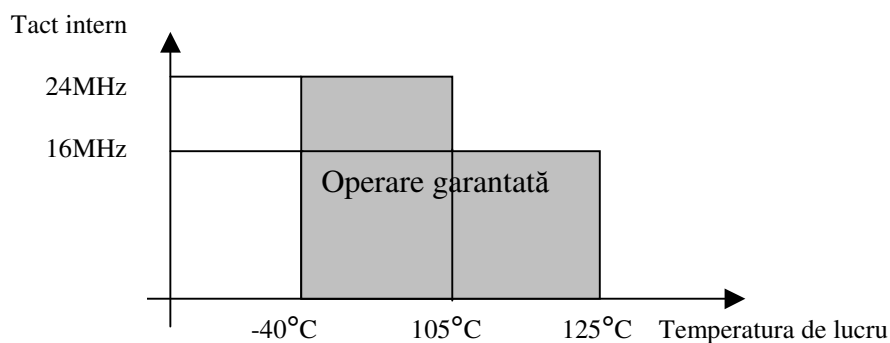


Figura 7.7: graficul temperaturii admise de lucru

Unele modele din noua familie a microcontrolerelor pe 8 biți (MB95100) au gama de temperaturi de lucru între -40°C și 85°C și gama temperaturilor de stocare între -55°C și 150°C .