



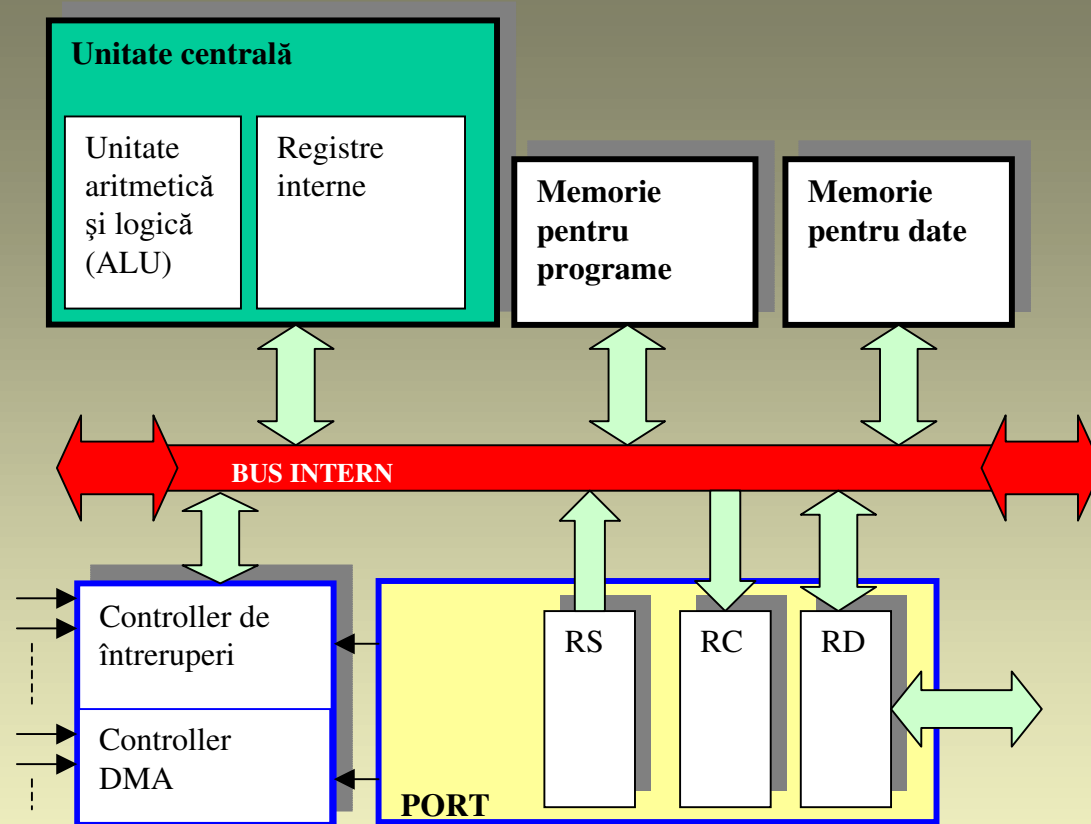
7. Interfețe integrate în microcontrolere

Cuprins și obiective

1. Microcontrollere
2. Unitatea centrală și memoria
3. Timerul
4. Interfață de comunicații seriale asincrone
5. Interfață de comunicații seriale sincrone
6. Interfața CAN
7. Ceasul de gardă
8. Generator PWM
9. Convertor analog digital
10. Proiectarea sistemelor cu MC în vederea siguranței în exploatare
11. Medii de programare și exemple

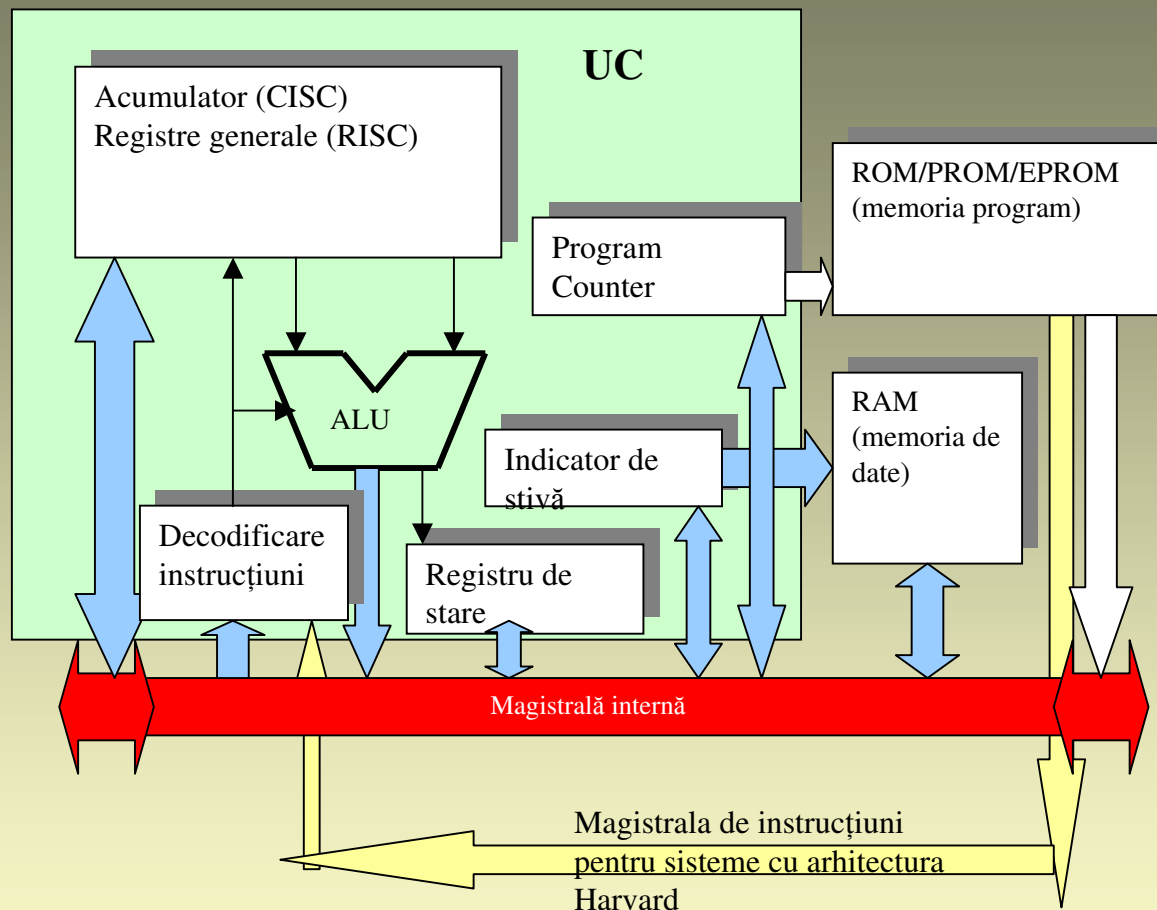
După parcurgerea acestui modul studenții își vor completa informațiile despre microcontrollere. Analiza caracteristicilor și a funcționării interfețelor interne îi va ajuta să înțeleagă că particularitățile fiecărei interfețe o recomandă pentru un anumit tip de aplicații, de o anumită complexitate și cu anumite cerințe de performanță.

Structura



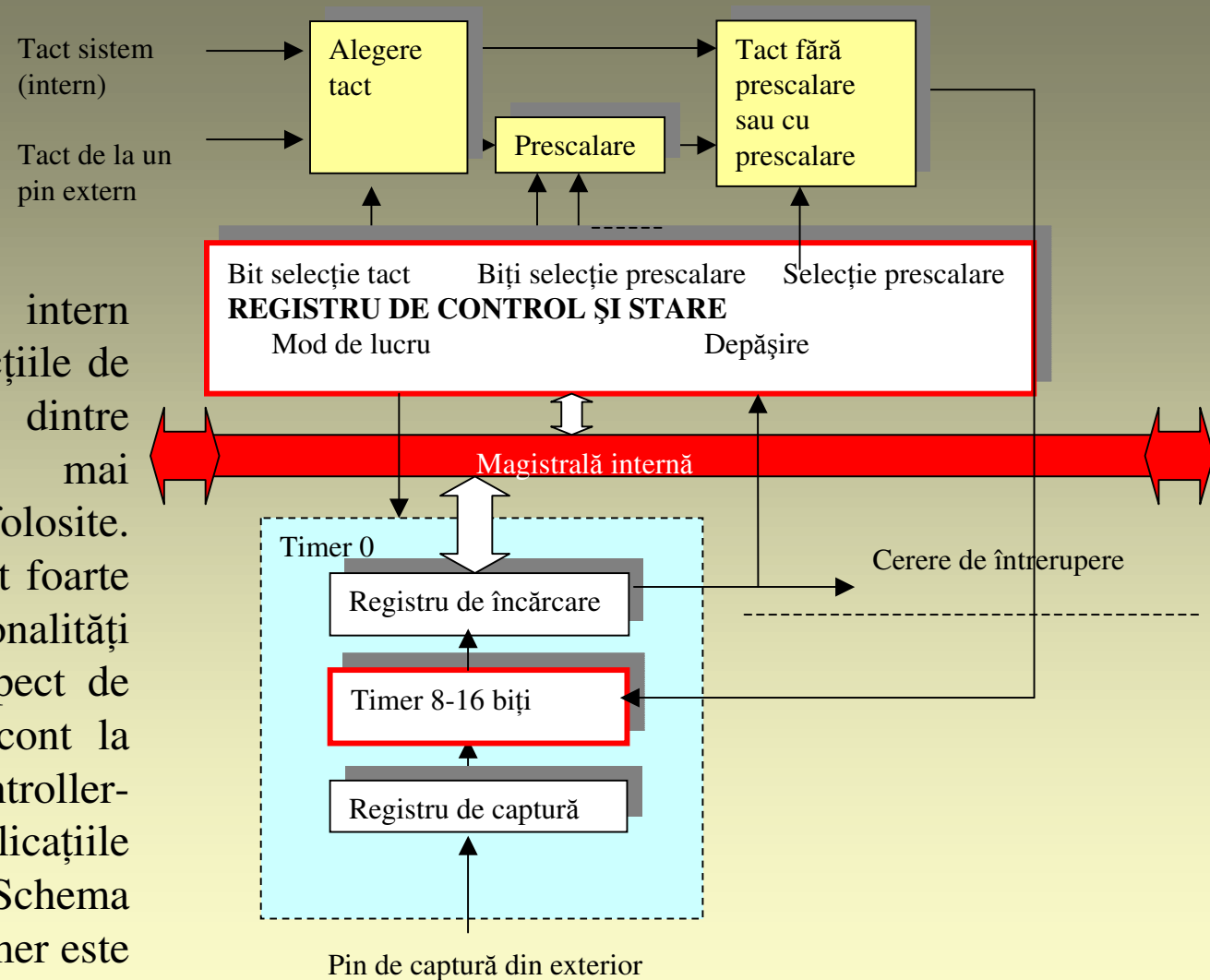
Microcontrollerul (MC) are schema bloc generală simplificată din figura de mai sus. În această schemă sunt puse în evidență interfețele, numite și dispozitive de intrare ieșire (I/O).

Unitatea centrală



Unitatea centrală este formată din ALU și un set de regiștrii interni, similari unor locații de memorie, folosiți pentru memorarea unor date des folosite sau pentru programarea unor anumite funcții. Diferitele familii de MC folosesc seturi diferite de regiștrii.

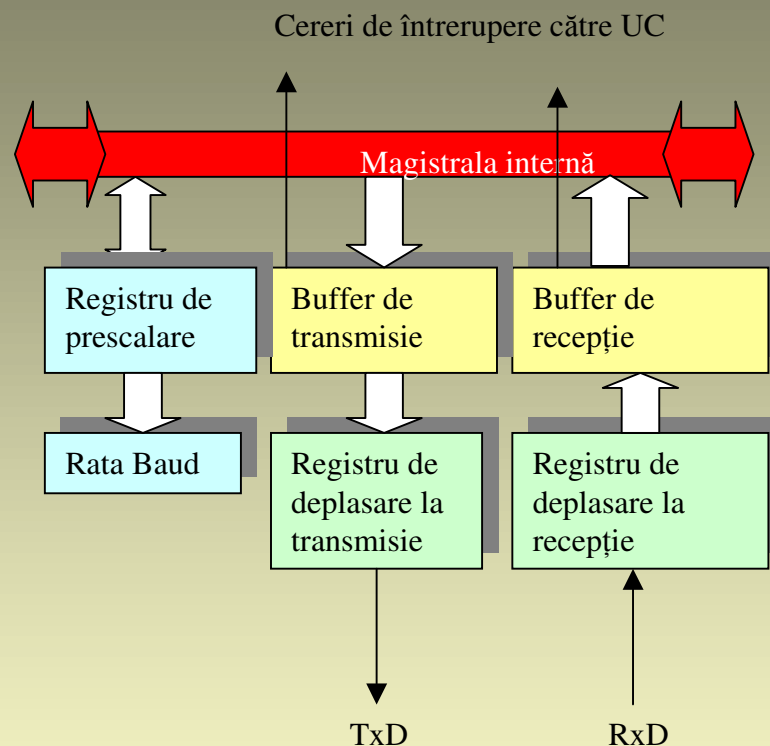
Timer



Timerul este modulul intern care realizează funcțiile de timp, fiind unul dintre modulele cele mai importante și des folosite. Modulele timer sunt foarte diferite ca și funcționalități și complexitate, aspect de care trebuie ținut cont la alegerea microcontroller-ului pentru aplicațiile sensibile la timp. Schema bloc a modulului timer este dată în figură.

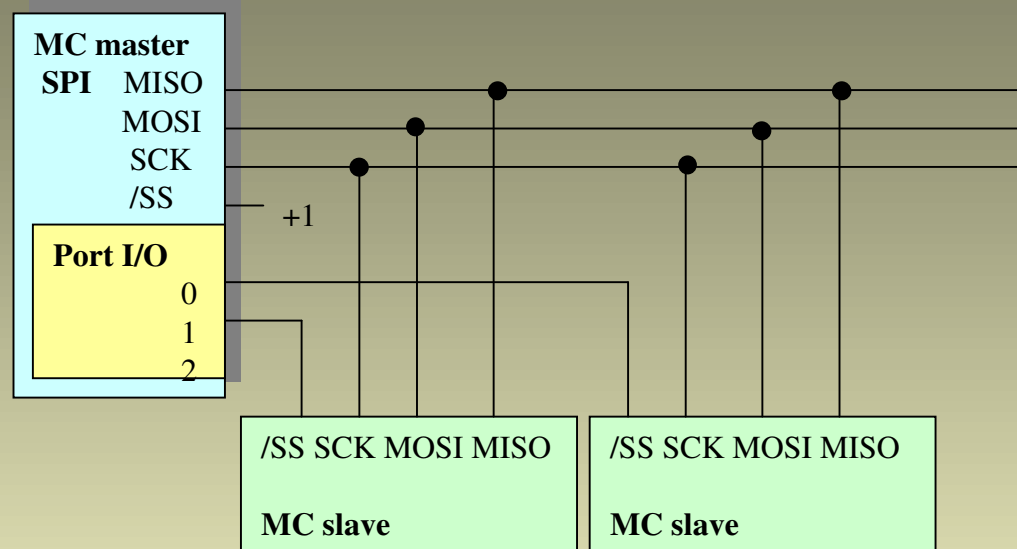
Port serial asincron

Acest port este numit UART (Universal Asynchronous Receiver Transmitter).



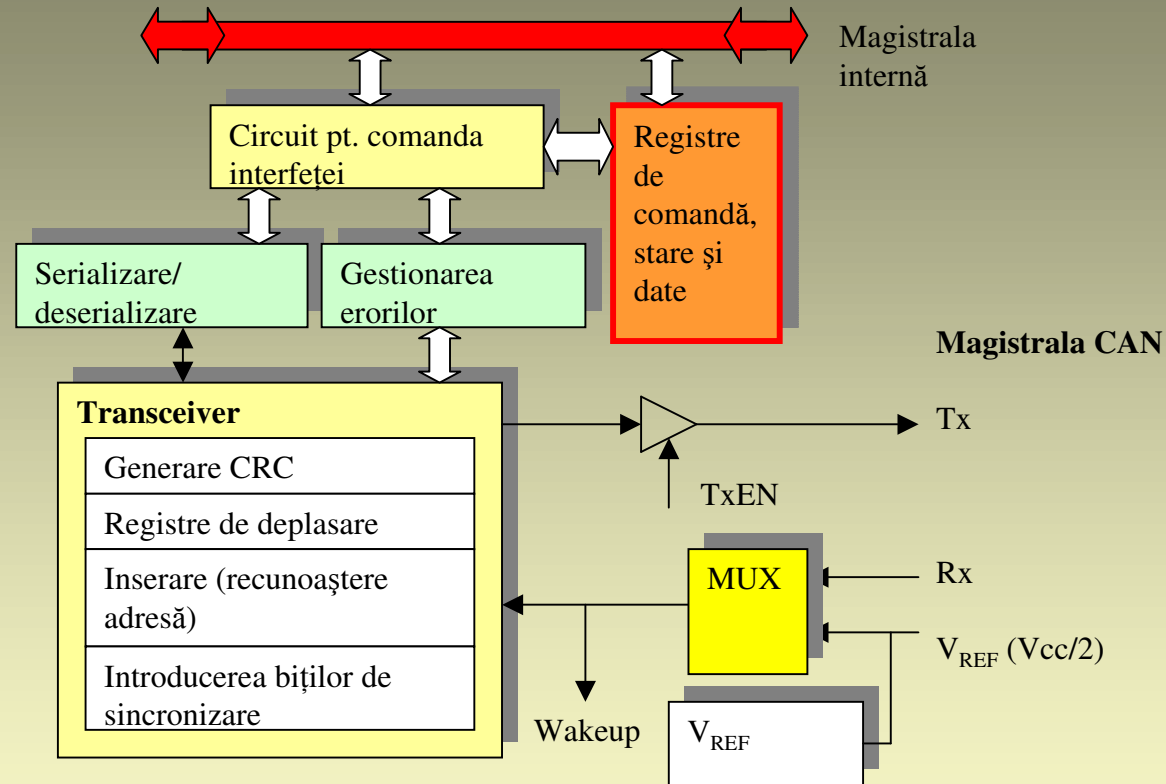
Caracterele seriale sunt transmise sau recepționate serial în registrele de transmisie sau recepție. La recepția unui caracter, acesta se încarcă în bufferul de recepție și se cere o întrerupere. La emisie, un caracter se introduce în bufferul de transmisie de unde este trecut în registrul de deplasare și se transmite serial, cerându-se și o întrerupere.

Port serial sincron SPI



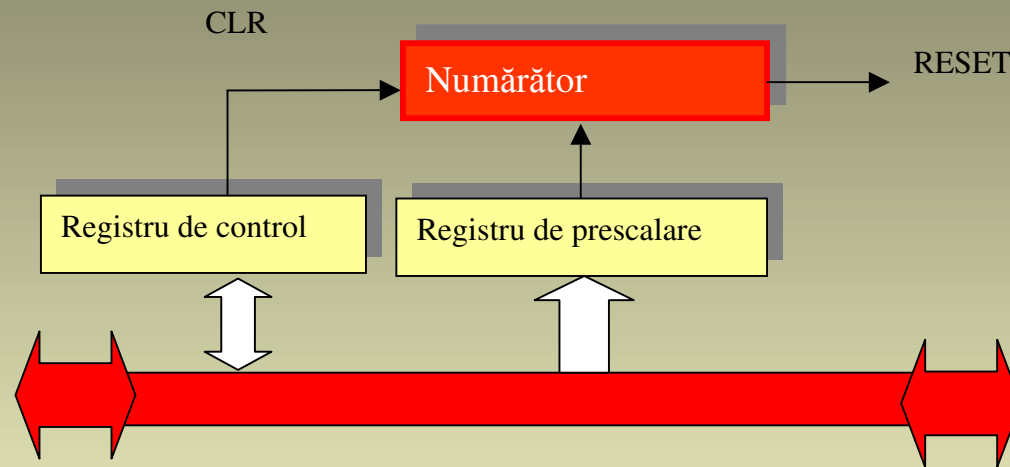
Cu portul SPI (**S**erial **P**eripheral **I**nterface) se poate realiza o comunicație sincronă simplă, folosită de regulă pentru a transfera date între circuite pe aceeași placă cu MC. Un transfer bidirecțional necesită 3 pini, unul dintre ei fiind alocat ceasului de transmisie generat de masterul SPI. Cu SPI se pot realiza transferuri și între MC. Transferurile pot fi full duplex.

Interfața CAN (Controller Area Network)



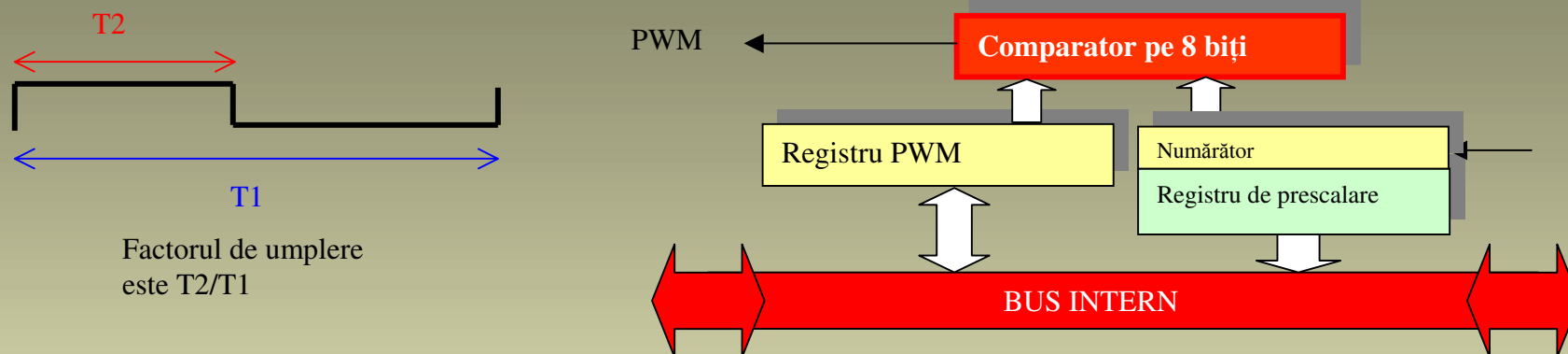
Protocolul CAN a fost definit de BOSCH în 1991 pentru utilizarea pe o magistrală la autoturisme, unde să îndeplinească condiții specifice: procesare în timp real, fiabilitate într-un mediu perturbat și preț mic.

Ceas de gardă (Watch Dog)



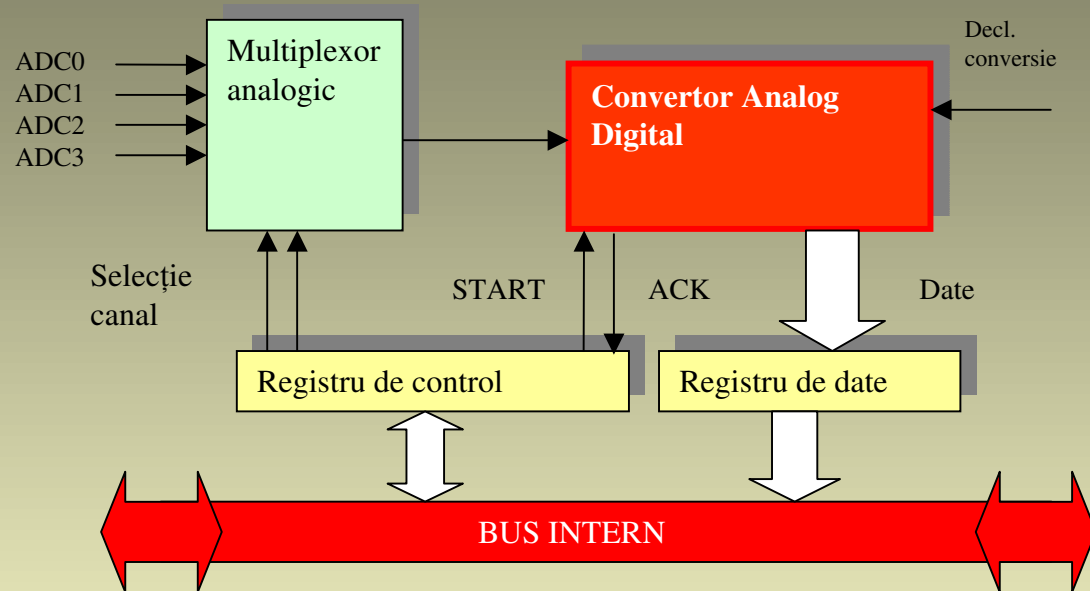
Ceasul de gardă este un timer care poate fi programat să numere un tact. Dacă numărătorul ajunge la capăt, semnalul de depășire declanșează un RESET al circuitului. Este sarcina programatorului să scrie în registrul de control un cuvânt care va reinițializa numărătorul. În cazul în care MC nu mai este sub controlul programului, el va fi resetat de către ceasul de gardă

Generator PWM



Modulația impulsurilor în lățime (Pulse Width Modulation) are multe aplicații, mai ales în comanda motoarelor de curent continuu sau a surselor de alimentare. Un semnal PWM arată ca în figura de sus, stânga. Un canal PWM este reprezentat în figura de sus, dreapta. Frecvența de repetiție este programată cu un registru de prescalare care generează un ceas pentru un numărător de 8 biți. Conținutul numărătorului este comparat cu cel al registrului PWM, dacă este mai mare ieșirea PWM este LOW, dacă este mai mic sau egal, PWM este HIGH. Factorul de umplere poate fi astfel modificat între 0 și 254/255.

Convertorul A/D



Activitate propusă:
Ce porturi, altele decât cele descrise cunoașteți în structura microcontrollerelor?

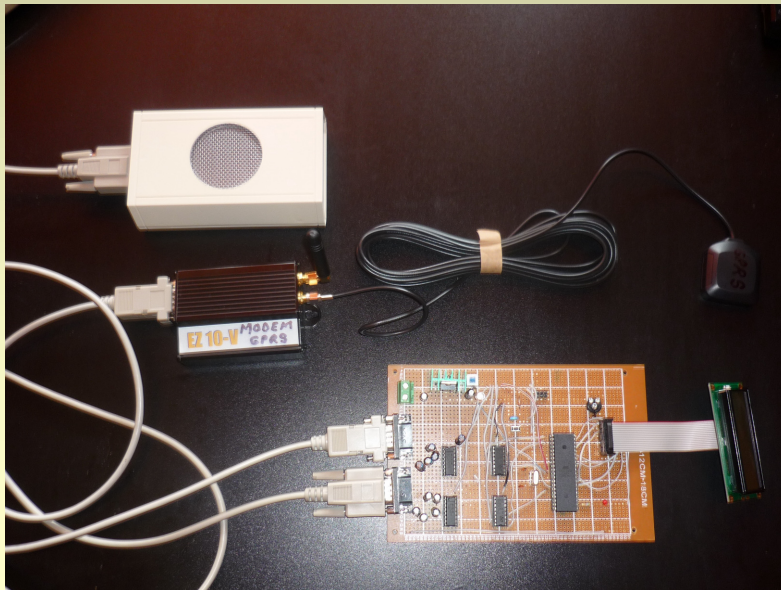
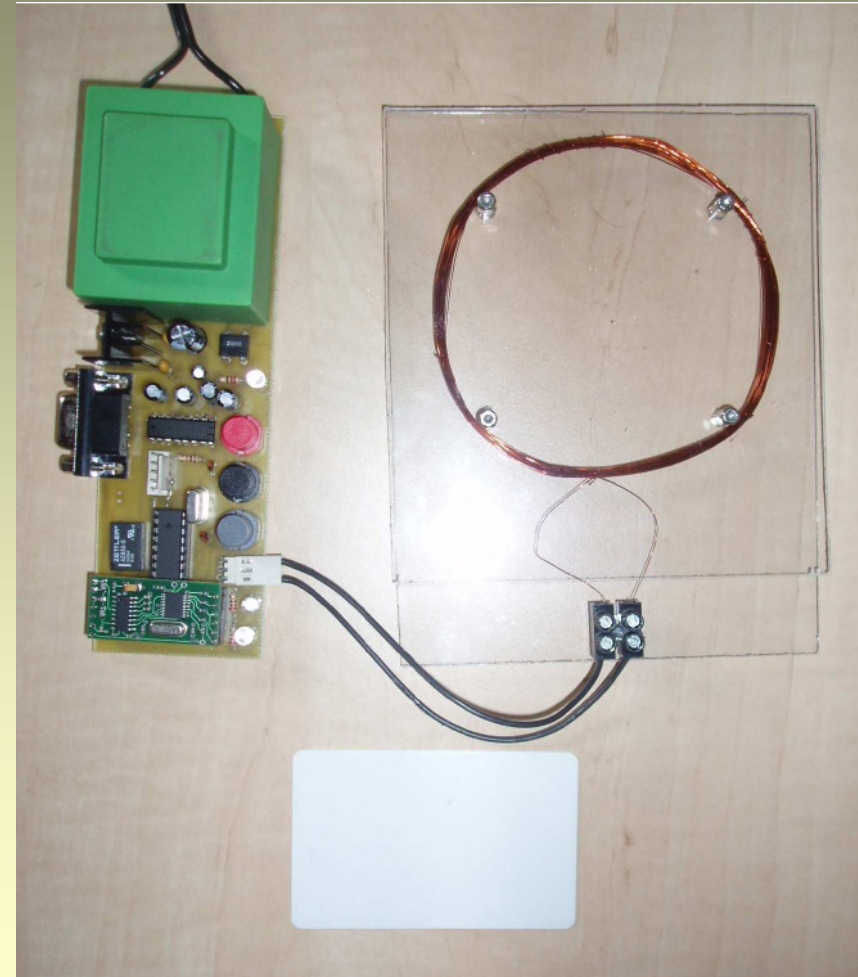
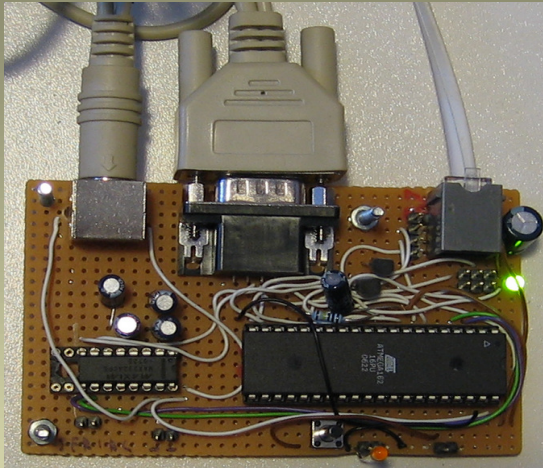
Schema bloc a unei interfețe tipice de conversie a semnalelor analogice în semnale digitale este dată în figură. Circuitul analogic de intrare constă într-un multiplexor analogic, iar convertorul AD este de regulă de 8-10-12 biți cu aproximații succesive.

Proiectarea sistemelor cu MC în vederea siguranței în exploatare

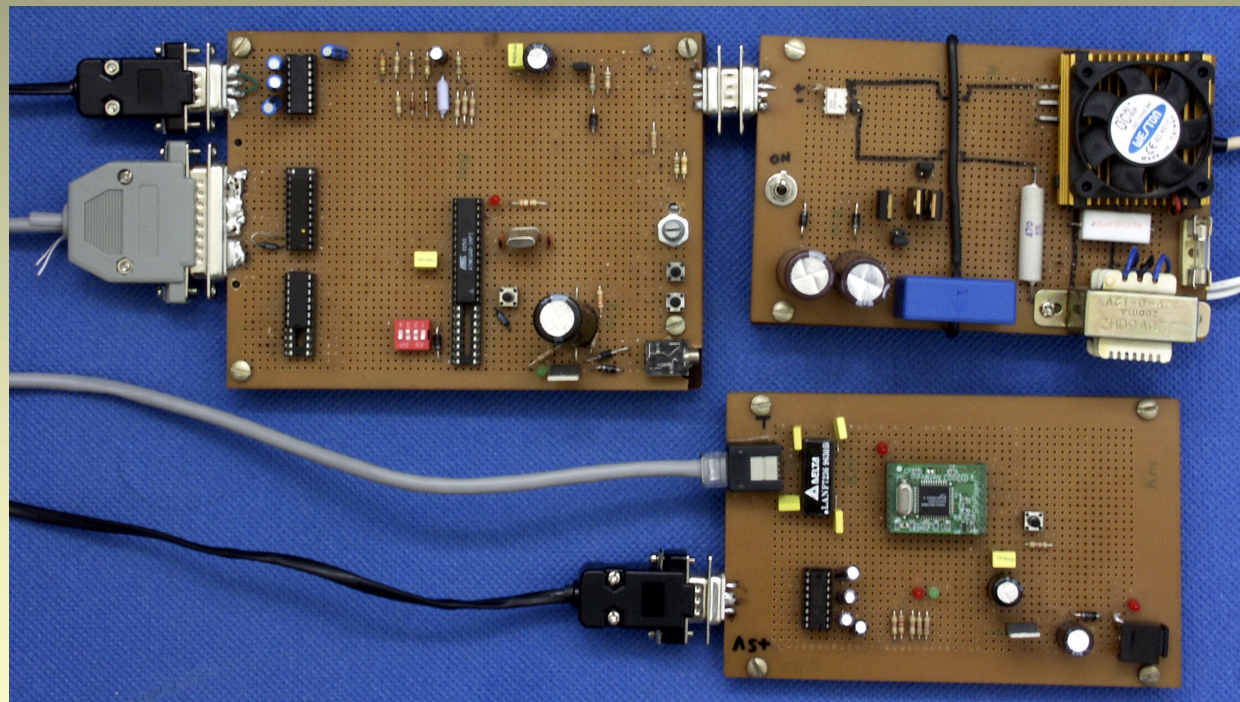
Prin metodele de **programare defensivă** se poate îmbunătăți mult siguranța în funcționare, fără nici un hardware suplimentar. Câteva exemple de metode sunt:

- reîncărcarea periodică a registrelor care comandă pinii de I/O și a celor mai importante registre.
- citirea repetată a semnalelor de intrare micșorează riscul unei citiri greșite.

Exemple de sisteme cu MC realizate la proiect



Exemple de sisteme cu MC realizate la proiect



Mediu de dezvoltare pentru MCS 51



The screenshot displays the PV32-A development environment. The main window shows assembly code for a program named 'pwm1.a51'. The code includes instructions like 'LJMP MAIN', 'ORG 8090h', and various 'MOV' and 'DJPZ' instructions. The 'Main Registers (A)' window is open, showing the current state of the CPU registers. The PC register is at 809B, and the ACC register is at FE. The status bar at the bottom indicates the program is running at 13:41.

CPU	Bank	Data	Hardware
PC	R8	@R0	P0
ACC	R0	@R1	P1
PSW	R1	@DPTR	P2
SP	R2	X@R0	P3
DPTR	R3	X@R1	TCON
B	R4	SPX	THL0
C	R5	XAREA	THL1
EA	R6	Task	THL2
IE	R7	TaskP	PCON

B8:	00	00	00	00	00	00	00
C0:	FF	00	00	00	FF	00	00
C8:	00	00	00	00	00	00	00
D0:	01	00	00	00	00	00	00
D8:	00	00	00	00	00	00	00
E0:	FE	00	00	00	00	00	00
E8:	00	00	00	00	00	00	00
F0:	00	00	00	00	00	00	00
F8:	00	00	00	00	FE	00	00

Franklin Software

Mediu de dezvoltare pentru ATMEL RISC



The screenshot displays the AVR Studio IDE interface. The main window shows a C program named 'blinking_led.c'. The code includes comments and preprocessor directives for setting the clock speed, including header files, and defining a main function with a while loop that toggles an LED on and off with 1000ms delays.

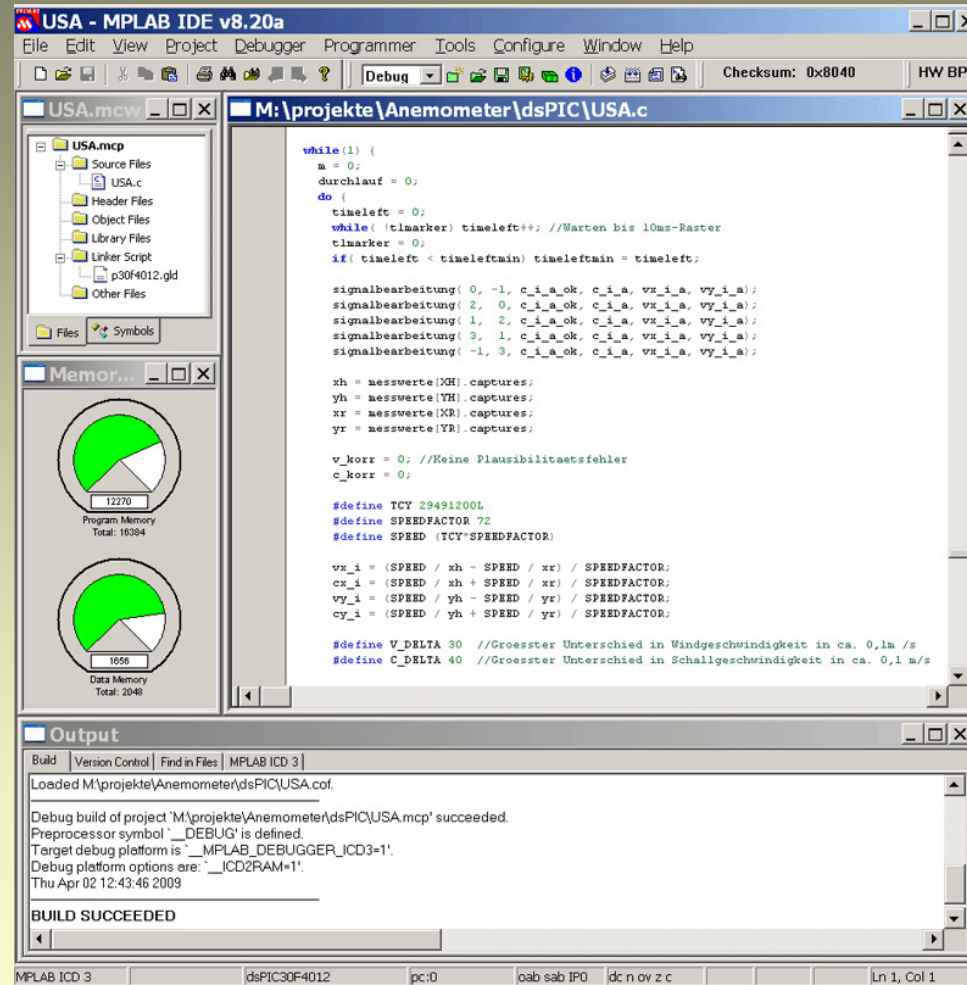
```
/*
Title: blinking_led.c
Author: you
Date Created: 01/01/01
Last Modified: 01/01/01
Purpose: This is a for loop that blinks a series of LEDs on PORTC
*/
//set your clock speed
#define F_CPU 4000000UL
//these are the include files. They are outside the project folder
#include <avr/io.h>
#include <avr/delay.h>
//this include is in your project folder
#include "myTimer.h"

int main (void)
{
//Set PORTC to all outputs
DDRC = 0xFF;

//create an infinite loop
while(1) {
//this turns pin C0 on and off
//turns C0 HIGH
PORTC |= (1<<0);
//PAUSE 1000 milliseconds
delay_ms(1000);
//turns C0 LOW
PORTC &= ~(1 << 0);
//PAUSE 1000 milliseconds
delay_ms(1000);
}
}
```

AVR STUDIO

Mediu de dezvoltare pentru PIC



MPLAB

Arduino



Cel mai simplu de construit un sistem pe bază de microcontroller este cu prin utilizarea modulelor Arduino. Există doi mari furnizori:

www.robofun.ro

www.teguna.ro

Se pot cumpăra o mare diversitate de sisteme de dezvoltare, traductori și elemente de execuție.



Două exemple de sisteme de dezvoltare:

Arduino Leonardo
(stânga) cu ATmega 32
și Arduino Due cu
microcontroller ARM pe
32 de biți (dreapta)



Proiect de diplomă cu Arduino



Sistem mobil echipat cu un senzor de gaz, un modul Arduino cu microcontroller, drivere pentru motoare de curent continuu și modul Bluetooth. Comanda de la distanță se face cu un telefon mobil, interfața grafică a programului fiind arătată în dreapta.

Temă



Activitate propusă:

descrieți o aplicație cu microcontroller și încercați să vă imaginați care sunt interfețele din microcontroller utilizate. De exemplu interfonul de acces în bloc.

Concluzii

Pentru a putea înțelege funcționarea interfețelor materialul începe cu o prezentare a unității centrale și a tipurilor de memorii care intră în structura microcontrollerelor. Sunt prezentate apoi interfețele de comunicații, începând cu cea mai simplă- interfața serială asincronă, apoi cea sincronă și mergând până la interfața CAN. Sunt prezentate în continuare și alte interfețe importante cum sunt timer-ul, convertorul analog digital, ceasul de gardă și generatorul PWM. Materialul se termină cu câteva indicații de proiectare a software-ului care să asigure stabilitatea operațiilor de intrare ieșire, o scurtă prezentare a câtorva medii de dezvoltare software și câteva fotografii ale unor aplicații realizate de studenți.

În acest modul se face legătura între noțiunile fundamentale din primele module și interfețele existente în microcontrolere.

Mulțumesc pentru atenție

