



BME



KJIT

Budapesti Műszaki és Gazdaságtudományi Egyetem

Közlekedésmérnöki és Járműmérnöki Kar

Közlekedés- és Járműirányítási Tanszék

Mechatronika és mikroszámítógépek

2020/2021 I. félév

8051, C8051F020 mikro
vezérlők és programozásuk

Fontos tudnivalók

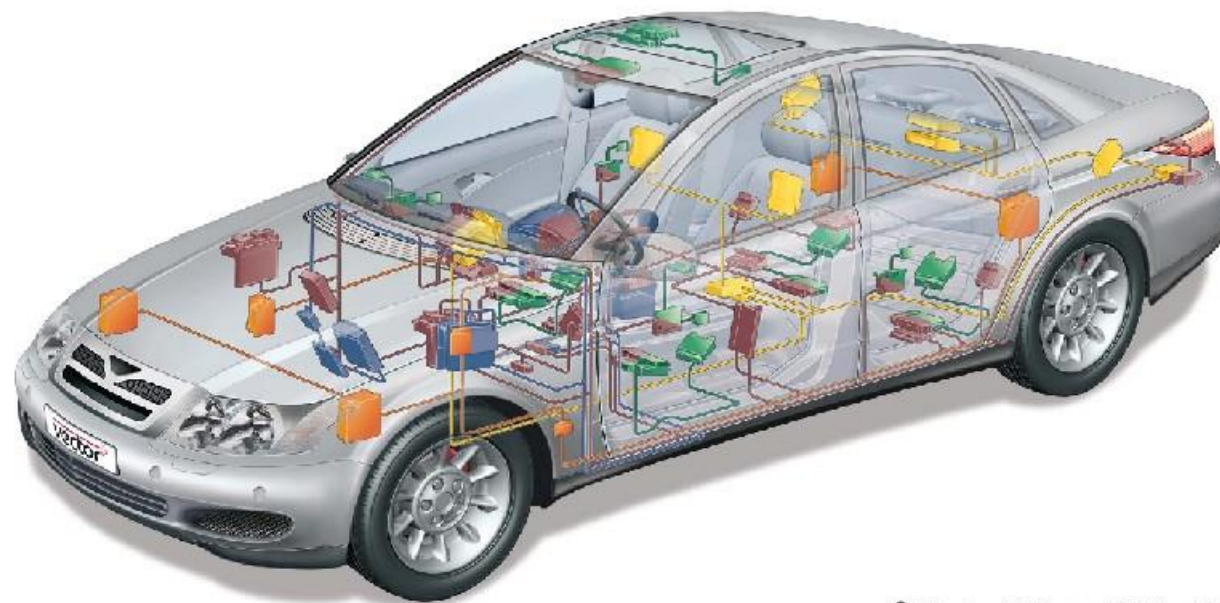
- Elérhetőség: ST. 108
- E-mail: lovetei.istvan@mail.bme.hu
- Teams: lovetei.istvan@edu.bme.hu
- Fontos tudnivalók: **MOODLE rendszer** (edu.kozlek.bme.hu)
- Félévközi jegy: 2 db sikeres lab. ZH + 2 db sikeres elméleti ZH átlaga alapján.
 - ASM ZH: 7. oktatási hét
 - C ZH: 14. oktatási hét
 - **Házi feladatok hiánytalan leadása a félévközi jegy megszerzésének feltétele!**
- Tananyag: **Embedded Programming Textbook** (honlap)
 - és: **C8051F020 datasheet** (honlap)
- **Órarendváltozás: 2. héten az EA és a LAB kurzus felcserélésre kerül!!!**

Játékszabályok

- Folyamatos számonkérés 😊
- Lab. ZH:
 - Minden kiadott segédanyag, órán elkészített program használható
- Házi feladatok
 - Leadási határidő gyakorlatot megelőző este 8 óra – e-mailben
 - Pontozás: 1-3 ig, késés esetén: pontlevonás
- Jelenlét
 - TVSZ: az órák 70%-án a jelenlét kötelező
- Folyamatos tanulás 😊 és számonkérés

A tantárgy célja

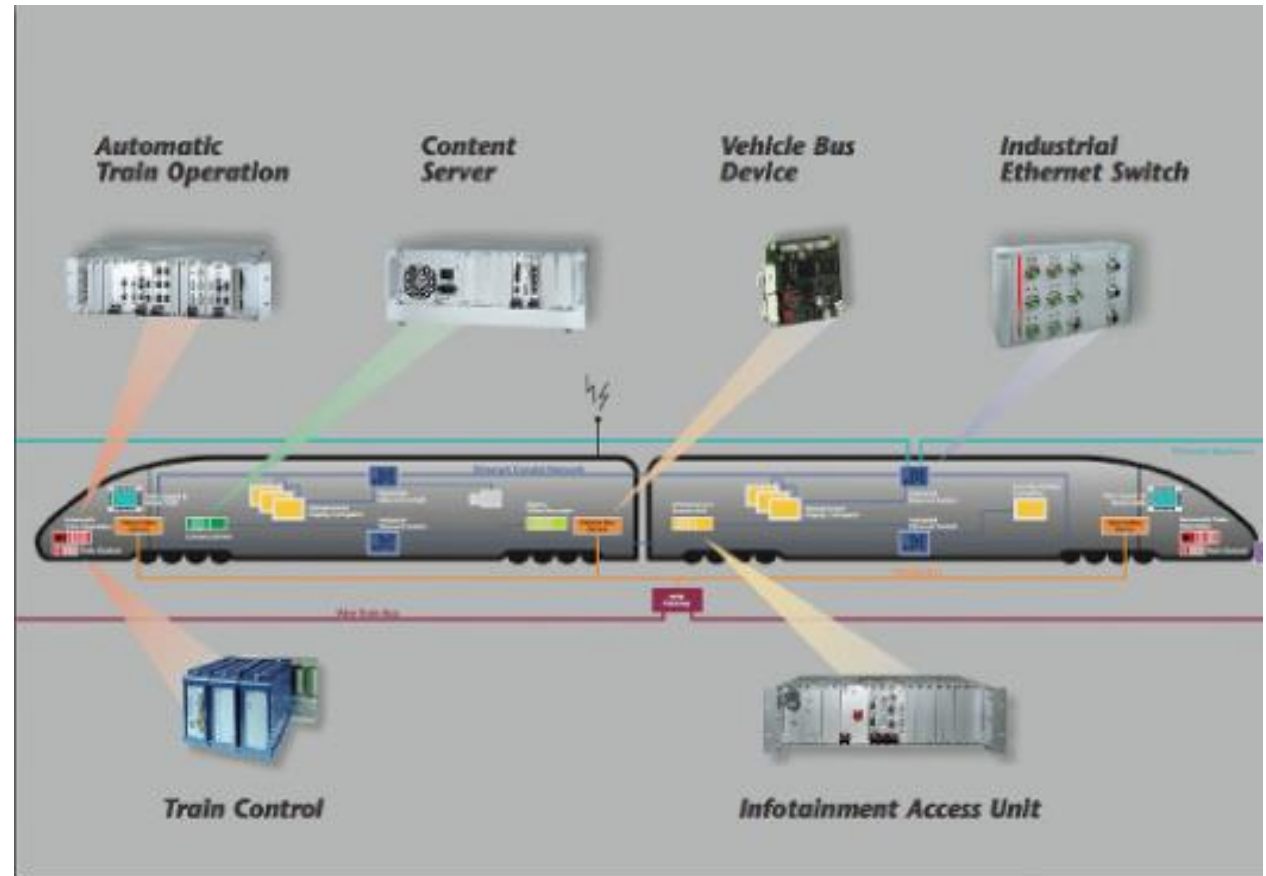
- Beágyazott (embedded) program készítése önállóan:
 - MCU alapkészletek elsajátítása;
- Eszköz:
 - Silicon Labs C8051F020 mikrokontroller (MCU);
 - Silicon Labs ToolStick University Daughter Card;
- Fejlesztő rendszer:
 - Silicon Laboratories IDE – release 4.90;
 - SDCC (Small Device C Compiler) 2.9.0;
- Program nyelvek:
 - 8051 assembly – Intel MCS[®] - 51 CISC utasításkészlettel;
 - C.



© Vector Informatik GmbH

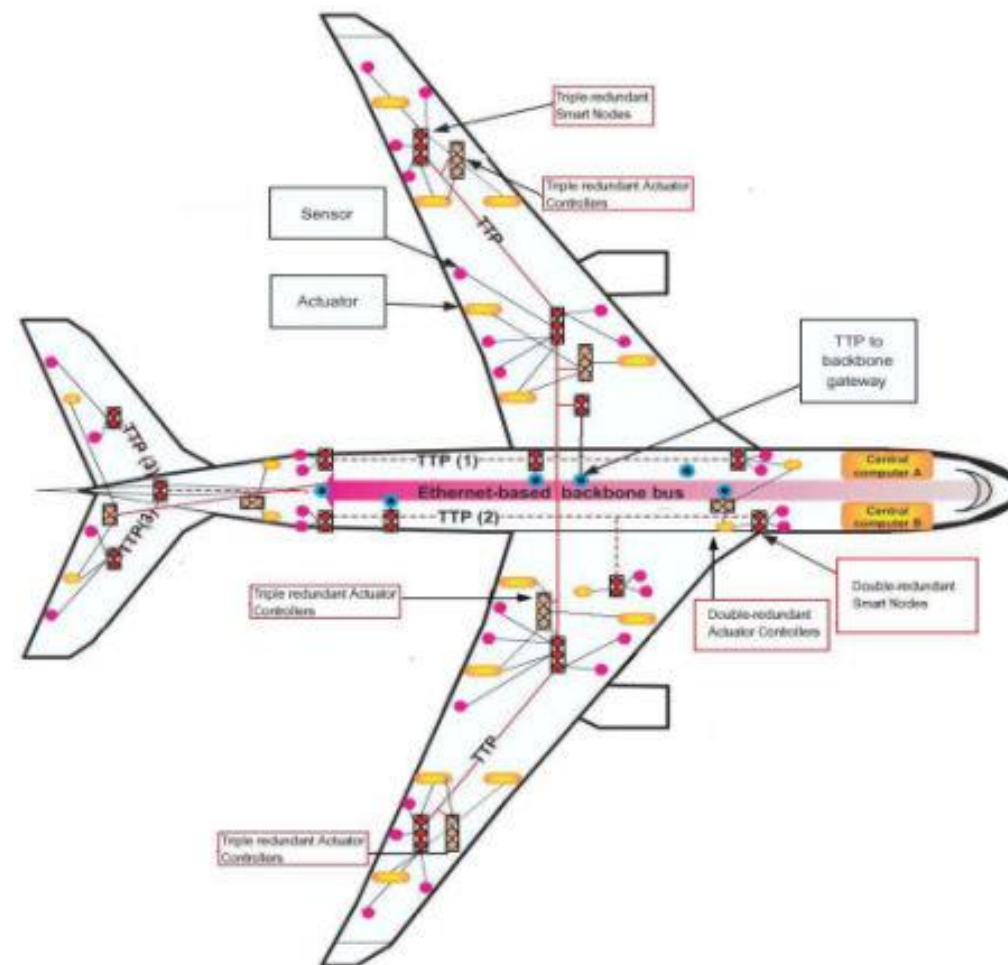
Beágyazott rendszerek

- Beágyazott rendszer (def): Egy beágyazott rendszer egy **speciális (dedikált) célú** számítógép rendszer.
 - jelzőlámpák és mérő rendszerek (közúti forgalom);
 - vasúti biztosítóberendezések, vonatbefolyásoló rendszerek, vonatok vezérlő rendszerei;
 - repülőgépek vezérlő elemei;
 - ipari folyamatok irányítása;
 - járműirányítási rendszerek megvalósítása;
 - stb...



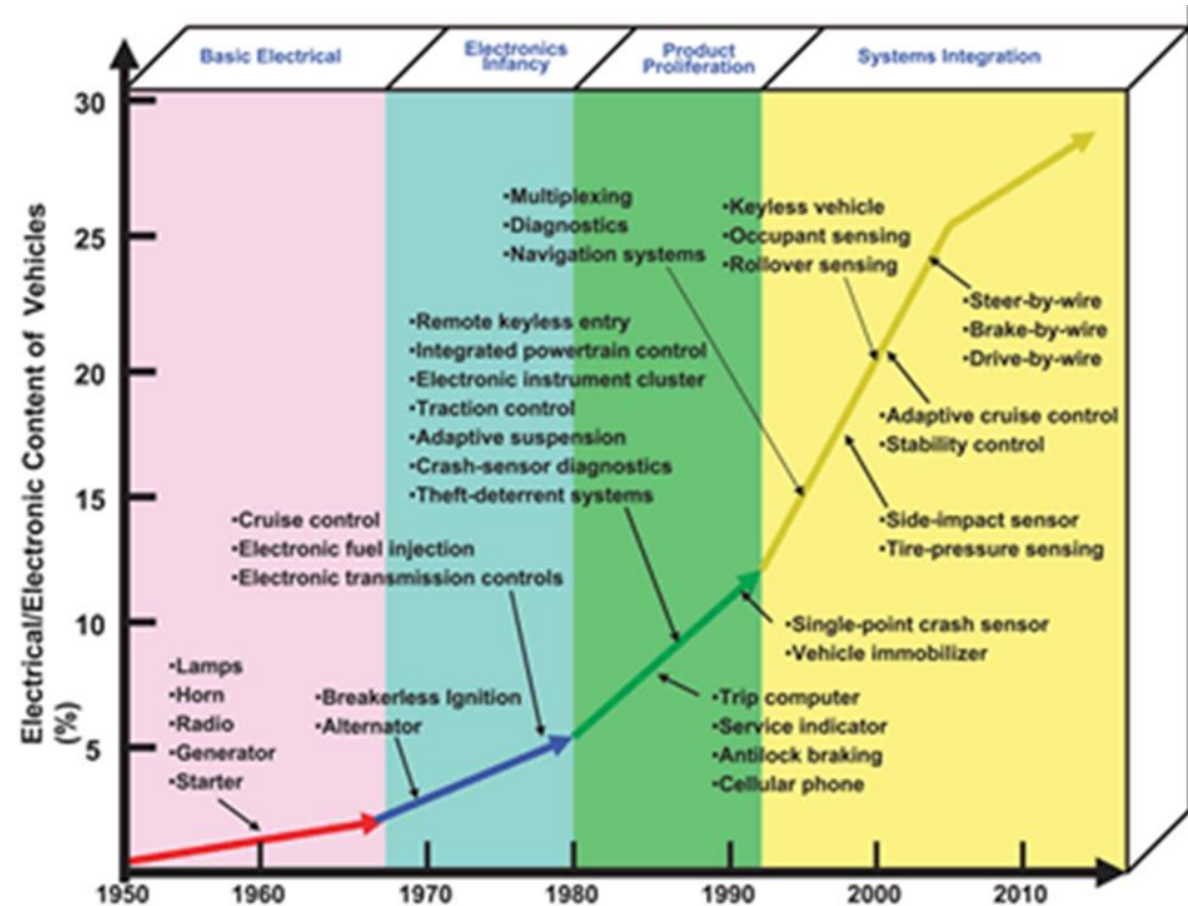
Beágyazott rendszerek

- elektronikus eszközök aránya egy autóban: kb. 25%,
 - pl. egy felső kategóriás autóban, az elektronikus vezérlő eszközök (ECU) száma kb. 80;
 - pl. egy modern repülőgépben több, mint 700 ECU fut egyidőben;
- általánosságban elmondható, hogy az ECU-k hálózatot (hálózatokat) alkotnak.



Beágyazott rendszerek

- Az első mikroprocesszorok az 1970-es években jelentek meg.
- 2000-ben átlagosan 15 processzor dolgozott egy járműben;
- 2010-re ez a szám 30-ra emelkedett.



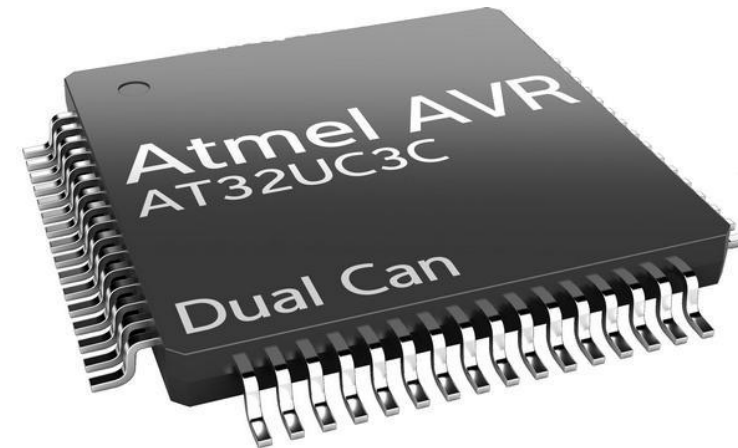
Mikroprocesszor vs. Mikrokontroller

- Mikroprocesszor (CPU)
 - Egy chip-ben egy számítógység és annak vezérlése,
 - Nincs ROM, RAM és perifériavezérlés,
 - Általános célokra,
 - Nagy, általános utasításkészlet és további speciális utasításkészlet,
 - Összetett memóriakezelés,
 - Komplex kiegészítő áramkört igényel, önmagában működésképtelen.

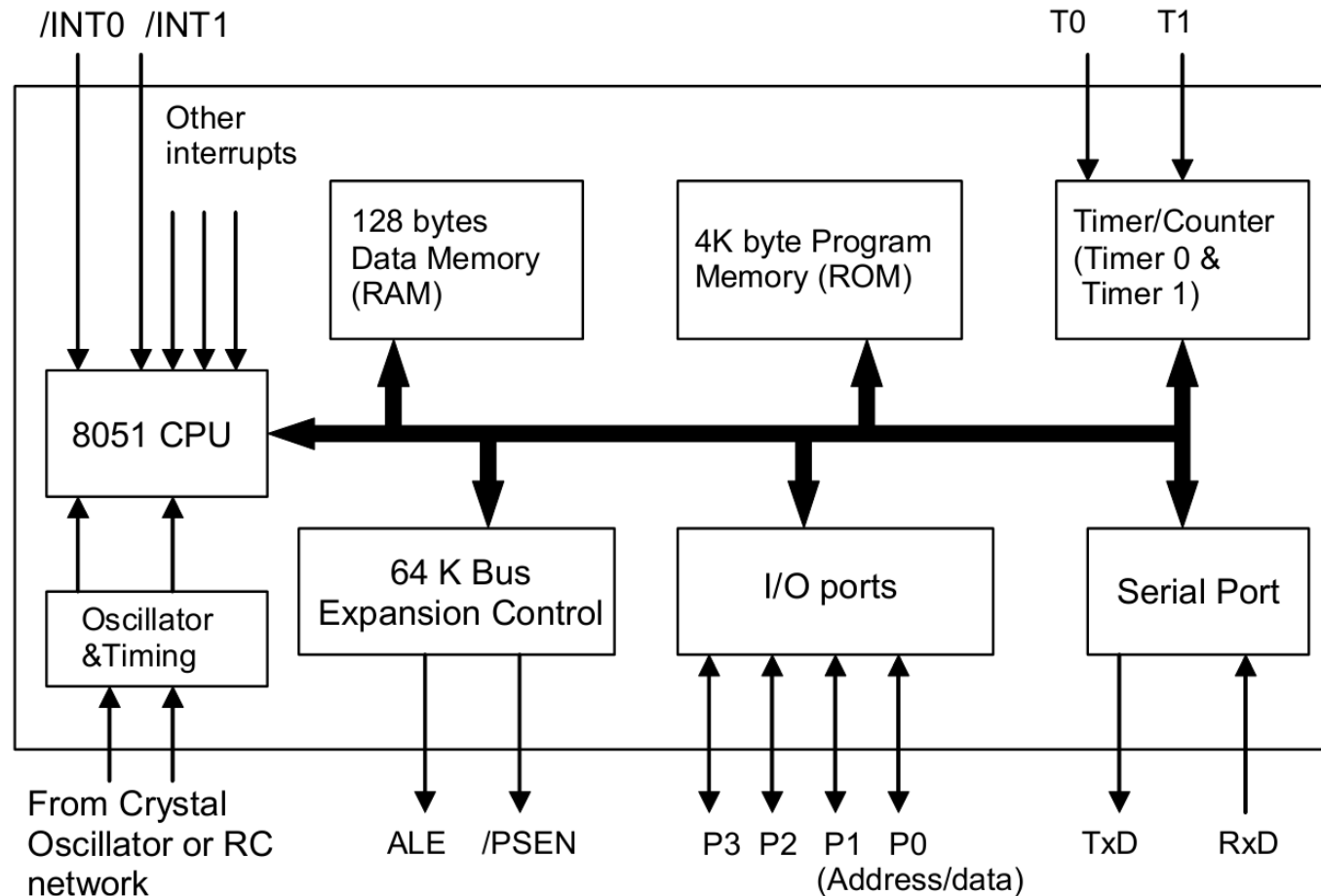


Mikroprocesszor vs. Mikrokontroller

- Mikrokontroller (MCU)
 - Mikroprocesszor és
 - Perifériavezérlő áramkörök, ROM, RAM egy közös egységben integrálva,
 - Önmagában is működőképes komplett „számítógép”,
 - Alacsonyabb számítási kapacitás,
 - Kiválóan alkalmas ipari irányítástechnikai feladatok elvégzésére,
 - Célszámítógép – **embedded**.

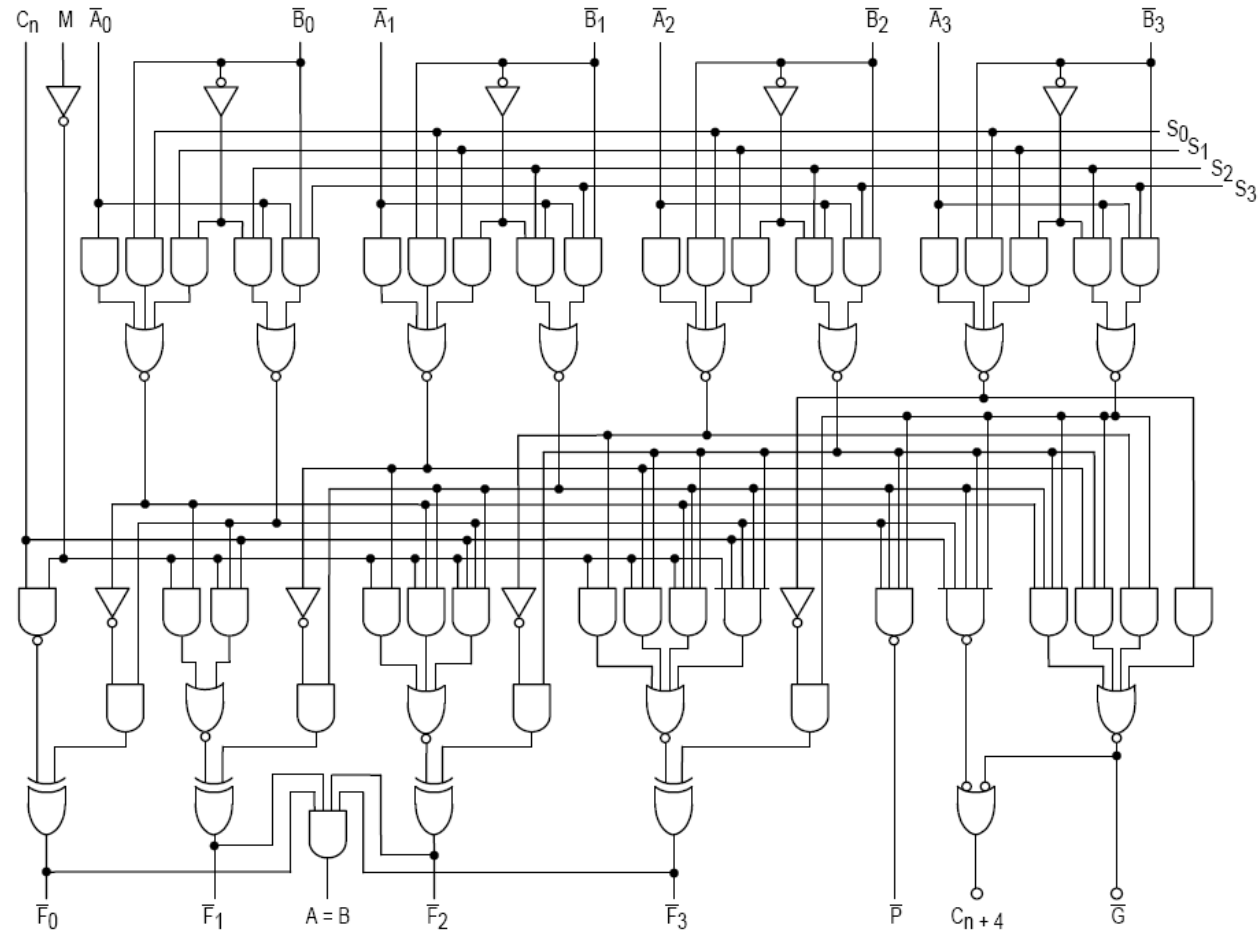


Általános (8051) mikrokontroller architektúra

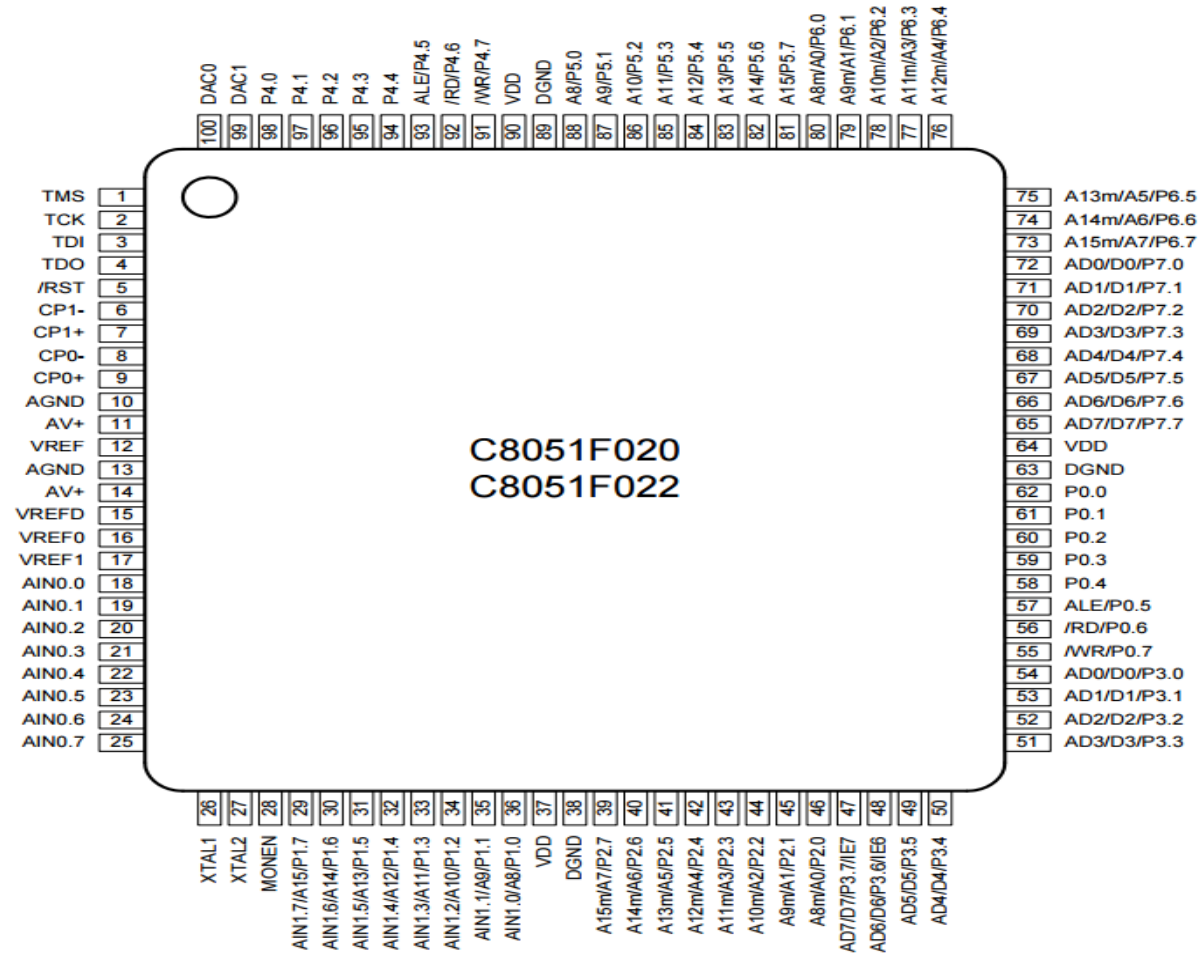


ALU, pl.

- Pl. a 74181 típusú IC kombinációs logikai áramkör, amely egy egyszerű 4-bites ALU.

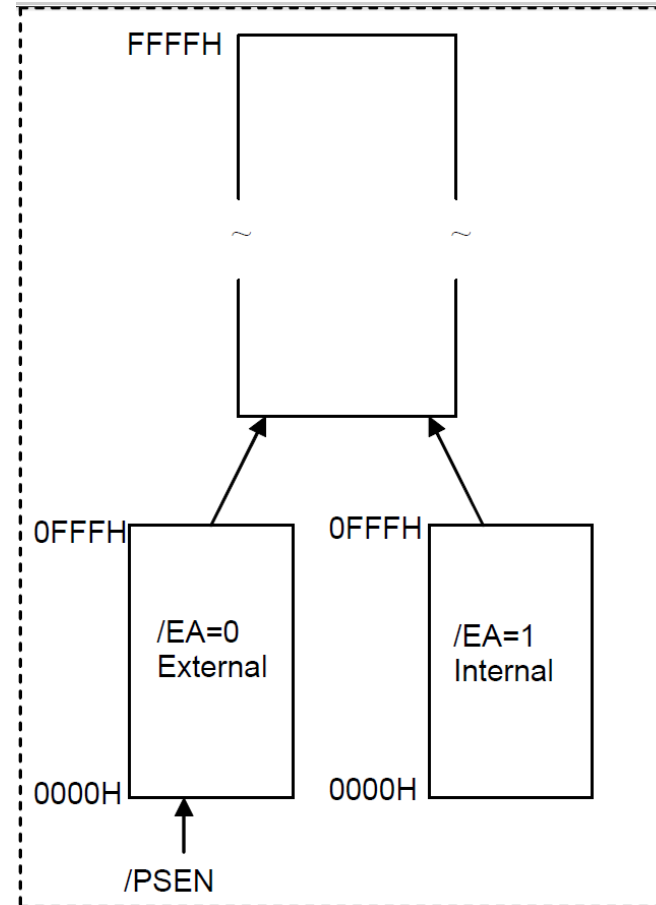


A C8051F020 mikrokontroller architektúrája



Memória felépítés

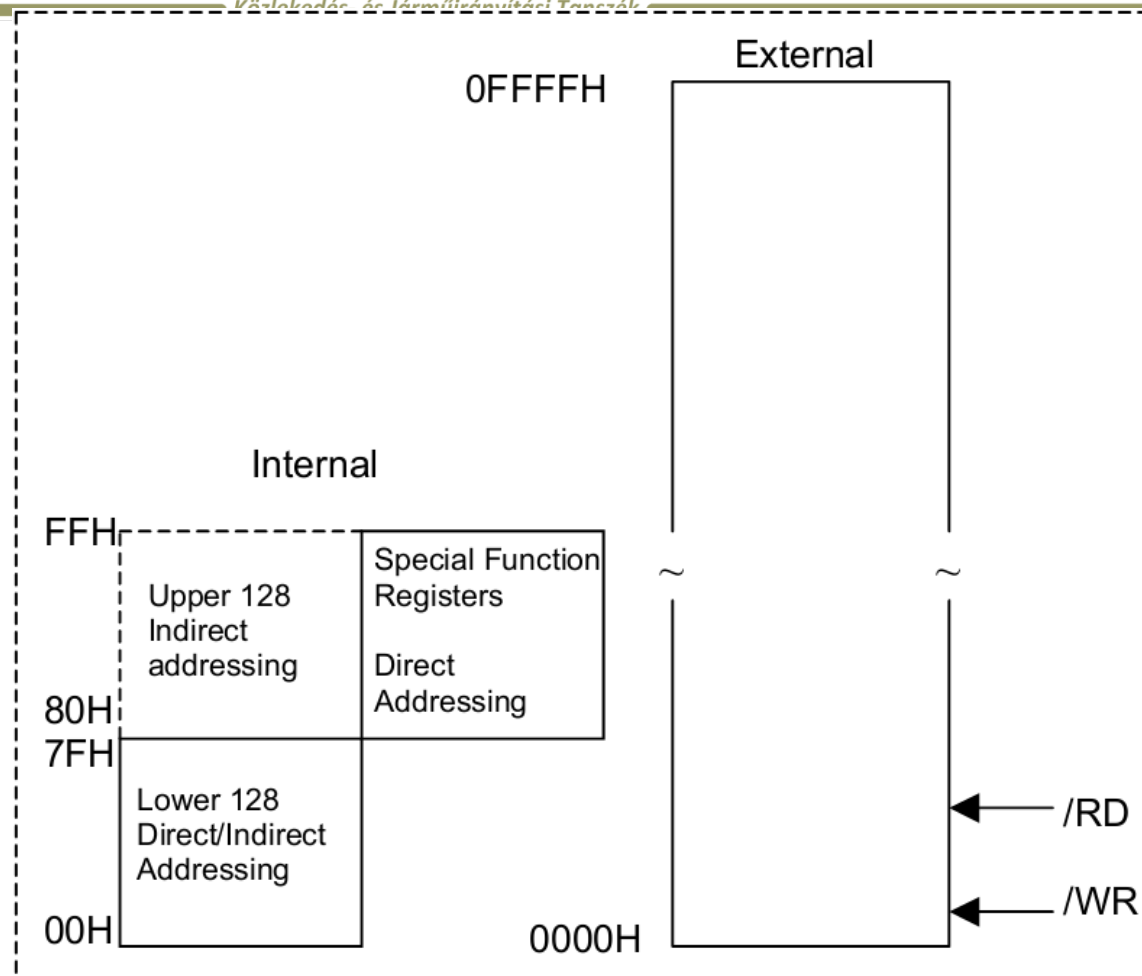
(FLASH) ROM
4 kB + 64 kB



Memória felépítés

RAM

384 B + 64 kB



Memória felépítés

Alsó 128 bájtt

Byte Address	Bit Address							
7F	General Purpose RAM							
30								
2F	7F	7E	7D	7C	7B	7A	79	78
2E	77	76	75	74	73	72	71	70
2D	6F	6E	6D	6C	6B	6A	69	68
2C	67	66	65	64	63	62	61	60
2B	5F	5E	5D	5C	5B	5A	59	58
2A	57	56	55	54	53	52	51	50
29	4F	4E	4D	4C	4B	4A	49	48
28	47	46	45	44	43	42	41	40

27	3F	3E	3D	3C	3B	3A	39	38
26	37	36	35	34	33	32	31	30
25	2F	2E	2D	2C	2B	2A	29	28
24	27	26	25	24	23	22	21	20
23	1F	1E	1D	1C	1B	1A	19	18
22	17	16	15	14	13	12	11	10
21	0F	0E	0D	0C	0B	0A	09	08
20	07	06	05	04	03	02	01	00
1F	Bank 3							
18								
17	Bank 2							
10								
0F	Bank 1							
08								
07	Default Register Bank for R0 – R7							
00								

*A regiszter a processzorba beépített nagyon gyors elérésű, kis méretű memória. A regiszterek addig (ideiglenesen) tárolják az információkat, utasításokat, amíg a processzor dolgozik velük.

Memória felépítés

SFR (Special Function Registers) - a mikrokontroller működését befolyásoló regiszterek

Byte Address	Bit Address							
FF								
F0	F7	F6	F5	F4	F3	F2	F1	F0
E0	E7	E6	E5	E4	E3	E2	E1	E0
D0	D7	D6	D5	D4	D3	D2	-	D0
B8	-	-	-	BC	BB	BA	B9	B8
B0	B7	B6	B5	B4	B3	B2	B1	B0
A8	AF	-	-	AC	AB	AA	A9	A8
A0	A7	A6	A5	A4	A3	A2	A1	A0

	99	Not bit-addressable								SBUF
	98	9F	96	95	94	93	92	91	90	SCON
B	90	97	96	95	94	93	92	91	90	P1
ACC	8D	Not bit-addressable								TH1
	8C	Not bit-addressable								TH0
PSW	8B	Not bit-addressable								TL1
	8A	Not bit-addressable								TL0
IP	89	Not bit-addressable								TMOD
	88	8F	8E	8D	8C	8B	8A	89	88	TCON
P3	87	Not bit-addressable								PCON
	83	Not bit-addressable								DPH
IE	82	Not bit-addressable								DPL
	81	Not bit-addressable								SP
P2	80	87	86	85	84	83	82	81	80	P0

Regiszter címek - C8051F020

F8	SPI0CN	PCA0H	PCA0CPH0	PCA0CPH1	PCA0CPH2	PCA0CPH3	PCA0CPH4	WDTCN
F0	B	SCON1	SBUF1	SADDR1	TL4	TH4	EIP1	EIP2
E8	ADC0CN	PCA0L	PCA0CPL0	PCA0CPL1	PCA0CPL2	PCA0CPL3	PCA0CPL4	RSTSRC
E0	ACC	XBR0	XBR1	XBR2	RCAP4L	RCAP4H	EIE1	EIE2
D8	PCA0CN	PCA0MD	PCA0M0	PCA0CPM1	PCA0CPM2	PCA0CPM3	PCA0CPM4	
D0	PSW	REF0CN	DAC0L	DAC0H	DAC0CN	DAC1L	DAC1H	DAC1CN
C8	T2CON	T4CON	RCAP2L	RCAP2H	TL2	TH2		SMB0CR
C0	SMB0CN	SMB0STA	SMB0DAT	SMB0ADR	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH
B8	IP	SADEN0	AMX0CF	AMX0SL	ADC0CF	P1MDIN	ADC0L	ADC0H
B0	P3	OSCXCN	OSCICN			P74OUT	FLSCL	FLACL
A8	IE	SADDR0	ADC1CN	ADC1CF	AMX1SL	P3IF	SADEN1	EMI0CN
A0	P2	EMI0TC		EMI0CF	P0MDOUT	P1MDOUT	P2MDOUT	P3MDOUT
98	SCON0	SBUF0	SPI0CFG	SPIODAT	ADC1	SPI0CKR	CPT0CN	CPT1CN
90	P1	TMR3CN	TMR3RLL	TMR3RLH	TMR3L	TMR3H	P7	
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH	P4	P5	P6	PCON
	0(8) Bit addressable	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

*: ha a sor érték második számjegye 8, akkor a zárójeles értéket kell figyelembe venni az oszlop értékénél!

Pl:

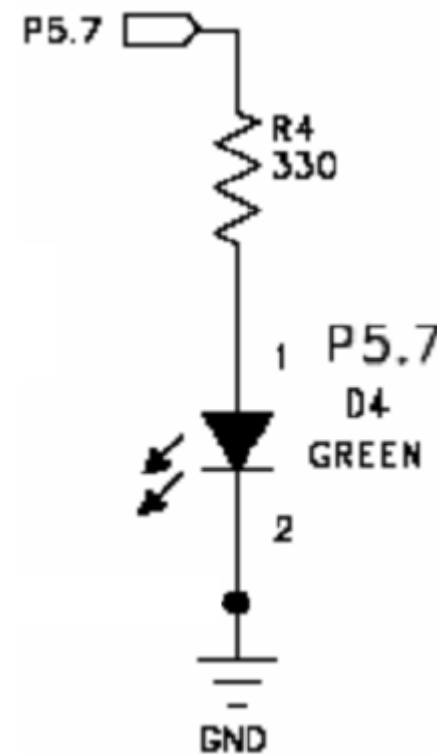
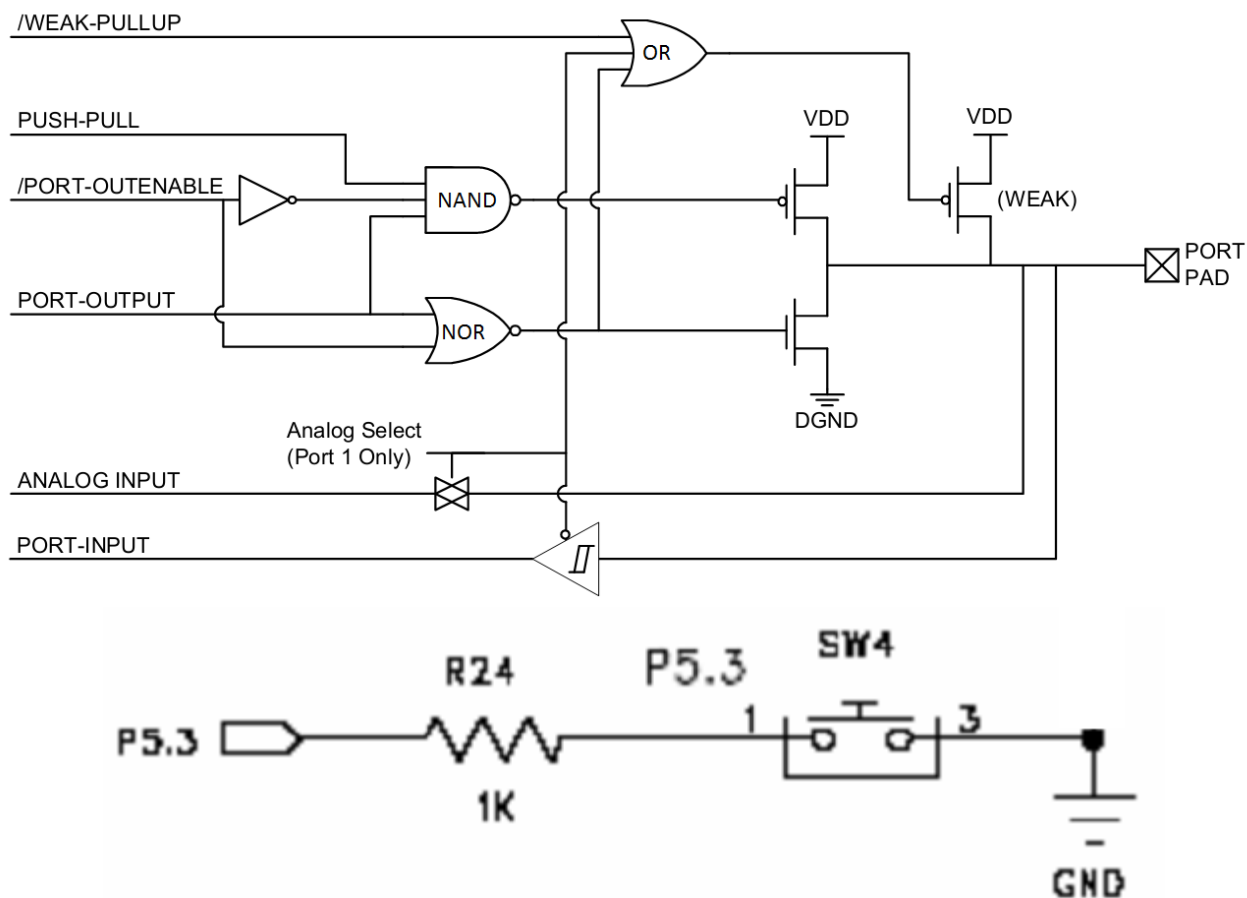
- P74OUT helye: B5H=0xB5
- WDTCN helye: FFH=0xFF

Legfontosabb regiszterek

- ACC
- B
- R0-R7
- PSW
- DPTR
- PC
- SP

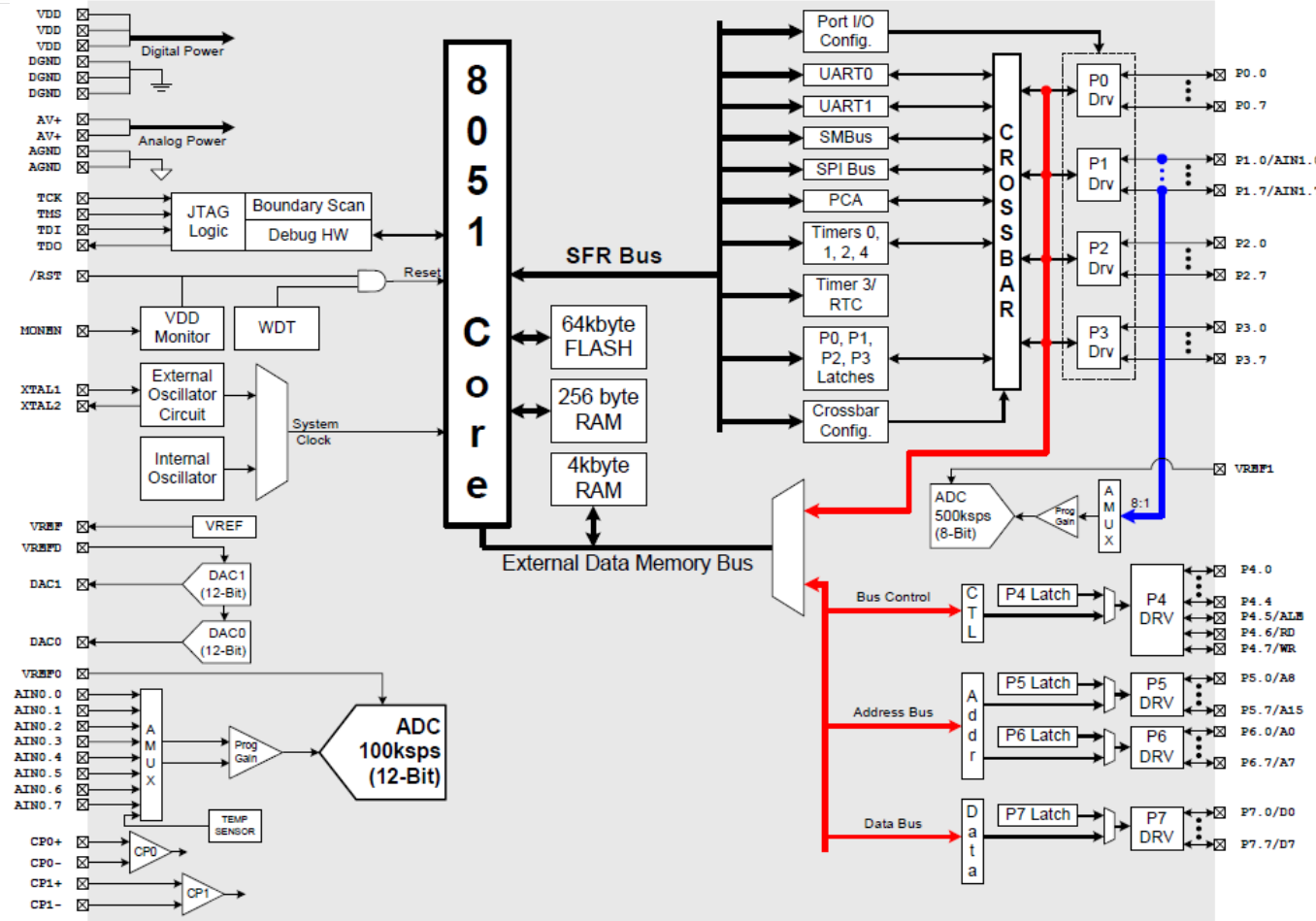
Bit	Sym	Bit	Description
PSW.7	CY	D7H	Carry flag
PSW.6	AC	D6H	Auxiliary carry flag (BCD)
PSW.5	F0	D5H	User Flag 0
PSW.4	RS1	D4H	Register bank select 1
			Register bank select 0
PSW.3	RS0	D3H	00 = bank 0; address 00H-07H 01 = bank 1; address 08H-0FH 10 = bank 2; address 10H-17H 11 = bank 3; address 18H-1FH
PSW.2	OV	D2H	Overflow flag
PSW.1	-	D1H	Reserved
PSW.0	P	D0H	Even parity flag (ACC)

Port I/O módok

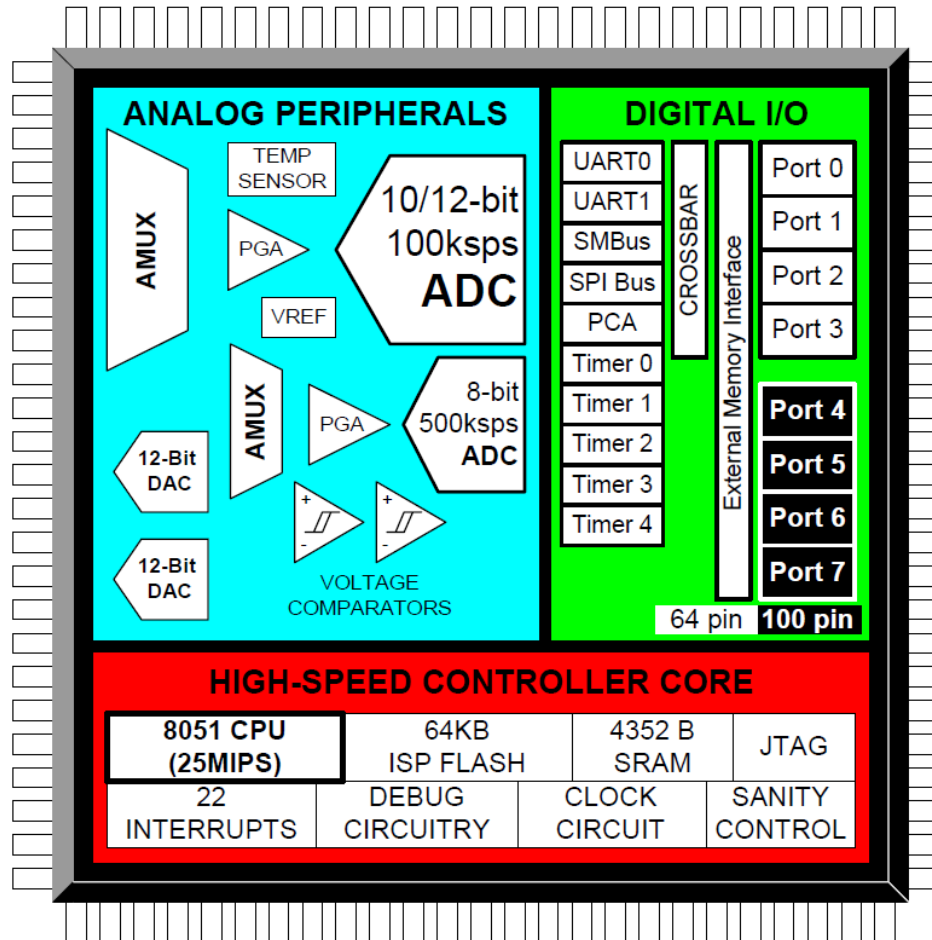


Silicon Labs C8051F020 MCU

Többlet:
 Memória
 Port 4-7
 Timer 2-4
 Oscillator
 ADC/DAC
 SFR-ek



Silicon Labs C8051F020 MCU

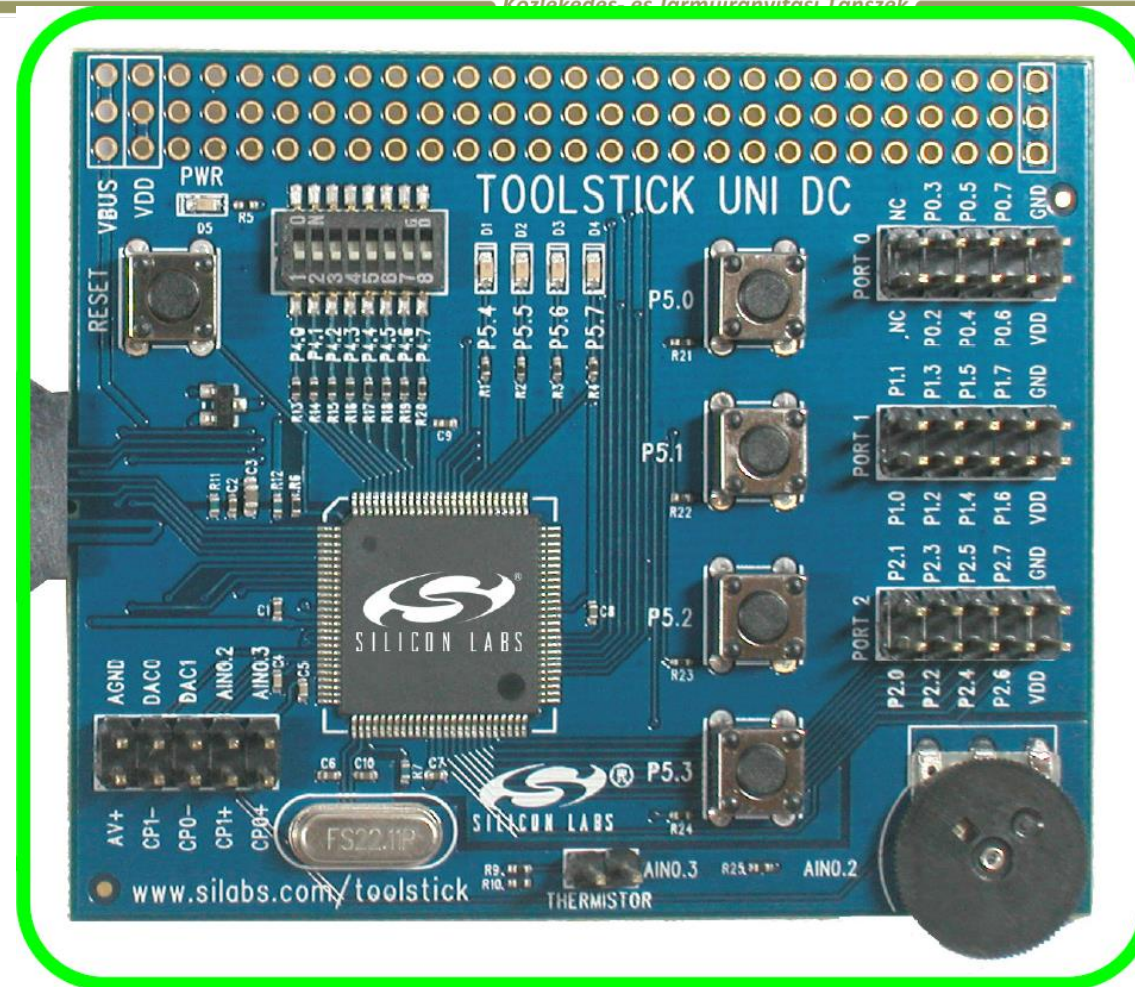


Silicon Labs ToolStick UniDC kártya

Budapesti Műszaki és Gazdaságtudományi Egyetem

Közlekedésmérnöki és Járműmérnöki Kar

Közlekedés- és Járműirányítási Tanszék



Programozás / Utasításkészlet - ASM

- Mikroprogramozott utasítás végrehajtás
 - Az utasítások egy beépített mikrokód (gépi kódú alak) segítségével kerülnek végrehajtásra
 - A mikrokód a ROM-ba „beégetve”
- Fejlesztői környezet:
 - IDE – Integrated Development Environment
- Complex Instruction Set Computer (CISC)
 - Több, bonyolultabb utasítások, általában több órajel ciklus alatt futnak le
 - A CISC processzorok utasításai általában több elemi műveletet végeznek egyszerre, így a gépi kódú programjaik rövidebbek, jobban átláthatóak egy ember számára Itt: Intel MCS[®]-51

Jelölések az utasításkészletben:

- ***Tanszéki honlap: 8051 utasításkészlet c. pdf!***
- Rn: n: 0-7, Default regiszterek, alapértelmezetten a RAM alsó 8 byte-ja, ha PSW 3. és 4. bitje 00
- @: indirekt címzés – lásd később
- direct: konkrét memóriacím 8 bit, formátum lehet: hexa, pl. – **0xFF**, bináris – pl. **0b11111111**, decimális – pl. **0d255**, vagy nincs jelölés 255
- #data: ebben az esetben adatként szerepel a kód, formátum lehet: hexa, pl. – **#0xFF**, bináris – pl. **#0b11111111**, decimális – pl. **#0d255**, vagy nincs jelölés **#255**
- bit: konkrét bit címe
- rel: ugrásoknál az ugrás helyét jelző címke neve, vagy konkrét cím a ROM területen

Makro utasítások felépítése

- Általános forma: **operation code operands**
 - (Általános forma: **instruction destination, source**)
- 3 című utasítás:
 - 1. operandus, 2. operandus, eredmény címe
 - CJNE A,direct,rel: hasonlítsd össze az acc regiszter tartalmát a direkt memóriacímű byte tartalmával, és ha a kettő nem egyenlő, akkor ugorj a rel (ROM) helyre
 - **cjne A,0x1F,vissza**
 - CJNE A,#data,rel: hasonlítsd össze az acc regiszter tartalmát a data értékkel, és ha a kettő nem egyenlő, akkor ugorj a rel helyre
 - **cjne A,#0x1F,vissza**

Makro utasítások felépítése

- 2 című utasítás :
 - 1. operandus, 2. operandus, eredmény az első operandus helyén
 - MOV Rn,direct: mozgasd be az Rn regiszterbe direct memóriacímű regiszter tartalmát
 - `mov R4,56`
 - MOV Rn,#data: mozgasd be az Rn regiszterbe a data adatot
 - `mov R4,#0x56`
- 1 című utasítás:
 - 1. operandus, eredmény az operandusban
 - SWAP A: cseréld fel az acc regiszter alsó és felső 4 bitjét
 - `swap a`

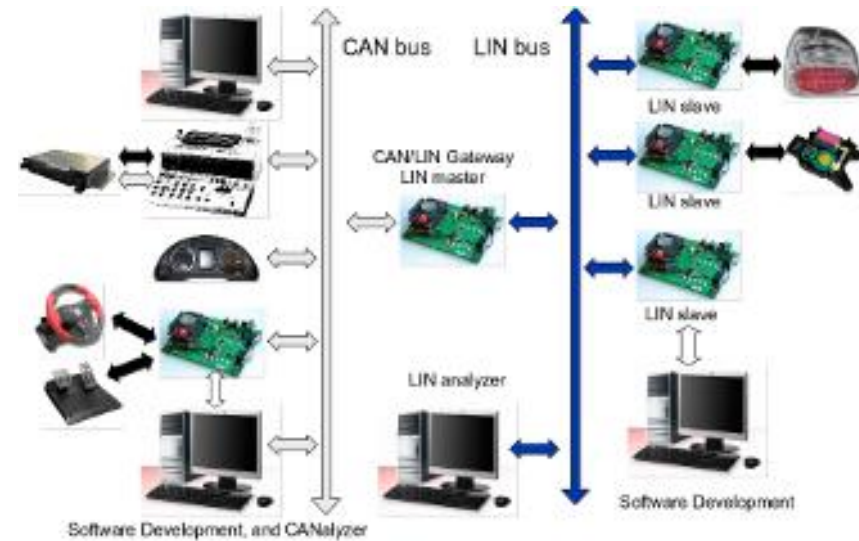
Utasítás csoportok

- Arithmetic Operations - Aritmetikai
 - add, subb, div, inc, dec
- Logical Operations - Logikai
 - anl, or, xrl, cpl, clr, rrc, rrl
- Data Transfer Operations - Adatmozgatás
 - mov, movx
- Boolean Variable Op. - Bit szintű logika
 - anl, or, cpl
- Program Branching Op. - Elágazások
 - jc, jnc, sjmp, cjne, djnz

Alkalmazási példák a járműiparban



CAN vezérlő - Freescale
MC9S12XDT512



CAN/LIN vezérlés

Alkalmazási példák a járműiparban

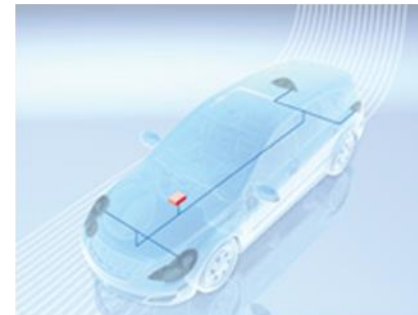
Budapesti Műszaki és Gazdaságtudományi Egyetem

Közlekedésmérnöki és Járműmérnöki Kar

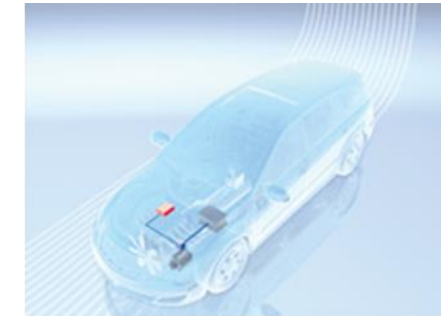
Közlekedés- és Járműirányítási Tanszék



UAV vezérlés



intelligent lighting



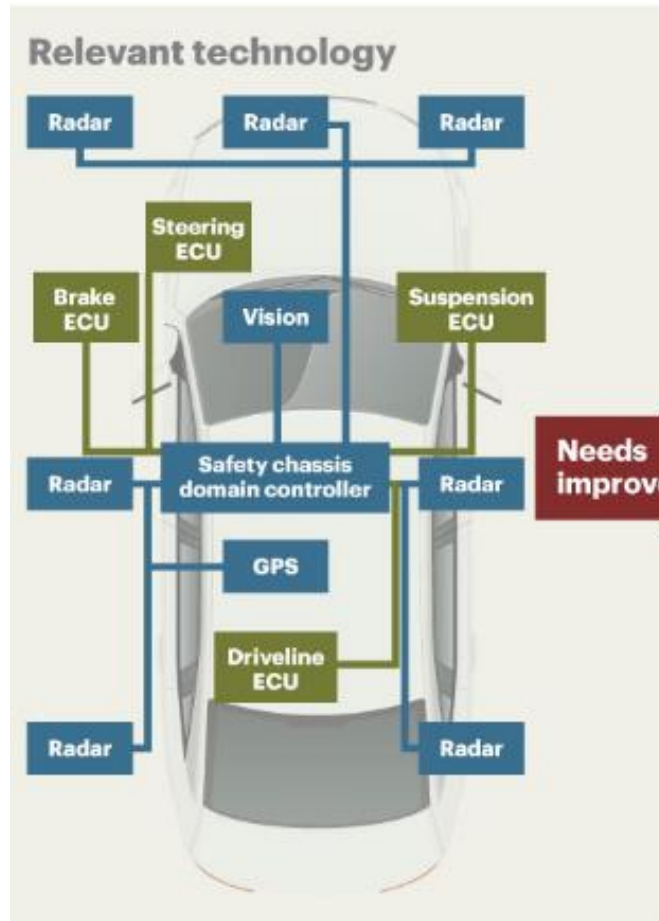
engine control

automotive sensors

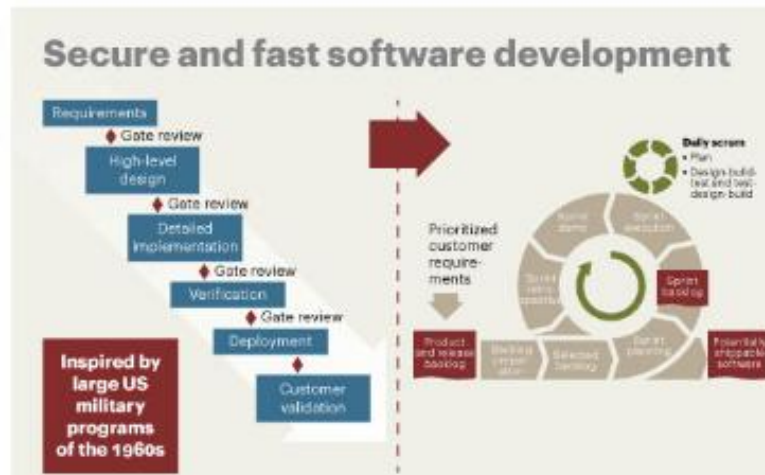
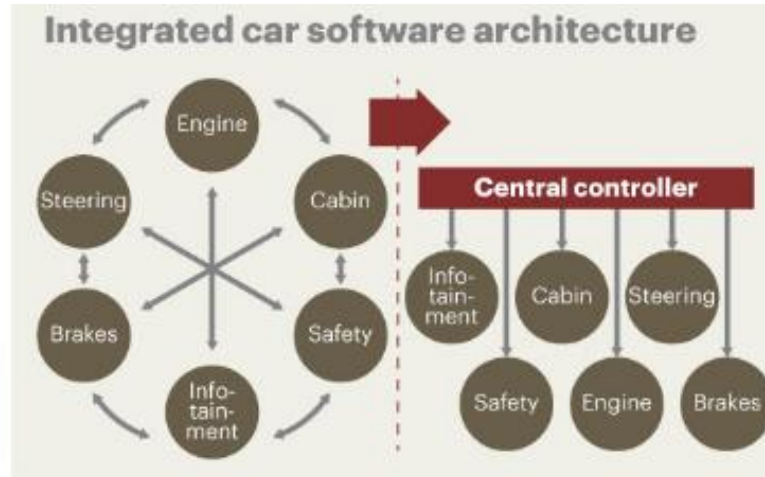
safety



Alkalmazási példák a járműiparban



Needs improvement



Alkalmazás más területen, példák

- Mosógépek
- Mikrohullámú sütők
- Parkoló automaták
- Vasúti biztonsági célú rendszerek
- Modellvasutak
- Robotok
- Fénymásolók
- Liftek
- Router
- Okostelefon
- GPS
- CT
- MRI
- Klíma
- Számítógép perifériák
- Digitális fényképezőgép, kamera
- Riasztók
- Stb...

A következő órára

- A felhasználói környezet telepítése a saját számítógépekre a honlapon található sorrendben, az ott lévő utasítások alapján:
 - **Alapértelmezett könyvtárakba telepítsük a programokat, 32 bites rendszer esetén az SDCC-hez hozzunk létre ProgramFiles(x86) mappát!**
 - **Csak a tanszéki honlapról töltsük le a program verziókat, mert előfordulhat, hogy újabb verziókkal és frissítésekkel az eszköz nem fog működni! Állítsunk be otthon egy üres ASM projektet a Telepítési útmutató alapján!**
- Windows 8/8.1/10 felhasználóknak ajánlott további beállítások:
 - <https://support.microsoft.com/hu-hu/kb/2900614> (újraindítás szükséges)
- Az IDE beállítása, lásd: Telepítési útmutató – tanszéki honlap
 - Probléma esetén keressetek meg!!! (mielőbb....)
- Otthoni feladat:
 - A ma leadott tananyag elsajátítása;
 - Utasításkészletet hozni kell a következő órára!!!!
 - Következő óra elején számonkérés lehetséges.
- Következő alkalom:
 - Az ASM utasításkészlete, bővebben;
 - Az első saját ASM program megírása.

8051, C8051F020 mikro vezérlők és programozásuk

Vége az első labornak.