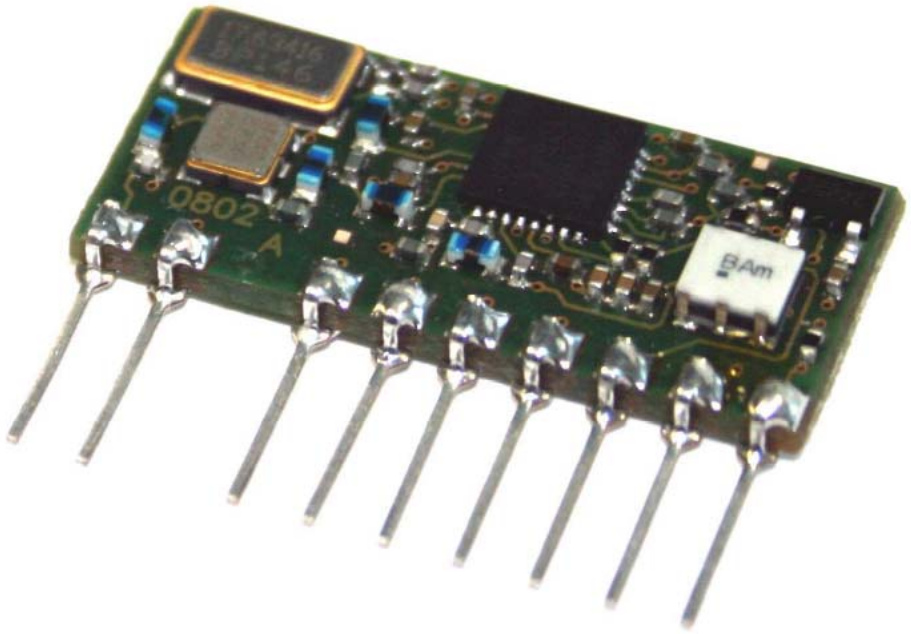
BACKEND TARGET PROPOSAL

(INFORMAL)

Real Serial Devices (e.g.: Full-Duplex 433MHz Transmitter)



Emulated Serial Devices (e.g.: Ethernet)

XPort® Pro™

World's Smallest Linux Networking Server

- Enable M2M edge computing and effortlessly handle demanding applications with the power of an advanced architecture 32-bit processor
- Run Linux or Lantronix Evolution OS for the ultimate in application development flexibility
- Production-ready hardware and software solution provides simple device set-up, configuration and monitoring, with powerful, industry-standard management tools
- Footprint compatible with the popular XPort, with everything you need in a tiny, integrated RJ45 package that is EMC/EMI/ RoHS compliant
- VIP Access technology seamlessly integrates with Lantronix' ManageLinx remote services enablement platform

Embedded Serial I/O

SPECIAL FUNCTION REGISTERS (SFR) PAGE 0

SFR Register	SFR Adrs	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Value
P0	80h	-	-	-	-	-	-	-	-	1111 1111b
SP	81h	-	-	-	-	-	-	-	-	0000 0111b
DPL0	82h	-	-	-	-	-	-	-	-	0000 0000b
DPH0	83h	-	-	-	-	-	-	-	-	0000 0000b
DPL1	84h									0000 0000b
DPH1	85h									0000 0000b
DPS	86h								DPSEL	0000 0000b
PCON	87h	OSCSTOP	INTMODEN	DEVCFGEN	SFRINDADR	GF1	GF0	PDOWN	IDLE	0110 0000b
INTEN1	88h	T1IEN	U1IEN	U0IEN	PCHGIEN0	T0IEN	SPIRXOVIEN	SPITXEIEN	-	0000 0000b
T0T1CFG	89h	-	T1GATE	T0GATE	T1CLKSRC	T1OUTEN	T1MODE8	T0OUTEN	T0MODE8	0000 0000b
UART0INT	A1h	COLEN	RXOVEN	RXAVAILEN	TXEMPTYEN	COLENF	RXOVF	RXAVENF	TXEMPTYF	0000 0001b
UART0CFG	A2h	BRADJ3	BRADJ2	BRADJ1	BRADJ0	BRCLKSRC	B9RXTX	B9EN	STOP2EN	1110 0000b
UART0BUF	A3h									0000 0000b
UART0BRL	A4h									0000 0000b
UART1INT	B1h	COLEN	RXOVEN	RXAVAILEN	TXEMPTYEN	COLENF	RXOVF	RXAVENF	TXEMPTYF	0000 0001b
UART1CFG	B2h	BRADJ3	BRADJ2	BRADJ1	BRADJ0	BRCLKSRC	B9RXTX	B9EN	STOP2EN	1110 0000b
UART1BUF	B3h									0000 0000b
UART1BRL	B4h									0000 0000b
UART1BRH	B5h									0000 0000b
UART1EXT	B6h	U1TIMERF	U1TIMEREN	U1RXSTATE	MULTIPROC	0	0	0	0	0010 0000b

Serial I/O Formalization (UART0, UART1)

- Given input stream (READ), model-dependent filtering

```
λM:MODEL.λINSTREAM:LIST BYTE8. λMEM.λADDR.
    MATCH EQC ADDR (UART0R M) WITH
        [ TRUE => MATCH INSTREAM WITH
            [ NIL => PAIR ?? CONST NIL | CONS H T => PAIR ?? H T ]
        | FALSE => ...
```

- Resulting output stream (WRITE), model-dependent filtering

```
λM:MODEL.λOUTSTREAM:LIST BYTE8. λMEM.λADDR. λVAL.
    MATCH EQC ADDR (UART0W M) WITH
        [ TRUE => PAIR ?? MEM (OUTSTREAM@[VAL])
        | FALSE => ...
```

Intel 8051

From Wikipedia, the free encyclopedia

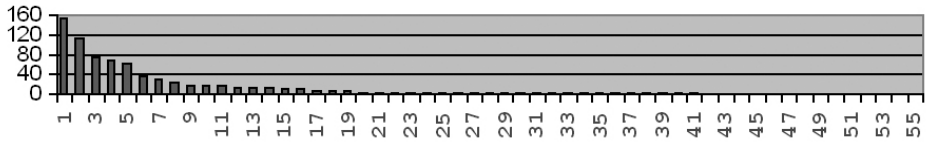
The **Intel 8051** is a Harvard architecture, single chip microcontroller (µC) which was developed by Intel in 1980 for use in embedded systems. Intel's original versions were popular in the 1980s and early 1990s, but has today largely been superseded by a vast range of faster and/or functionally enhanced 8051-compatible devices manufactured by more than 20 independent manufacturers including Atmel, Infineon Technologies (formerly Siemens AG), Maxim Integrated Products (via its Dallas Semiconductor subsidiary), NXP (formerly Philips Semiconductor), Nuvoton (formerly Winbond), ST Microelectronics, Silicon Laboratories (formerly Cygnal), Texas Instruments and Cypress Semiconductor. Intel's official designation for the 8051 family of µCs is **MCS 51**.

Intel's original 8051 family was developed using NMOS technology, but later versions, identified by a letter C in their name (e.g., 80C51) used CMOS technology and were less power-hungry than their NMOS predecessors. This made them more suitable for battery-powered devices.

Intel P8051 microcontroller.

SAB-C515-LN by Infineon is based on the 8051

Keil µVision3 8051 Support List



BRAND	MCUS	BRAND	MCUS
1 - NXP	154	29 - OKI	3
2 - Silicon Laboratories Inc.	111	30 - Teridian Semiconductor Co.	3
3 - Atmel	74	31 - Tezzaron Semiconductor	3
4 - Infineon	68	32 - Acer Labs	2
5 - Winbond	61	33 - Cast Inc.	2
6 - Hynix Semiconductor	39	34 - CybraTech	2
7 - Texas Instruments	29	35 - Daewoo	2
8 - Dallas Semiconductor	25	36 - EasyPlug	2
9 - Ramtron	17	37 - Intel	2
10 - SMSC	17	38 - Maxim	2
11 - Megawin	16	39 - Syntek Semiconductor Co.	2
12 - Analog Devices	14	40 - Tekmos	2
13 - SST	13	41 - Zensys	2
14 - Digital Core Design	12	42 - Actel	1
15 - SyncMOS	11	43 - Aeroflex	1
16 - ISSI	10	44 - CML Microcircuits	1
17 - Myson Technology	8	45 - Cybernetic Microsystems	1
18 - Chipcon	6	46 - Genesis Microchip	1
19 - MXIC	6	47 - Handshake Solutions	1
20 - Cypress Semiconductor	5	48 - Honeywell	1
21 - RDC Semiconductor	5	49 - InnoASIC	1
22 - Dolphin	4	50 - Oregano Systems	1
23 - Mentor Graphics Co.	4	51 - Sanyo	1
24 - Micronas	4	52 - Sharp	1
25 - Triscend	4	53 - Siliconians	1
26 - Zylogic Semiconductor Co.	4	54 - Synopsys	1
27 - EM Microelectronic	3	55 – Vitesse	1
28 - Evatronix	3	**TOTAL**	**770**

NXP (ex Philips) [154 models]

Key Statistics

Top Locations

- **Eindhoven Area, Netherlands** (500+)
- **Nijmegen Area, Netherlands** (500+)
- **San Francisco Bay Area** (359)
- **Hamburg Area, Germany** (317)
- **Bangalore Area, India** (314)
- **Le Havre Area, France** (274)

▸ Headquarters Address

HQ Region	Eindhoven Area, Netherlands
Industry	Semiconductors
Type	Privately Held
Status	Operating
Company Size	30,000 employees
2008 Revenue	$5,443,000
Website	http://www.nxp.com

Silicon Laboratories [111 models]

Key Statistics

Top Locations

- **Austin, Texas Area** (315)
- **San Francisco Bay Area** (32)

▸ Headquarters Address

HQ Region	Austin, Texas Area
Industry	Semiconductors
Type	Public Company
Status	Operating
Company Size	727 employees
2006 Revenue	$465 mil (9%)
Founded	1996
Website	http://www.silabs.com

Atmel [74 models]

Key Statistics

Top Locations

- Colorado Springs, Colorado Area (324)
- San Francisco Bay Area (296)
- Marseille Area, France (201)
- Nantes Area, France (51)

▸ Headquarters Address

HQ Region	San Francisco Bay Area
Industry	Semiconductors
Type	Public Company
Status	Operating
Company Size	6,000 employees
2006 Revenue	$1,671 mil (7%)
Founded	1984
Website	http://www.atmel.com

Infineon [68 models]

Key Statistics

Top Locations

- **Munich Area, Germany** (500+)
- **Bangalore Area, India** (304)
- **San Francisco Bay Area** (293)
- **Ålborg Area, Denmark** (143)

▸ Headquarters Address

HQ Region	Munich Area, Germany
Industry	Semiconductors
Type	Public Company
Status	Operating
Company Size	29,100 employees
2008 Revenue	5,000,000,000 [EUR]
Founded	1999
Website	http://www.infineon.com

Winbond [61 models]

Key Statistics

Top Locations

- **San Francisco Bay Area** (50)

▸ Headquarters Address

HQ Region	San Francisco Bay Area
Industry	Semiconductors
Type	Public Company
Status	Operating
Company Size	5001-10,000 employees
2006 Revenue	35,477 mil [TWD] (24%)
Founded	1987
Website	http://www.winbond.com

Hynix Semiconductors [39 models]

Key Statistics

Top Locations

- **Eugene, Oregon Area** (65)
- **San Francisco Bay Area** (43)

▸ Headquarters Address

HQ Region	Eugene, Oregon Area
Industry	Semiconductors
Type	Privately Held
Status	Operating
Company Size	113 employees
2006 Revenue	7,731,932 mil [KRW] (30%)
Founded	1983
Website	http://hsa.hynix.com/us_s...

Texas Instruments [29 models]

Key Statistics

Top Locations

- **Dallas/Fort Worth Area** (500+)
- **Nice Area, France** (500+)
- **Houston, Texas Area** (497)
- **Bangalore Area, India** (319)

▸ Headquarters Address

HQ Region	Dallas/Fort Worth Area
Industry	Semiconductors
Type	Public Company
Status	Operating
Company Size	32,000 employees
2007 Revenue	$13,835 mil
Founded	1930
Website	http://www.ti.com

Dallas Semiconductors [25 models]

Key Statistics

Top Locations

- **Dallas/Fort Worth Area** (89)

▸ Headquarters Address

HQ Region	Dallas/Fort Worth Area
Industry	Semiconductors
Type	Public Company
Status	Operating Subsidiary
Company Size	1001-5000 employees
Founded	1984
Website	http://www.dalsemi.com

Ramtron [17 models]

Key Statistics

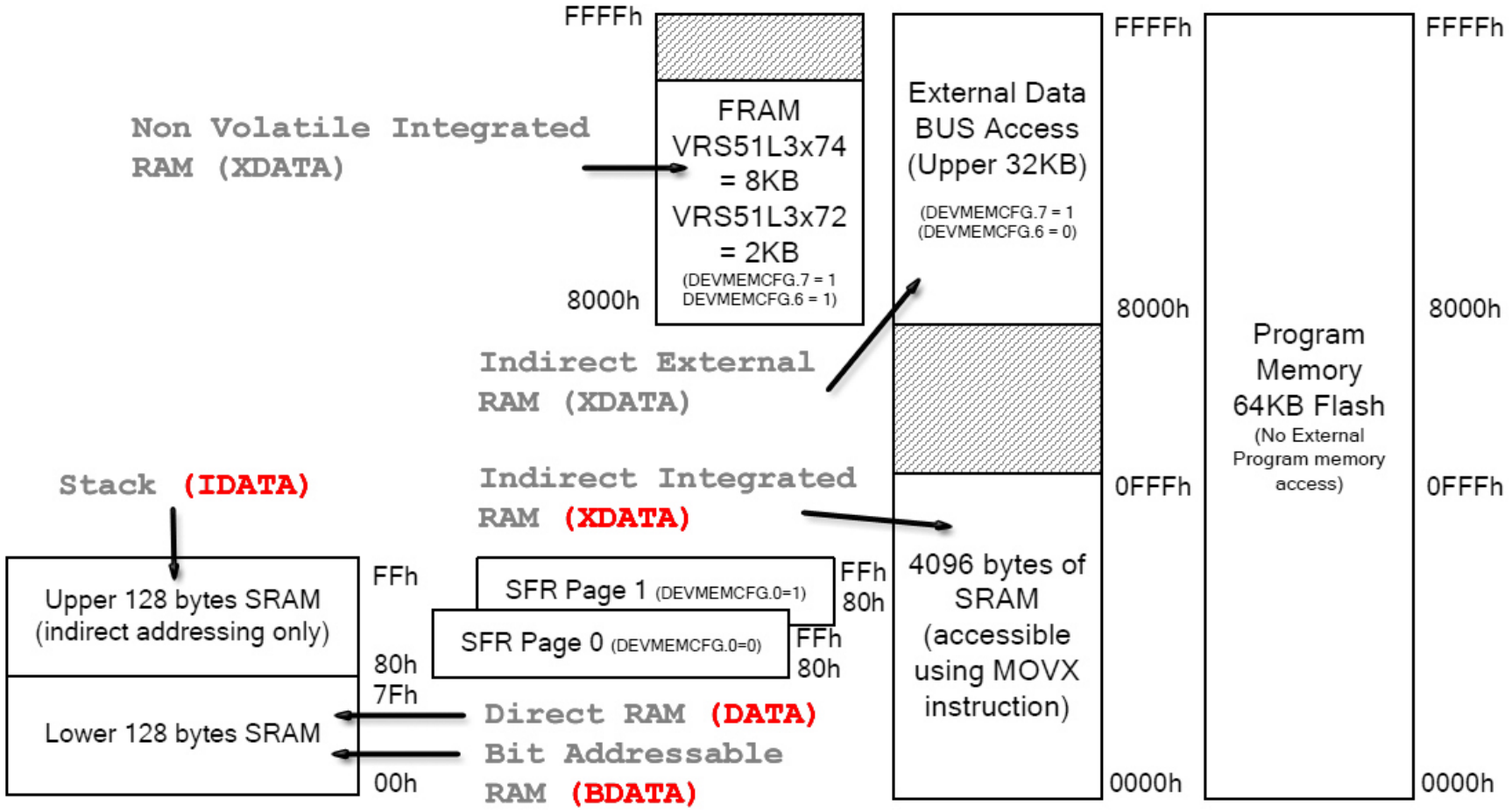
Top Locations

- Colorado Springs, Colorado Area (38)
- Greater San Diego Area (7)

▸ Headquarters Address

HQ Region	Colorado Springs, Colorado Area
Industry	Semiconductors
Type	Public Company
Status	Operating
Company Size	51-200 employees
2006 Revenue	$40 mil (18%)
Founded	1984
Website	http://www.ramtron.com

Memory Architecture



SRAM 1/2

R0-R7 Bank Registers [0x00-0x1F]

- [0x00-0x07]: R0-R7 Bank0 [0x08-0x0F]: R0-R7 Bank1
- [0x10-0x17]: R0-R7 Bank2 [0x18-0x1F]: R0-R7 Bank3

Bit Addressable Fast Memory [0x20-0x2F]

- Code example [var]: ***bdata*** *char bitVar;*
- Code example [bit]: ***sbit*** *fastBit = bitVar ^ 0;*

Direct Addressable Fast Memory [0x30-0x7F]

- Code example [var]: ***data*** *char dirVar;*

Indirect Addressable Fast Memory [0x80-0xFF]

- Code example [var]: ***idata*** *char indirVar;*

SRAM 2/2

Special Function Registers (SFR) [0x80-0xFF]

- [0x81] memory mapped **SP**
- [0x82] memory mapped **DP-Low**
- [0x83] memory mapped **DP-High**
- [0xD0] memory mapped **STATUS** (*CHURROUP*)
  C *= Carry,* ***H*** *= Half-carry,* ***O*** *= Overflow,* ***P*** *= Parity,*
  RR *= R0-R7 Bank select,* ***UU*** *= User1 & User2 flag*
- [0xE0] memory mapped **A**
- [0xF0] memory mapped **B**

Hardware Stack, PUSH/POP [0x00-0xFF]

- Usually decreasing 0xFF → 0x??

XRAM

Integrated External Memory [0x0000-0x????]

- Software emulated STACK
- Code example [var]: ***xdata*** *char slowVar;*

FLASH

Program Memory [0x0000-0x????]

- Code example [var]: ***code*** *char slowVar;*

Bit Address [1 Byte Operand]

- [0x00-0x7F] bit addressable memory → [0x20-0x2F]
- [0x80-0xFF] aligned sfr → [0x80, 0x88, 0x90, 0x98, …, 0xF8]

Register R0-R7 [3 Bits Implicit Operand]

Direct [1 Byte Operand]

- [0x00-0x7F] internal SRAM
- [0x80-0xFF] sfr

Indirect (@R0-R1) [1 Bit Implicit Operand]

- [0x00-0xFF] internal SRAM

Some Irregular Instructions…

ASSEMBLER [WWW.NXP.COM]

Mnemonic	Description	Size	Hex Code
ADD A, Rn	Add register to A	1	28h-2Fh
ADD A, direct	Add direct byte to A	2	25h
ADD A, @Ri	Add data memory to A	1	26h-27h
ADD A, #data	Add immediate to A	2	24h
ADDC A, Rn	Add register to A with carry	1	38h-3Fh
ADDC A, direct	Add direct byte to A with carry	2	35h
ADDC A, @Ri	Add data memory to A with carry	1	36h-37h
ADDC A, #data	Add immediate to A with carry	2	34h
SUBB A, Rn	Subtract register from A with borrow	1	98h-9Fh
SUBB A, direct	Subtract direct byte from A with borrow	2	95h
SUBB A, @Ri	Subtract data mem from A with borrow	1	96h-97h
SUBB A, #data	Subtract immediate from A with borrow	2	94h
INC A	Increment A	1	04h
INC Rn	Increment register	1	08h-0Fh
INC direct	Increment direct byte	2	05h
INC @Ri	Increment data memory	1	06h-07h
DEC A	Decrement A	1	14h
DEC Rn	Decrement register	1	18h-1Fh
DEC direct	Decrement direct byte	2	15h
DEC @Ri	Decrement data memory	1	16h-17h
INC DPTR	Increment data pointer	1	A3h
MUL AB	Multiply A by B	1	A4h
DIV AB	Divide A by B	1	84h
DA A	Decimal adjust A	1	D4h

ASSEMBLER [WWW.NXP.COM]

Mnemonic	Description	Size	Hex Code
ANL A, Rn	AND register to A	1	58h-5Fh
ANL A, direct	AND direct byte to A	2	55h
ANL A, @Ri	AND data memory to A	1	56h-57h
ANL A, #data	AND immediate to A	2	54h
ANL direct, A	AND A to direct byte	2	52h
ANL direct, #data	AND immediate data to direct byte	3	53h
ORL A, Rn	OR register to A	1	48h-4Fh
ORL A, direct	OR direct byte to A	2	45
ORL A, @Ri	OR data memory to A	1	46h-47h
ORL A, #data	OR immediate to A	2	44h
ORL direct, A	OR A to direct byte	2	42h
ORL direct, #data	OR immediate data to direct byte	3	43h
XRL A, Rn	Exclusive-OR register to A	1	68h-6Fh
XRL A, direct	Exclusive-OR direct byte to A	2	65h
XRL A, @Ri	Exclusive-OR data memory to A	1	66h-67h
XRL A, #data	Exclusive-OR immediate to A	2	64h
XRL direct, A	Exclusive-OR A to direct byte	2	62h
XRL direct, #data	Exclusive-OR immediate to direct byte	3	63h
CLR A	Clear A	1	E4h
CPL A	Compliment A	1	F4h
SWAP A	Swap nibbles of A	1	C4h
RL A	Rotate A left	1	23h
RLC A	Rotate A left through carry	1	33h
RR A	Rotate A right	1	03h
RRC A	Rotate A right through carry	1	13h

Mnemonic	Description	Size	Hex Code
CLR C	Clear Carry bit	1	C3h
CLR bit	Clear bit	2	C2h
SETB C	Set Carry bit to 1	1	D3h
SETB bit	Set bit to 1	2	D2h
CPL C	Complement Carry bit	1	B3h
CPL bit	Complement bit	2	B2h
ANL C,bit	Logical AND between Carry and bit	2	82h
ANL C,#bit	Logical AND between Carry and not bit	2	B0h
ORL C,bit	Logical ORL between Carry and bit	2	72h
ORL C,#bit	Logical ORL between Carry and not bit	2	A0h
MOV C,bit	Copy bit value into Carry	2	A2h
MOV bit,C	Copy Carry value into Bit	2	92h
NOP	No operation	1	00h

Mnemonic	Description	Size	Hex Code
ACALL addr 11	Absolute call to subroutine	2	11h-F1h
LCALL addr 16	Long call to subroutine	3	12h
RET	Return from subroutine	1	22h
RETI	Return from interrupt	1	32h
AJMP addr 11	Absolute jump unconditional	2	01h-E1h
LJMP addr 16	Long jump unconditional	3	02h
SJMP rel	Short jump (relative address)	2	80h
JC rel	Jump on carry = 1	2	40h
JNC rel	Jump on carry = 0	2	50h
JB bit, rel	Jump on direct bit = 1	3	20h
JNB bit, rel	Jump on direct bit = 0	3	30h
JBC bit, rel	Jump on direct bit = 1 and clear	3	10h
JMP @A+DPTR	Jump indirect relative DPTR	1	73h
JZ rel	Jump on accumulator = 0	2	60h
JNZ rel	Jump on accumulator 1= 0	2	70h
CJNE A, direct, rel	Compare A, direct JNE relative	3	B5h
CJNE A, #d, rel	Compare A, immediate JNE relative	3	B4h
CJNE Rn, #d, rel	Compare reg, immediate JNE relative	3	B8h-BFh
CJNE @Ri, #d, rel	Compare ind, immediate JNE relative	3	B6h-B7h
DJNZ Rn, rel	Decrement register, JNZ relative	2	D8h-DFh
DJNZ direct, rel	Decrement direct byte, JNZ relative	3	D5

ASSEMBLER [WWW.NXP.COM]

Mnemonic	Description	Size	Hex Code
MOV A, Rn	Move register to A	1	E8h-EFh
MOV A, direct	Move direct byte to A	2	E5h
MOV A, @Ri	Move data memory to A	1	E6h-E7h
MOV A, #data	Move immediate to A	2	74h
MOV Rn, A	Move A to register	1	F8h-FFh
MOV Rn, direct	Move direct byte to register	2	A8h-AFh
MOV Rn, #data	Move immediate to register	2	78h-7Fh
MOV direct, A	Move A to direct byte	2	F5h
MOV direct, Rn	Move register to direct byte	2	88h-8Fh
MOV direct, direct	Move direct byte to direct byte	3	85h
MOV direct, @Ri	Move data memory to direct byte	2	86h-87h
MOV direct, #data	Move immediate to direct byte	3	75h
MOV @Ri, A	Move A to data memory	1	F6h-F7h
MOV @Ri, direct	Move direct byte to data memory	2	A6h-A7h
MOV @Ri, #data	Move immediate to data memory	2	76h-77h
MOV DPTR, #data	Move immediate to data pointer	3	90h
MOVC A, @A+DPTR	Move code byte relative DPTR to A	1	93h
MOVC A, @A+PC	Move code byte relative PC to A	1	83h
MOVX A, @Ri	Move external data (A8) to A	1	E2h-E3h
MOVX A, @DPTR	Move external data (A16) to A	1	E0h
MOVX @Ri,A	Move A to external data (A8)	1	F2h-F3h
MOVX @DPTR, A	Move A to external data (A16)	1	F0h
PUSH direct	Push direct byte onto stack	2	C0h
POP direct	Pop direct byte from stack	2	D0h
XCH A, Rn	Exchange A and register	1	C8h-CFh
XCH A, direct	Exchange A and direct byte	2	C5h
XCH A, @Ri	Exchange A and data memory	1	C6h-C7h
XCHD A, @Ri	Exchange A and data memory lower nibble	1	D6h-D7h

Memory

- 256B SRAM, organized as *bdata, data, idata*
- Max 64Kb XRAM as (*model* → *xdata*)
- Max 64Kb FLASH as (*model* → *code*)

Stack

- Hardware (SRAM) or software emulated (XRAM)

ALU

- 4 memory mapped *R0-R7* bank registers
- Memory mapped sfr (*SP, DP, STATUS, A, B,* (*model* → *UART0/1*))

Unified Assembler (Instructions & Addressing Modes & Opcodes)

- Discard *RETI* (no interrupt handling)
- Discard *MOVC A,@A+PC* (no switch-like jump tables)
- Model dependent execution timing

IDE

- µVision3 [**www.keil.com**] (*Compare benchmark!*)
- RIDE [**www.raisonance.com**]
- SPJ IDE51 [**www.spjsystems.com**]

Freeware

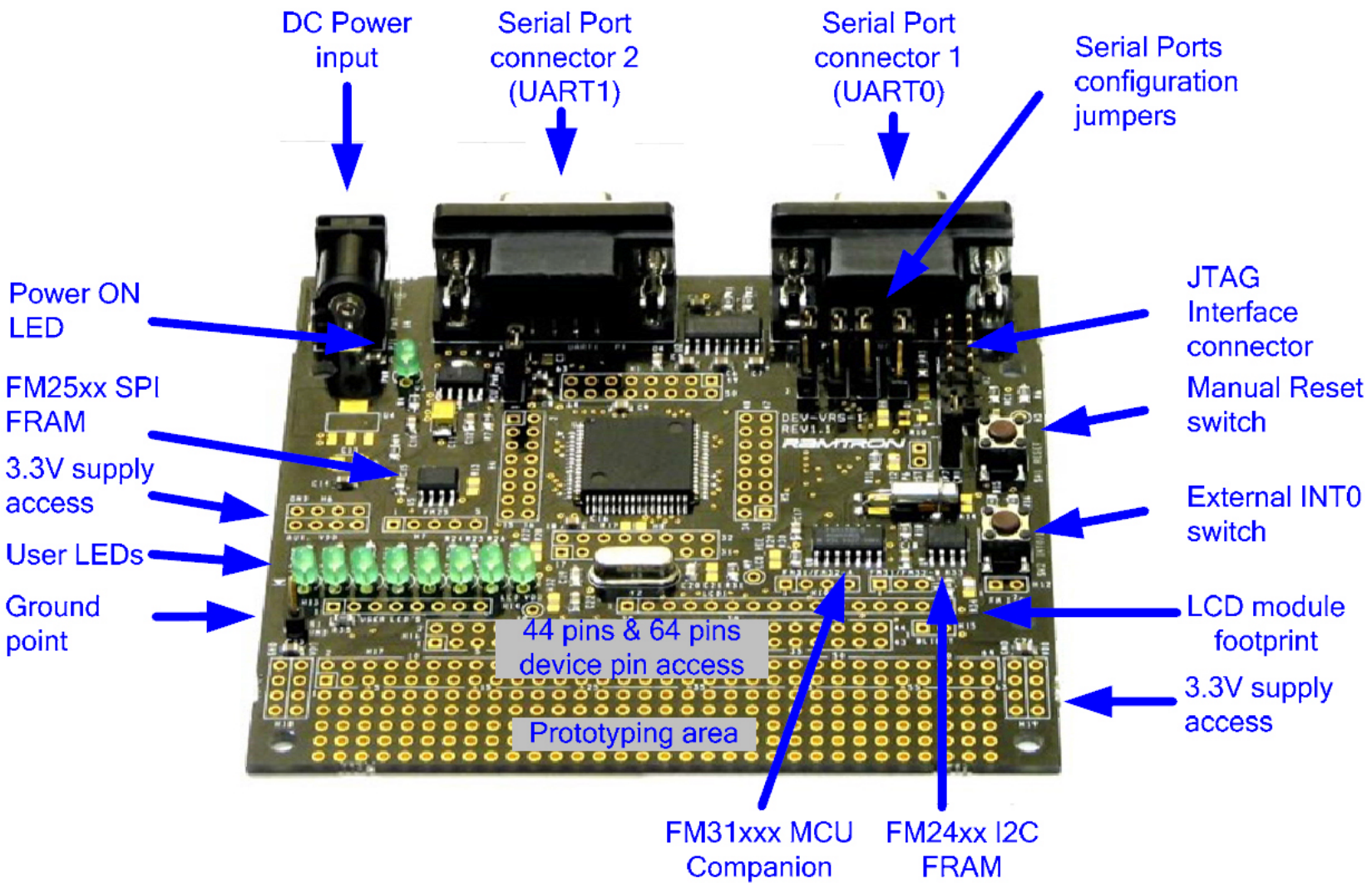
- SDCC [**sdcc.sourceforge.net**] (*Compare benchmark!*)
- Small-C [**www.rigelcorp.com**]

Emulation

- Embedded benchmark/testing refers to *in-circuit* emulation **only**!
- Choose some *development boards* from different manufacturers (easy USB/parallel programming/debugging)!

E.G.: VERSAKIT-30XX (RAMTRON) ≈ 100€

DEVELOPMENT BOARD [WWW.RAMTRON.COM]



Preliminary

- Ask Keil for a µVision3 academic license…
- Choose some 8051 European manufacturers…
- Get a dev-board from each chosen manufacturer…
- Get full assembly semantic from manufacturers or via dev-board…

Implementation - Optimization level [e.g.: µVision3]

- 0 = *Constant folding* 1 = *Dead code elimination*
- 2 = *Data overlaying* 3 = *Peephole optimization*
- 4 = *Register variables* 5 = *Common subexpression elimination*
- 6 = *Loop rotation* 7 = *Extended index access optimization*
- 8 = *Reuse common entry code* (**default**)
- 9 = *Common block subroutines* (dangerous, need robust code!)

Benchmark

- Compile test program with µVision3, SDCC, and CerCo…
- Size & Speed comparison, run real targets…