COVENTRY UNIVERSITY

Faculty Of Engineering & Computing

A brief introduction to the PIC18 microcontroller

Objectives: To provide a pre-Lab introduction to the PIC18's hardware and software resources. The aim is to help in understanding the experimentation that will follow. This will involve the use of hardware and software tools associated with the various aspects of microcontroller interfacing.

Introduction: The PIC family of micros is manufactured by Microchip Technology Inc. It was formed from a group that stemmed from a semiconductor division of General Instruments back in 1989. The PIC name comes from the initials of **P**rogrammable **I**nterface **C**ontroller. Microchip manufactures a wealth of PIC devices containing a wealth of embedded peripheral devices. A product map of available devices is shown below:

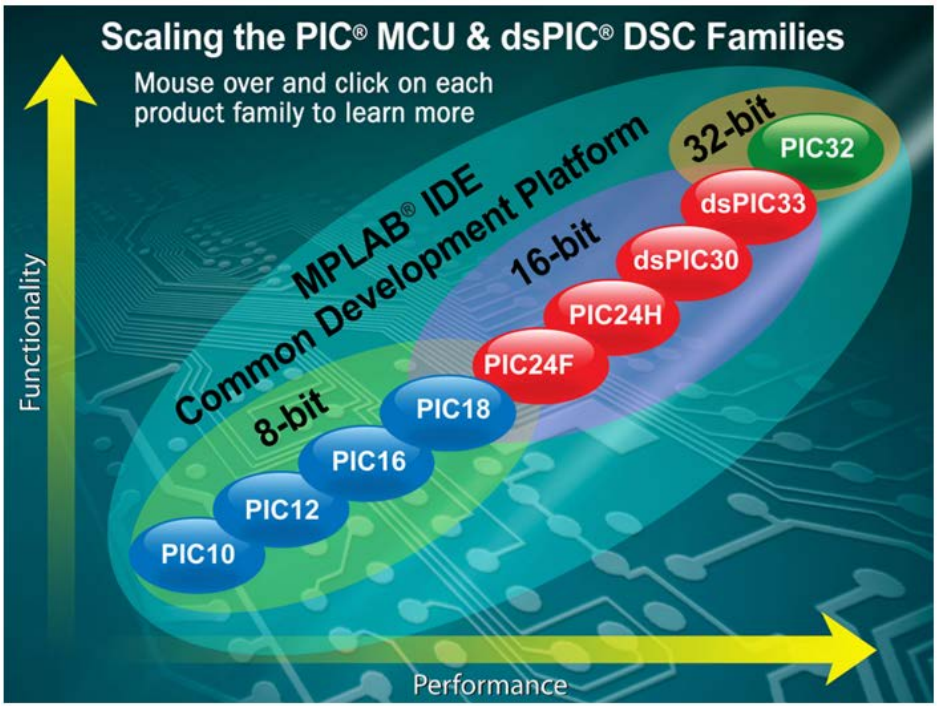


Fig. 1 PIC Microcontroller family

http://www.microchip.com/

Figure 1 above shows the current PIC micro product family. At the low end of the family we have the PIC10 micro. This is an 8-bit micro is offered in 6 and 8 pin configuration operating at around 5 MIPS (Million Instructions per Second). The next higher member of the family is the PIC12 and

PIC16. These offer midrange architecture, operating at 5-8 MIPS with a reasonable size of memory. At a higher level we have the PIC18 family. These are advanced high-performance devices operating at 10-16 MIPS and offer large amounts of memory and a diverse variety of on-chip peripheral devices. In the experimentation that will follow we will focus on this particular micro.
At the top of the family tree we have the 16-bit PIC24x the dsPIC family members and followed at the very top by the dsPIC33 and PIC32 series of micros which are 32-bit devices.

The PIC18 that we will be using consists of three architectures:

The standard PI18F SERIES,
The PIC18J series which are 10-12 MIPS low voltage devices with USB, Ethernet or LCD
The PIC18K series 16 MIPS high performance, low power micros.

As already mentioned the processor involved in our experimentation will be the standard PIC18F4520. Some of the other family members are also available for final year project work.

Table 1 PIC18F family members

Feature	PIC18F242	PIC18F252	PIC18F442	PIC18F452
Program memory (bytes)	16K	32K	16K	32K
Data memory (bytes)	768	1536	768	1536
EEPROM (bytes)	256	256	256	256
I/O Ports	A,B,C	A,B,C	A,B,C,D,E	A,B,C,D,E
Timers	4	4	4	4
Interrupt sources	17	17	18	18
Capture/compare/PWM	2	2	2	2
Serial communication	MSSP USART	MSSP USART	MSSP USART	MSSP USART
A/D converter (10 bits)	5 channels	5 channels	8 channels	8 channels
Low-voltage detect	Yes	Yes	Yes	Yes
Brown-out reset	Yes	Yes	Yes	Yes
Packages	28-pin DIP 28-pin SOIC	28-pin DIP 28-pin SOIC	40-pin DIP 44-pin PLCC 44-pin TQFP	40-pin DIP 44-pin PLCC 44-pin TQFP

PIC18F2420/2520/4420/4520

Device	Program Memory		Data Memory		I/O	10-bit A/D (ch)	CCP/ ECCP (PWM)	MSSP		EUSART	Comp.	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI	Master I^2C			
PIC18F2420	16K	8192	768	256	25	10	2/0	Y	Y	1	2	1/3
PIC18F2520	32K	16384	1536	256	25	10	2/0	Y	Y	1	2	1/3
PIC18F4420	16K	8192	768	256	36	13	1/1	Y	Y	1	2	1/3
PIC18F4520	32K	16384	1536	256	36	13	1/1	Y	Y	1	2	1/3

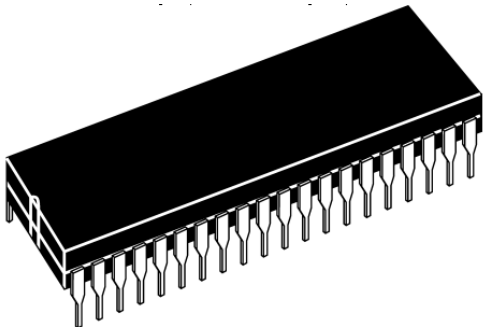
Data Bus Width	8Bit
Device Core	PIC
Family Name	PIC18
Instruction Set Architecture	RISC
Interface Type	MSSP/SPI/I2C/PSP/USART
Maximum Clock Rate	40MHz
Maximum Operating Temperature	85°C
Maximum Speed	40MHz
Minimum Operating Temperature	-40°C
Mounting	Surface Mount
Number of Programmable I/Os	36
Number of Timers	4
On-Chip ADC	13-chx10-bit
Pin Count	44
Product Height	1
Product Length	10
Product Width	10
Program Memory Size	32
Program Memory Type	Flash
RAM Size	1.5KB
Supplier Package	TQFP
Typical Operating Supply Voltage	5V

Data Bus Width	8Bit
Device Core	PIC
Family Name	PIC18
Instruction Set Architecture	RISC
Interface Type	MSSP/SPI/I2C/PSP/USART
Maximum Clock Rate	40MHz
Maximum Operating Temperature	85°C
Maximum Speed	40MHz
Minimum Operating Temperature	-40°C
Mounting	Through Hole
Number of Programmable I/Os	36
Number of Timers	4
On-Chip ADC	13-chx10-bit
Pin Count	40
Product Height	3.81
Product Length	52.26
Product Width	13.84
Program Memory Size	32
Program Memory Type	Flash
RAM Size	1.5KB
Supplier Package	PDIP
Typical Operating Supply Voltage	5V

Device Footprint and Pin-outs

SMD version

Fig. 2 (a)



DIP version

Fig. 2 (b)

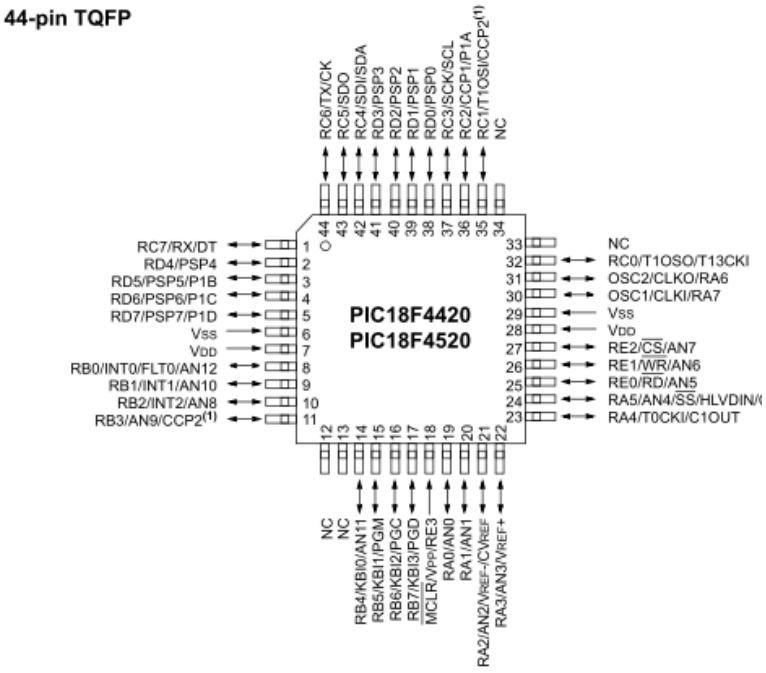


Fig. 3 (a)

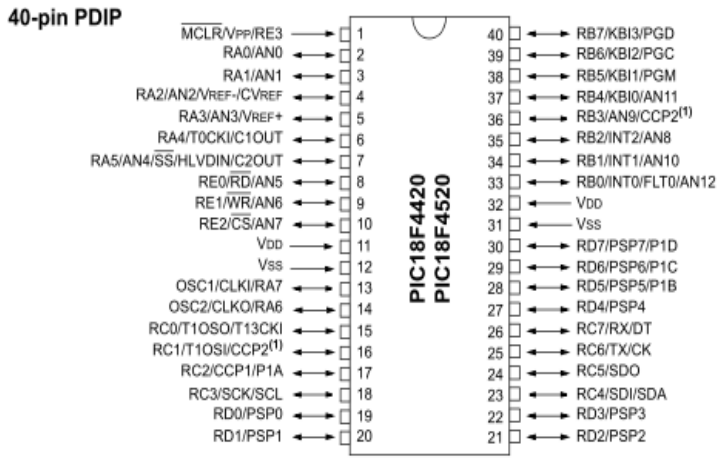


Fig. 3 (b)

The PIC18F architecture

There are mainly two types of microcontroller architectures. The Von Newmann and the Harvard . A great number of micros use the Von Newmann architecture. In this architecture the data and program instructions share the same bus and are as shown below in Fig. 4 (a). The PIC family of micros use the Harvard architecture as shown in Fig. 4b. Here instructions and data have different buses. This allows code and data to be fetched simultaneously resulting in an improved performance.

The PIC micro is a RISC (Reduced Instruction Set Computer) micro. The PIC18F4520 has 77 instructions as compared to the 209 instructions of the 68HC12 which is CISC (Complex Instruction Set) micro.

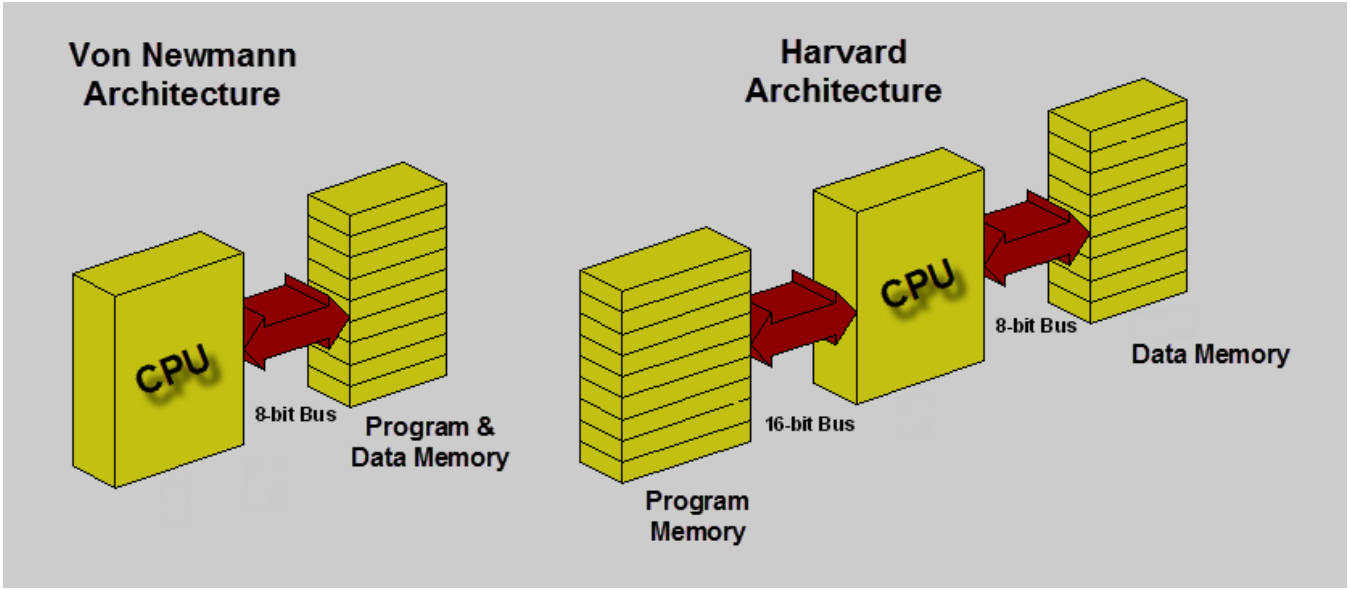


Fig. 4(a)

Fig. 4(b)

A simple block diagram of the PIC18 is as shown below in Fig. 5. The CPU consists of an 8-bit ***ALU*** (**A**rithmetic **L**ogic **U**nit), a ***Working Register*** (WREG), the ***Instruction Decoder,*** the ***Status Register,*** the ***Program Counter*** (PC), the ***Bank Select Register*** (BSR) the ***File Select Registers*** (FSR), and the ***Control Unit.***

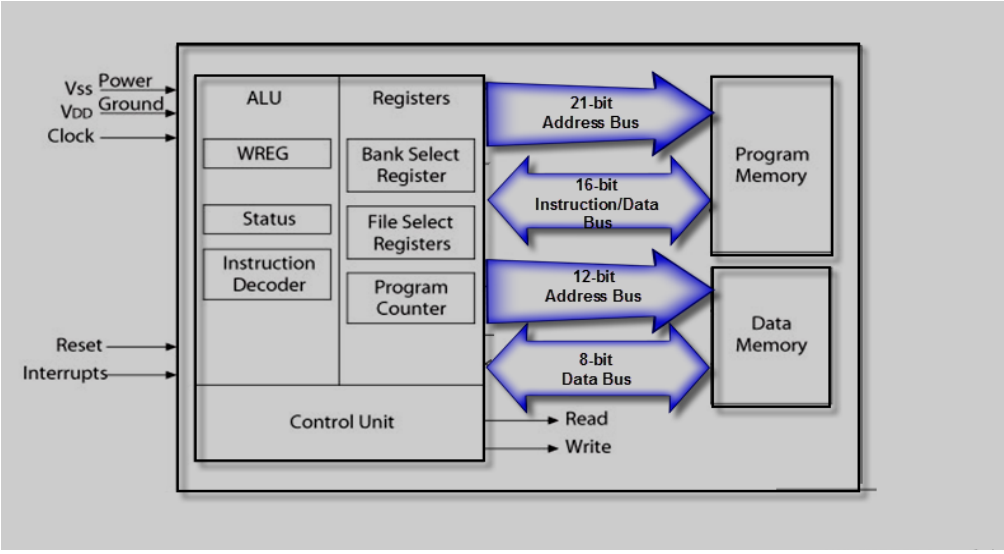


Fig. 5 Block Diagram PIC18F4520

ALU

The ALU section of the CPU is responsible for carrying out arithmetic functions such as add, subtract, multiply and divide as well as logic functions such as AND, OR, and NOT.

Working Register (WREG)

In the CPU, registers are used to store temporary information. This information could be a byte of information to be processed, or an address to be used in fetching data. The majority of the PIC registers are 8-bit. On most micros the corresponding WREG is the Accumulator (A). Most data transfers will pass through the WREG. It is an 8-bit register. Any data larger than 8-bits will have to be broken into 8-bit chunks before they are processed.

Status Register

The Status Register contains the arithmetic status of the Arithmetic Logic unit. It is an 8-bit register but only the Least Significant 5-bits are used that indicate operational results as follows:

N : ***Negative flag***, set when bit 7 is one as a result of an arithmetic or logic operation
OV: ***Overflow flag***, set when the result of an operation of signed numbers is beyond 7-bits
Z: ***Zero flag*** set when the result of an operation is zero
DC: ***Digit Carry*** or Half Carry set when carry generated from Bit3 to Bit4 in an arithmetic operation
C: ***Carry flag***, set when an addition generates a carry

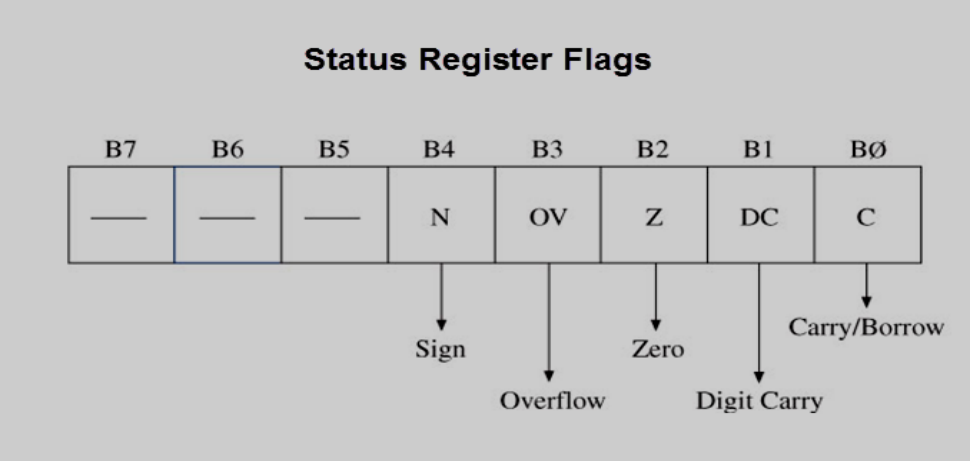


Fig. 6

Instruction Decoder

To execute an instruction the processor copies the instruction code from the program memory into the instruction register. It can then be decoded by the instruction decoder. The Instruction Decoder is a combinatorial logic block which sets up the CPU control lines as needed.

Program Counter

The Program Counter is a 21-bit register that specifies the address of the instruction to fetch for execution. Being 21-bits wide is formed by three separate 8-bit registers. The low byte is known as the PCL register and both readable and writable, the high byte is known as PCH register and contains the bits 8-15. It is not directly readable or writable. The upper byte is called the PCU. This register contains bits 16-20 and it also not readable or writable.

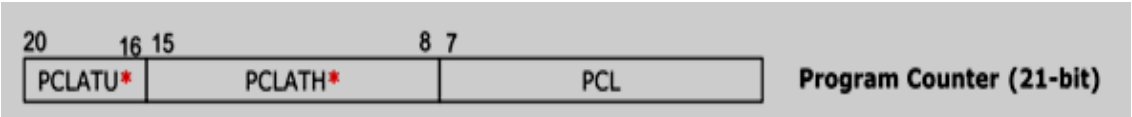


Fig. 7 Program Counter

Control Unit

The Control Unit provides and control signals to the various read and write operations.

File Select Registers

The File Select Registers (FSR) are 16-bit registers used as memory pointers for indirect addressing of data memory. Indirect addressing (see in later work) allows the user to access a location in data memory without giving a fixed address in the instruction. This as mentioned is achieved by the using the File Select Registers as pointers to the specific locations to be read or written to. FSRs are located in RAM and can be manipulated under program control.

Data Memory organisation and other registers

The PIC18's Data Memory is implemented as Static RAM (sRAM). Each location in the data memory is also referred to as a ***File Register*****.** As already mentioned this is a 4Kbyte address space addressed by a 12-bit address bus. The data memory map is as shown in Fig. 9. The 4k PIC18 data memory bank is divided into 16 banks. Only one bank of the 256 file registers is active at any

given time. Additional 4-bits need to be placed in the Bank Select Register (BSR) in order to select the active bank.
There are two types of registers in the data memory, the ***General Purpose Registers*** (GPRs) and the ***Special-Function Registers*** (SFRs not be confused the File Select Registers) . The General Purpose Registers are used to hold dynamic data during program execution. The SFRs are used for controlling the various peripheral modules and selected operations of the CPU. These are CPU core registers (such as Stack Pointer, Status Register and Program Counter). They also include the various peripheral registers for the various IO modules of the PIC18 such as Parallel ports, Serial communication ports, registers that control the read and write operations of EEDATA areas etc.
The GPRs start from address 00h upwards to address EFFh. The SFRs occupy the highest data memory block starting at hex location F60h. See Fig.9 and the complete SFR map table in the appendix.

The first 127 bytes in bank 0 (000h – 07Fh) of the GPRs and the last 127 bytes in bank 16 (0F80h-0FFFh) of the SFRs are grouped into a special bank called ***Access Bank***.

Note: Half of bank 0 and half of bank 15 form a Virtual Bank that is accessible no matter which bank is selected (Fig.9).

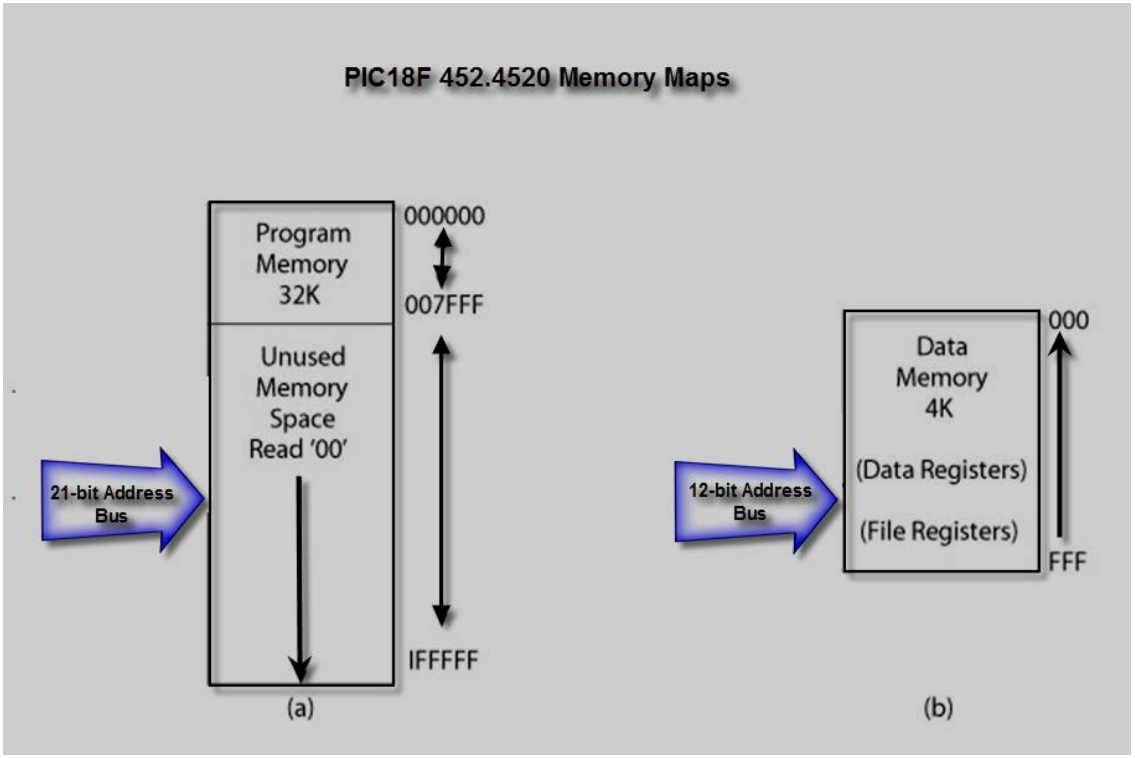


Fig. 8 PIC18Fxxxx simplified Memory Maps

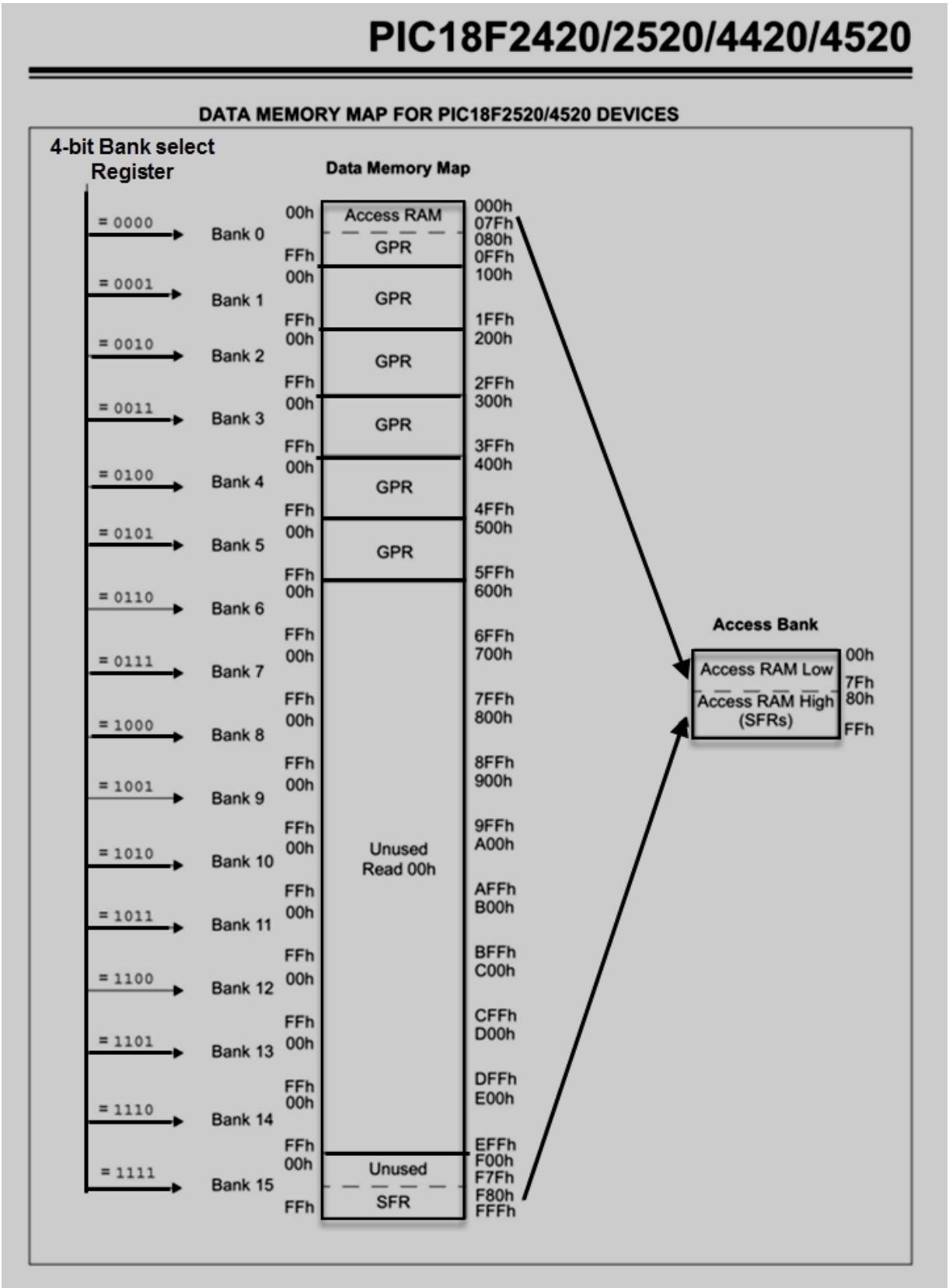


Fig. 9 Data Memory Map

A PIC18 assembly programming model is shown in figure 10 .

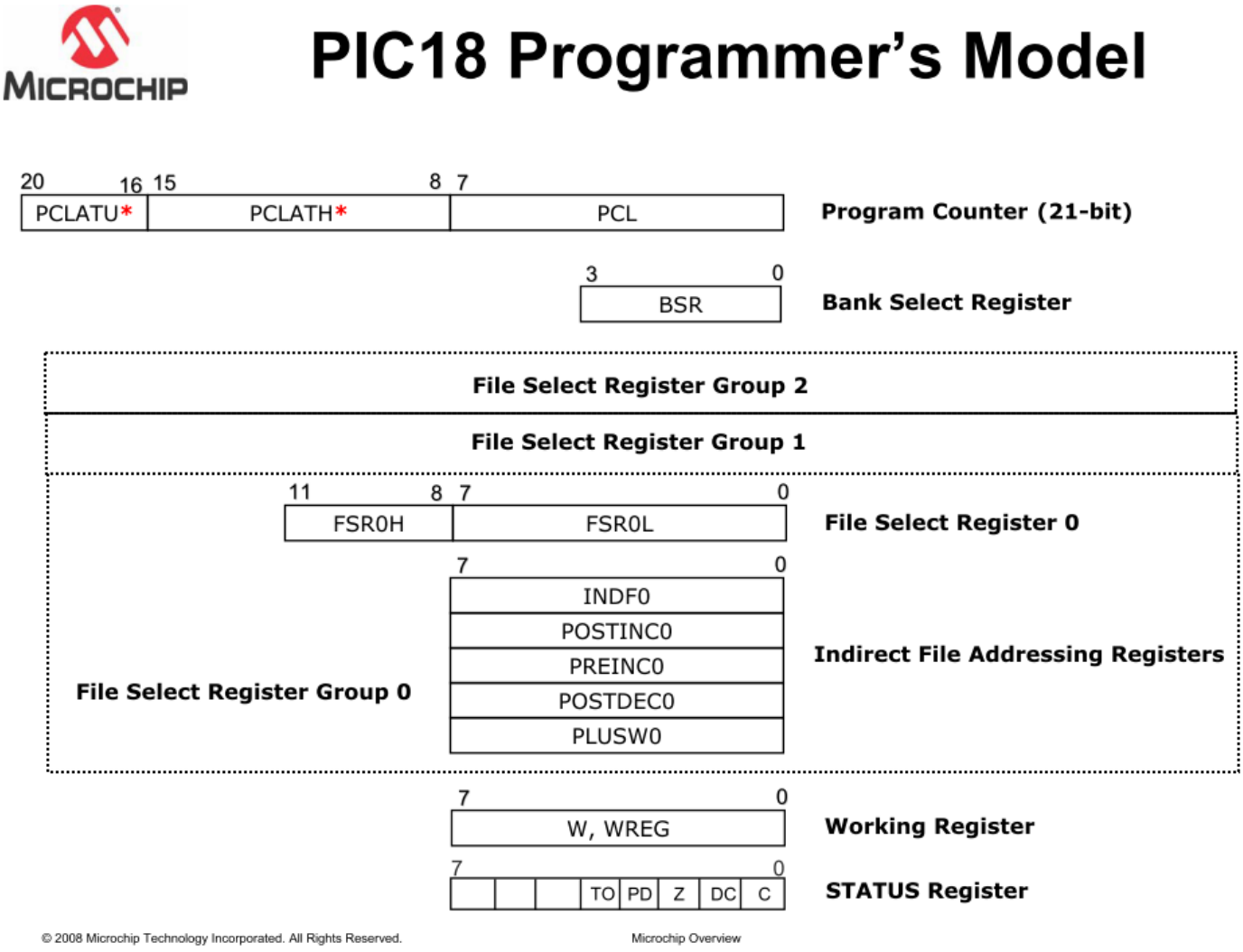


Fig. 10

EEPROM Data Memory.

All PIC18 devices that have on-chip flash program memory also have 256 or 1024 bytes of data EEPROM. This is readable and writable and is not mapped directly in the register file space. It can only be indirectly addressed through a Special Function Register.

Program Memory

The PIC18 family has a 21-bit program counter which gives the CPU the ability to address a 2-MB program memory space. Inaccessible memory locations will return a zero read value.
The PIC18 MCU has a 31-entry return address ***Stack*** that is used to hold return addresses for subroutine calls and interrupt processing. The stack is NOT part of the program memory space. The stack is controlled by a 5-bit ***Stack Pointer***. After a CPU reset the stack is initialised to 000h. During the subroutine call or interrupt, the stack pointer is first incremented and the memory location pointed to by the stack pointer is written using the contents of the program counter.

During the return from a subroutine call or interrupt routine the memory location pointed to by the stack is decremented.
The ***Reset Vector*** where the program starts after a reset is at location 0000h. Addresses 0008h and 0018h are reserved for the vectors of ***high*** and ***low priority interrupts*** and the corresponding interrupt routines have to be written to start at these locations.
The program memory map is shown in Fig. 11 below:

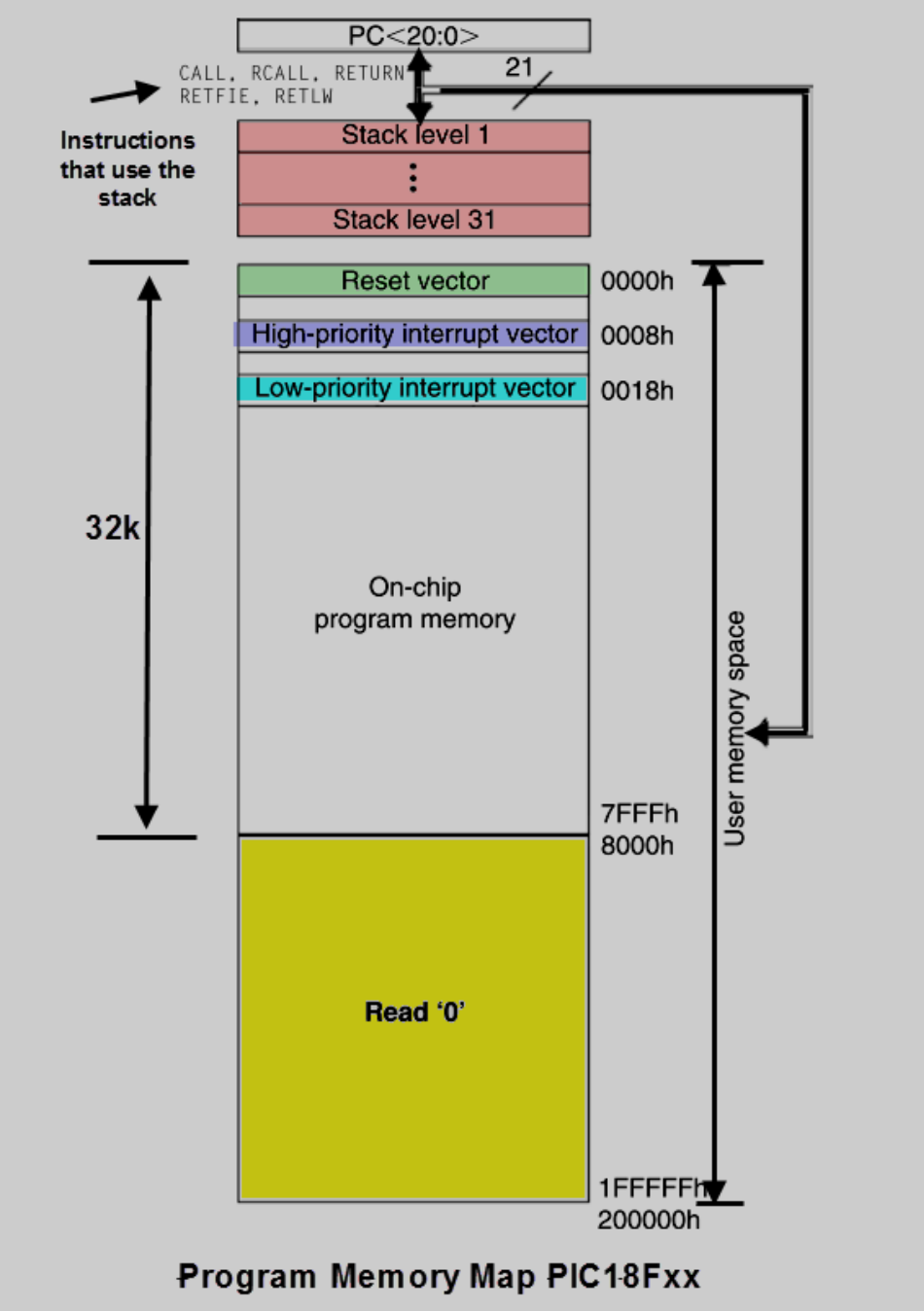
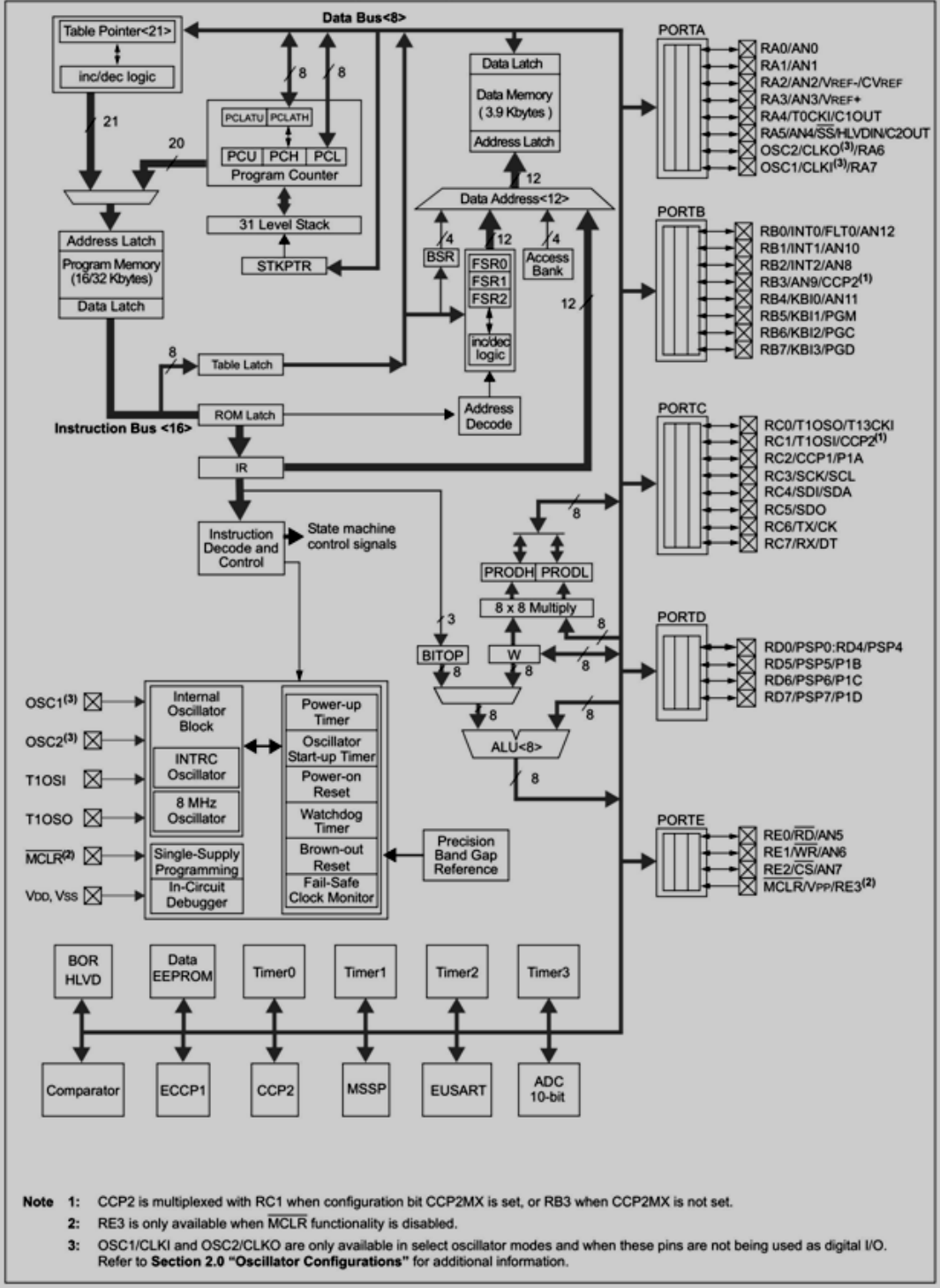


Fig. 11 Program Memory Map

Appendix

PIC18F4420/4520 (40/44-PIN) BLOCK DIAGRAM



Note 1: CCP2 is multiplexed with RC1 when configuration bit CCP2MX is set, or RB3 when CCP2MX is not set.

2: RE3 is only available when MCLR functionality is disabled.

3: OSC1/CLKI and OSC2/CLKO are only available in select oscillator modes and when these pins are not being used as digital I/O. Refer to **Section 2.0 "Oscillator Configurations"** for additional information.

DEVICE FEATURES

Features	PIC18F2420	PIC18F2520	PIC18F4420	PIC18F4520
Operating Frequency	DC – 40 MHz	DC – 40 MHz	DC – 40 MHz	DC – 40 MHz
Program Memory (Bytes)	16384	32768	16384	32768
Program Memory (Instructions)	8192	16384	8192	16384
Data Memory (Bytes)	768	1536	768	1536
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	19	19	20	20
I/O Ports	Ports A, B, C, (E)	Ports A, B, C, (E)	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	1	1
Enhanced Capture/Compare/PWM Modules	0	0	1	1
Serial Communications	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART
Parallel Communications (PSP)	No	No	Yes	Yes
10-bit Analog-to-Digital Module	10 Input Channels	10 Input Channels	13 Input Channels	13 Input Channels
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT
Programmable High/Low-Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled
Packages	28-pin PDIP 28-pin SOIC 28-pin QFN	28-pin PDIP 28-pin SOIC 28-pin QFN	40-pin PDIP 44-pin QFN 44-pin TQFP	40-pin PDIP 44-pin QFN 44-pin TQFP

SPECIAL FUNCTION REGISTER MAP FOR PIC18F2420/2520/4420/4520 DEVICES

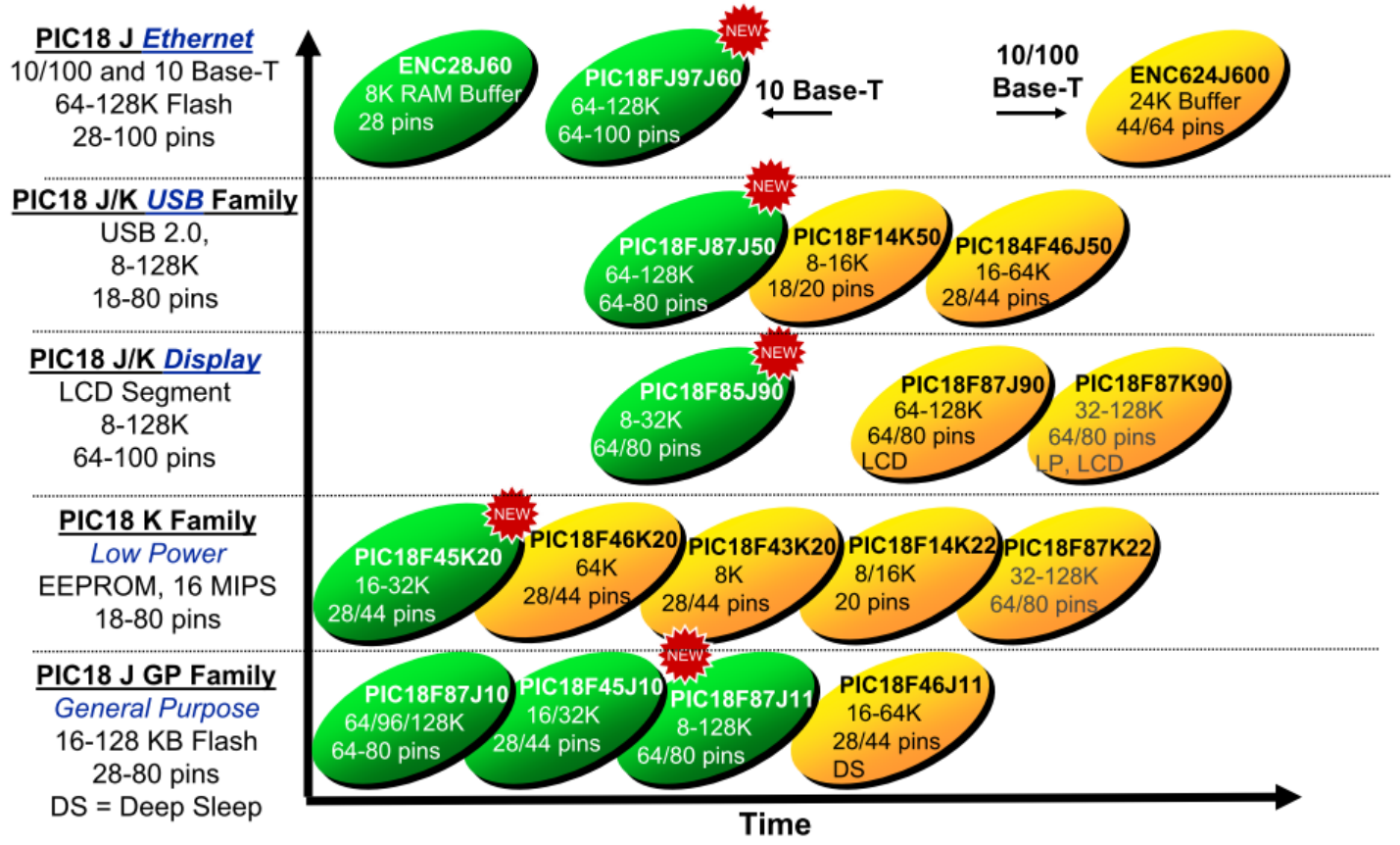
Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSU	FDFh	INDF2(1)	FBFh	CCPR1H	F9Fh	IPR1
FFEh	TOSH	FDEh	POSTINC2(1)	FBEh	CCPR1L	F9Eh	PIR1
FFDh	TOSL	FDDh	POSTDEC2(1)	FBDh	CCP1CON	F9Dh	PIE1
FFCh	STKPTR	FDCh	PREINC2(1)	FBCh	CCPR2H	F9Ch	—(2)
FFBh	PCLATU	FDBh	PLUSW2(1)	FBBh	CCPR2L	F9Bh	OSCTUNE
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON	F9Ah	—(2)
FF9h	PCL	FD9h	FSR2L	FB9h	—(2)	F99h	—(2)
FF8h	TBLPTRU	FD8h	STATUS	FB8h	BAUDCON	F98h	—(2)
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	PWM1CON(3)	F97h	—(2)
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	ECCP1AS(3)	F96h	TRISE(3)
FF5h	TABLAT	FD5h	T0CON	FB5h	CVRCON	F95h	TRISD(3)
FF4h	PRODH	FD4h	—(2)	FB4h	CMCON	F94h	TRISC
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB
FF2h	INTCON	FD2h	HLVDCON	FB2h	TMR3L	F92h	TRISA
FF1h	INTCON2	FD1h	WDTCON	FB1h	T3CON	F91h	—(2)
FF0h	INTCON3	FD0h	RCON	FB0h	SPBRGH	F90h	—(2)
FEFh	INDF0(1)	FCFh	TMR1H	FAFh	SPBRG	F8Fh	—(2)
FEEh	POSTINC0(1)	FCEh	TMR1L	FAEh	RCREG	F8Eh	—(2)
FEDh	POSTDEC0(1)	FCDh	T1CON	FADh	TXREG	F8Dh	LATE(3)
FECh	PREINC0(1)	FCCh	TMR2	FACh	TXSTA	F8Ch	LATD(3)
FEBh	PLUSW0(1)	FCBh	PR2	FABh	RCSTA	F8Bh	LATC
FEAh	FSR0H	FCAh	T2CON	FAAh	—(2)	F8Ah	LATB
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA
FE8h	WREG	FC8h	SSPADD	FA8h	EEDATA	F88h	—(2)
FE7h	INDF1(1)	FC7h	SSPSTAT	FA7h	EECON2(1)	F87h	—(2)
FE6h	POSTINC1(1)	FC6h	SSPCON1	FA6h	EECON1	F86h	—(2)
FE5h	POSTDEC1(1)	FC5h	SSPCON2	FA5h	—(2)	F85h	—(2)
FE4h	PREINC1(1)	FC4h	ADRESH	FA4h	—(2)	F84h	PORTE(3)
FE3h	PLUSW1(1)	FC3h	ADRESL	FA3h	—(2)	F83h	PORTD(3)
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB
FE0h	BSR	FC0h	ADCON2	FA0h	PIE2	F80h	PORTA

Note 1: This is not a physical register.
2: Unimplemented registers are read as '0'.
3: This register is not available on 28-pin devices.

CPU Registers

Address (Hex)	Name	Description
0FFF	TOSU	Top Of Stack Upper
0FFE	TOSH	Top Of Stack High
0FFD	TOSL	Top Of Stack Low
0FFC	STKPTR	Stack Pointer
0FFB	PCLATU	Upper Program Counter Latch
0FFA	PCLATH	High Program CounterLatch
0FF9	PCL	Program Counter Low byte
0FF8	TBLPTRU	Table Pointer Upper byte
0FF7	TBLPTRH	Table Pointer High byte
0FF6	TBLPTRL	Table Pointer Low byte
0FF5	TABLAT	Table Latch
0FF4	PRODH	Product Register High
0FF3	PRODL	Product Register Low
0FF2	INTCON	Interrupt Control Register
0FF1	INTCON2	Interrupt Control Reg 2
0FF0	INTCON3	Interrupt Control Reg 3
0FEF	INDF0	Indirect File Reg Pointer 0
0FEE	POSTINC0	Post Increment Pointer 0
0FED	POSTDEC0	Post Decrement Pointer 0
0FEC	PREINC0	Pre Increment pointer 0
0FEB	PLUSSW0	Add W Reg to FSR0
0FEA	FSR0H	File Select Register 0 High byte
0FE9	FSR0L	File Select Register 0 Low byte
0FE8	WREG	Working Register
0FE7	INDF1	Indirect File Register pointer 1
0FE6	POSTINC1	Post Increment Pointer 1
0FE5	POSTDEC1	Post Decrement Pointer1
0FE4	PREINC1	Pre Increment Pointer 1
0FE3	PLUSW1	Add W Register to FSR1
0FE2	FSR1H	File Select Register High byte
0FE1	FSR1L	File Select Register Low byte
0FE0	BSR	Bank Select Register
0FDF	INDF2	Indirect File Register Pointer 2
0FDE	POSTINC2	Post Increment Pointer 2
0FDD	POSTDEC2	Post Decrement Pointer 2
0FDC	PREINC2	Pre Increment Pointer 2
0FDB	PLUSW2	Add W Register to FSR2
0FDA	FSR2H	File Select Register 2 High byte
0FD9	FSR2L	File Select Register Low byte
0FD8	STATUS	Status Register

Some other advanced PIC18F family members



PIC18 Instruction Set

Byte Oriented

ADDWF f,d,a Add W to f where d=0->W, d=1->f, a is generally not specified (access bank stuff)
ADDWFC f,d,a Add W and Carry bit to f
ANDWF f,d,a And W with f
CLRF f,a Clear f
COMF f,a Complement f
CPFSEQ Compare, skip if f==W
CPFSGT Compare, skip if f > W
CPFSLT Compare, skip if f < W
DECF f,d,a Decrement f
DECFSZ f,d,a Dec f, skip if 0
DCFSNZ f,d,a Dec f, skip if not 0
INCF f,d,a Increment f
INCFSZ f,d,a Increment f, skip if zero
INFSNZ f,d,a Increment f, skip if not zero
IORWF f inclusive-OR W with f
MOVF f,d,a Move f (usually to W)
MOVFF f,ff Move f to ff
MOVWF f,a Move W to f
MULWF f,a W x f
NEGF f,a Negate f
RLCF f,d,a Rotate left f through Carry (not-quite multiply by 2 with carry)
RLNCF f,d,a Rotate left (no carry)
RRCF f,d,a Rotate right through Carry
RRNCF f,d,a Rotate right f (no carry)
SETF f,a Set f = 0xff
SUBFWB f,d,a Subtract f FROM w with Borrow
SUBWF f,d,a Subtract W from f
SUBWFB f,d,a Subtract W from f with Borrow
SWAPF f,d,a Swap nibbles of f
XORWF f,d,a W XOR f

Bit Oriented

BCF f,b,a Bit clear, bit is indexed 0 to 7
BSF f,b,a Bit set
BTFSC f,b,a Bit test, skip if clear
BTFSS f,b,a Bit test, skip if set
BTG f,b,a Bit toggle

Control Operations

BC n	Branch if Carry, n is either a relative or a direct address
BN n	Branch if Negative
BNC n	Branch if Not Carry
BNN n	Branch if Not Negative
BNOV n	Branch if Not Overflow
BNZ n	Branch if Not Zero
BOV n	Branch if Overflow
BRA n	Branch Unconditionally
BZ n	Branch if Zero
CALL n, s	Call Subroutine
CLRWDT	Clear Watchdog Timer
DAW	Decimal Adjust W
GOTO n	Go to address
NOP	No operation
POP	Pop top of return stack (TOS)
PUSH	Push top of return stack (TOS)
RCALL n	Relative Call
RESET	Software device reset
RETFIE	Return from Interrupt and Enable Interrupts
RETURN s	Return from subroutine
SLEEP	Enter SLEEP Mode

Literals

ADDLW kk	Add literal to W
ANDLW kk	And literal with W
IORLW kk	Inclusive-OR literal with W
LFSR r,kk	Move literal (12 bit) 2nd word to FSRr 1st word
MOVLB k	Move literal to BSR<3:0>
MOVLW kk	Move literal to W
MULLW kk	Multiply literal with W
RETLW kk	Return with literal in W
SUBLW kk	Subtract W from literal
XORLW kk	XOR literal with W

Memory

TLBRD	Table Read

d = 0 means destination is WREG. d = 1 means destination is FILE and this is the default.