Instruction Set (x14)

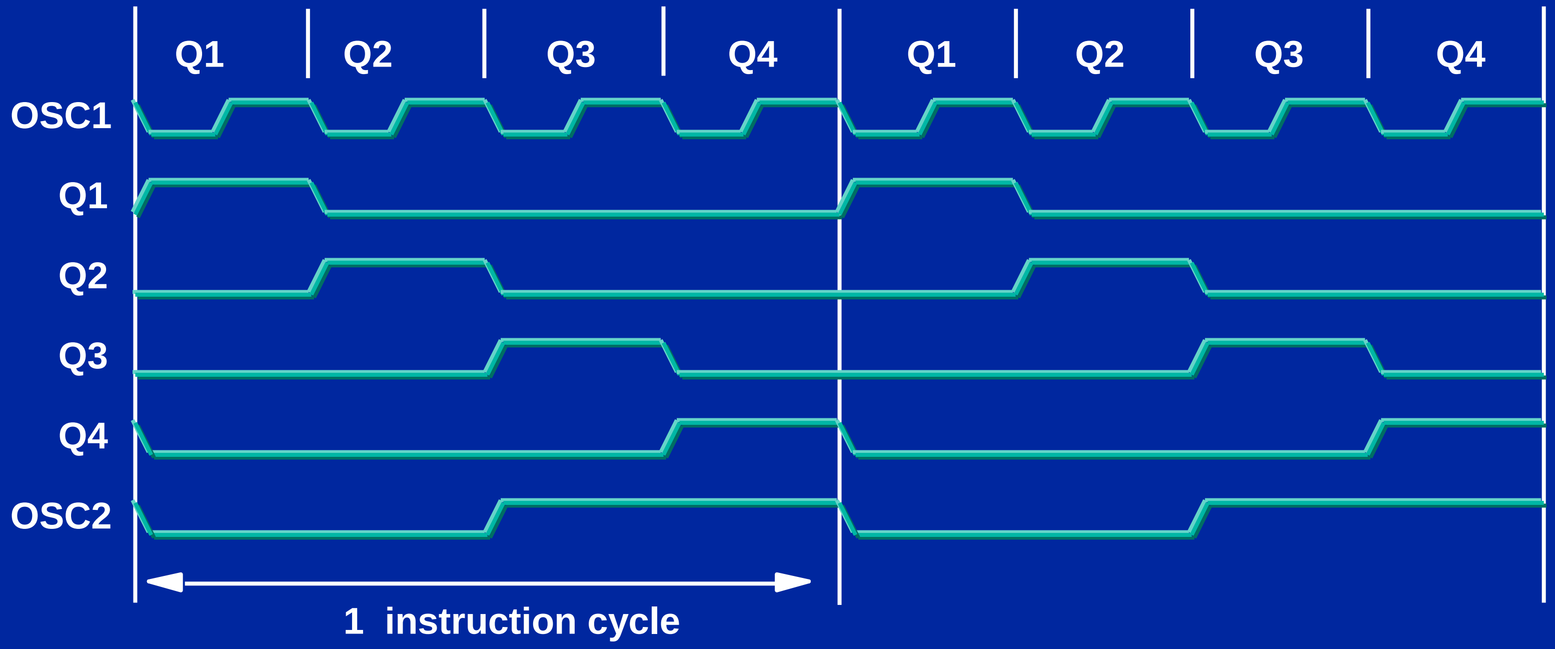
PICmicro® MCU Instruction Set

- 12-bit core → **33 instructions**
- 14-bit core → **35 instructions**
- 16-bit core → **58 instructions**
- 16-bit enh. core → **77 instructions**

- **Easy to learn**
- **High compaction**
- **Very powerful single-word instructions**
- **All single- cycle except program branches**
- **Upward compatibility of instructions**

PIC16CXX Architecture
Clocking Scheme

- Instruction cycle rate is 1/4 of clock input frequency
- Instruction cycle is 200 ns at 20 MHz clock rate



Legend: Typically: Q1 = Read Q2 = Decode Q3 = Execute Q4 = Write

PICmicro MCU Instruction Set

Summary (14-bit core)

Byte-Oriented Operations

NOP	-	No Operation
MOVWF	f	Move W to f
CLRW	-	Clear W
CLRF	f	Clear f
SUBWF	f,d	Subtract W from f
DECF	f,d	Decrement f
IORWF	f,d	Inclusive OR W and f
ANDWF	f,d	AND W and f
XORWF	f,d	Exclusive OR W and f
ADDWF	f,d	Add W and f
MOVF	f,d	Move f
COMF	f,d	Complement f
INCF	f,d	Increment f
DECFSZ	f,d	Decrement f, skip if zero
RRF	f,d	Rotate right f through carry
RLF	f,d	Rotate left f through carry
SWAPF	f,d	Swap nibbles of f
INCFSZ	f,d	Increment f, skip if zero

Bit-Oriented Operations

BCF	f,b	Bit clear f
BSF	f,b	Bit set f
BTFSC	f,b	Bit test f, skip if clear
BTFSS	f,b	Bit test f, skip if set

Literal and Control Operations

SLEEP	-	Go into standby mode
CLRWDT	-	Clear watchdog timer
RETLW	k	Return, place literal in W
RETFIE	-	Return from interrupt
RETURN	-	Return from subroutine
CALL	k	Call subroutine
GOTO	k	Go to address (k is 9-bit)
MOVLW	k	Move literal to W
IORLW	k	Inclusive OR literal with W
ADDLW	k	Add literal with W
SUBLW	k	Subtract W from literal
ANDLW	k	AND literal with W
XORLW	k	Exclusive OR literal with W

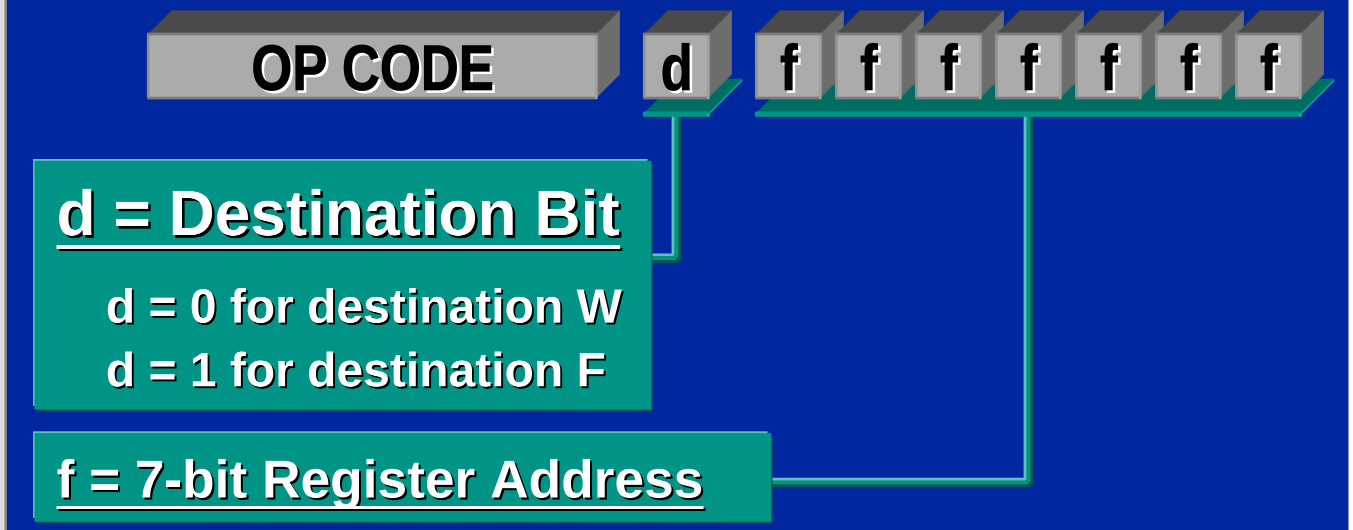
f = File Register, k = literal value (8-bit), b = bit address <0,7>, d = destination (0=f, 1=W)

PICmicro MCU Instruction Set

Byte-Oriented Operations

Byte-Oriented Operations	
NOP	-
MOVWF	f
CLRW	-
CLRF	f
SUBWF	f,d
DECF	f,d
IORWF	f,d
ANDWF	f,d
XORWF	f,d
ADDWF	f,d
MOVF	f,d
COMF	f,d
INCF	f,d
DECFSZ	f,d
RRF	f,d
RRL	f,d
SWAPF	f,d
INCFSZ	f,d

14-bit Instruction for Byte Oriented Operations



d = Destination Bit

d = 0 for destination W
d = 1 for destination F

f = 7-bit Register Address

Example:

ADDWF REG, W

ADDWF f, d

PICmicro MCU Instruction Set

Byte-Oriented Operations

NOP	No Operation
Syntax:	NOP
Operands:	None
Operation:	No operation
Status:	None
Encoding:	00 0000 0000 0000
Words:	1
Cycles:	1

- Example:

  NOP

PICmicro MCU Instruction Set

Byte-Oriented Operations

MOVWF	Move W to f
Syntax:	MOVWF f
Operands:	0 <= f <= 127
Operation:	(W) -> (f)
Status:	None
Encoding:	00 0000 1fff fff
Words:	1
Cycles:	1

- Example:

  MOVWF FSR

  Before Instruction

  FSR = 0xFF

  W = 0x4F

  After Instruction

  FSR = 0x4F

  W = 0x4F

PICmicro MCU Instruction Set

Byte-Oriented Operations

CLRW	Clear W
Syntax:	CLRW
Operands:	None
Operation:	00h -> (W) 1 -> Z
Status:	Z
Encoding:	00 0001 0000 0000
Words:	1
Cycles:	1

- Example:

  CLRW

  Before Instruction

  W = 0x4F

  After Instruction

  W = 0x00

  Z = 1

PICmicro MCU Instruction Set

Byte-Oriented Operations

CLRF	Clear f
Syntax:	CLRF f
Operands:	0 <= f <= 127
Operation:	00h -> (f) 1 -> Z
Status:	Z
Encoding:	00 0001 1fff ffff
Words:	1
Cycles:	1

- Example:

  CLRF FSR

  Before Instruction

  FSR = 0x4F

  After Instruction

  FSR = 0x00

  Z = 1

PICmicro MCU Instruction Set

Byte-Oriented Operations

SUBWF	Subtract W from f
Syntax:	SUBWF f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	(f) - (W) -> dest
Status:	C,DC,Z
Encoding:	00 0010 dfff ffff
Words:	1
Cycles:	1

- Example:

  SUBWF FSR,0

  Before Instruction

  FSR = 0x03
  W = 0x02
  C = ?
  Z = ?

  After Instruction

  FSR = 0x03
  W = 0x01
  C = 1
  Z = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

DECF	Decrement f
Syntax:	DECF f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	(f) - 1 -> dest
Status:	Z
Encoding:	00 0011 dfff ffff
Words:	1
Cycles:	1

- Example:

DECF FSR,1

Before Instruction

FSR = 0x01
Z = 0

After Instruction

FSR = 0x00
Z = 1

PICmicro MCU Instruction Set

Byte-Oriented Operations

IORWF	Inclusive OR W & f
Syntax:	IORWF f,d
Operands:	0 <= f <= 127
	d = {0,1}
Operation:	(W) .OR. (f) -> dest
Status:	Z
Encoding:	00 0100 dfff ffff
Words:	1
Cycles:	1

- Example:

IORWF CNT,0

Before Instruction

CNT = 0x13
W = 0x91

After Instruction

CNT = 0x13
W = 0x93
Z = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

ANDWF AND W with f

Syntax: ANDWF f,d

Operands: 0 <= f <= 127

d = {0,1}

Operation: (W).AND.(f) -> dest

Status: Z

Encoding: 00 0101 dfff ffff

Words: 1

Cycles: 1

- Example:

ANDWF CNT,1

Before Instruction

CNT = 0x17

W = 0xC2

After Instruction

CNT = 0x02

W = 0xC2

Z = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

XORWF	Exclusive OR W & f
Syntax:	XORWF f,d
Operands:	0 <= f <= 127
	d = {0,1}
Operation:	(W).XOR.(f) -> dest
Status:	Z
Encoding:	00 0110 dfff ffff
Words:	1
Cycles:	1

- Example:

XORWF FSR,0

Before Instruction

FSR = 0xAF
W = 0xB5

After Instruction

FSR = 0xAF
W = 0x1A
Z = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

ADDWF	Add W with f
Syntax:	ADDWF f,d
Operands:	0 <= f <= 127
	d = {0,1}
Operation:	(W) + (f) -> dest
Status:	Z
Encoding:	00 0111 dfff ffff
Words:	1
Cycles:	1

- Example:

ADDWF FSR,0

Before Instruction

FSR = 0x17
W = 0xC2

After Instruction

FSR = 0x17
W = 0xD9
Z = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

MOVF	Move f
Syntax:	MOVF f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	(f) -> dest
Status:	Z
Encoding:	00 1000 dfff ffff
Words:	1
Cycles:	1

- Example:

MOVF FSR,1

Before Instruction

FSR = 0x00
Z = 0

After Instruction

FSR = 0x00
Z = 1

PICmicro MCU Instruction Set

Byte-Oriented Operations

COMF	Complement f
Syntax:	MOVF f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	$(\overline{f})$ -> dest
Status:	Z
Encoding:	00 1001 dfff ffff
Words:	1
Cycles:	1

- Example:

COMF REG,0

Before Instruction

REG = 0x13
W = 0x58
Z = 0

After Instruction

REG = 0x13
W = 0xEC
Z = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

INCF	Increment f
Syntax:	INCF f,d
Operands:	0 <= f <= 127
	d = {0,1}
Operation:	(f) + 1 -> dest
Status:	Z
Encoding:	00 1010 dfff ffff
Words:	1
Cycles:	1

- Example:

INCF REG,1

Before Instruction

REG = 0xFF

Z = 0

After Instruction

FSR = 0x00

Z = 1

PICmicro MCU Instruction Set

Byte-Oriented Operations

DECFSZ	Dec. f, Skip if 0
Syntax:	DECFSZ f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	(f) - 1 -> dest skip if result = 0
Status:	None
Encoding:	00 1011 dfff ffff
Words:	1
Cycles:	1(2)

Example:

```
Loop    DECFSZ  CNT,1
        GOTO      Loop
Continue
```

Before Instruction

PC = address *Loop*

After Instruction

CNT = CNT - 1

if CNT = 0,

PC = address *Continue*

if CNT != 0,

PC = address *Loop*+1

PICmicro MCU Instruction Set

Byte-Oriented Operations

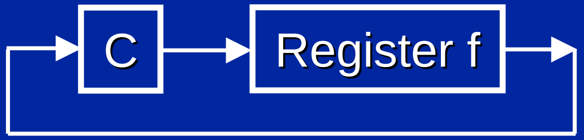
RRF Rotate Right f -> C

Syntax: RRF f,d

Operands: 0 <= f <= 127

d = {0,1}

Operation:



Status: C

Encoding: 00 1100 dfff ffff

Words: 1

Cycles: 1

Example:

RRF CNT,0

Before Instruction

CNT = 1110 0110

C = 0

After Instruction

CNT = 1110 0110

W = 0111 0011

C = 0

PICmicro MCU Instruction Set

Byte-Oriented Operations

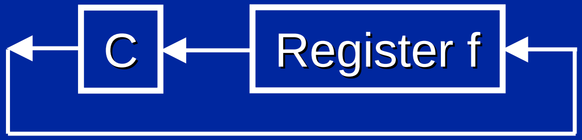
RLF Rotate Left f -> C

Syntax: RLF f,d

Operands: 0 <= f <= 127

d = {0,1}

Operation:



Status: C

Encoding: 00 1101 dfff ffff

Words: 1

Cycles: 1

Example:

RLF CNT,0

Before Instruction

CNT = 1110 0110

C = 0

After Instruction

CNT = 1110 0110

W = 1100 1100

C = 1

PICmicro MCU Instruction Set

Byte-Oriented Operations

SWAPF	Swap Nibbles in f
Syntax:	SWAPF f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	f<3:0> -> dest<7:4> f<7:4> -> dest<3:0>
Status:	None
Encoding:	00 1110 dfff ffff
Words:	1
Cycles:	1

Example:

SWAPF REG,0

Before Instruction

REG = 0xA5

After Instruction

REG = 0xA5

W = 0x5A

PICmicro MCU Instruction Set

Byte-Oriented Operations

INCFSZ	Inc. f, Skip if 0
Syntax:	INCFSZ f,d
Operands:	0 <= f <= 127 d = {0,1}
Operation:	(f) + 1 -> dest skip if result = 0
Status:	None
Encoding:	00 1111 dfff ffff
Words:	1
Cycles:	1(2)

Example:

```
Loop    INCFSZ  CNT,1
        GOTO     Loop
Continue
```

Before Instruction

PC = address *Loop*

After Instruction

CNT = CNT + 1

if CNT = 0,

PC = address *Continue*

if CNT != 0,

PC = address *Loop*+1

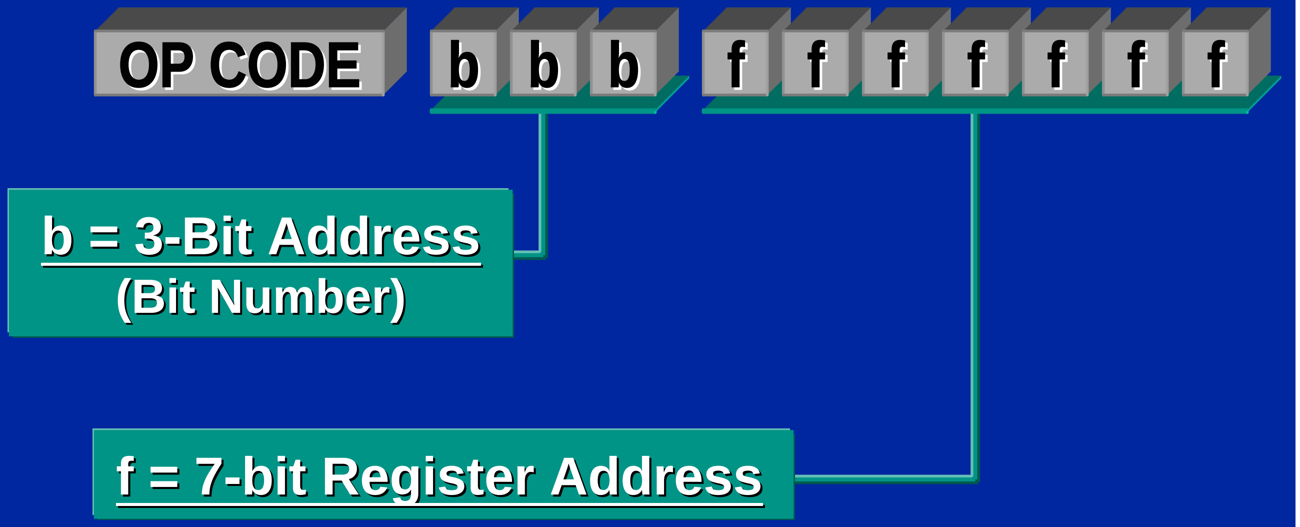
PICmicro MCU Instruction Set

Bit-Oriented Operations

Bit-Oriented Operations

BCF	f,b
BSF	f,b
BTFSC	f,b
BTFSS	f,b

14-bit Instruction for Bit Oriented Operations



b = 3-Bit Address
(Bit Number)

f = 7-bit Register Address

Example:

BTFSC STATUS, C

BTFSC f, b

PICmicro MCU Instruction Set

Bit-Oriented Operations

BCF	Bit Clear f
Syntax:	BCF f,b
Operands:	0 <= f <= 127 0 <= b <= 7
Operation:	0 -> (f<b>)
Status:	None
Encoding:	01 00bb bfff ffff
Words:	1
Cycles:	1

- Example:

  BCF FSR,4

  Before Instruction

  FSR = 0011 0000

  After Instruction

  FSR = 0010 0000

PICmicro MCU Instruction Set

Bit-Oriented Operations

BSF	Bit Set f
Syntax:	BSF f,b
Operands:	0 <= f <= 127 0 <= b <= 7
Operation:	1 -> (f<b>)
Status:	None
Encoding:	01 01bb bfff ffff
Words:	1
Cycles:	1

- Example:

  BSF FSR,4

  Before Instruction

  FSR = 0010 0000

  After Instruction

  FSR = 0011 0000

PICmicro MCU Instruction Set

Bit-Oriented Operations

BTFSC	Bit Test f, Skip if 0
Syntax:	BTFSC f,b
Operands:	0 <= f <= 127 0 <= b <= 7
Operation:	skip if (f<b>) = 0
Status:	None
Encoding:	01 10bb bfff ffff
Words:	1
Cycles:	1(2)

Example:

```
Here   BTFSC   CNT,1
False  GOTO     Done
True
```

Before Instruction

PC = address *Here*

After Instruction

if CNT<1> = 0,

PC = address *True*

if CNT<1> = 1,

PC = address *False*

PICmicro MCU Instruction Set

Bit-Oriented Operations

BTFSS	Bit Test f, Skip if 1
Syntax:	BTFSS f,b
Operands:	0 <= f <= 127 0 <= b <= 7
Operation:	skip if (f<b>) = 1
Status:	None
Encoding:	01 11bb bfff ffff
Words:	1
Cycles:	1(2)

Example:

```
Here   BTFSS   CNT,1
False  GOTO    Done
True
```

Before Instruction

PC = address *Here*

After Instruction

if CNT<1> = 1,

PC = address *True*

if CNT<1> = 0,

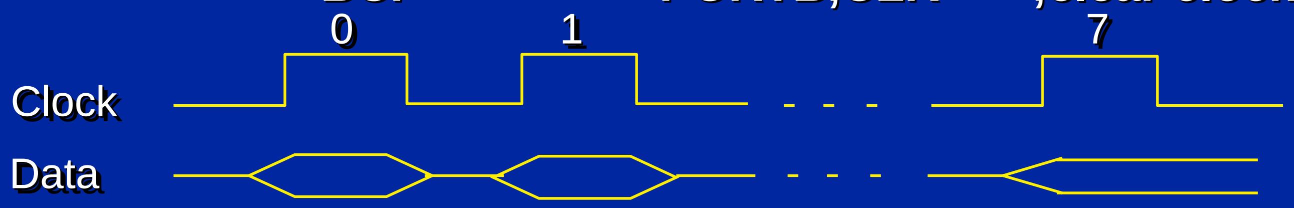
PC = address *False*

PICmicro MCU Instruction Set

Example: Bit Manipulation

- Synchronous serial transmission of eight bits of data from file register XDATA to I/O Pin:

```
XMIT:      MOVLW    0x08          ;Bit count = 8
           MOVWF    bit_count
XM_LOOP:
           BCF      PORTB,DT      ;preset clock & data lines to 0
           BCF      PORTB,CLK     ;preset clock & data lines to 0
           RRF      XDATA,F       ;rotate data right thru Carry
           BTFSC    STATUS,C      ;test carry bit
           BSF      PORTB,DT      ;set 1 → Data pin
           BSF      PORTB,CLK     ;set 1 → Clock pin
           DECFSZ   bit_count,F   ;decrement count
           GOTO     XM_LOOP       ;Not done then repeat
           BCF      PORTB,CLK     ;clear clock line and exit
```

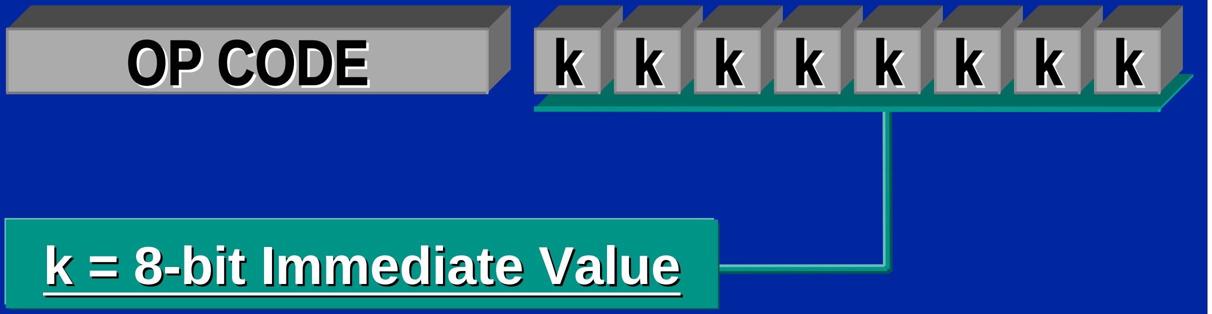


PICmicro MCU Instruction Set

Literal Operations

Literal Operations

MOVLW k
IORLW k
ADDLW k
SUBLW k
ANDLW k
XORLW k



14-bit Instruction for Literal Operations

k = 8-bit Immediate Value

Example:

MOVLW 0x2F

MOVLW k

PICmicro MCU Instruction Set

Literal Operations

MOVLW	Move Literal to f
Syntax:	MOVLW k
Operands:	0 <= k <= 255
Operation:	k -> (W)
Status:	None
Encoding:	11 0000 kkkk kkkk
Words:	1
Cycles:	1

- Example:

  MOVLW 0x5A

  After Instruction

  W = 0x5A

PICmicro MCU Instruction Set

Literal Operations

IORLW	Inclusive OR Literal with f
Syntax:	IORLW k
Operands:	0 <= k <= 255
Operation:	(W) .OR. k -> (W)
Status:	Z
Encoding:	11 1000 kkkk kkkk
Words:	1
Cycles:	1

- Example:

  IORLW 0x35

  Before Instruction

  W = 0x9A

  After Instruction

  W = 0xBF

  Z = 0

PICmicro MCU Instruction Set

Literal Operations

ADDLW	Add Literal with f
Syntax:	ADDLW k
Operands:	0 <= k <= 255
Operation:	(W) + k -> (W)
Status:	C,DC,Z
Encoding:	11 1110 kkkk kkkk
Words:	1
Cycles:	1

- Example:

  ADDLW 0x15

  Before Instruction

  W = 0x10

  After Instruction

  W = 0x25

PICmicro MCU Instruction Set

Literal Operations

SUBLW	Subtract W from Literal
Syntax:	SUBLW k
Operands:	0 <= k <= 255
Operation:	k - (W) -> (W)
Status:	C,DC,Z
Encoding:	11 1100 kkkk kkkk
Words:	1
Cycles:	1

- Example:

  SUBLW 0x02

  Before Instruction

  W = 2
  C = ?
  Z = ?

  After Instruction

  W = 0x00
  C = 1
  Z = 1

PICmicro MCU Instruction Set

Literal Operations

ANDLW	AND Literal with f
Syntax:	ANDLW k
Operands:	0 <= k <= 255
Operation:	(W) .AND. k -> (W)
Status:	Z
Encoding:	11 1001 kkkk kkkk
Words:	1
Cycles:	1

- Example:

  ANDLW 0x5F

  Before Instruction

  W = 0xA3

  After Instruction

  W = 0x03

PICmicro MCU Instruction Set

Literal Operations

XORLW	Exclusive OR Literal with f
Syntax:	XORLW k
Operands:	0 <= k <= 255
Operation:	(W) .OR. k -> (W)
Status:	Z
Encoding:	11 1010 kkkk kkkk
Words:	1
Cycles:	1

- Example:

XORLW 0xAF

Before Instruction

W = 0xB5

After Instruction

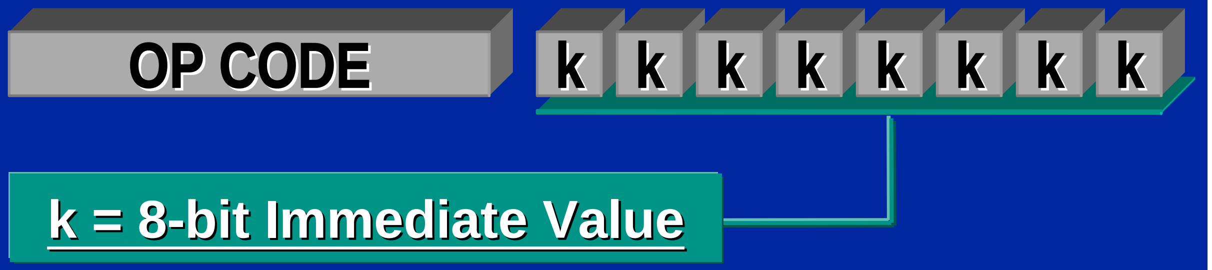
W = 0x1A

PICmicro MCU Instruction Set

Control Operations

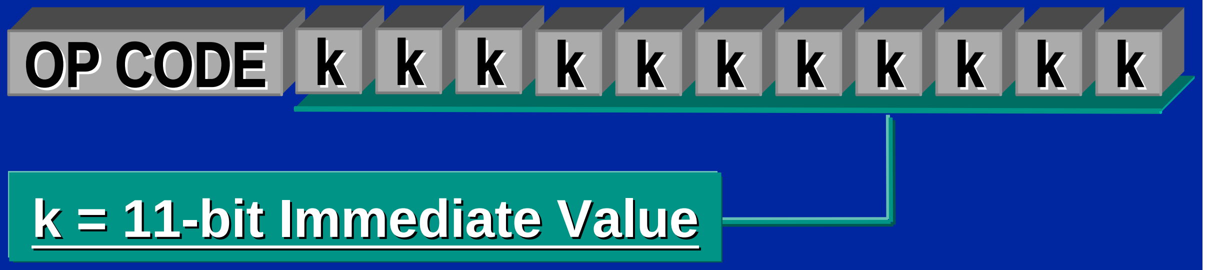
Control Operations	
SLEEP	-
CLRWDT	-
RETLW	k
RETFIE	-
RETURN	-
CALL	k
GOTO	k

14-bit Instruction for RETLW



k = 8-bit Immediate Value

14-bit Instruction for CALL and GOTO



k = 11-bit Immediate Value

PICmicro MCU Instruction Set

Control Operations

SLEEP	Enter SLEEP mode
Syntax:	SLEEP
Operands:	None
Operation:	00h -> WDT 1 -> $\overline{TO}$ 0 -> $\overline{PD}$
Status:	$\overline{TO}$, $\overline{PD}$
Encoding:	00 0000 0110 0011
Words:	1
Cycles:	1

- Example:

  SLEEP

PICmicro Instruction Set

Control Operations

CLRWDT	**Clear Watchdog**
Syntax:	CLRWDT
Operands:	None
Operation:	00h -> WDT 0 -> WDT prescaler 1 -> $\overline{\text{TO}}$ 1 -> $\overline{\text{PD}}$
Status:	$\overline{\text{TO}}$, $\overline{\text{PD}}$
Encoding:	00 0000 0110 0100
Words:	1
Cycles:	1

- Example:

  CLRWDT

PICmicro MCU Instruction Set

Control Operations

RETLW	**Return with Literal in f**
Syntax:	RETLW k
Operands:	0 <= k <= 255
Operation:	k -> (W) TOS -> PC
Status:	None
Encoding:	11 0100 kkkk kkkk
Words:	1
Cycles:	2

- Example:

  RETLW 0x5A

  After Instruction

  W = 0x5A

PICmicro MCU Instruction Set

Control Operations

RETFIE	Return from Interrupt
Syntax:	RETFIE
Operands:	None
Operation:	TOS -> PC 1 -> GIE
Status:	None
Encoding:	00 0000 0000 1001
Words:	1
Cycles:	2

- Example:

  RETFIE

PICmicro MCU Instruction Set

Control Operations

RETURN Return from Subroutine

Syntax: RETURN

Operands: None

Operation: TOS -> PC

Status: None

Encoding: 00 0000 0000 1000

Words: 1

Cycles: 2

- Example:

  RETURN

PICmicro MCU Instruction Set
Control Operations

CALL Call Subroutine

Syntax: CALL k

Operands: 0 <= f <= 2047

Operation: (PC) + 1 -> TOS
k -> PC<10:0>
PCLATH<4:3 ->
PC<12:11>

Status: None

Encoding: 10 0kkk kkkk kkkk

Words: 1

Cycles: 2

Example:

Here *CALL* *There*

Before Instruction
PC = address *Here*

After Instruction
PC = address *There*
TOS = address *Here+1*

PICmicro MCU Instruction Set

Control Operations

GOTO Unconditional Branch

Syntax: GOTO k

Operands: 0 <= f <= 2047

Operation: k -> PC<10:0>
PCLATH<4:3> -> PC<12:11>

Status: None

Encoding: 10 1kkk kkkk kkkk

Words: 1

Cycles: 2

Example:

GOTO *There*

After Instruction
PC = address *There*

PICmicro MCU Instruction Set

Instruction Set Demo