Microcontroladores

Engenharia de Computação

Maio – 2019

Prof. Mardson Freitas de Amorim

Arquitetura:

Trabalhando com PIC

- **Introdução**
- ✓ O ambiente de programação: MPLab
- ✓ Um programa:
  - ✓ A estrutura
  - ✓ Entendendo um exemplo

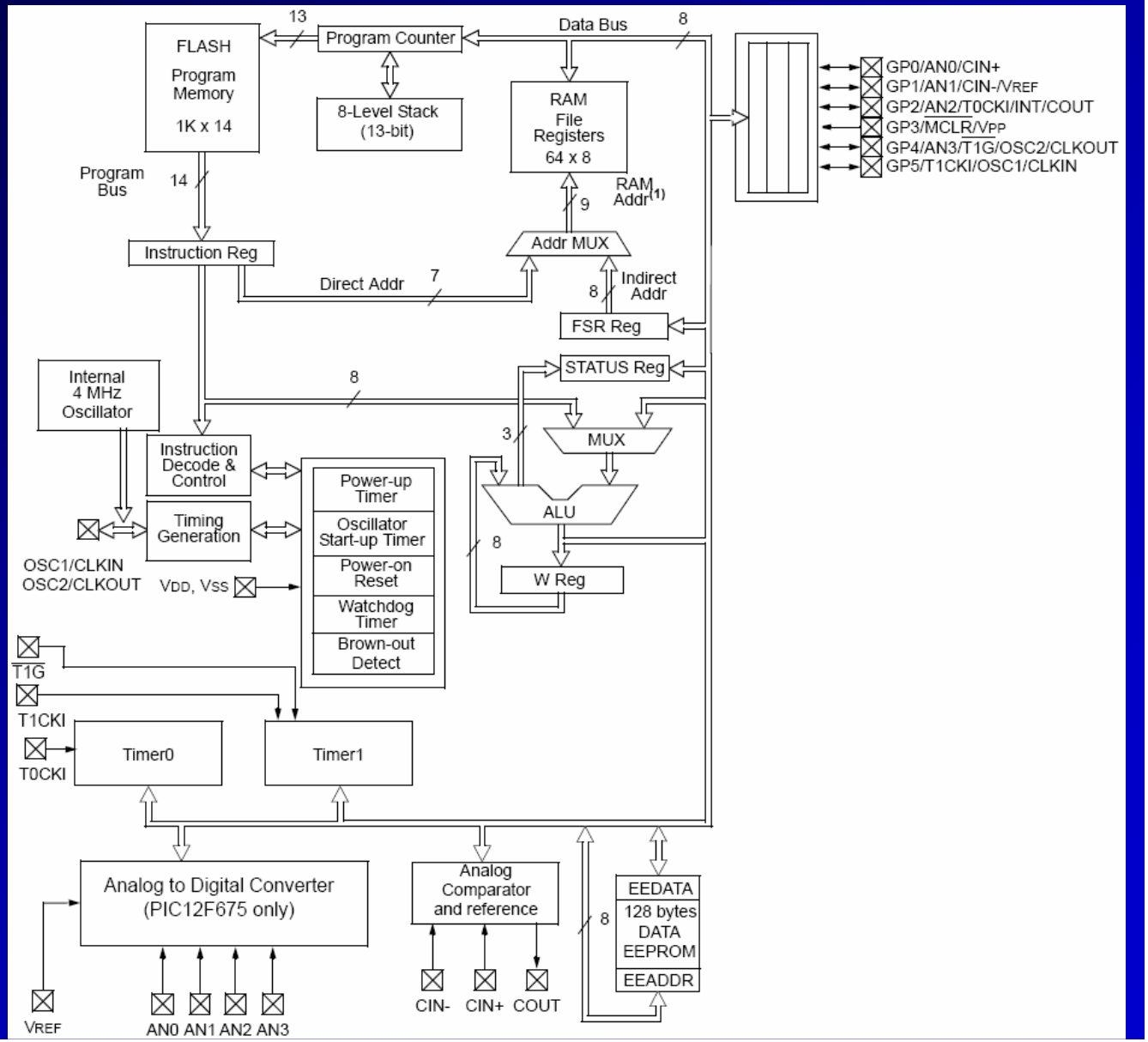
Arquitetura

Trabalhando com o **PIC**

- Arquitetura
- Memórias
- Interrupções
- O microcontrolador PIC12F675
- Registradores especiais
- Conjunto de instruções

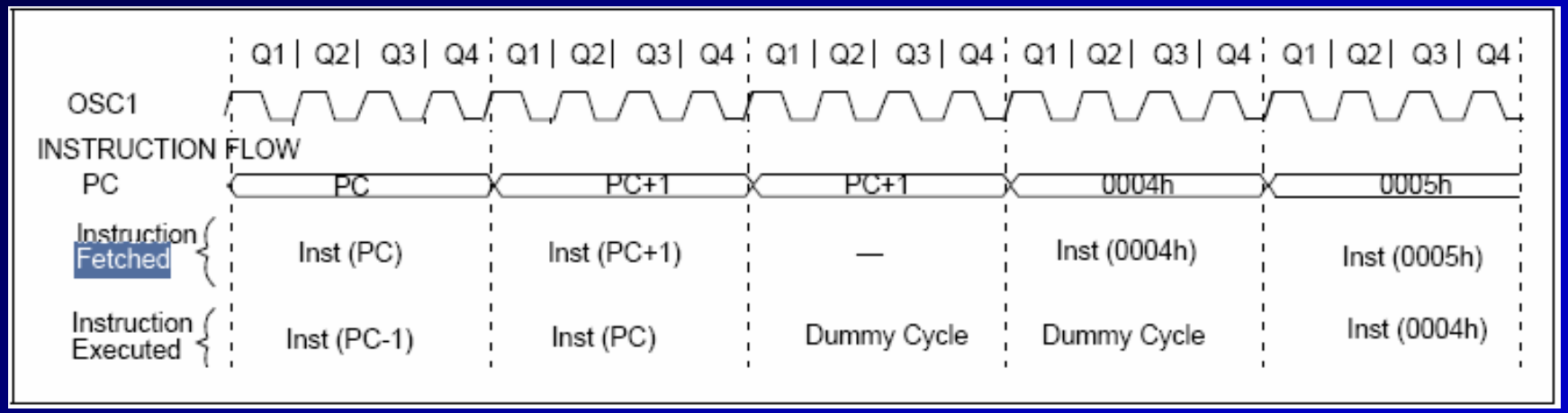
Arquitetura

- Arquitetura
  - Harvard
  - Estruturação interna



Arquitetura

- Arquitetura
  - Ciclos de máquina



Arquitetura

Trabalhando com o **PIC**

Memórias de programa

- 14 bits
- Vetor de RESET
  - Endereço: 0x00
- Vetor de interrupção
  - Endereço: 0x04
- Pilha (*STACK*)
  - 8 níveis

Arquitetura

Trabalhando com o **PIC**

Memórias de dados

- Registradores especiais
  - SFR (*Special Function Registers*)
- Registradores de uso geral
  - Armazenamento de dados do usuário
- EEPROM
  - 128 bytes

Arquitetura

Trabalhando com o **PIC**

- Interrupções
  - Internas e externa
    - *Timers* - Contadores internos (2)
    - Externa
    - Mudança de estado
    - Fim de escrita na EEPROM
    - Comparador
  - Desvio do “PC” para 0x04

Arquitetura: PIC12F675

Registradores especiais

- Gerais (alguns)
  - STATUS
  - OPTIONS_REG
  - INTCON, PIR, PIE
  - PCL
- Portas
  - TRISIO, GPIO
- Contadores
  - TMR0, TMR1L, TMR1H

- EEPROM, comparador, tensão de referência, endereçamento indireto

	File Address		File Address
Indirect addr.(1)	00h	Indirect addr.(1)	80h
TMR0	01h	OPTION_REG	81h
PCL	02h	PCL	82h
STATUS	03h	STATUS	83h
FSR	04h	FSR	84h
GPIO	05h	TRISIO	85h
PCLATH	0Ah	PCLATH	8Ah
INTCON	0Bh	INTCON	8Bh
PIR1	0Ch	PIE1	8Ch
TMR1L	0Eh	PCON	8Eh
TMR1H	0Fh		8Fh
T1CON	10h	OSCCAL	90h
	15h	WPU	95h
	16h	IOC	96h
CMCON	19h	VRCON	99h
	1Ah	EEDATA	9Ah
	1Bh	EEADR	9Bh
	1Ch	EECON1	9Ch
	1Dh	EECON2(1)	9Dh
ADRESH(2)	1Eh	ADRESL(2)	9Eh
ADCON0(2)	1Fh	ANSEL(2)	9Fh
General Purpose Registers 64 Bytes	20h – 5Fh	accesses 20h-5Fh	A0h – DFh
	60h – 7Fh		E0h – FFh
Bank 0		Bank 1	

Arquitetura: PIC12F675

Registradores especiais gerais (alguns)

- STATUS
  - ULA
  - RESET
  - Banco de memória

Reserved	Reserved	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit 7							bit 0

bit 7 **IRP:** This bit is reserved and should be maintained as '0'

bit 6 **RP1:** This bit is reserved and should be maintained as '0'

bit 5 **RP0:** Register Bank Select bit (used for direct addressing)
0 = Bank 0 (00h - 7Fh)
1 = Bank 1 (80h - FFh)

bit 4 **$\overline{TO}$:** Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3 **$\overline{PD}$**: Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2 **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC**: Digit carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
For $\overline{borrow}$, the polarity is reversed.
1 = A carry-out from the 4th low order bit of the result occurred
0 = No carry-out from the 4th low order bit of the result

bit 0 **C**: Carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Arquitetura: PIC12F675

Registradores especiais gerais (alguns)

- OPTIONS_REG
  - *Prescaler*
  - Interrupção
  - Timer 0
  - *Pull-up*

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

bit 7 **$\overline{\text{GPPU}}$:** GPIO Pull-up Enable bit
1 = GPIO pull-ups are disabled
0 = GPIO pull-ups are enabled by individual port latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of GP2/INT pin
0 = Interrupt on falling edge of GP2/INT pin

bit 5 **T0CS:** TMR0 Clock Source Select bit
1 = Transition on GP2/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE:** TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on GP2/T0CKI pin
0 = Increment on low-to-high transition on GP2/T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the TIMER0 module

bit 2-0 **PS2:PS0:** Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Arquitetura: PIC12F675

Registradores especiais gerais (alguns)

- INTCON
  - Timer 0
  - Portas
  - Interrupção

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF
bit 7							bit 0

bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5 **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE:** GP2/INT External Interrupt Enable bit
1 = Enables the GP2/INT external interrupt
0 = Disables the GP2/INT external interrupt

bit 3 **GPIE:** Port Change Interrupt Enable bit(1)
1 = Enables the GPIO port change interrupt
0 = Disables the GPIO port change interrupt

bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit(2)
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** GP2/INT External Interrupt Flag bit
1 = The GP2/INT external interrupt occurred (must be cleared in software)
0 = The GP2/INT external interrupt did not occur

bit 0 **GPIF:** Port Change Interrupt Flag bit
1 = When at least one of the GP5:GP0 pins changed state (must be cleared in software)
0 = None of the GP5:GP0 pins have changed state

Arquitetura: PIC12F675

Registradores especiais gerais (alguns)

- PIE1
  - Interrupção

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIE	ADIE	—	—	CMIE	—	—	TMR1IE
bit 7							bit 0

bit 7	**EEIE:** EE Write Complete Interrupt Enable bit 1 = Enables the EE write complete interrupt 0 = Disables the EE write complete interrupt
bit 6	**ADIE:** A/D Converter Interrupt Enable bit 1 = Enables the A/D converter interrupt 0 = Disables the A/D converter interrupt
bit 5-4	**Unimplemented:** Read as '0'
bit 3	**CMIE:** Comparator Interrupt Enable bit 1 = Enables the comparator interrupt 0 = Disables the comparator interrupt
bit 2-1	**Unimplemented:** Read as '0'
bit 0	**TMR1IE:** TMR1 Overflow Interrupt Enable bit 1 = Enables the TMR1 overflow interrupt 0 = Disables the TMR1 overflow interrupt

Arquitetura: PIC12F675

Registradores especiais gerais (alguns)

- PIR1
  - Interrupção

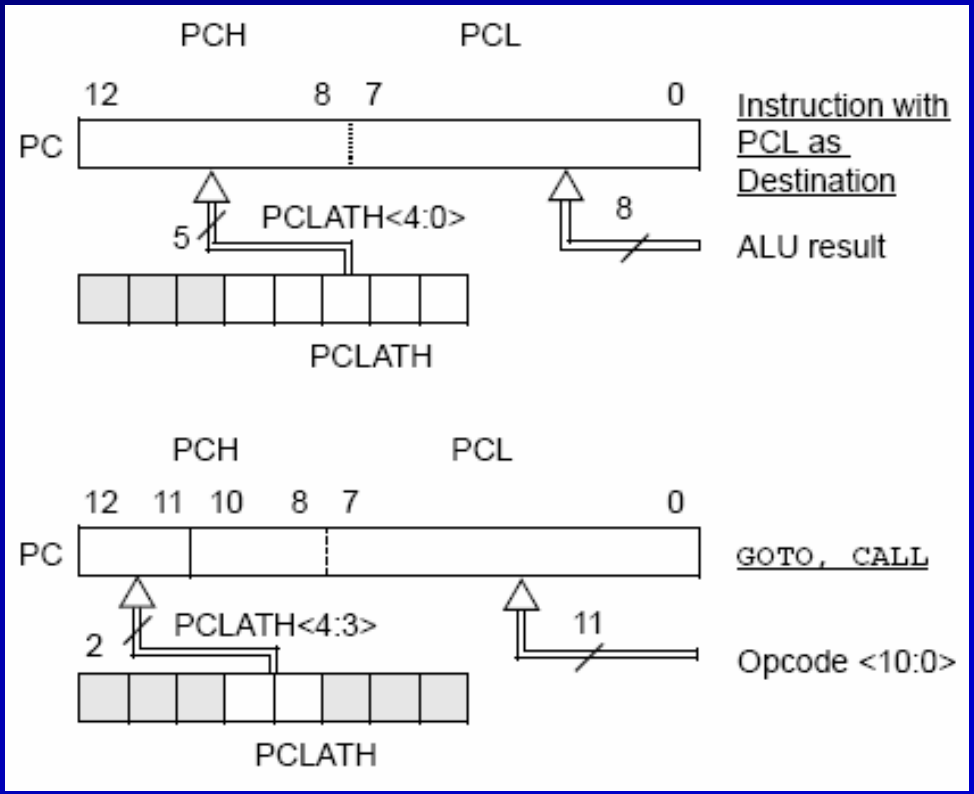
R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIF	ADIF	—	—	CMIF	—	—	TMR1IF
bit 7							bit 0

bit 7	**EEIF:** EEPROM Write Operation Interrupt Flag bit 1 = The write operation completed (must be cleared in software) 0 = The write operation has not completed or has not been started
bit 6	**ADIF:** A/D Converter Interrupt Flag bit (PIC12F675 only) 1 = The A/D conversion is complete (must be cleared in software) 0 = The A/D conversion is not complete
bit 5-4	**Unimplemented**: Read as '0'
bit 3	**CMIF**: Comparator Interrupt Flag bit 1 = Comparator input has changed (must be cleared in software) 0 = Comparator input has not changed
bit 2-1	**Unimplemented**: Read as '0'
bit 0	**TMR1IF**: TMR1 Overflow Interrupt Flag bit 1 = TMR1 register overflowed (must be cleared in software) 0 = TMR1 register did not overflow

Arquitetura: PIC12F675

Registradores especiais gerais (alguns)

- PCL
  - Parte acessível do PC



Arquitetura

Trabalhando com o **PIC** – RISC (35 instruções)

Conjunto de instruções – termos utilizados

Work → W

Literal → L e k

Bit → B e b

Skip → S

Clear → C

File → F e f

Destino → F e W

Teste → T

Set → S

Zero → Z

Conjunto de instruções – Mnemônicos base

ADD → soma

CLR → limpar, zerar (clear)

DEC → decremento de um

IOR → lógica “OU”

RL → rotacionar 1 bit p/ esquerda

SUB → subtração

XOR → lógica “OU exclusivo”

AND → lógica “E”

COM → complemento

INC → incremento de um

MOV → mover, transferir para algum lugar

RL → rotacionar 1 bit p/ direita

SWAP → inversão

Arquitetura

Trabalhando com o **PIC**

Exemplos

- Decrementar um registrador →

Conjunto de instruções – termos utilizados

Work → W
Literal → L e k
Bit → B e b
Skip → S
Clear → C
File → F e f
Destino → F e W
Teste → T
Set → S
Zero → Z

Conjunto de instruções – Mnemônicos base

ADD → soma
CLR → limpar, zerar (clear)
DEC → decremento de um
IOR → lógica “OU”
RL → rotacionar 1 bit p/ esquerda
SUB → subtração
XOR → lógica “OU exclusivo”
AND → lógica “E”
COM → complemento
INC → incremento de um
MOV → mover, transferir para algum lugar
RL → rotacionar 1 bit p/ direita
SWAP → inversão

Arquitetura

Trabalhando com o **PIC**

Exemplos

- Decrementar um registrador → DECF

Conjunto de instruções – termos utilizados

Work → W
Literal → L e k
Bit → B e b
Skip → S
Clear → C
File → F e f
Destino → F e W
Teste → T
Set → S
Zero → Z

Conjunto de instruções – Mnemônicos base

ADD → soma
CLR → limpar, zerar (clear)
DEC → decremento de um
IOR → lógica “OU”
RL → rotacionar 1 bit p/ esquerda
SUB → subtração
XOR → lógica “OU exclusivo”
AND → lógica “E”
COM → complemento
INC → incremento de um
MOV → mover, transferir para algum lugar
RL → rotacionar 1 bit p/ direita
SWAP → inversão

Arquitetura

Trabalhando com o **PIC**

Exemplos

- Decrementar um registrador → DECF
- DECFSZ →

Conjunto de instruções – termos utilizados

Work → W
Literal → L e k
Bit → B e b
Skip → S
Clear → C
File → F e f
Destino → F e W
Teste → T
Set → S
Zero → Z

Conjunto de instruções – Mnemônicos base

ADD → soma
CLR → limpar, zerar (clear)
DEC → decremento de um
IOR → lógica "OU"
RL → rotacionar 1 bit p/ esquerda
SUB → subtração
XOR → lógica "OU exclusivo"
AND → lógica "E"
COM → complemento
INC → incremento de um
MOV → mover, transferir para algum lugar
RL → rotacionar 1 bit p/ direita
SWAP → inversão

Prof. Mardson

Arquitetura

Trabalhando com o **PIC**

Exemplos

- Decrementar um registrador → DECF
- DECFSZ → decrementa (DEC) o registrador (F) e pula (S) se o resultado for zero (Z)

Conjunto de instruções – termos utilizados

Work → W
Literal → L e k
Bit → B e b
Skip → S
Clear → C
File → F e f
Destino → F e W
Teste → T
Set → S
Zero → Z

Conjunto de instruções – Mnemônicos base

ADD → soma
CLR → limpar, zerar (clear)
DEC → decremento de um
IOR → lógica "OU"
RL → rotacionar 1 bit p/ esquerda
SUB → subtração
XOR → lógica "OU exclusivo"
AND → lógica "E"
COM → complemento
INC → incremento de um
MOV → mover, transferir para algum lugar
RL → rotacionar 1 bit p/ direita
SWAP → inversão

Arquitetura

Trabalhando com o **PIC**

Exemplos

- Decrementar um registrador → DECF
- DECFSZ → decrementa (DEC) o registrador (F) e pula (S) se o resultado for zero (Z)

DECFSZ	**Decrement f, Skip if 0**
Syntax:	[*label*] DECFSZ f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(f) - 1 → (destination); skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are decremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruc-tion is executed. If the result is 0, then a NOP is executed instead, making it a 2TCY instruction.

Arquitetura

Trabalhando com o **PIC**

Conjunto de instruções: operações com registradores

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C,DC,Z	**1,2**
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z	**1,2**
CLRF	f	Clear f	1	00 0001 lfff ffff	Z	**2**
CLRW	-	Clear W	1	00 0001 0xxx xxxx	Z	
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z	**1,2**
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z	**1,2**
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00 1011 dfff ffff		**1,2,3**
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z	**1,2**
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00 1111 dfff ffff		**1,2,3**
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z	**1,2**
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z	**1,2**
MOVWF	f	Move W to f	1	00 0000 lfff ffff		
NOP	-	No Operation	1	00 0000 0xx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C	**1,2**
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C	**1,2**
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C,DC,Z	**1,2**
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff		**1,2**
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z	**1,2**

Arquitetura

Trabalhando com o **PIC**

Conjunto de instruções: operações com bits

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BIT-ORIENTED FILE REGISTER OPERATIONS						
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff		1,2
BSF	f, b	Bit Set f	1	01 01bb bfff ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01 10bb bfff ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01 11bb bfff ffff		3

Arquitetura

Trabalhando com o **PIC**

Conjunto de instruções: controle e operações com literais

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
LITERAL AND CONTROL OPERATIONS						
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z	
CALL	k	Call subroutine	2	10 0kkk kkkk kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$,$\overline{PD}$	
GOTO	k	Go to address	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk		
RETFIE	-	Return from interrupt	2	00 0000 0000 1001		
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk		
RETURN	-	Return from Subroutine	2	00 0000 0000 1000		
SLEEP	-	Go into Standby mode	1	00 0000 0110 0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z	