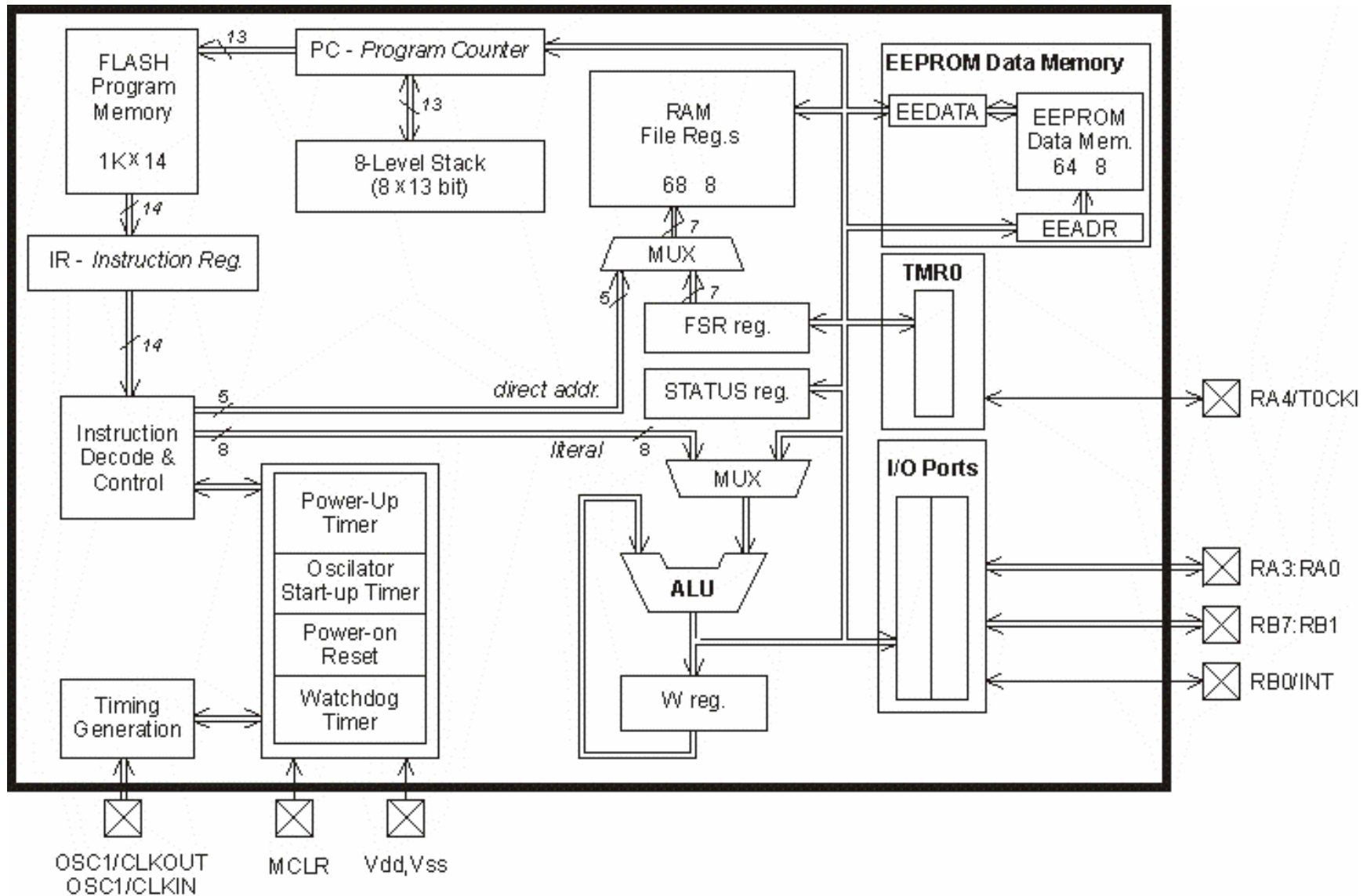
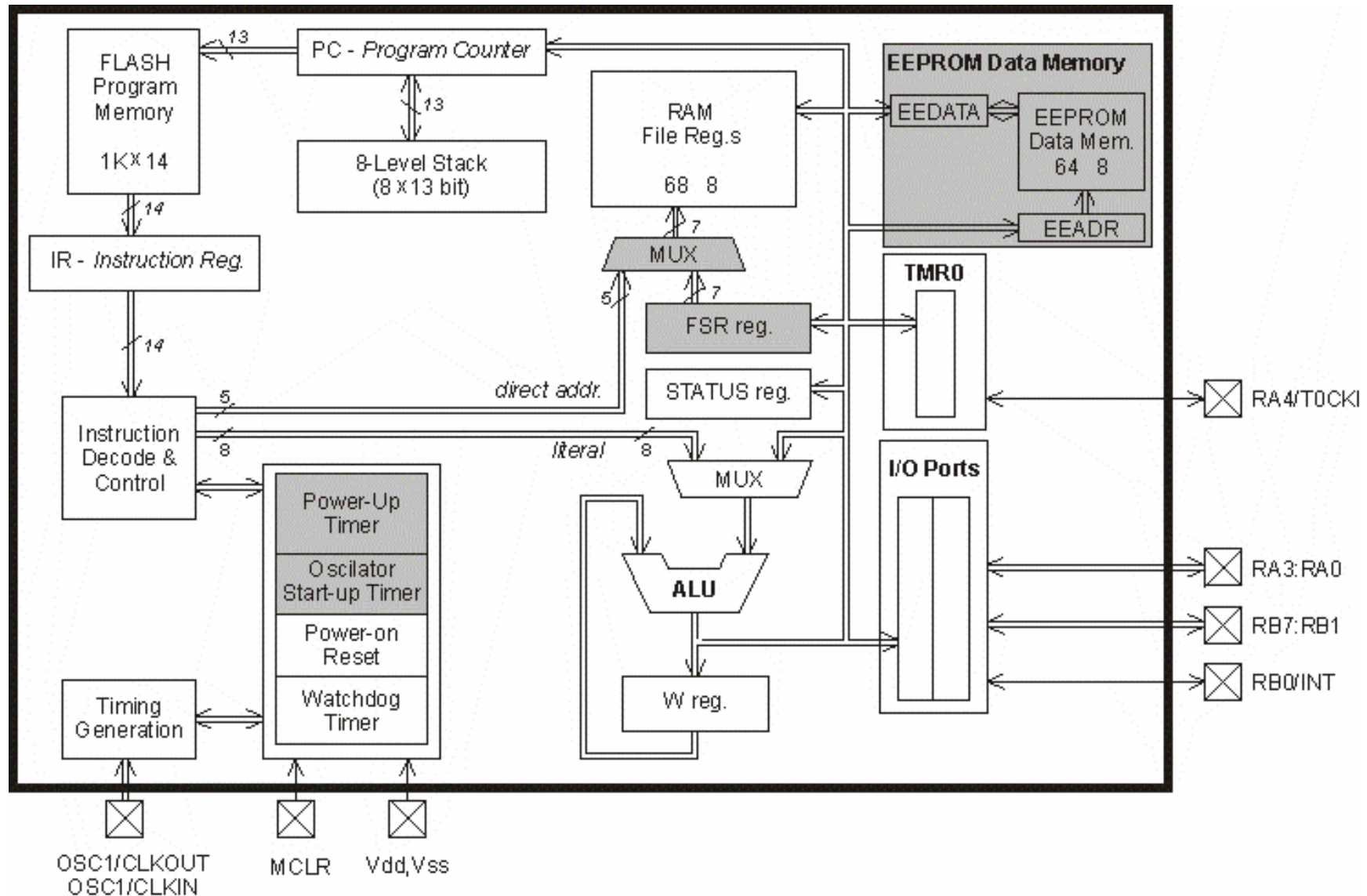


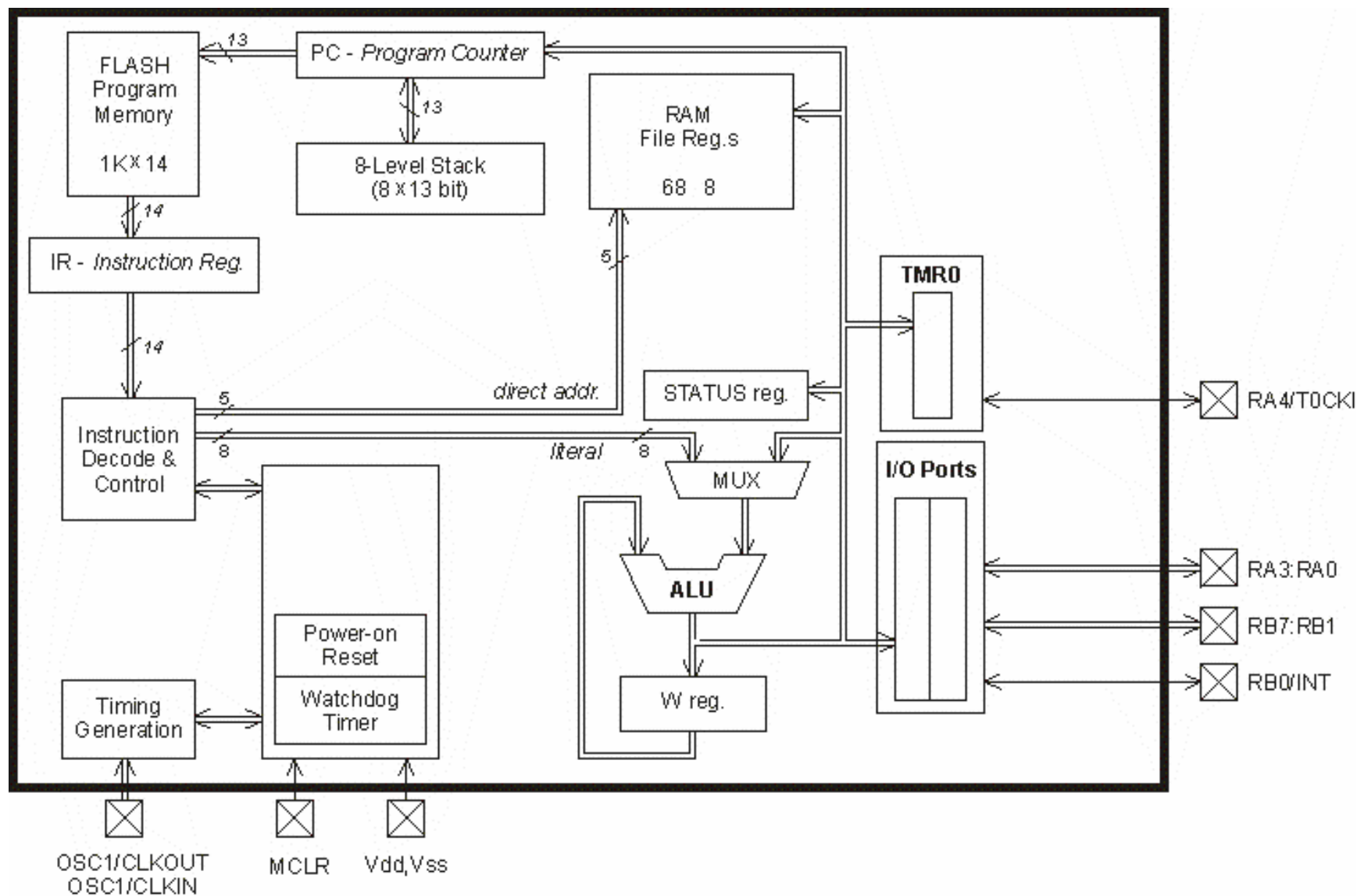
# Architektura procesora PIC16F84A



# Uproszczenia architektury



# Uproszczona architektury



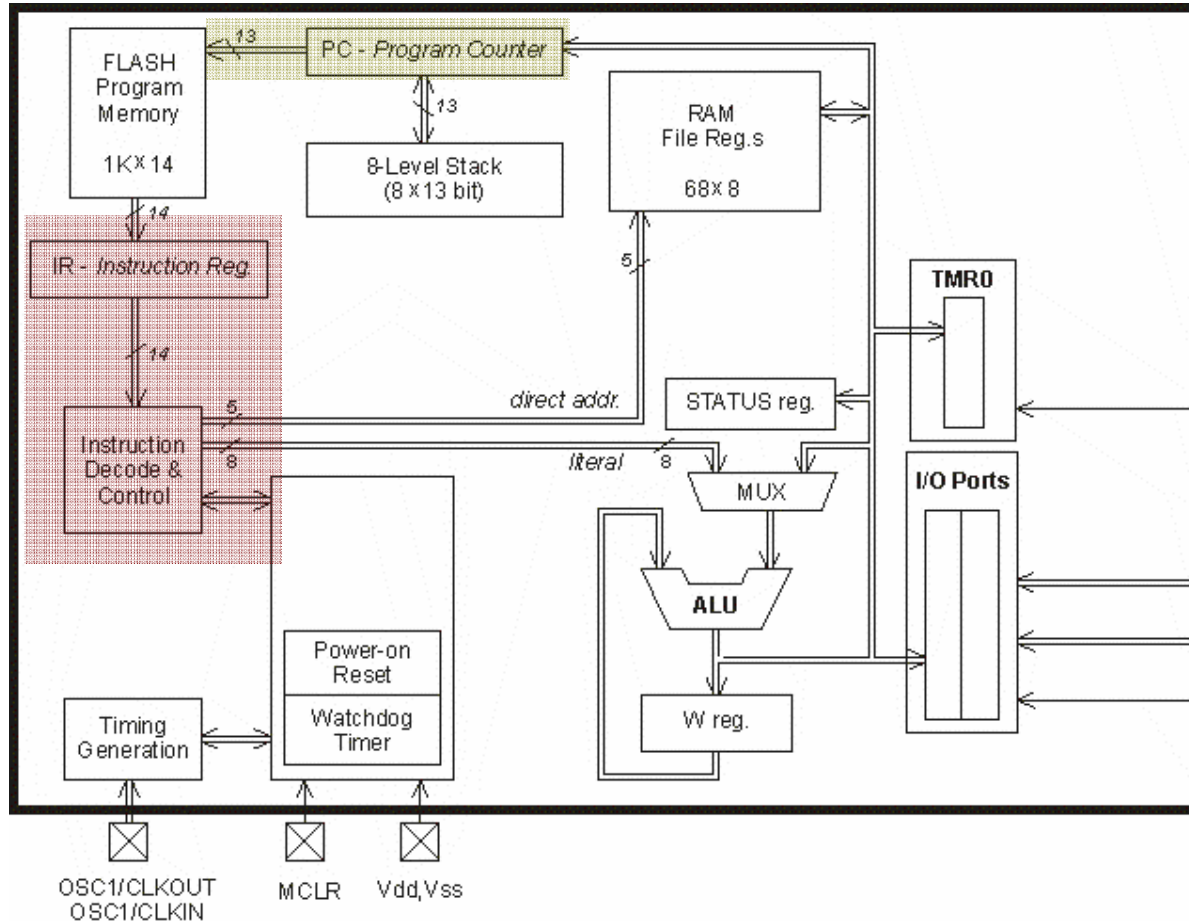
# Typy instrukcji

FIGURE 7-1: GENERAL FORMAT FOR INSTRUCTIONS

<b>1</b>	<p>Byte-oriented file register operations</p> <p style="text-align: center;">13                      8 7 6                      0</p> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: fit-content;"> <span style="border-right: 1px solid black; padding: 0 10px;">OPCODE</span> <span style="padding: 0 5px;">d</span> <span style="padding: 0 10px;">f (FILE #)</span> </div> <p>d = 0 for destination W            d = 1 for destination f            f = 7-bit file register address</p>
<b>2</b>	<p>Bit-oriented file register operations</p> <p style="text-align: center;">13                      10 9 7 6                      0</p> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: fit-content;"> <span style="border-right: 1px solid black; padding: 0 10px;">OPCODE</span> <span style="padding: 0 5px;">b (BIT #)</span> <span style="padding: 0 10px;">f (FILE #)</span> </div> <p>b = 3-bit bit address            f = 7-bit file register address</p>
<b>3</b>	<p>Literal and control operations</p> <p>General</p> <p style="text-align: center;">13                      8 7                      0</p> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: fit-content;"> <span style="border-right: 1px solid black; padding: 0 10px;">OPCODE</span> <span style="padding: 0 10px;">k (literal)</span> </div> <p>k = 8-bit immediate value</p>
<b>4</b>	<p>CALL and GOTO instructions only</p> <p style="text-align: center;">13                      11 10                      0</p> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: fit-content;"> <span style="border-right: 1px solid black; padding: 0 10px;">OPCODE</span> <span style="padding: 0 10px;">k (literal)</span> </div> <p>k = 11-bit immediate value</p>

Microchip PIC16F84A  
 Data Sheet  
 strona 35

# 3 i 4 - Operacje na literałach



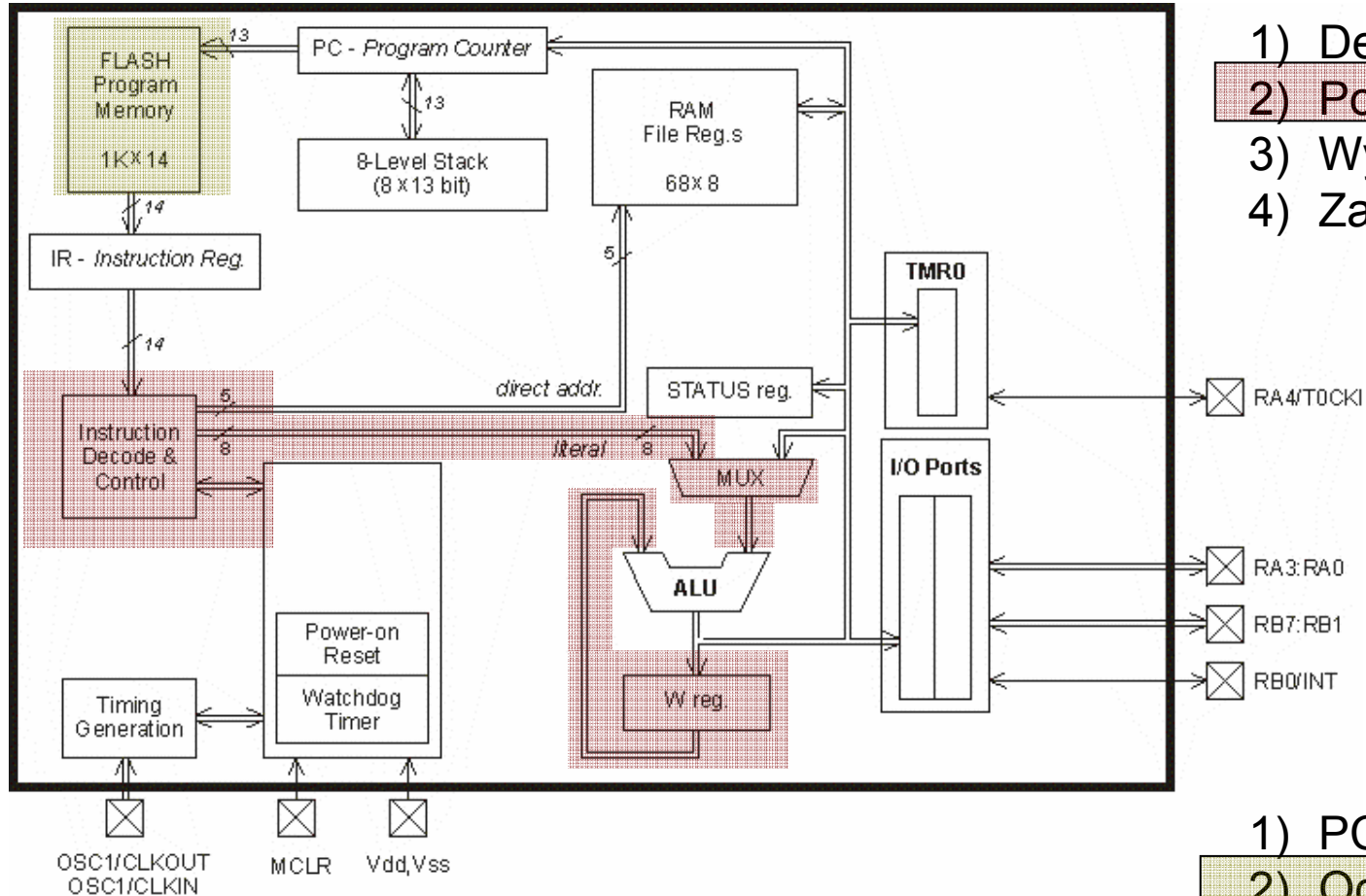
- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDLW k*  
*SUBLW k*  
*MOVLW k*  
*ANDLW k*  
*IORLW k*  
*XORLW k*  
*GOTO k*  
*RETLW k*  
*CALL k*

- 1) PC->szyna; PC++

- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# 3 i 4 - Operacje na literałach

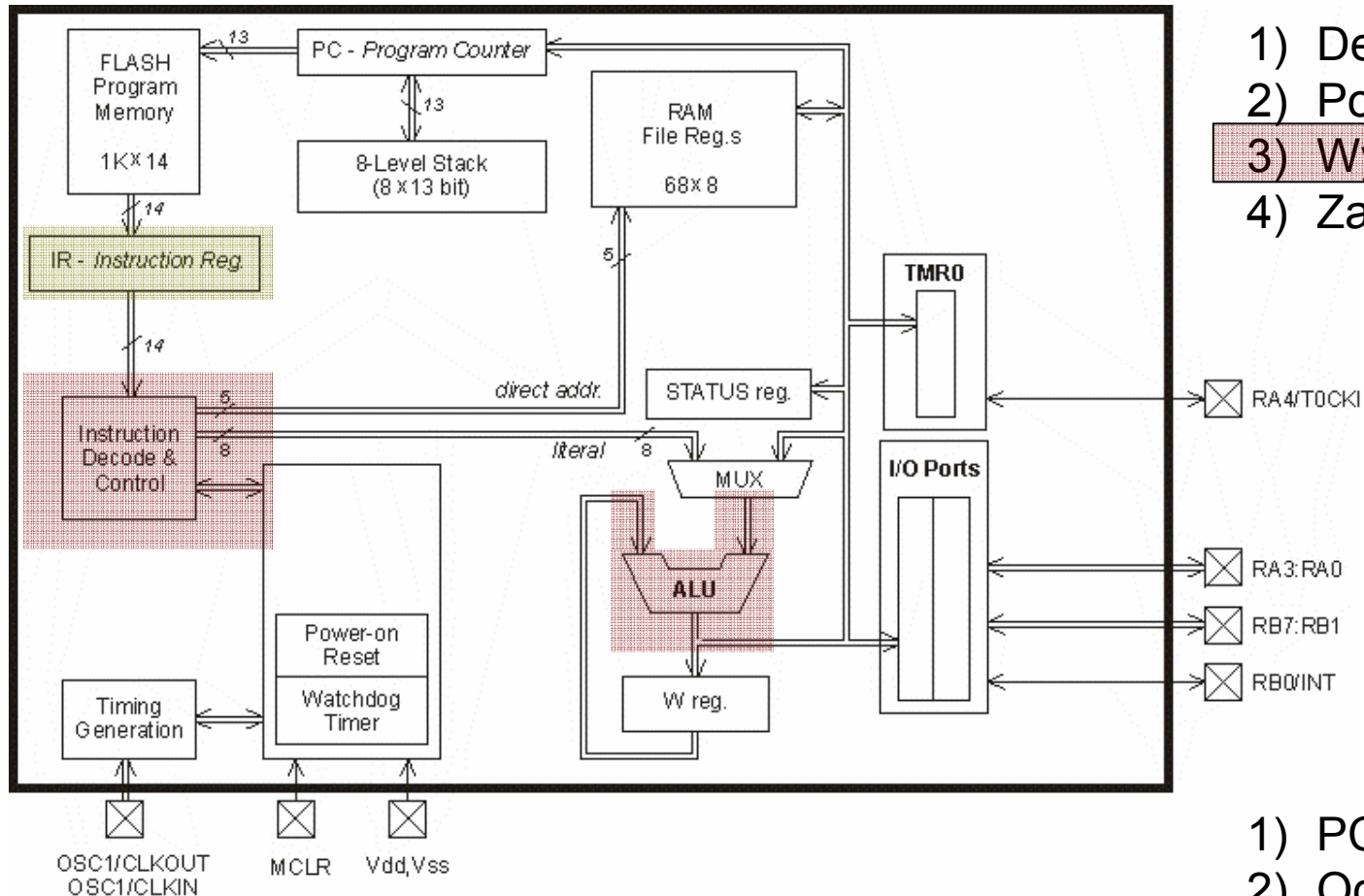


- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDLW k*  
*SUBLW k*  
*MOVLW k*  
*ANDLW k*  
*IORLW k*  
*XORLW k*  
*GOTO k*  
*RETLW k*  
*CALL k*

- 1) PC->szyna; PC++
- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# 3 i 4 - Operacje na literałach

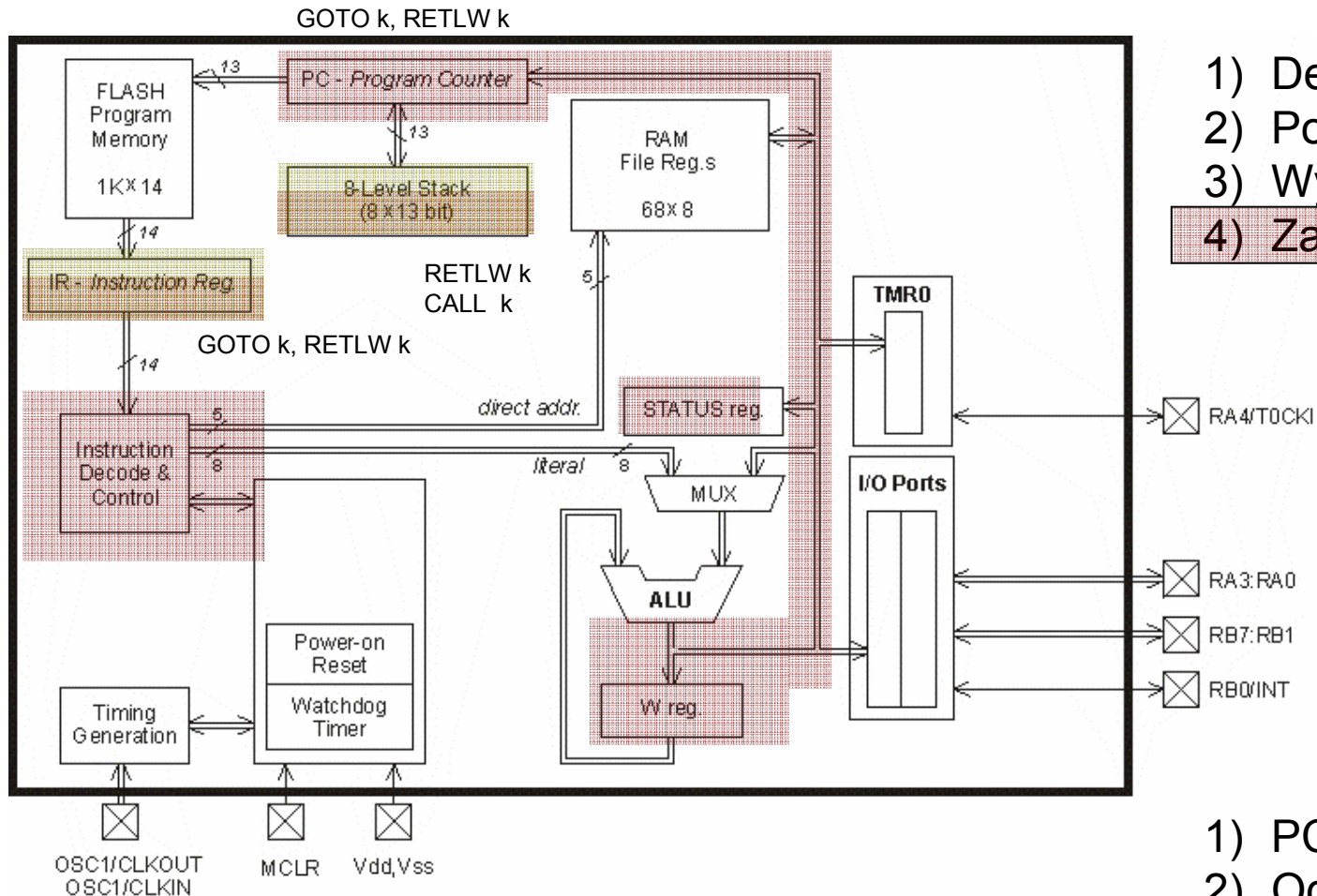


- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDLW k*  
*SUBLW k*  
*MOVLW k*  
*ANDLW k*  
*IORLW k*  
*XORLW k*  
*GOTO k*  
*RETLW k*  
*CALL k*

- 1) PC->szyna; PC++
- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# 3 i 4 - Operacje na literałach



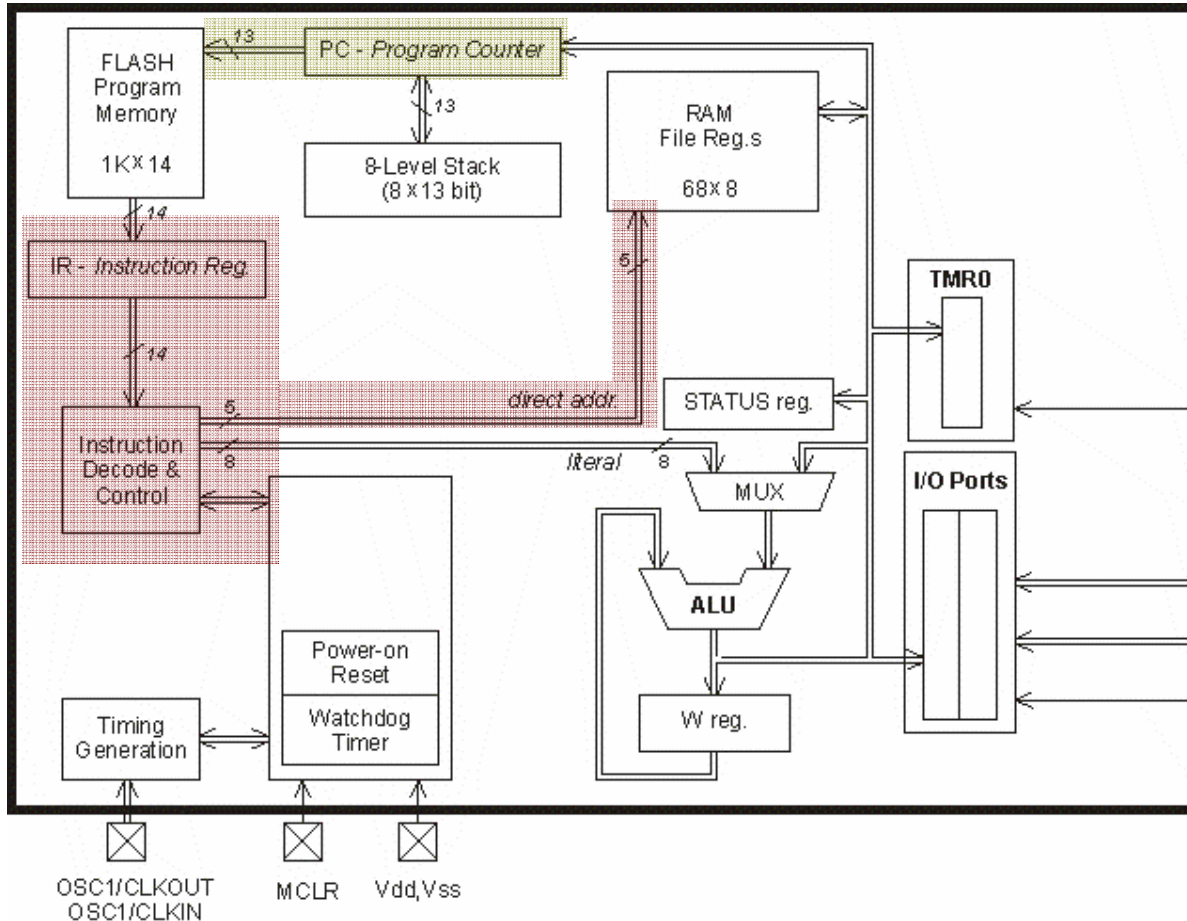
- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDLW k*  
*SUBLW k*  
*MOVLW k*  
*ANDLW k*  
*IORLW k*  
*XORLW k*  
*GOTO k*  
*RETLW k*  
*CALL k*

- 1) PC->szyna; PC++
- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych



# 1 i 2 - Operacje na rejestrach



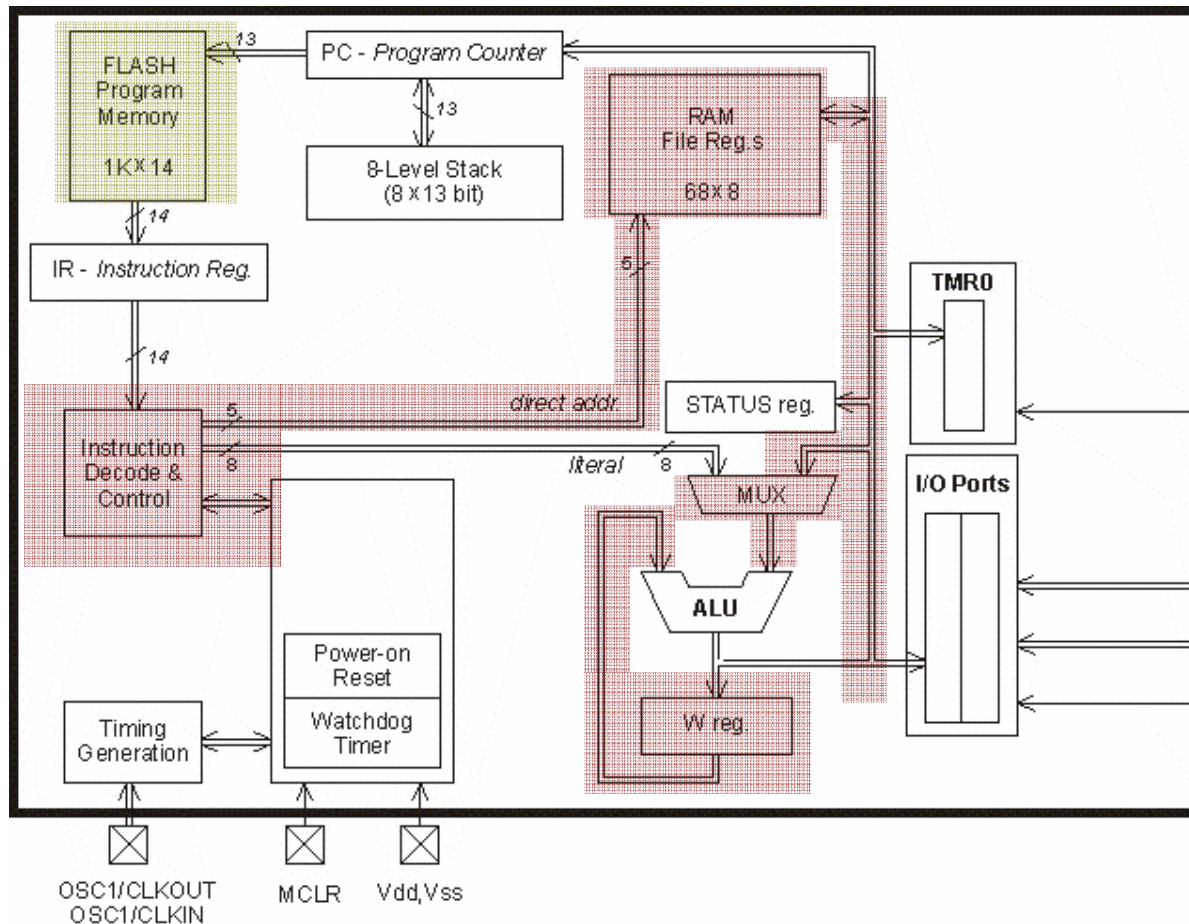
- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDWF F,d*  
*ANDWF F,d*  
*DECF F,d*  
*DECFSZ F,d*  
*MOVEF F,d*  
*MOVWF F*  
*RLF F,d*  
*BSF F,b*  
*BTFSS F,b*

- 1) PC->szyna; PC++

- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# 1 i 2 - Operacje na rejestrach

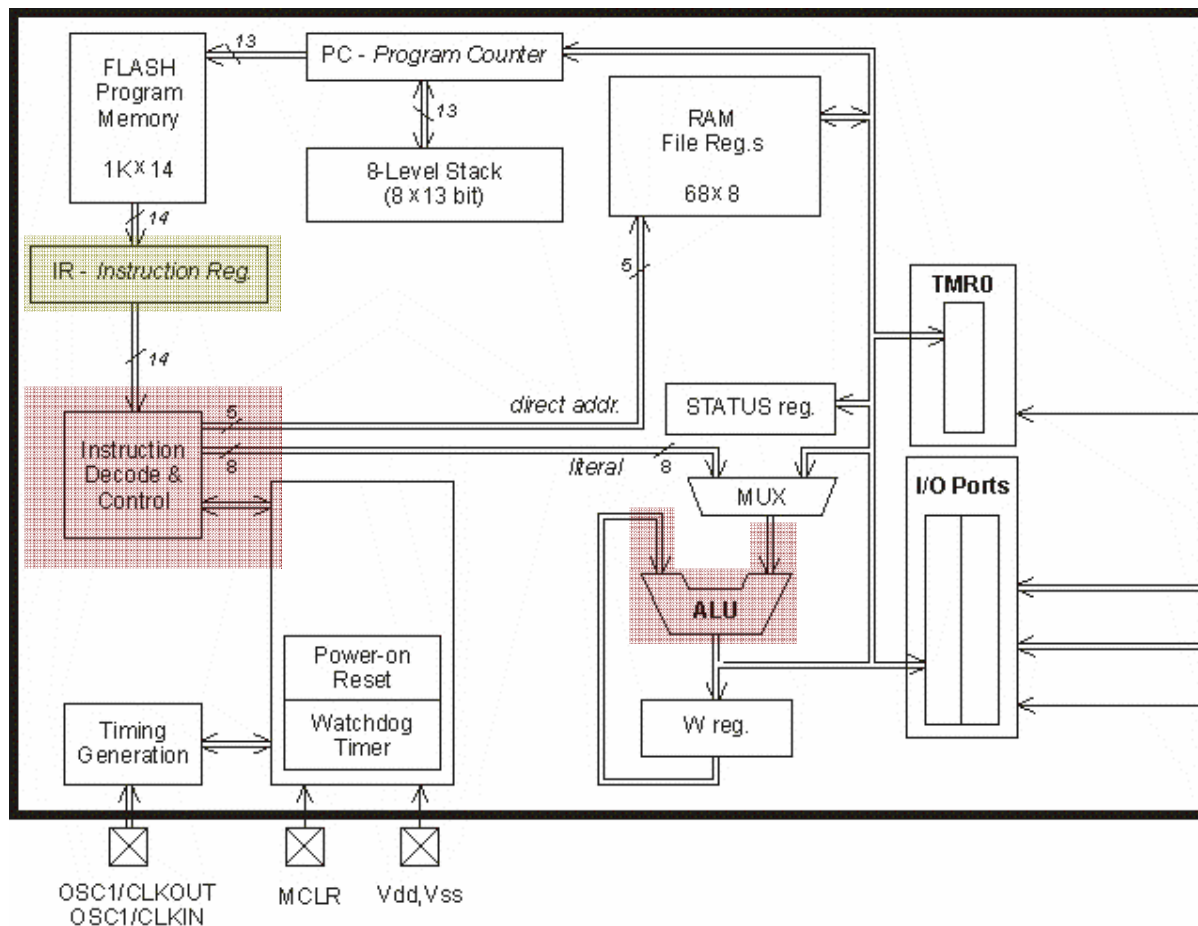


- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDWF F,d*  
*ANDWF F,d*  
*DECF F,d*  
*DECFSZ F,d*  
*MOVEF F,d*  
*MOVWF F*  
*RLF F,d*  
*BSF F,b*  
*BTFSS F,b*

- 1) PC->szyna; PC++
- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# 1 i 2 - Operacje na rejestrach

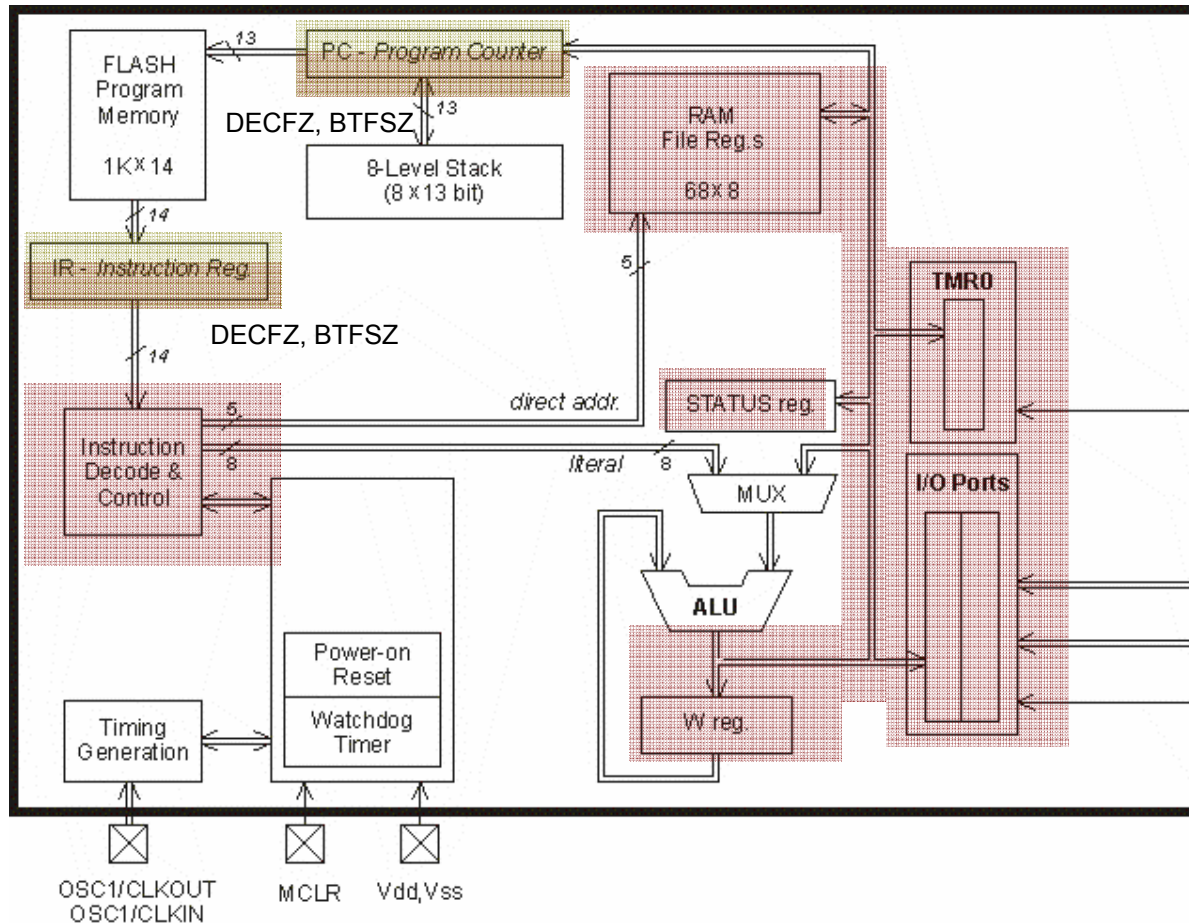


- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDWF F,d*  
*ANDWF F,d*  
*DECF F,d*  
*DECFSZ F,d*  
*MOVEF F,d*  
*MOVWF F*  
*RLF F,d*  
*BSF F,b*  
*BTFSS F,b*

- 1) PC->szyna; PC++
- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# 1 i 2 - Operacje na rejestrach



- 1) Dekodowanie instr.
- 2) Pobranie argumentów
- 3) Wykonanie (ALU)
- 4) Zapis wyników

*ADDWF F,d*  
*ANDWF F,d*  
*DECWF F,d*  
*DECFSZ F,d*  
*MOVEF F,d*  
*MOVWF F*  
*RLF F,d*  
*BSF F,b*  
*BTFSS F,b*

- 1) PC->szyna; PC++
- 2) Odczyt FLASH
- 3) Wypełnienie IR
- 4) Zmiany IR i PC  
zależne od danych

# Konkurs

---

**Zaproponować własną  
organizację procesora  
PIC16F84A**