inżynier przyszłości
Wzmocnienie potencjału dydaktycznego
Politechniki Poznańskiej

Marcin Rodziewicz

INSTRUCTIONS FOR 8051 MICROCONTROLLER

Materiały dydaktyczne dzięki dofinansowaniu ze środków
Europejskiego Funduszu Społecznego dystrybuowane są bezpłatnie

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

Poznan University of Technology
Faculty of Electronics and Telecommunications

8051 MICROCONTROLLER

Input/Output Pins, Ports and Internal Memory of the 8051 Microcontroller

Poznan 2014

1. Introduction

The 8051 microcontroller is equipped with four 8-bit ports: P0, P1, P2, and P3. All of these ports can be addressed either as a whole byte or as single bit. Therefore, there are 32 input/output pins for disposal. However, in the case of the DSM-51 evaluation board, only the pins of the P1 port and 6 pins of the P3 port can be controlled. To have control over all 32 pins, the program has to be written into the internal memory of the microcontroller.

In the internal structure of the microcontroller, the ports are placed in the Special Function Registers (SFR) area. Each register has its own address that identifies it. The special functions registers reside in the internal part of RAM (256x8) starting from the address 80H (128) ending at the address FFH (256). Each port has its own register that can be found at the following addresses: P0 - 80H, P1 - 90H, P2 - A0H, P3 - B0H.
The table below contains all special function registers of the 8051 microcontroller.

Table 1.1

Address	Symbol	Name
0E0H	* ACC	Accumulator
0F0H	* B	B register
0D0H	* PSW	Program Status Word register
81H	SP	Stack pointer
	DPTR	Data pointer
82H	DPL	Data pointer lower byte
83H	DPH	Data pointer higher byte
80H	* P0	Port 0
90H	* P1	Port 1
0A0H	* P2	Port 2
0B0H	* P3	Port 3
0B8H	* IP	Interrupt Priority register
0A8H	* IE	Interrupt Enable register
89H	TMOD	Timer Mode register
88H	* TCON	Timer Control register
8CH	TH0	Timer 0 higher byte
8AH	TL1	Timer 0 lower byte
8DH	TH1	Timer 1 higher byte
8BH	TL1	Timer 1 lower byte
98H	* SCON	Serial port Control register
99H	SBUF	Serial port buffer
87H	PCON	Power Control register

The RESET signal of the microcontroller sets all the bits of microcontrollers' ports to 1.
All the internal registers of the 8051 microcontroller are 8-bit registers. Apart from the special function registers, the 8051 microcontroller is equipped with 128 accessible bytes of RAM

located in the 00H – 7FH address area. The first 32 bytes of this memory range are dedicated to four register banks. Each register bank consists of 8 registers: R0 – R7. To access any of these registers, one can use their direct addresses or their names, e.g., MOV A, R0
The current register bank is selected by the RS0 and RS1 bits located in the PSW register. The selection procedure is shown in the table below.

Table 1.2

RS1	RS0	Bank	Address
0	0	0	00H-07H
0	1	1	08H-0FH
1	0	2	10H-17H
1	1	3	18H-1FH

The SFR range of RAM can be addressed directly or by name. Additionally, one can access single bits of the special functions registers. Apart from the direct addressing mode, to access the RAM one can use the indirect addressing mode. This can be achieved by using the R0 and R1 registers of any of the four register banks. Any byte from the 00H-7FH RAM area can be accessed in this way. Another special area of RAM is the 20H-2FH address range where single bits can be addressed similarly to the SFR bit addressing.

2. Exercise

1. Write a program which will light a LED diode connected to the pin 7 of the P1 port of the 8051 microcontroller.

   Note: Any pin of any port can be accessed by the following syntax:
   Px.y where x is the number of the port (0 – 3) and y is the number of the pin (0 – 7).

   Note: Prior to writing the program to the microcontroller's memory, it has to be assembled. To perform the assembling, one has to use the DSM51ASS program (the name of the file that contains the source code should be passed as a parameter of the program). As a result of the assembly process, two files will be generated, i.e., HEX file containing actual program translated to microprocessor's instructions, and LST file containing program listing.

2. Modify the program from Exercise 1, so that the light of the LED diode is toggled with a certain delay. The delay can be achieved by using the DELAY_100MS subroutine which is pre-written to the DSM-51 memory. The length of the delay can be modified by placing a required value in the Accumulator register.

3. Modify the program from Exercise 2, so that the light of the LED is alternately turned on with the buzzer connected to the pin 5 of the port P1. Draw an algorithm of this program. Use direct bit addressing to access the diode and the buzzer.

4. Write the program from Exercise 3 using direct byte addressing instead of direct bit addressing.

5. Write a program which will read and display the identifying numbers of successive register banks on the LCD of the DSM-51 evaluation board. The identifying number of the register bank should be written into one register of each bank, e.g., R7. To clear the LCD display, use the LCD_CLR subroutine (LCALL LCD_CLR). To write the value to the LCD display, use the WRITE_HEX subroutine (LCALL WRITE_HEX). WRITE_HEX displays the current value stored in the ACC register.

   Note: LCD-related procedures use the ACC for operation, so the value in the ACC register is modified after each call.

6. Write a program which will set the values of 10 subsequent bytes of RAM (starting from the 40H address) to zero. Use indirect addressing to access the appropriate bytes.
7. Write a program that writes the numbers 0-7 into subsequent registers of the third register bank. Use the register (name) addressing. Display the content of one of the registers on the LCD using the direct addressing method.
8. Write the program from Exercise 7 using indirect addressing for writing the numbers to the registers.