מדינת ישראל
משרד החינוך

סוג הבחינה: גמר לבתי־ספר לטכנאים ולהנדסאים
מועד הבחינה: אביב תשע"א, 2011
נספח לשאלונים: 711911, 711921

נוסחאון במיקרו־בקר 8051
לכיתה י"ג

(15 עמודים)

אין להעביר את הנוסחאון לנבחן אחר

מקום למדבקת נבחן

נוסחאון בשפת ASM–51

ARITHMETIC OPERATIONS

Mnemonic		Description	Byte	Cyc
ADD	A,Rn	Add register to Accumulator	1	1
ADD	A,direct	Add direct byte to Accumulator	2	1
ADD	A,@Ri	Add indirect RAM to Accumulator	1	1
ADD	A,#data	Add immediate data to Accumulator	2	1
ADDC	A,Rn	Add register to Accumulator with Carry	1	1
ADDC	A,direct	Add direct byte to A with Carry flag	2	1
ADDC	A,@Ri	Add indirect RAM to A with Carry flag	1	1
ADDC	A,#data	Add immediate data to A with Carry flag	2	1
SUBB	A,Rn	Subtract register from A with Borrow	1	1
SUBB	A,direct	Subtract direct byte from A with Borrow	2	1
SUBB	A,@Ri	Subtract indirect RAM from A w/Borrow	1	1
SUBB	A,#data	Subtract immed. data from A w/Borrow	2	1
INC	A	Increment Accumulator	1	1
INC	Rn	Increment register	1	1
INC	direct	Increment direct byte	2	1
INC	@Ri	Increment indirect RAM	1	1
DEC	A	Decrement Accumulator	1	1
DEC	Rn	Decrement register	1	1
DEC	direct	Decrement direct byte	2	1
DEC	@Ri	Decrement indirect RAM	1	1
INC	DPTR	Increment Data Pointer	1	2
MUL	AB	Multiply A & B	1	4
DIV	AB	Divide A by B	1	4
DA	A	Decimal Adjust Accumulator	1	1

LOGICAL OPERATION

Mnemonic		Destination	Byte	Cyc
ANL	A,Rn	AND register to Accumulator	1	1
ANL	A,direct	AND direct byte to Accumulator	2	1
ANL	A,@Ri	AND indirect RAM to Accumulator	1	1
ANL	A,#data	AND immediate data to Accumulator	2	1
ANL	direct,A	AND Accumulator to direct byte	2	1
ANL	direct,#data	AND immediate data to direct byte	3	2
ORL	A,Rn	OR register to Accumulator	1	1
ORL	A,direct	OR direct byte to Accumulator	2	1
ORL	A,@Ri	OR indirect RAM to Accumulator	1	1
ORL	A,#data	OR immediate data to Accumulator	2	1
ORL	direct,A	OR Accumulator to direct byte	2	1
ORL	direct,#data	OR immediate data to direct byte	3	2
XRL	A,Rn	Exclusive-OR register to Accumulator	1	1
XRL	A,direct	Exclusive-OR direct byte to Accumulator	2	1
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1
XRL	A,#data	Exclusive-OR immediate data to A	2	1
XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	1
XRL	direct,#data	Exclusive-OR immediate data to direct	3	2
CLR	A	Clear Accumulator	1	1
CPL	A	Complement Accumulator	1	1
RL	A	Rotate Accumulator Left	1	1
RLC	A	Rotate A Left through the Carry flag	1	1
RR	A	Rotate Accumulator Right	1	1
RRC	A	Rotate A Right through Carry flag	1	1
SWAP	A	Swap nibbles within the Accumulator	1	1

DATA TRANSFER

Mnemonic		Description	Byte	Cyc
MOV	A,Rn	Move register to Accumulator	1	1
MOV	A,direct	Move direct byte to Accumulator	2	1
MOV	A,@Ri	Move indirect RAM to Accumulator	1	1
MOV	A,#data	Move immediate data to Accumulator	2	1
MOV	Rn,A	Move Accumulator to register	1	1
MOV	Rn,direct	Move direct byte to register	2	2
MOV	Rn,#data	Move immediate data to register	2	1
MOV	direct,A	Move Accumulator to direct byte	2	1
MOV	direct,Rn	Move register to direct byte	2	2
MOV	direct,direct	Move direct byte to direct	3	2
MOV	direct,@Ri	Move indirect RAM to direct byte	2	2
MOV	direct,#data	Move immediate data to direct byte	3	2
MOV	@Ri,A	Move Accumulator to indirect RAM	1	1
MOV	@Ri,direct	Move direct byte to indirect RAM	2	2
MOV	@Ri,#data	Move immediate data to indirect RAM	2	1
MOV	DPTR,#data 16	Load Data Pointer with a 16-bit constant	3	2

DATA TRANSFER (cont.)

Mnemonic		Description	Byte	Cyc
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to A	1	2
MOVC	A,@A+PC	Move Code byte relative to PC to A	1	2
MOVX	A,@Ri	Move External RAM (8-bit addr) to A	1	2
MOVX	A,@DPTR	Move External RAM (16-bit addr) to A	1	2
MOVX	@Ri,A	Move A to External RAM (8-bit addr)	1	2
MOVX	@DPTR,A	Move A to External RAM (16-bit addr)	1	2
PUSH	direct	Push direct byte onto stack	2	2
POP	direct	Pop direct byte from stack	2	2
XCH	A,Rn	Exchange register with Accumulator	1	1
XCH	A,direct	Exchange direct byte with Accumulator	2	1
XCH	A,@Ri	Exchange indirect RAM with A	1	1
XCHD	A,@Ri	Exchange low-order Digit ind. RAM w/A	1	1

BOOLEAN VARIABLE MANIPULATION

Mnemonic		Description	Byte	Cyc
CLR	C	Clear Carry flag	1	1
CLR	bit	Clear direct bit	2	1
SETB	C	Set Carry flag	1	1
SETB	bit	Set direct Bit	2	1
CPL	C	Complement Carry flag	1	1
CPL	bit	Complement direct bit	2	1
ANL	C,bit	AND direct bit to Carry flag	2	2
ANL	C,/bit	AND complement of direct bit to Carry	2	2
ORL	C,bit	OR direct bit to Carry flag	2	2
ORL	C,/bit	OR complement of direct bit to Carry	2	2
MOV	C,bit	Move direct bit to Carry flag	2	1
MOV	bit,C	Move Carry flag to direct bit	2	2

PROGRAM AND MACHINE CONTROL

Mnemonic		Description	Byte	Cyc
ACALL	addr11	Absolute Subroutine Call	2	2
LCALL	addr16	Long Subroutine Call	3	2
RET		Return from subroutine	1	2
RETI		Return from interrupt	1	2
AJMP	addr11	Absolute Jump	2	2
LJMP	addr16	Long Jump	3	2
SJMP	rel	Short Jump (relative addr)	2	2
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2
JZ	rel	Jump if Accumulator is Zero	2	2
JNZ	rel	Jump if Accumulator is Not Zero	2	2
JC	rel	Jump if Carry flag is set	2	2
JNC	rel	Jump if No Carry flag	2	2
JB	bit,rel	Jump if direct Bit set	3	2
JNB	bit,rel	Jump if direct Bit Not set	3	2
JBC	bit,rel	Jump if direct Bit is set & Clear bit	3	2
CJNE	A,direct,rel	Compare direct to A & Jump if Not Equal	3	2
CJNE	A,#data,rel	Comp. immed. to A & Jump if Not Equal	3	2
CJNE	Rn,#data,rel	Comp. immed. to reg & Jump if Not Equal	3	2
CJNE	@Ri,#data,rel	Comp. immed. to ind. & Jump if Not Equal	3	2
DJNZ	Rn,rel	Decrement register & Jump if Not Zero	2	2
DJNZ	direct,rel	Decrement direct & Jump if Not Zero	3	2
NOP		No operation	1	1

Notes on data addressing modes:

Rn	-Working register R0-R7
direct	-128 internal RAM locations, any I/O port, control or status register
@Ri	-Indirect internal RAM location addressed by register R0 or R1
#data	-8-bit constant included in instruction
#data16	-16-bit constant included as bytes 2 & 3 of instruction
bit	-128 software flags, any I/O pin, control or status bit

Notes on program addressing modes:

addr16	-Destination address for LCALL & LJMP may be anywhere within the 64-Kilobyte program memory address space.
addr11	-Destination address for ACALL & AJMP will be within the same 2-Kilobyte page of program memory as the first byte of the following instruction.
rel	-SJMP and conditional jumps include an 8-bit offset byte. Range is +127/-128 bytes relative to first byte of the following instruction.

INSTRUCTIONS THAT AFFECT FLAG SETTINGS'

INSTRUCTION	FLAG C	0V	AC	INSTRUCTION	FLAG C	0V	AC
ADD	X	X	X	CLR C	0		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C,bit	X		
MUL	0	X		ANL C,/bit	X		
DIV	0	X		ORL C,bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C,bit	X		
RLC	X			CJNE	X		
SETB C	1						

המשך בעמוד 2

Special Function Registers

P3-Alternate Special Functions of Port 3

(MSB) (LSB)

RD	WR	T1	T0	INT1	INT0	TXD	RXD

Symbol	Position	Name and Significance	Symbol	Position	Name and Significance
RD	P3.7	Read data control output. Active low pulse generated by hardware when external data memory is read.	INT1	P3.3	Interrupt 1 input pin. Low-level or falling-edge triggered.
			INT0	P3.2	Interrupt 0 input pin. Low-level or falling-edge triggered.
WR	P3.6	Write data control output. Active low pulse generated by hardware when external data memory is written.	TXD	P3.1	Transmit Data pin for serial port in UART mode. Clock output in shift register mode.
T1	P3.5	Timer/counter 1 external input or test pin.	RXD	P3.0	Receive Data pin for serial port in UART mode. Data I/O pin in shift register mode.
T0	P3.4	Timer/counter 0 external input or test pin.			

המשך בעמוד 3

TMOD-Timer/Counter Mode Register

(MSB) (LSB)

GATE	C / $\overline{T}$	M1	M0	GATE	C / $\overline{T}$	M1	M0
TIMER1				TIMER0			

GATE — Gating control. When set, Timer/counter "x" is enabled only while "INTx" pin is high and "TRx" control bit is set. When cleared, timer/counter is enabled whenever "TRx" control bit is set.

C / $\overline{T}$ — Timer or Counter Selector. Cleared for Timer operation (input from internal system clock).
Set for Counter operation (input from "Tx" input pin).

M1	M0	Operating Mode
0	0	MCS-48 Timer "TLx" serves as five bit prescaler.
0	1	16-bit timer/counter. "THx" and "TLx" are cascaded; there is no prescaler.
1	0	8-bit auto-reload timer/counter. "THx" holds a value which is to be reloaded into "TLx" each time it overflows.
1	1	(Timer 0) TL0 is an eight-bit timer/counter controlled by the standard Timer 0 control bits. TH0 is an eight-bit timer only controlled by Timer 1 control bits.
1	1	(Timer 1) Timer/counter 1 stopped.

המשך בעמוד 4

TCON-Timer/Counter Control/Status Register

(MSB) (LSB)

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Symbol	Position	Name and Significance	Symbol	Position	Name and Significance
TF1	TCON.7	Timer 1 overflow Flag. Set by hardware on timer/ counter overflow. Cleared when interrupt processed.	IE1	TCON.3	Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TR1	TCON.6	Timer 1 Run control bit. Set/cleared by software to turn timer/counter on/off.	IT1	TCON.2	Interrupt 1 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TF0	TCON.5	Timer 0 overflow Flag. Set by hardware on timer/ counter overflow. Cleared when interrupt processed.	IE0	TCON.1	Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TR0	TCON.4	Timer 0 Run control bit. Set/cleared by software to turn timer/counter on/off.	IT0	TCON.0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

המשך בעמוד 5

SCON-Serial Port Control/Status Register

(MSB) (LSB)

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Symbol	Position	Name and Significance
SM0	SCON.7	Serial port Mode control bit 0. Set/cleared by software (see note).
SM1	SCON.6	Serial port Mode control bit 1. Set/cleared by software (see note).
SM2	SCON.5	Serial port Mode control bit 2. Set by software to disable reception of frames for which bit 8 is zero.
REN	SCON.4	Receiver Enable control bit. Set/cleared by software to enable/disable serial data reception.
TB8	SCON.3	Transmit Bit 8. Set/ cleared by hardware to determine state of ninth data bit transmitted in 9-bit UART mode.
RB8	SCON.2	Receive Bit 8. Set/ cleared by hardware to indicate state of ninth data bit received.
TI	SCON.1	Transmit Interrupt flag. Set by hardware when byte transmitted. Cleared by software after servicing.
RI	SCON.0	Receive Interrupt flag. Set by hardware when byte received. Cleared by software after servicing.

Note — the state of (SM0, SM1) selects:
(0,0) — Shift register I/O expansion.
(0,1) — 8 bit UART, variable data rate.
(1,0) — 9 bit UART, fixed data rate.
(1,1) — 9 bit UART, variable data rate.

המשך בעמוד 6

IE-Interrupt Enable Register

(MSB) (LSB)

EA	—	—	ES	ET1	EX1	ET0	EX0

Symbol	Position	Name and Significance
EA	IE.7	Enable All control bit. Cleared by software to disable all interrupts, independent of the state of IE.4-IE.0.
—	IE.6	(reserved)
—	IE.5	(reserved)
ES	IE.4	Enable Serial port control bit. Set/cleared by software to enable/disable interrupts from TI or RI flags.
ET1	IE.3	Enable Timer 1 control bit. Set/cleared by software to enable/disable interrupts from timer/ counter 1.
EX1	IE.2	Enable External interrupt 1 control bit. Set/cleared by software to enable/disable interrupts from INT1.
ET0	IE.1	Enable Timer 0 control bit. Set/cleared by software to enable/disable interrupts from timer/ counter 0.
EX0	IE.0	Enable External interrupt 0 control bit. Set/cleared by software to enable/disable interrupts from INT0.

המשך בעמוד 7

IP-Interrupt Priority Control Register

(MSB)	—	—	—	PS	PT1	PX1	PT0	PX0	(LSB)

Symbol	Position	Name and Significance
—	IP.7	(reserved)
—	IP.6	(reserved)
—	IP.5	(reserved)
PS	IP.4	Serial port Priority control bit. Set/cleared by software to specify high/low priority interrupts for Serial port.
PT1	IP.3	Timer 1 Priority control bit. Set/cleared by software to specify high/low priority interrupts for timer/counter 1.
PX1	IP.2	External interrupt 1 Priority control bit. Set/cleared by software to specify high/low priority interrupts for INT1.
PT0	IP.1	Timer 0 Priority control bit. Set/cleared by software to specify high/low priority interrupts for timer/counter 0.
PX0	IP.0	External interrupt 0 Priority control bit. Set/cleared by software to specify high/low priority interrupts for INT0.

Register	Address	Function
P0	80H*	Port 0
SP	81H	Stack Pointer
DPL	82H	Data Pointer (Low)
DPH	83H	Data Pointer (High)
TCON	88H*	Timer register
TMOD	89H	Timer Mode register
TL0	8AH	Timer 0 Low byte
TL1	8BH	Timer 1 Low byte
TH0	8CH	Timer 0 High byte
TH1	8DH	Timer 1 High byte
P1	90H*	Port 1
SCON	98H*	Serial Port Control register
SBUF	99H	Serial Port data Buffer
P2	0A0H*	Port 2
IE	0A8H*	Interrupt Enable register
P3	0B0H*	Port 3
IP	0B8H*	Interrupt Priority register
PSW	0D0H*	Program Status Word
ACC	0E0H*	Accumulator (direct address)
B	0F0H*	B register

Interrupt Source	Service Routine Starting Address
(Reset)	0000H
External 0	0003H
Timer/Counter 0	000BH
External 1	0013H
Timer/Counter 1	001BH
Serial Port	0023H

*=bit addressable register

המשך בעמוד 8

נוסחאון בשפת C של המיקרו־בקר 8051

נוסחאון זה מתאים למהדר Keil uVision3 . חלקים ממנו מתאימים גם למהדרים אחרים.

Data Types (טיפוסי נתונים)

Name	Description	תאור	Size	Range
bit	One Bit	ביט בודד	1 bit	0 to 1
char	Character or small integer	תו בודד או בית	1 byte	–128 to 127
unsigned char	Unsigned small integer	בית אחד ללא סימן	1 byte	0 to 255
int	Integer	מספר שלם	2 bytes	–32768 to 32767
unsigned int	Unsigned integer	מספר שלם ללא סימן	2 bytes	0 to 65535
long	Long integer	מספר שלם ארוך	4 bytes	–2147483648 to 2147483647
unsigned long	Unsigned long integer	מספר שלם ארוך ללא סימן	4 bytes	0 to 4294967295
float	Floating point number	מספר ממשי	4 bytes	+/– 1.175494E–38 to +/– 3.402823E+38
sbit	Special Bit	ביט מיוחד	1 bit	0 to 1
sfr	8 bits special Function Registers	בית מיוחד	1 byte	0 to 255
sfr16	16 bits special Function Registers	בית כפול מיוחד	2 bytes	0 to 65535

דוגמאות:

```
unsigned char a;
int b, c;
sfr P1=0x90;
sbit P1_7 = 0x97;
```

המשך בעמוד 9

Memory Areas (אזורי הזיכרון)

Name	Description	Example
data	places the variable in directly addressable RAM in the micro core	unsigned int data dnum;
xdata	places the variable in external RAM	unsigned char xdata xnum_at_0x8000; *
idata	places the variable in indirectly addressable memory within the micro core	int idata inum;
code	places the variable in program memory	unsigned char code cnum=0xAA;

* _at_ – places the variable in absolute address.

Preprocessor-directives (הנחיות לקדם־מהדר)

Description	Syntax	Example
macro definitions	#define identifier replacement	#define LED P 1_7

Operators (אופרטורים)

Description	תאור	Operator
Assignment	השמה	=

Initalization of variables (אתחול משתנים)

```
unsigned char  d=0;
d=75;        // decimal  number
d=0x4B;      // hexadecimal  number
```

identifier - מזהה ; replacement - תחליף

המשך בעמוד 10

Arithmetic Operators (אופרטורים חשבוניים)

Operator	תאור	Description
+	חיבור	addition
−	חיסור	subtraction
*	כפל	multiplication
/	חילוק	division
%	שארית	modulo

Relational and equality operators (אופרטורים להשוואה ויחסים)

Operator	תאור	Description
= =	שווה	Equal to
! =	שונה	Not equal to
>	גדול מ־	Greater than
<	קטן מ־	Less than
> =	גדול מ־/ שווה	Greater than or equal to
< =	קטן מ־/ שווה	Less than or equal to

Logical operators (אופרטורים לוגיים בין ביטויים)

Operator	תאור	Description
!	היפוך	NOT
&&	וגם	AND
||	או	OR

Bitwise Operators (אופרטורים על סיביות)

Operator	תאור	Description
&	וגם	AND
|	או כולל	Inclusive OR
^	או מוציא	Exclusive OR (XOR)
~	היפוך בית	Byte inversion
!	היפוך סיבית	Bit inversion
<<	הזזה שמאלה	Shift Left
>>	הזזה ימינה	Shift Right

המשך בעמוד 11

Conditional Structures (מבני בקרה – משפטי תנאי)

Description	Syntax	Example

if	if (condition)
	{
	statements ;
	}
		if (d == 100)
		{
		P1 = 0xFF;
		}

if .. else	if (condition)
	statement ;
	else
	statement ;
		if (d == 100)
		P1 = 0xFF;
		else
		P1=0;

if .. else if .. else	if (condition)
	statement ;
	else if (condition)
	statement ;
	else
	statement ;
		if (d > 0)
		P1=4;
		else if (d < 0)
		P1=2;
		else
		P1=1;

condition – תנאי ; statement – הצהרה

המשך בעמוד 12

Iteration Structures (מבני בקרה – לולאות)

Description	Syntax	Example

while loop	while (expression)
	{
	statements ;
	}
	
		while (n>0)
		{
		P1=n;
		n– –;
		}

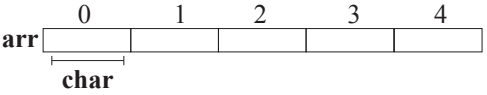
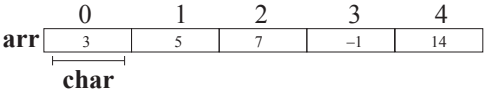
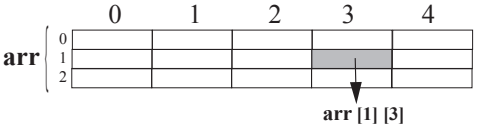
do-while loop	do
	{
	statements ;
	}
	while (condition);
		do
		{
		n=P1;
		}
		while (n != 0);

for loop	for (initialization; condition; increase)
	{
	statements ;
	}
	
		for (i=0; i<10; i++)
		{
		P1=i;
		}

condition – תנאי ; statement – הצהרה ; initialization – אתחול ; increase – קידום

המשך בעמוד 13

Arrays (מערכים)

Description	Syntax	Example
הגדרת מערך חד מימדי 	type name [number of elements];	char arr[5];
אתחול והצבת ערכים במערך 	type name [number of elements] = {value1,..valueN};	char arr[5] = {3,5,7,–1, 14};
הגדרת מערך דו מימדי 	type name [number of elements] [number of elements];	char arr[3][5];

elements – פרטים ; value – ערך

Structure of a program (מבנה כללי של תוכנית)

```
#include <8051.h>        //including headers for the SFR definitions
                           of your microcontroller

void main()

{

  while(1)

  {

      //Your code

  }

}
```

המשך בעמוד 14

Functions (פונקציות)

Description	Syntax	Example

Functions with no type and no argument	void name (void)
	{
	statements;
	
	}
		void OutData(void)
		{
		P1=0xAB;
		}
		void main()
		{
		OutData();
		}

Functions with no type	void name (parameter1, parameter2, ...)
	{
	statements;
	
	}
		void OutData(unsigned char a, unsigned char b)
		{
		P0=a;
		P1=b;
		}
		void main()
		{
		OutData(0xF,0xF0);
		}

Functions with type and argument	type name (parameter1, parameter2, ...)
	{
	statements;
	
	}
		unsigned char InOutData(unsigned char a)
		{
		P1=a;
		return P0;
		}
		void main()
		{
		unsigned char r;
		r = InOutData(63);
		P2=r;
		}

parameter – ערך המועבר לפונקציה ; statement – הצהרה ; type – טיפוס ; argument – טיעון

המשך בעמוד 15

Interrput Service Routines (שגרות לטיפול בפסיקות)

Interrupt Number	Description	Address
0	External INT 0	0003h
1	Timer 0	000Bh
2	External INT 1	0013h
3	Timer 1	001Bh
4	Serial port	0023h

דוגמה:

```
void int2sub () interrupt 2
{
      // Interrupt code
}
```