

Лабораторна робота №7

Тема: **ВИВЧЕННЯ АРИФМЕТИЧНИХ МОЖЛИВОСТЕЙ ОЕВМ**

Мета роботи: Ознайомлення з арифметичними можливостями ОЕВМ без застосування процедур множення та ділення. Отримання практичних навичок програмування МК.

1 Завдання

Задача1. Розробіть алгоритм і напишіть програму на мові Асемблера для обробки даних за допомогою не рекурсивного фільтру, де y_n - вихідне значення в n -й момент часу, $x_n, x_{n-1} \dots$ - 8-розрядні вхідні значення в n -й, $n-1$ -й ... моменти часу. Програму розробіть без застосування процедур множення і ділення.

Таблиця 7.1

Варіант	Функція
1	$y_n = \frac{1,25(x_n + x_{n-1})}{4} - 9,5x_{n-2} + \frac{3}{4}x_{n-3}$
2	$y_n = 0,5 \cdot \left(x_n + 2 \left(x_{n-1} - \frac{3,25}{4} x_{n-2} \right) \right) - \frac{1}{2} x_{n-3}$
3	$y_n = \frac{\left(\frac{4x_n - x_{n-1}}{0,25} + \frac{(3x_{n-2} - x_{n-3})}{0,5} \right)}{0,25}$
4	$y_n = x_n + \frac{(x_{n-1} + (0,5x_{n-2} - 2x_{n-3}))}{0,5}$
5	$y_n = 0,25 \cdot \left(x_n + 9 \cdot \left(x_{n-1} - 0,5x_{n-2} + \frac{1}{8}x_{n-3} \right) \right)$
6	$y_n = 2 \cdot \left(\frac{0,75x_n + 2x_{n-1}}{0,5} + \frac{(3x_{n-2} - x_{n-3})}{0,25} \right) - 5$
7	$y_n = 0,25 \cdot \left(x_n + 3 \cdot \left(\frac{1,5}{2} x_{n-1} - 0,5x_{n-2} \right) + \frac{1}{4} x_{n-3} \right)$
8	$y_n = \frac{1}{4} \cdot \left(0,25x_n + 6x_{n-1} - 3,25x_{n-2} + \frac{9}{8}x_{n-3} \right)$
9	$y_n = 0,75x_n + \frac{\left(6x_{n-1} - \frac{(3x_{n-2} - 2x_{n-3})}{0,5} - 45 \right)}{0,25}$

10	$y_n = 2 \cdot \left(\left(x_n + 0,5 \cdot \left(\frac{3,25}{4} x_{n-1} - x_{n-2} \right) \right) - \frac{3}{4} x_{n-3} \right)$
11	$y_n = \frac{(2x_n + 0,5x_{n-1})}{2} + 9,5 \cdot \left(x_{n-2} + \frac{3}{4} x_{n-3} \right)$
12	$y_n = \frac{\left(2x_n + \frac{\left(3x_{n-1} + \frac{(x_{n-2} + 3x_{n-3})}{0,5} \right)}{0,25} \right)}{0,5}$
13	$y_n = 1,25x_n + 3(x_{n-1} - 3,25x_{n-2}) + 0,75x_{n-3}$
14	$y_n = 0,5 \cdot \left(6x_n + 2 \cdot \left((9x_{n-1} - 0,75x_{n-2}) + \frac{1}{4} x_{n-3} \right) \right)$
15	$y_n = 0,25x_n + 2 \left(x_{n-1} - \frac{3,25}{4} x_{n-2} \right) - \frac{0,25}{2} x_{n-3}$
16	$y_n = \frac{(x_n - 0,25x_{n-1})}{2} + \left(\frac{1,5 \cdot (2x_{n-2} - x_{n-3})}{4} \right)$
17	$y_n = 1,5x_n + 8 \left(0,5x_{n-1} - \frac{0,75}{2} x_{n-2} + 0,125x_{n-3} \right)$
18	$y_n = (1,25x_n + 2x_{n-1} + 1) - 0,25 \cdot \left(x_{n-2} + \frac{5}{2} x_{n-3} \right) - 5$
19	$y_n = 0,5 \cdot \left(1,25x_n + 2 \cdot \left(\frac{3}{4} x_{n-1} - 0,5 \cdot (2x_{n-2} + x_{n-3}) \right) \right)$
20	$y_n = \frac{\left(0,625x_n + \frac{\left(\frac{x_{n-1} + 3x_{n-2}}{0,5} \right)}{0,5} - 0,5x_{n-3} \right)}{0,5}$
21	$y_n = \frac{1}{2} \cdot \left(0,375x_n + 2x_{n-1} - 2,25x_{n-2} + \frac{3}{8} x_{n-3} \right)$
22	$y_n = 7x_n + \frac{(3x_{n-1} - 2,5x_{n-2})}{0,5} - 0,375x_{n-3}$
23	$y_n = 3,25x_n + 2 \cdot \left(x_{n-1} - 1,25 \left(x_{n-2} + \frac{3}{2} x_{n-3} \right) \right)$
24	$y_n = \frac{3 \cdot (4x_n + 0,375x_{n-1} - x_{n-2})}{2} + \frac{(x_{n-3} - 1)}{0,5}$
25	$y_n = 1,75x_n + \frac{(x_{n-1} - 1,25(x_{n-2} + 2x_{n-3}))}{0,25}$

3 Порядок виконання роботи

- 3.1 Запуск програми емулятора SCM.
- 3.2 Складання та введення програми на Асемблері.
- 3.3 Налаштування програми.
- 3.4 Виконання програми.

4 Зміст звіту

- 4.1 Назва та мета роботи
- 4.2 Лістинг налагодженої програми (з коментарями).
- 4.3 Результат виконання програми
- 4.4 Висновки по роботі

5 Контрольні питання

- 5.1 Як здійснити множення без процедури множення на Асемблері для МК51?
- 5.2 Як здійснити ділення без процедури множення на Асемблері для МК51?
- 5.3 Як здійснити арифметичний суму на Асемблері для МК51?
- 5.4 Як здійснити арифметичний різницю на Асемблері для МК51?
- 5.5 Як здійснити скидання прапора переносу C?

ДОДАТОК

Таблиця 7.2 - Система команд

Мнемокод	КОП	Мнемокод	КОП	Мнемокод	КОП
ACALL 0xxH	11	AJMP 5XXH	A1	DA A	D4
ACALL 1xxH	31	AJMP 6XXH	C1	DEC A	14
ACALL 2xxH	51	AJMP 7XXH	E1	DEC ad	15
ACALL 3xxH	71	ANL A, ad	55	DEC R0	18
ACALL 4xxH	91	ANL A, R0	58	DEC R1	19
ACALL 5xxH	B1	ANL A, R1	59	DEC R2	1A
ACALL 6xxH	D1	ANL A, R2	5A	DEC R3	1B
ACALL 7xxH	F1	ANL A, R3	5B	DEC R4	1C
ADD A, ad	25	ANL A, R4	5C	DEC R5	1D
ADD A, R0	28	ANL A, R5	5D	DEC R6	1E
ADD A, R1	29	ANL A, R6	5E	DEC R7	1F
ADD A, R2	2A	ANL A, R7	5F	DEC @R0	16
ADD A, R3	2B	ANL A, @R0	56	DEC @R1	17
ADD A, R4	2C	ANL A, @R1	57	DIV AB	84
ADD A, R5	2D	ANL A, #d	54	DJNZ ad, rel	D5
ADD A, R6	2E	ANL ad, A	52	DJNZ R0, rel	D8

ADDA, R7	2F	ANL ad, #d	S3	DJNZ R1, rel	D9
ADD A, @R0	26	ANL C, bit	82	DJNZ R2, rel	DA
ADD A, @R1	27	ANL C, /bit	BO	DJNZ R3, rel	DB
ADD A, #d	34	CJNE A, ad, rel	B5	DJNZ R4, rel	DC
ADDC A, ad	35	CJNE A, #d, rel	B4	DJNZ R5, rel	DD
ADDC A, R0	38	CJNE R0, #d, rel	B8	DJNZ R6, rel	DE
ADDC A, R0	39	CJNE R1, #d, rel	B9	DJNZ R7, rel	DF
ADDC A, R0	3A	CJNE R2, #d, rel	BA	INC a	04
ADDC A, R0	3B	CJNE R3, #d, rel	BB	INC ad	05
ADDC A, R0	3C	CJNE R4, #d, rel	BC	INC DPTR	A3
ADDC A, R0	3D	CJNE R5, #d, rel	BD	INC R0	08
ADDC A, R0	3E	CJNE R6, #d, rel	BE	INC R1	09
ADDC A, R0	3F	CJNE R7, #d, rel	BF	INC R2	0A
ADDC A, @R0	36	CJNE @R0, #d, rel	B6	INC R3	0B
ADDC A, @R1	37	CJNE @R1, #d, rel	B7	INCR4	0C
ADDC A, #d	24	CLR A	E4	INC R5	0D
AJMP 0XXH	01	CLR bit	C2	INC R6	0E
AJMP 1XXH	21	CLR C	C3	INC R7	0F
AJMP 2XXH	41	CPL A	F4	INC @R0	06
AJMP 3XXH	61	CPL bit	B2	INC @R1	07
AJMP 4XXH	81	CPL C	B3	JB bit, rel	20
				JBC bit, rel	10

Продовження таблиці 7.2

Мнемокол	КОП	Мнемокол	КОП	Мнемокол	КОП
JC rel	40	MOV ad, @R0	86	MOV R7, ad	AF
JMP @A+DPTR	73	MOV ad, @R1	87	MOV R7, #d	7F
JNB bit, rel	30	MOV ad, #d	75	MOV @R0, A	F6
JNC rel	50	MOV ad, ads	85	MOV @R0, ad	A6
JNZ rel	70	MOV bit, C	92	MOV @R0, #d	76
JZ rel	60	MOV C, bit	A2	MOV @R1, A	F7
LCALL ad16	12	MOV DPTR, #dl6	90	MOV @R1, ad	A7
LJMP ad 16	02	MOV R0, A	F8	MOV @R1, #d	77
MOV A, ad	E5	MOV R0, ad	A8	MOVC A,	93
MOV A, R0	E8	MOV R0, #d	78	MOVC A, @+PC	83
MOV A, R1	E9	MOV R1, A	F9	MOVX A,	EO
MOV A, R2	EA	MOV R1, ad	A9	MOVX A, @R0	E2
MOV A, R3	EB	MOV R1, #d	79	MOVX A, @R1	E3
MOV A, R4	EC	MOV R2, A	FA	MOVX @DPTR,	F0
MOV A, R5	ED	MOV R2, ad	AA	MOVX @R0, A	F2
MOV A, R6	EE	MOV R2, #d	7A	MOVX @R1, A	F3
MOV A, R7	EF	MOV R3, A	FB	MUL AB	A4
MOV A, @R0	E6	MOV R3, ad	AB	NOP	00
MOV A, @R1	E7	MOV R3, #d	7B	ORL A, ad	45
MOV a, #d	74	MOV R4, A	FC	ORL A, R0	48
MOV ad, A	F5	MOV R4, ad	AC	ORL A, R1	49
MOV ad, R0	88	MOV R4, #d	7C	ORL A, R2	4A
MOV ad, R1	89	MOV R5, A	FD	ORL A, R3	4B
MOV ad, R2	8A	MOV R5, ad	AD	ORL A, R4	4C
MOV ad, R3	8B	MOV R5, #d	7D	ORL A, R5	4D
MOV ad, R4	8C	MOV R6, A	FE	ORL A, R6	4E

MOV ad , R5	8D	MOV R6, ad	AE	ORL A, R7	4F
MOV ad , R6	8E	MOV R6, #d	7E	ORL A, @R0	46
MOV ad , R7	8F	MOV R7 , A	FF	ORL A , @R0	47
ORL A, #d	44	RRC A	13	SUBB A , R7	9F
ORL ad , A	42	SETB bit	D2	SUBB A , @R0	96
ORL ad , #d	43	setb c	D3	SUBB A, @R1	97
ORL C , bit	72	SJMP rel	80	SWAP A	C4
ORL C, /bit	AO	SUBB A, ad	95	XCH A , ad	C5
POP ad	DO	SUBB A, R0	98	XCH A, R0	C8
PUSH ad	CO	SUBB A, R1	99	XCH A, R1	C9
RET	22	SUBB A , R2	9A	XCH A , R2	CA
RETI	32	SUBB A , R3	9B	XCH A , R3	CB
RL A	23	SUBB A, R4	9C	XCH A , R4	CC
RLC A	33	SUBB A , R5	9D	XCH A , R5	CD
RR A	03	SUBB A, R6	9E	XCH A , R6	CE
XCH A, R7	CF	XRL A, R1	69	XRL A , R7	6F
XCH A, @R0	06	XRL A,	6A	XRL A , @R0	66
XCH A, @R1	C7	XRL A, R3	6B	XRL A, @R1	67
XCHD A, @R0	D6	XRL A, R4	6C	XRL A, #d	64
XCHD A, @R1	D7	XRL A, R5	6D	XRL ad , A	62
XRL A, ad	65	XRL A , R6	6E	XRL ad, #d	63
XRL A, R0	68				