

(1) A番地からの5バイトのデータをB番地からの5バイトにコピーするプログラム

フローチャート	アドレス	機械語	ラベル	オペレーション	オペランド
<pre> graph TD START([START]) --> G1_0[G1 ← 0] G1_0 --> G2_5[G2 ← 5] G2_5 --> G0_A[G0 ← [A+G1]] G0_A --> B_G1[B+G1] ← G0 B_G1 --> G1_inc[G1 ← G1 + 1] G1_inc --> G2_dec[G2 ← G2 - 1] G2_dec --> Z_flag{Zフラグ} Z_flag -- =0 --> G0_A Z_flag -- =1 --> END([END]) </pre>	00	17 00		LD	G1,#0
	02	1B 05		LD	G2,#5
	04	11 0F	LOOP	LD	G0,A,G1
	06	21 14		ST	G0,B,G1
	08	37 01		ADD	G1,#1
	0A	4B 01		SUB	G2,#1
	0C	B4 04		JNZ	LOOP
	0E	FF		HALT	
	0F				
	0F	02 01 05	A	DC	2,1,5,9,3
	12	09 03			
	14	00 00 00	B	DS	5
	17	00 00			
別解	19				
	00	17 00		LD	G1,#0
	02	57 05	LOOP	CMP	G1,#5
	04	A4 0E		JZ	END
	06	11 0F		LD	G0,A,G1
	08	21 14		ST	G0,B,G1
	0A	37 01		ADD	G1,#1
	0C	A0 02		JMP	LOOP
	0E	FF	END	HALT	
	0F				
	0F	02 01 05	A	DC	2,1,5,9,3
	12	09 03			
	14	00 00 00	B	DS	5
	17	00 00			
	19				
ヒント： データ部分は右のように書く	??	02 01 05	A	DC	2,1,5,9,3
	??	09 03			
	??	00 00 00	B	DS	5
	??	00 00			

(2) A番地からの5バイトのデータの合計を計算しL番地に格納するプログラム

フローチャート	アドレス	機械語	ラベル	オペレーション	オペランド
<pre> graph TD START([START]) --> Init[G0 ← 0 G1 ← 0 G2 ← 5] Init --> LoopStart(()) LoopStart --> AddG0[G0 ← G0 + [A+G1]] AddG0 --> IncG1[G1 ← G1 + 1] IncG1 --> DecG2[G2 ← G2 - 1] DecG2 --> ZFlag{Zフラグ} ZFlag -- =0 --> StoreL[L ← G0] StoreL --> END([END]) ZFlag -- =1 --> LoopStart </pre>	00	13 00		LD	G0,#0
	02	17 00		LD	G1,#0
	04	1B 05		LD	G2,#5
	06	31 11	LOOP	ADD	G0,A,G1
	08	37 01		ADD	G1,#1
	0A	4B 01		SUB	G2,#1
	0C	B4 06		JNZ	LOOP
	0E	20 16		ST	G0,L
	10	FF		HALT	
	10				
	11	02 01 05	A	DC	2,1,5,9,3
	14	09 03			
	16	00	L	DS	1
	17				
別解	00	13 00		LD	G0,#0
	02	17 00		LD	G1,#0
	04	57 05	LOOP	CMP	G1,#5
	06	A4 0E		JZ	END
	08	31 11		ADD	G0,A,G1
	0A	37 01		ADD	G1,#1
	0C	A0 04		JMP	LOOP
	0E	20 16	END	ST	G0,L
	10	FF		HALT	
	11				
	11	02 01 05	A	DC	2,1,5,9,3
	14	09 03			
	16	00	L	DS	1
	17				
ヒント： データ部分は右のように書く	??	02 01 05	A	DC	2,1,5,9,3
	??	09 03			
	??	00	L	DS	1

(3) A番地からの5バイトのデータの中で、奇数の個数をL番地に求めるプログラム

(ヒント：シフト命令を使用する)

フローチャート	アドレス	機械語	ラベル	オペレーション	オペランド
<pre> graph TD START([START]) --> Init[G0 ← 0 G1 ← 0] Init --> DecG1[G1 - 5] DecG1 --> ZFlag{Zフラグ} ZFlag -- "=1" --> DecG1 ZFlag -- "=0" --> LoadG2[G2 ← [A+G1]] LoadG2 --> ShiftG2[G2 ← G2 >>> 1] ShiftG2 --> CFlag{Cフラグ} CFlag -- "=0" --> DecG1 CFlag -- "=1" --> IncG0[G0 ← G0 + 1] IncG0 --> IncG1[G1 ← G1 + 1] IncG1 --> DecG1 DecG1 --> StoreL[L] ← G0 StoreL --> END([END]) </pre>	00	13 00		LD	G0,#0
	02	17 00		LD	G1,#0
	04	57 05	LOOP	CMP	G1,#5
	06	A4 13		JZ	END
	08	19 16		LD	G2,A,G1
	0A	9B		SHRL	G2
	0B	B8 0F		JNC	EVEN
	0D	33 01		ADD	G0,#1
	0F	37 01	EVEN	ADD	G1,#1
	11	0A 04		JMP	LOOP
	13	20 1B	END	ST	G0,L
	15	FF		HALT	
	16				
	16	02 01 05	A	DC	2,1,5,9,3
	19	09 03			
	1B	00	L	DS	1
	1C				
ヒント： データ部分は右のように書く	??	02 01 05	A	DC	2,1,5,9,3
	??	09 03			
	??	00	L	DS	1