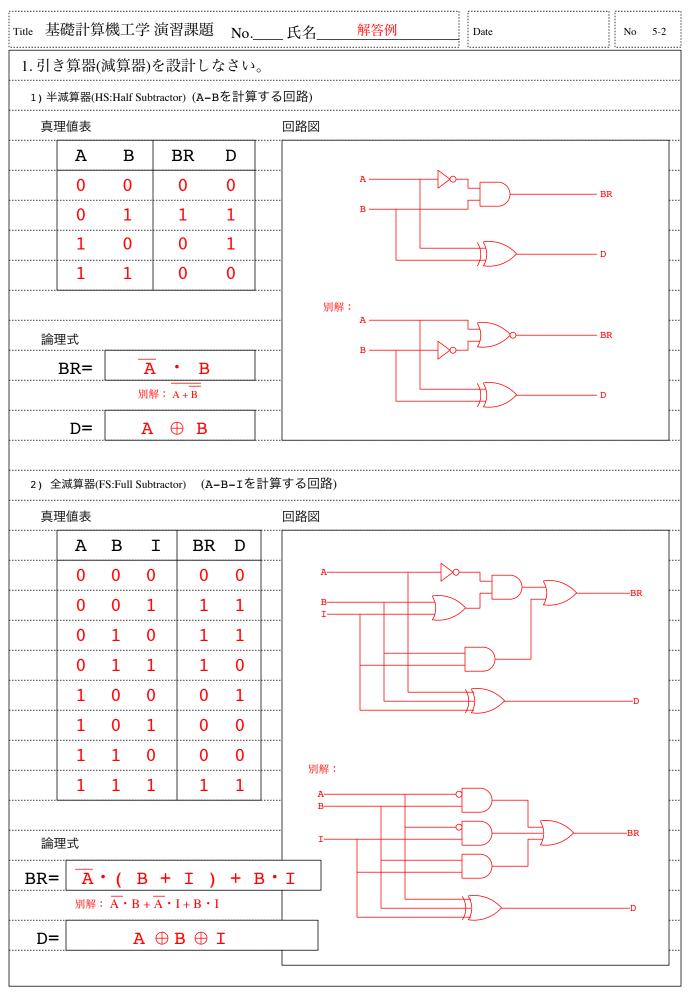
Title	基礎計算機工学 演習課題	No	_ 氏名	解答例	Date	No	1
(1) 言葉の確認(次の言葉の意味:	を再度確	認し、簡単に	:説明しなさい。)			
	ビット:情報の最小単位、「	2つのう	ちのどちらか	ゝ」を表現すること	ができる情報		
	ニブル: 4ビットのこと						
	バイト: 8ビットのこと						
(2) 空白に当てはまる数値/式を	書きなさ	ر ۲°				
	ノイマン型のコンピュータ	は情報を	ビットの組合	合せで表現します。	1 ビットで表現できる情	報は	
	2種類です。2ビットでは(4)	種類の情報を	表現できます。逆	に、16種類の情報を表	現す	
	るためには(4)ビ	`ット必要	長になります。	一般に、nビット	で表現できる情報の種類	損は次	
	の式で表すことができます。						
	表現できる情	報の種類	= (2 ⁿ)		
				······································			
(3) 次の表を暗記しなさい。	2	2進数	10進数			
		(0000	0			
		(0001	1			
		(0010	2			
		(0011	3			
		(0100	4			
		(0101	5			
		(0110	6			
		(0111	7			
			1000	8			
			1001	9			
			L010	10			
			1011	11			
]	L100	12			
]	1101	13			
			L110	14			
			1111	15			



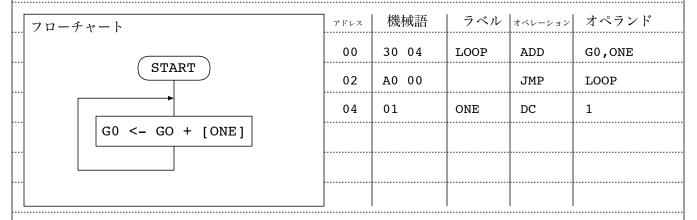
(1) 10 H番地と11 H番地の和を12 H番地に、差を13 H番地に格納するプログラム

L	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
		00	10 10		LD	G0,10H
	START	02	30 11		ADD	G0,11H
	G0 <- [10H]	04	20 12		ST	G0,12H
	GO <- GO + [11H]	06	10 10		LD	G0,10H
		08	40 11		SUB	G0,11H
	[12H] <- G0	0A	20 13		ST	G0,13H
	G0 <- [10H]	0C	FF		HALT	
	G0 <- G0 - [11H]	0D				
ļ	[13H] <- G0					
-	END					
-						
-						

(2) 10 H番地の値の 3 倍を 11 H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
	00	10 10		LD	G0,10H
START	02	30 10		ADD	G0,10H
GO <- [10H]	04	30 10		ADD	G0,10H
G0 <- G0 + [10H]	06	20 11		ST	G0,11H
	08	FF		HALT	
 G0 <- G0 + [10H]	09				
 [11H] <- G0					
 (END					
	}				

次のプログラムを実行して、ランプの変化を観察しなさい。



(1) PCを表示したままステップモードで実行する。PCはどのように変化しましたか?

00 -> 02 -> 00 -> 02 の順に変化した。

(ADD命令と、JMP命令を交互に実行している様子が分かった。)

正解は教えた範囲では 分からないので、 想像で答える。

- (2) PCを表示したまま通常実行する。PCはどのように変化していると考えられますか?(停止にはSTOPボタンを使用)D2, D1, D0 が点灯した。D0 が点灯したことから、命令の実行途中にPCは1番地づつ増加していると想像できる。D2が点灯したことからJMP命令実行時に、一旦、PCが04になっていることが想像できる。以上を総合するとPCは、00 -> 01 -> 02 -> 03 -> 04 -> 00 の順に変化していると考えられる。
- (3) G0を表示したままステップモードで実行する。G0はどのように変化しましたか?

00 -> 01 -> 01 -> 02 -> 02 -> 03 -> 03 の順に変化した。

(1命令実行する毎に止まっているのでADD命令を実行するのは2回に1回だけ)

(4) PCを表示したまま(ADD命令を実行する毎に止まるように)ブレークモードで実行する。

PCはどのようになりましたか?

02 のまま変化しないように見える。

(ADDを実行後に止まるので、PCが次のJMP命令を実行する直前の02になった状態で止まる。)

(5) G0を表示したまま(ADD命令を実行する毎に止まるように)ブレークモードで実行する。

GOはどのようになりましたか?

00 -> 01 -> 02 -> 03 -> 04 -> 05 -> 06 の順に変化した。

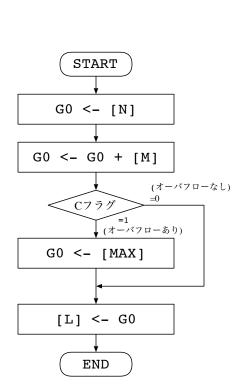
(ADD命令を実行する度に止まるので)

(6) G0を表示したまま(ADD命令を実行する毎に止まるように)ブレークモードで実行する。

各フラグが「1になる条件」は何ですか?

	1 になる条件	
Cフラグ	FFHに1加えた時と、00Hから1引いた時(符号無し2進数と考えたときのオーバーフロー)	
Zフラグ	FFHに1加えた時と、01Hから1引いた時(計算結果が0になったとき)	
sフラグ	計算結果のD7が1のとき(符号付き2進数と考えたときのマイナス)	

N番地とM番地の値の合計を計算し、L番地に結果を格納するプログラム(但し、オーバーフローが 発生した場合は255(FFH)をL番地に格納する)を作りなさい。



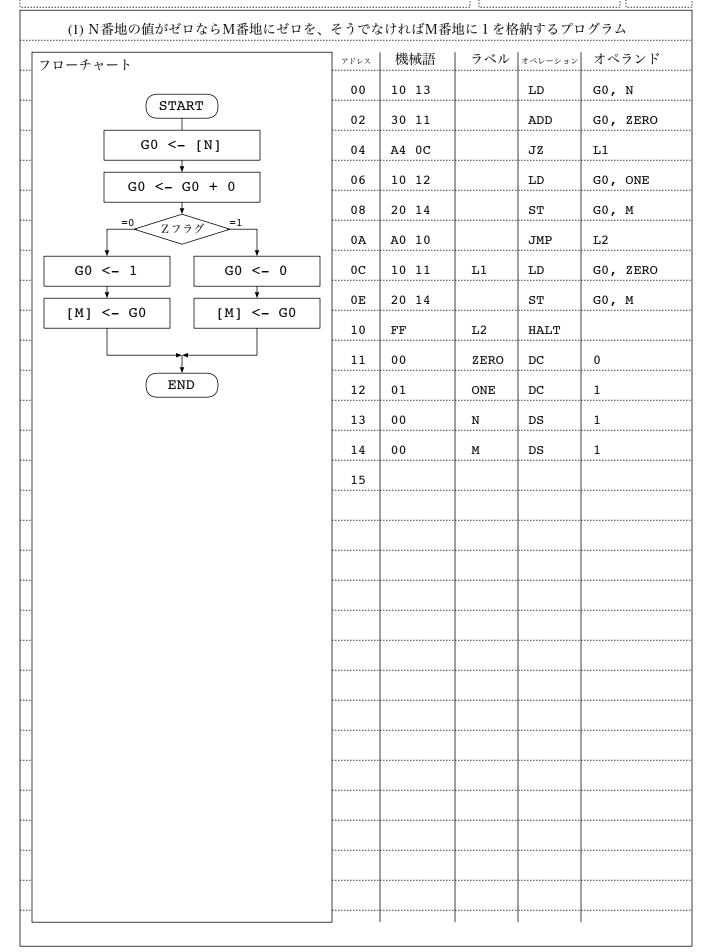
フローチャート

N,M:データが置いてある場所 L:結果を格納する場所

MAX:255のデータが置いてある場所

	: 格納するフ Eりなさい。	ロクフム(1	且し、オー	バーフローが
アドレス	機械語	ラベル	オペレーション	オペランド
00	10 OD		LD	GO,N
02	30 OE		ADD	G0,M
04	A8 08		JC	L1
06	A0 0A		JMP	L2
08	10 OF	L1	LD	G0,MAX
0A	20 10	L2	ST	G0,L
0C	FF		HALT	
0D				
0D	AA	N	DC	176
0E	64	M	DC	100
0F	FF	MAX	DC	255
10	00	L	DS	1
J	I	I		

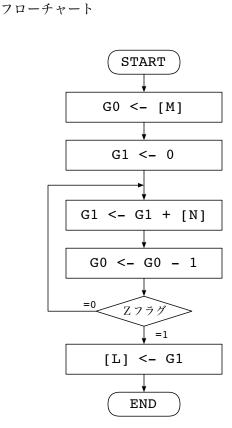
		Date		No	8-1
--	--	------	--	----	-----



(2) N番地とM番地の値の大きい方を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
START	00	10 11		LD	G0, N
START	02	40 12		SUB	G0, M
G0 <- [N]	04	AC 0C		JM	L1
GO <- GO - [M]	06	10 11		LD	G0, N
	08	20 13		ST	G0, L
=0 Sフラグ =1	0A	A0 10		JMP	L2
G0 <- [N] G0 <- [M]	0C	10 12	L1	LD	G0, M
[L] <- G0 [L] <- G0	0E	20 13		ST	G0, L
	10	FF	L2	HALT	
	11	00	N	DS	1
END	12	00	М	DS	1
	13	00	L	DS	1
	14				

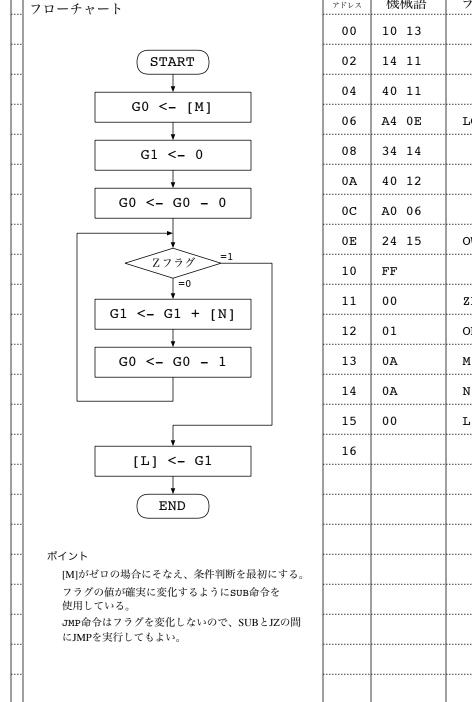
(1) N番地のデータと、M番地のデータのかけ算を計算しL番地に格納するプログラムを作りなさい。



G0かける数[M]回数繰り返すためのカウンタ G1:かけられる数[N]の合計(積)を求めるレジスタ

	у, од ш, ш,	二倍利19る	7 4 7 7 4	を作りなさい。
アドレス	機械語	ラベル	オペレーション	オペランド
00	10 12		LD	G0,M
02	14 OF		LD	G1,ZERO
04	34 11	LOOP	ADD	G1,N
06	40 10		SUB	G0,ONE
08	A4 0C		JZ	OWARI
0A	A0 04		JMP	LOOP
0C	24 13	OWARI	ST	G1 , L
0E	FF		HALT	
0F	00	ZERO	DC	0
10	01	ONE	DC	1
11	0A	N	DC	10
12	0A	М	DC	10
13	00	L	DS	1
14				
ļ	<u> </u>	p		

(2) 前のプログラムをN番地、または、M番地のデータがゼロの場合も正常に動作するように改良しなさい。



ファレス 機械語 ラベル オペレーション オペランド 00 10 13	· · · · · · · · · · · · · · · · · · ·	0-6日7799日	り正市に到	11 1 2 2 1	に以及しなるV ₃ 。
02 14 11 LD G1,ZERO 04 40 11 SUB G0,ZERO 06 A4 0E LOOP JZ OWARI 08 34 14 ADD G1,N 0A 40 12 SUB G0,ONE 0C A0 06 JMP LOOP 0E 24 15 OWARI ST G1,L 10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	アドレス	機械語	ラベル	オペレーション	オペランド
04 40 11 SUB G0,ZERO 06 A4 0E LOOP JZ OWARI 08 34 14 ADD G1,N 0A 40 12 SUB G0,ONE 0C A0 06 JMP LOOP 0E 24 15 OWARI ST G1,L 10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	0.0	10 13		LD	G0,M
06 A4 0E LOOP JZ OWARI 08 34 14 ADD G1,N 0A 40 12 SUB G0,ONE 0C A0 06 JMP LOOP 0E 24 15 OWARI ST G1,L 10 FF HALT HALT 11 00 ZERO DC 0 12 01 ONE DC 1 1 13 0A M DC 10 14 0A N DC 10 10 15 00 L DS 1	02	14 11		LD	G1,ZERO
08 34 14 ADD G1,N 0A 40 12 SUB G0,ONE 0C A0 06 JMP LOOP 0E 24 15 OWARI ST G1,L 10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	04	40 11		SUB	G0,ZERO
0A 40 12 SUB G0,ONE 0C A0 06 JMP LOOP 0E 24 15 OWARI ST G1,L 10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	06	A4 0E	LOOP	JZ	OWARI
OC A0 06 JMP LOOP 0E 24 15 OWARI ST G1,L 10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	08	34 14		ADD	G1,N
OE 24 15 OWARI ST G1,L 10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	0A	40 12		SUB	G0,ONE
10 FF HALT 11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	0C	A0 06		JMP	LOOP
11 00 ZERO DC 0 12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	0E	24 15	OWARI	ST	G1,L
12 01 ONE DC 1 13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	10	FF		HALT	
13 0A M DC 10 14 0A N DC 10 15 00 L DS 1	11	00	ZERO	DC	0
14 0A N DC 10 15 00 L DS 1	12	01	ONE	DC	1
15 00 L DS 1	13	0A	M	DC	10
	14	0A	N	DC	10
	15	00	L	DS	1
	16				
h					
]	·	·····	·····	

(1) M番地のデータをN番地のデータで割り、商をK番地、余りをL番地に格納するプログラムを作りなさい。

フローチャート	アドレス	機械語			オペランド
	0.0	10 15		LD	G0,M
START	02	14 13		LD	G1,ZERO
	04	50 16	LOOP	СМР	G0,N
G0 <- [M]	06	AC 0E		JM	OWARI
G1 <- 0	08	40 16		SUB	G0,N
	0A	34 14		ADD	G1,ONE
G0 - [N]	0C	A0 04		JMP	LOOP
… sフラグ =1(負)	0E	24 17	OWARI	ST	G1,K
=0(正)	10	20 18		ST	G0,L
G0 <- G0 - [N]	12	FF	-	HALT	
	13	00	ZERO	DC	0
G1 <- G1 + 1	14	01	ONE	DC	1
	15	64	М	DC	100
[K] <- G1	16	0A	N	DC	10
	17	00	K	DS	1
[L] <- G0	18	00	L	DS	1
END END	19				
 G0:割られる数 G1:引いた回数					
			-		
···					
	ļ				
	ļ				

(1) G0 の値を 1 ビット右回転するプログラムを作	りなさい	0			
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
START	00	93	L0	SHRL	G0
G0 <- G0 >> 1	01	A8 05		JC	L1
= 0	03	A0 07		JMP	L2
$\begin{array}{c} \cdots \\ \cdots \\ \cdots \\ \cdots \\ \end{array} = 1$	05	30 OA	L1	ADD	GO, BIT
G0 <- G0 + 80H	07	FF	L2	HALT	
	08	A0 00		JMP	L0
END	0A	80	BIT	DC	80Н
(2) M番地のデータの 7 倍をL番地に格納する	アドレス	機械語	ラベル	オペレーション	オペランド
プログラムを作りなさい。	00	10 OD		LD	G0, M
フローチャート	02	91		SHLL	G0
解答例 1	03	20 OF		ST	G0, TMP
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	05	91		SHLL	G0
····	06	30 OF		ADD	G0, TMP
 	08	30 OD		ADD	G0, M
····	0A	20 0E		ST	G0, L
····	0C	FF		HALT	
····	0D	05	M	DC	5
····	0E	00	L	DS	1
	0F	00	TMP	DS	1
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
	00	10 0A		LD	G0, M
$L = M \times 7$	02	91		SHLL	G0
= M × 8 - M	03	91		SHLL	G0
····	04	91		SHLL	G0
 	05	40 0A		SUB	G0, M
 	07	20 OB		ST	G0, L
 	09	FF		HALT	
 	0A	05	M	DC	5
	ļ		L	DS	1

フローチャート				るプログラ		
ノローナヤート		アドレス	機械語	ラベル	オペレーション	オペランド
		00	10 07		LD	G0, A
		02	93		SHRL	G0
		03	93		SHRL	G0
		04	20 08		ST	G0, B
		06	FF		HALT	
		07	23	A	DC	35
		08	00	В	DS	1
		09				
]				
(4) A番地の内容を符号無	既し2進数と見なし、1.5	AをB番	 地に求めるフ	プログラム	 を作りなさ	γ ₂ °
フローチャート		アドレス	機械語		オペレーション	オペランド
		F	r			オペラント
		00	10 08		LD	G0, A
B = 1.5	× ъ	00	10 08 93			
B = 1.5 = A ×	× A 1/2 + A				LD	G0, A
		02	93		LD SHRL	G0, A G0
		02	93		LD SHRL ADD	G0, A G0 G0, A
		02	93 30 08 20 09	A	LD SHRL ADD ST	G0, A G0 G0, A
		02 03 05	93 30 08 20 09 FF	A B	LD SHRL ADD ST HALT	G0, A G0 G0, A G0, B
		02 03 05 07	93 30 08 20 09 FF 1E		LD SHRL ADD ST HALT DC	G0, A G0 G0, A G0, B
		02 03 05 07	93 30 08 20 09 FF 1E		LD SHRL ADD ST HALT DC	G0, A G0 G0, A G0, B
		02 03 05 07	93 30 08 20 09 FF 1E		LD SHRL ADD ST HALT DC	G0, A G0 G0, A G0, B
		02 03 05 07	93 30 08 20 09 FF 1E		LD SHRL ADD ST HALT DC	G0, A G0 G0, A G0, B
		02 03 05 07	93 30 08 20 09 FF 1E		LD SHRL ADD ST HALT DC	G0, A G0 G0, A G0, B
		02 03 05 07	93 30 08 20 09 FF 1E		LD SHRL ADD ST HALT DC	G0, A G0 G0, A G0, B

(1) A番地からの5バイトのデータの和をB番地に求めるプログラムを作りなさい。						
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	
··· 解答例 1		解答例 1				
所合例 I START	00	13 00		LD	G0, #0	
	02	17 00		LD	G1, #0	
G0 <- 0 G1 <- 0	04	57 05	LOOP	СМР	G1, #5	
	06	A4 0E		JZ	STOP	
G1 - 5	08	31 11		ADD	G0, A, G1	
Z7 7 / = 1	0A	37 01		ADD	G1, #1	
= 0	0C	A0 04		JMP	LOOP	
G0 <- G0 + [A + G1]	0E	20 16	STOP	ST	G0, B	
G1 <- G1 + 1	10	FF		HALT		
	11	02 01 05	A	DC	2,1,5,9,3	
	14	09 03				
[B] <- G0	16	00	В	DS	1	
END	17					
		解答例 2				
··· 解答例 2 START	00	13 00		LD	G0, #0	
	02	17 00		LD	G1, #0	
G0 <- 0 G1 <- 0 G2 <- 5	04	1B 05		LD	G2, #5	
	06	31 13	LOOP	ADD	G0, A, G1	
G0 <- G0 + [A + G1]	08	37 01		ADD	G1, #1	
G1 <- G1 + 1	0A	4B 01		SUB	G2, #1	
	0C	A4 10		JZ	STOP	
G2 <- G2 - 1	0E	A0 06		JMP	LOOP	
=0 Z7 ¬¬ /¬	10	20 18	STOP	ST	G0, B	
= 1	12	FF		HALT		
[B] <- G0	13	02 01 05	A	DC	2,1,5,9,3	
END	16	09 03				
	18	00	В	DS	1	
	19					

(2) A番地からの 5 バイトをB番地からの 5 バイト(にコピー	するプログラ	ムを作り	なさい。	
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
解答例 1		解答例 1			
START	00	17 00		LD	G1 , #0
G1 <- 0	02	57 05	LOOP	СМР	G1 , #5
——————————————————————————————————————	04	A4 0E		JZ	STOP
G1 - 5	06	11 OF		LD	G0,A,G1
	08	21 14		ST	G0,B,G1
	0A	37 01		ADD	G1,#1
G0 <- [A + G1]	0C	A0 02		JMP	LOOP
	0E	FF	STOP	HALT	
[B + G1] <- G0	0F	02 01 05	A	DC	2,1,5,9,3
G1 <- G1 + 1	12	09 03			
	14	00 00 00	В	DS	5
END	17	00 00			
	19				
解答例 2		解答例 2			
START	00	17 00		LD	G1 , #0
G1 <- 0	02	1B 05		LD	G2 , #5
G2 <- 5	04	11 11	LOOP	LD	G0,A,G1
G0 <- [A + G1]	06	21 16		ST	G0,B,G1
[B + G1] <- G0	08	37 01		ADD	G1 , #1
	0A	4B 01		SUB	G2 , #1
G1 <- G1 + 1	0C	A4 10		JZ	STOP
G2 <- G2 - 1	0E	A0 04		JMP	LOOP
= 0	10	FF	STOP	HALT	
Z75/	11	02 01 05	A	DC	2,1,5,9,3
= 1 END	14	09 03			
	16	00 00 00	В	DS	5
	19	00 00			
	1B				

ファレス 機械語 ラベル オペレーションオペランド 00 17 00	(1) A番地からの5バイトのデータのビットを反転するプログラムを作りなさい。						
START 02 57 05 LOOP CMP G1, #5 04 A4 10 JZ STOP 06 11 11 LD G0, A, G1 08 83 FF XOR G0, #0FFH 0A 21 11 ST G0, A, G1 0C 37 01 ADD G1, #1 0E A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH	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
04 A4 10 JZ STOP G1 <- 0 06 11 11 LD G0, A, G1 08 83 FF XOR G0, #0FFH 0A 21 11 ST G0, A, G1 0C 37 01 ADD G1, #1 0E A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH			00	17 00		LD	G1, #0
G1 <- 0 06 11 11 LD G0, A, G1 08 83 FF XOR G0, #0FFH 0A 21 11 ST G0, A, G1 0C 37 01 ADD G1, #1 0E A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH		START	02	57 05	LOOP	CMP	G1, #5
O6			04	A4 10		JZ	STOP
OA 21 11 ST G0, A, G1 OC 37 01 ADD G1, #1 OE A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH		G1 <- 0	06	11 11		LD	G0, A, G1
OC 37 01 ADD G1, #1 OE A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH		∀ G1 - 5	08	83 FF		XOR	G0, #0FFH
OC 37 01 ADD G1, #1 OE A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH			0A	21 11		ST	G0, A, G1
0E A0 02 JMP LOOP 10 FF STOP HALT 11 00 88 A DC 00H, 88H 13 CC EE DC 0CCH, 0EEH 15 FF DC 0FFH		2777	0C	37 01		ADD	G1, #1
10 FF STOP HALT 11		▼	0E	A0 02		JMP	LOOP
11			10	FF	STOP	HALT	
13 CC EE DC OCCH, OEEH 15 FF DC OFFH		G0 <- G0 ^ FFH	11				
G1 <- G1 + 1 15 FF DC 0FFH		[A + G1] <- G0	11	00 88	A	DC	00н, 88н
15 FF DC OFFH			13	CC EE		DC	OCCH, OEEH
END		G1 <- G1 + 1	15	FF		DC	0FFH
END							
		END					

(2) A番地とB番地からの5バイトの論理和をC番地からの5バイトに求めるプログラムを作りなさい。							
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		
	00	17 00		LD	G1, #0		
START	02	57 05	LOOP	CMP	G1, #5		
G1 <- 0	04	A4 10		JZ	STOP		
	06	11 11		LD	G0, A, G1		
G1 - 5	08	71 16		OR	G0, B, G1		
= 1	0A	21 1B		ST	G0, C, G1		
Z7 ¬ ¬ ¬ = 1	0C	37 01		ADD	G1, #1		
G0 <- [A + G1]	0E	A0 02		JMP	LOOP		
	10	FF	STOP	HALT			
G0 <- G0 [B + G1]	11						
[C + G1] <- G0	11	12 34	A	DC	12н, 34н		
G1 <- G1 + 1	13	56 78		DC	56н, 78н		
	15	9A		DC	9АН		
	16						
END	16	0F 0F	В	DC	OFH, OFH		
	18	0F 0F		DC	0FH, 0FH		
	1A	0F		DC	0FH		
	1B						
	1B	00 00	С	DS	5		
	1D	00 00					
	1F	00					
····							
····							
 							
 					······		
 					······		

(1) A番地からの5バイトのデータのなかで、奇数の個数をB番地に求めるプログラムを作りなさい。							
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		
	00	13 00		LD	G0,#0		
START	02	17 00		LD	G1,#0		
	04	57 05	LOOP	CMP	G1,#5		
G0 <- 0	06	A4 14		JZ	FIN		
G1 <- 0	08	19 17		LD	G2,A,G1		
	0A	6B 01		AND	G2,#01H		
G1 - 5	0C	A4 10		JZ	EVEN		
Z7 7 7 = 1	0E	33 01		ADD	G0,#1		
	10	37 01	EVEN	ADD	G1,#1		
G2 <- [A + G1]	12	A0 04		JMP	LOOP		
G2 <- G2 & 01H	14	20 1C	FIN	ST	G0,B		
1./11 440	16	FF		HALT			
	17						
go <- g0 + 1	17	01 02 03	A	DC	1,2,3,4,5		
	1A	04 05					
G1 <- G1 + 1	1C	00	В	DS	1		
[B] <- GO							
END							
<u> </u>							
<u> </u>							
				·····			

(2) A番地からの 5 バイトのデータの最大値をB番地に求めるプログラムを作りなさい。							
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	
		00	10 17		LD	G0,A	
	START	02	17 01		LD	G1,#1	
		04	57 05	L0	CMP	G1,#5	
	G0 <- [A]	06	A4 14		JZ	L3	
	G1 <- 1	08	51 17		CMP	G0,A,G1	
		0A	AC 0E		JM	L1	
	G1 - 5	0C	A0 10		JMP	L2	
	Z777 =1	0E	11 17	L1	LD	G0,A,G1	
	= 0	10	37 01	L2	ADD	G1,#1	
	G0 - [A + G1]	12	A0 04		JMP	L0	
	S777 =0	14	20 1C	L3	ST	G0,B	
	=1	16	FF		HALT		
	G0 <- [A + G1]	17					
	ļ	17	01 03 02	A	DC	1,3,2,5,4	
	G1 <- G1 + 1	1A	05 04				
		1C	00	В	DS	1	
	[B] <- G0						
	END						
		}		l	l	·····	
		-					

フローチャート

	イトデータの合計をB番地に求めるプロ	っ がニ トナ ノン ヴカット	シェー ドシコ ル かんひょ
- (3) A 否則(/) ら ハイ	「トナータの合計をR香地に来める)」	コクフムタインエクスト	ナートなしに作りなるい

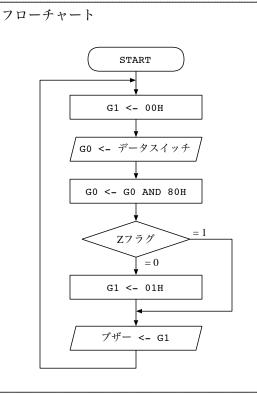
START
G0 <- 0 G1 <- A G2 <- 5
次のADD命令のアドレス部 に G1 の値を書き込む
G0 <- G0 + [前の命令が書き込んだ値]
G1 <- G1 + 1
<u> </u>
G2 <- G2 - 1
=0 Z7 ¬ ¬ ¬ ¬
= 1
[B] <- G0
END

アドレス	機械語	ラベル	オペレーショ	ンオペランド
00	13 00		LD	G0,#0
02	17 15		LD	G1,#A
04	1B 05		LD	G2 , #5
06	24 09	LOOP	ST	G1,ADDR
08	30		DC	30Н
09	00	ADDR	DS	1
0A	37 01		ADD	G1,#1
0C	4B 01		SUB	G2,#1
0E	A4 12		JZ	STOP
10	A0 06		JMP	LOOP
12	20 1A	STOP	ST	G0,B
14	FF		HALT	
15	02 01 05	A	DC	2,1,5,9,3
19	09 03			
1A	00	В	DS	1
1B				
別の書き	方(途中まで)			
00	13 00		LD	G0,#0
02	17 15		LD	G1,#A
04	1B 05		LD	G2 , #5
06	24 09	LOOP	ST	G1,X+1
08	30 00	Х	ADD	G0,0
0A	37 01		ADD	G1 , #1
0C	4B 01		SUB	G2,#1
0E	A4 12		JZ	STOP
•••				
]				

の、4至時よさのにぶりした	ゴ カフノッイにない	11マナ 上は公田はつ	ノナプログニノた佐りももい
(2) A番地からの5バイトを、	アークスイツアにセツ	「トしくめつに個で埋めつ」	くりノロクフムを作りなさい。

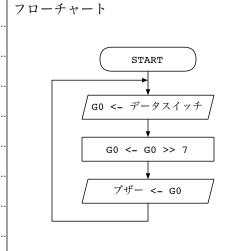
(2) A番地	からの5バイトを、データスイッチにも	2ットして	てあった値で埋	里めつくす	プログラム	ムを作りなさい。
フローチ	ニャート	アドレス	機械語	ラベル	オペレーション	オペランド
		00	C0 00		IN	G0,0
	START	02	17 00		LD	G1,#0
	↓ G0 <- データスイッチ	04	1B 05		LD	G2 , #5
		06	21 11	L1	ST	G0,A,G1
	G1 <- 0	08	37 01		ADD	G1 , #1
	G2 <- 5	0A	4B 01		SUB	G2 , #1
		0C	A4 10		JZ	L2
	[A+G1] <- G0	0E	A0 06		JMP	L1
	G1 <- G1 + 1	10	FF	L2	HALT	
		01	00 00 00	A	DS	5
	G2 <- G2 - 1	14	00 00			
	=0 Z7 ¬ ¬ ¬					
	=1					
	END					
					······	
					······	
····			-			
····			-			
l						
ļ <u> </u>						
		-	1	1	1	1

...(3) データスイッチのビット7(D7)をオンにしている間、ブザーを鳴らすプログラムを作りなさい。............. (但し、論理演算命令を用いること。)



		l	<u> </u>		[
	アドレス	機械語	ラベル	オペレーション	オペランド
	00	17 00	START	LD	G1,#00H
	02	C0 00		IN	G0,00H
	04	63 80		AND	G0,#80H
	06	A4 0A		JZ	OUTP
	08	17 01		LD	G1,#01H
	0A	C7 00	OUTP	OUT	G1,00H
	0C	A0 00		JMP	START
	0E				
_		t	h	r	

...(4) データスイッチのビット7(D7)をオンにしている間、ブザーを鳴らすプログラムを作りなさい。............ (但し、シフト命令を用いること。)



アドレス	機械語	ラベル	オペレーショ	ンオペランド
00	C0 00	START	IN	G0,00H
02	93		SHRL	G0
03	93		SHRL	G0
04	93		SHRL	G0
05	93		SHRL	G0
06	93		SHRL	G0
07	93		SHRL	G0
08	93		SHRL	G0
09	C3 00		OUT	G0,00H
0В	A0 00		JMP	START
0 D				

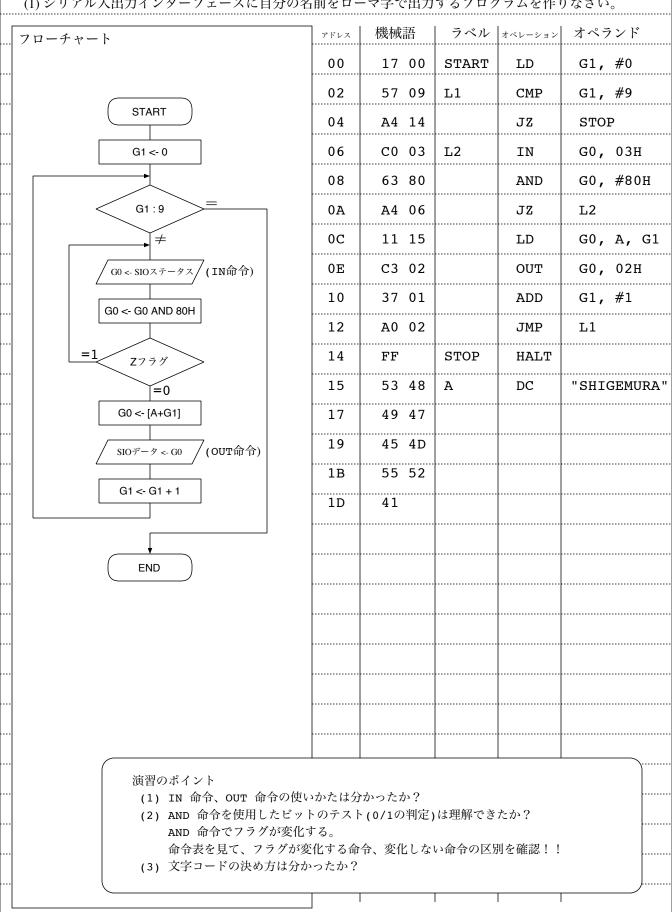
...(5) データスイッチのビット 0 (D0) が ON している間ブザーを鳴らすプログラムを作りなさい。 (但し、データスイッチの D7 が ON になったら、ブザーを止めて終了するようにすること)

START
G0 <- データスイッチ
ブザー <- G0
↓
G0 <- G0 AND 80H
=1 Zフラグ
= 0
G0 <- 00H
↓
ブザー <- G0
—
END

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
00	C0 00	START	IN	G0,00H
02	C3 00		OUT	G0,00H
04	63 80		AND	G0,#80H
06	A4 00		JZ	START
08	13 00		LD	G0,#00H
0A	C3 00		OUT	G0,00H
0C	FF		HALT	
	l			
	<u> </u>			
	<u> </u>			

(1) シリアル入出力インターフェースに自分の名前をローマ字で出力するプログラムを作りなさい。



(3) アルファベット小文字を大文字に変換し表示するプログラム(第2版)を作りなさい。

 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
		00	C0 03	L1	IN	G0,03H
	START	02	63 40		AND	G0,#40H
		04	A4 00		JZ	L1
<u> </u>	G0 <- SIOステータス	06	C4 02		IN	G1,02H
	G0 <- G0 and 40H	08				
	1	08	57 61		СМР	G1,#'a'
	=1 <u>Z75</u> /j	0A	AC 14		JM	L2
	=0	0C	57 7B		CMP	G1,#'z'+1
	<u> </u>	0E	AC 12		JM	L3
	G1 - 'a'	10	A0 14		JMP	L2
		12	47 20	L3	SUB	G1,#20H
	ウ文字 8フラグ =1	14				
	G1 - ('z'+1)	14	C0 03	L2	IN	G0,03Н
	G1 - (Z 11)	16	63 80		AND	G0,#80H
	S7¬¬// =0	18	A4 14		JZ	L2
	=1	1A	C7 02		OUT	G1,02H
	G1 <- G1 - 20H	1C	A0 00		JMP	L1
···· <u> </u>	/ GO <- SIOステータス/					
	G0 <- G0 and 80H					
	G0 <= G0 and 0011					
	=1 Z77/j					
	=0					
	<u> </u>					
·			1			1
ļ						
		J	l 	I	 	l