

SOFTSPOT SPECIAL

Sixteen pages of readers own program submissions for your enjoyment.

The next sixteen pages are packed with useful programs, games and ideas, for a variety of systems. All have been sent in by readers for our Softspot feature, and because of this they are untested by us.

If you think your software is of good enough quality to be included, send it to the following address :

CT Softspot,
Computing Today,
145 Charing Cross Road,
London WC2H 0EE.

and we'll take a look. We pay for all accepted programs, if you want yours returned please enclose an SAE. We do not demand line printer output, fine if you have one, all we ask is that the manuscript is legible and includes diagrams and explanations where necessary. A full (written) listing is mandatory.

So, for the next few pages, it's all yours!

J.F.Kendall.

TI59 ROUTINES

Anyone who is in the position of programming a microprocessor in machine code will have surely found calculating in hexadecimal rather tedious, to put it mildly, especially in the case of negative displacements for relative jumps where the two's complement has to be found. Hence this program was written to make hex. working much easier.

It incorporates the following features :-

- (1) Converts hex. to decimal and vice versa.
- (2) Finds two's complement of a hex. number.
- (3) Input any two hex. addresses and it will output the signed hex. displacement index for a relative jump from the first to the second.
- (4) Input an address followed by the displacement index and it will output the address to which a relative jump will be made.
- (5) Performs addition, subtraction, multiplication and division on hex. numbers.

Using The Program

The program was written for a TI59 programmable calculator, but it will fit into a TI58 if the memory is repartitioned. The number of data registers used will depend upon the size of the hex. number processed, but for the average micro. using four digit addresses up to ten registers are used. Access to a print/security cradle is not required and, as it stands, the program contains no print commands.

To use the program it must first be initialized by the key sequence RST, CMS, R/S. The display will now show 0 and the TI59 is ready to receive its first number. The entry and readout of decimal numbers follows normal calculator practice. Hexadecimal numbers are a little different; firstly the six numerals A to F are represented by their decimal equivalents ie A=10 F=15; secondly multidigit numbers are entered one digit at a time, starting at the most significant, separated by R/S, thus C50 is entered by the key sequence 12, R/S, 5, R/S, 0, R/S. The output of a hex. number takes place in a similar fashion. After a hex. calculation the most significant figure is in the display and operation of R/S brings each successive digit into the display unit a "flashing 1" indicates that all the information has been taken out. After the display has been cleared the calculator is ready for the next calculation.

When using the arithmetic routine it is necessary to enter the following codes for the arithmetic functions :-

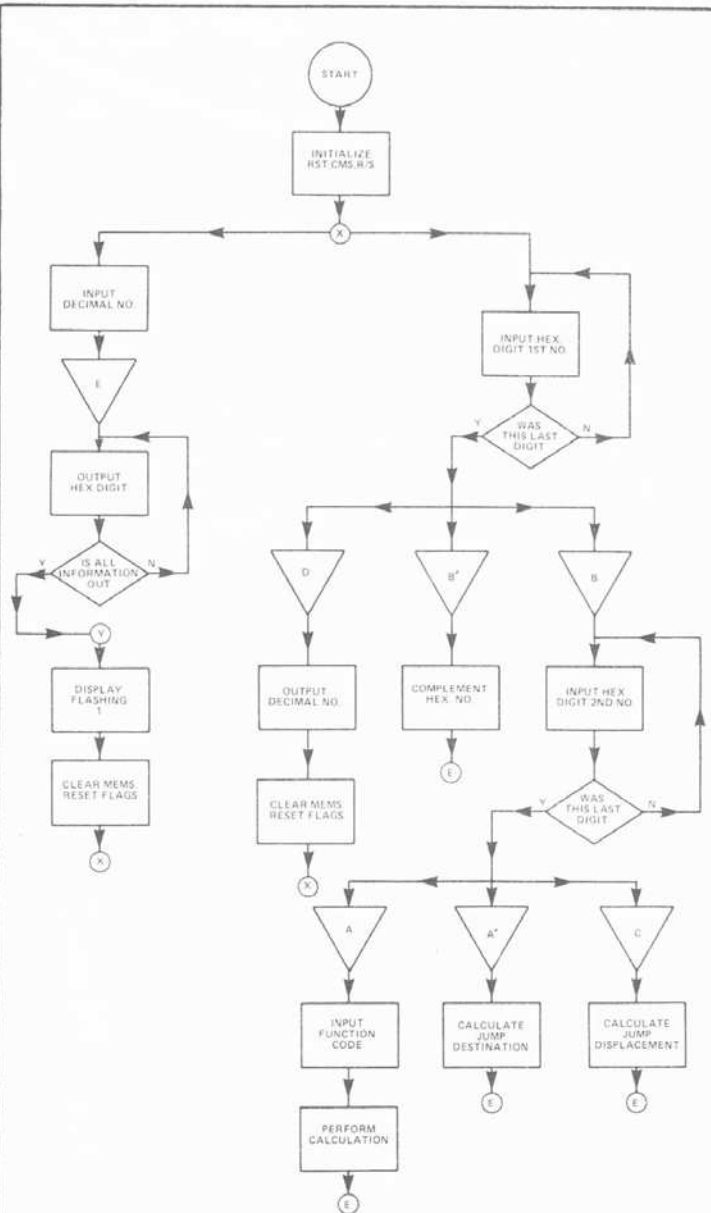
addition = 0
subtraction = 1
multiplication = 2
division = 3

All other functions are obtained by the use of the user Labels and their use should be clear from the accompanying flow chart.

Sample Runs

	Keys Pressed	Display
	Program entered from magnetic cards.	
	RST, CMS, R/S	0
1)	15 R/S } (Input F50)	15
	5 R/S }	5
	0 R/S }	0
	B (Signifies start of 2nd digit)	0
	15 R/S } (Input F4)	15
	4 R/S }	4
	A (Calculate jump destination)	0
	R/S	15
	R/S	4
	R/S	6
	R/S	1 (Flashing)
	Thus a relative jump command with displacement of F4 (eg Z80 op.code 18 F4) located at 0F50 would cause a jump to location 0F46.	
2)	CLR	0
	2 R/S } (Input 2B)	2
	11 R/S }	11
	D (Convert hex. to decimal)	43
3)	CLR	0
	13 R/S } (Input D0B)	13
	0 R/S }	0
	11 R/S }	11
	B (Signifies start of 2nd digit)	11
	3 R/S }	3
	10 R/S } (Input 3A)	10
	A (Select arithmetic routine)	10
	0 R/S (Select addition)	0
	R/S	13
	R/S	4
	R/S	5
	R/S	1 (Flashing)
	CLR	0

Thus D0B + 3A = D45



028	00	00	081	04	04
029	00	00	082	86	STF
030	76	LBL	083	01	01
031	13	C	084	76	LBL
032	01	1	085	43	RCL
033	94	+/-	086	07	7
034	32	X!T	087	42	STO
035	43	RCL	088	06	06
036	02	02	089	01	1
037	75	-	090	42	STO
038	43	RCL	091	05	05
039	01	01	092	25	CLR
040	95	=	093	32	X!T
041	42	STO	094	76	LBL
042	03	03	095	42	STO
043	69	DP	096	43	RCL
044	10	E*	097	04	4
045	67	EQ	098	55	÷
046	75	-	099	01	1
047	01	1	100	06	6
048	03	3	101	95	=
049	00	0	102	42	STO
050	32	X!T	103	03	03
051	43	RCL	104	59	INT
052	03	03	105	42	STO
053	77	GE	106	04	04
054	24	CE	107	43	RCL
055	75	-	108	03	03
056	02	2	109	22	INV
057	95	=	110	59	INT
058	76	LBL	111	65	×
059	15	E	112	01	1
060	42	STO	113	06	6
061	04	04	114	95	=
062	61	GTO	115	72	ST*
063	43	RCL	116	06	06
064	76	LBL	117	69	DP
065	75	-	118	26	26
066	01	1	119	69	DP
067	02	2	120	25	25
068	06	6	121	43	RCL
069	94	+/-	122	04	04
070	32	X!T	123	22	INV
071	43	RCL	124	67	EQ
072	03	03	125	42	STO
073	22	INV	126	22	INV
074	77	GE	127	87	IFF
075	24	CE	128	01	01
076	75	-	129	60	DEG
077	02	2	130	43	RCL
078	95	=	131	05	05
079	50	I×I	132	42	STO
080	42	STO	133	03	03

PROGRAM LISTING

000	91	R/S	014	61	GTO
001	42	STO	015	00	00
002	00	00	016	00	00
003	87	IFF	017	76	LBL
004	00	00	018	65	×
005	65	×	019	01	1
006	01	1	020	06	6
007	06	6	021	49	PRD
008	49	PRD	022	02	02
009	01	01	023	43	RCL
010	43	RCL	024	00	00
011	00	00	025	44	SUM
012	44	SUM	026	02	02
013	01	01	027	61	GTO

SOFTSPOT SPECIAL

134	43	RCL	187	43	RCL	240	47	CMS	293	16	A*
135	06	06	188	15	15	241	81	RST	294	22	INV
136	42	STO	189	95	=	242	76	LBL	295	87	IFF
137	04	04	190	77	GE	243	17	B*	296	00	00
138	01	1	191	52	EE	244	43	RCL	297	24	CE
139	06	6	192	72	ST*	245	01	01	298	01	1
140	32	X:T	193	04	04	246	42	STO	299	02	2
141	76	LBL	194	25	CLR	247	04	04	300	08	8
142	35	1/X	195	42	STO	248	86	STF	301	32	X:T
143	01	1	196	15	15	249	01	01	302	43	RCL
144	05	5	197	61	GTO	250	61	GTO	303	02	02
145	75	-	198	58	FIX	251	43	RCL	304	22	INV
146	73	RC*	199	76	LBL	252	76	LBL	305	77	GE
147	04	04	200	52	EE	253	11	A	306	70	RAD
148	95	=	201	25	CLR	254	22	INV	307	75	-
149	72	ST*	202	72	ST*	255	87	IFF	308	02	2
150	04	4	203	04	04	256	00	00	309	05	5
151	69	DP	204	76	LBL	257	24	CE	310	04	4
152	34	34	205	58	FIX	258	91	R/S	311	85	+
153	97	DSZ	206	69	DP	259	65	X	312	43	RCL
154	03	03	207	24	24	260	05	5	313	01	01
155	35	1/X	208	97	DSZ	261	85	+	314	95	=
156	69	DP	209	03	03	262	02	2	315	15	E
157	24	24	210	48	EXC	263	07	7	316	76	LBL
158	43	RCL	211	76	LBL	264	02	2	317	70	RAD
159	05	05	212	60	DEG	265	95	=	318	85	+
160	42	STO	213	73	RC*	266	42	STO	319	02	2
161	03	03	214	06	06	267	00	00	320	85	+
162	73	RC*	215	91	R/S	268	43	RCL	321	43	RCL
163	04	04	216	69	DP	269	01	01	322	01	01
164	85	+	217	36	36	270	83	GD*	323	95	=
165	01	1	218	97	DSZ	271	00	00	324	15	E
166	95	=	219	05	05	272	85	+	325	00	0
167	22	INV	220	60	DEG	273	43	RCL	326	00	0
168	77	GE	221	76	LBL	274	02	02	018	65	X
169	49	PRD	222	24	CE	275	95	=	031	13	C
170	01	1	223	25	CLR	276	15	E	059	15	E
171	42	STO	224	55	+	277	75	-	065	75	-
172	15	15	225	00	0	278	43	RCL	085	43	RCL
173	25	CLR	226	95	=	279	02	02	095	42	STO
174	76	LBL	227	47	CMS	280	95	=	142	35	1/X
175	49	PRD	228	81	RST	281	15	E	175	49	PRD
176	72	ST*	229	76	LBL	282	65	X	183	48	EXC
177	04	04	230	12	B	283	43	RCL	200	52	EE
178	69	DP	231	86	STF	284	02	02	205	58	FIX
179	24	24	232	00	00	285	95	=	212	60	DEG
180	69	DP	233	61	GTO	286	15	E	222	24	CE
181	33	33	234	00	00	287	55	+	230	12	B
182	76	LBL	235	00	00	288	43	RCL	237	14	D
183	48	EXC	236	76	LBL	289	02	02	243	17	B*
184	73	RC*	237	14	D	290	95	=	253	11	A
185	04	04	238	43	RCL	291	15	E	293	16	A*
186	85	+	239	01	01	292	76	LBL	317	70	RAD