

	C	E	F	G	H	I	J	K	L	M
1	Label	Memory Mapped I/O	C statement	Comment	Opcode	Operand	Byte 1	Byte 2	Byte 3	Byte 4
2			RM11 = 0x00	Start of firmware: E000						
3	RESET		noInterrupts()	Disable interrupts						
4				Set stack: 256 bytes	LDS	#\$:00FF				
5		Write: MC6821: Port B Ctl	mm_PiAB2 = 0x00							
6		Write: MC6821: Port B DDR	mm_PiAB1 = 0x00							
7		Write: MC6821: Port B Ctl	mm_PiAB2 = 0x04							
8		Read: MC6821: Port B DDR	RM17 = mm_PiAB1	Get status of PB0 (freq detect)						
9		Write: MC6821: Port A Ctl	mm_PiAA2 = 0x3E							
10		Write: MC6821: Port B Ctl	mm_PiAB2 = 0x3E							
11		Write: MC6821: Port B DDR	mm_PiAB1 = 0x00							
12		Write: MC6821: Port A DDR	mm_PiAA1 = 0x10							
13		Write: MC6821: Port A Ctl	mm_PiAA2 = 0x3A							
14		Write: MC6821: Port B Ctl	mm_PiAB2 = 0x3A							
15		Write: MC6821: Port A DDR	mm_PiAA1 = 0x1E							
16		Write: MC6821: Port B DDR	mm_PiAB1 = 0xFF							
17		Write: MC6821: Port A Ctl	mm_PiAA2 = 0x3E							
18		Write: MC6821: Port B Ctl	mm_PiAB2 = 0x3E							
19		Write: MC6840: Timer 3 MSB	mm_PTM6 = 0x00							
20		Write: MC6840: Timer 3 latch	mm_PTM7 = 0x27	Frequency timer = 12.5kHz (80uS)						
21		Write: MC6840: Timer 2 MSB	mm_PTM4 = 0x01							
22		Write: MC6840: Timer 2 latch	mm_PTM5 = 0xAC							
23		Write: MC6840: Timer 1 MSB	mm_PTM2 = 0x21							
24		Write: MC6840: Timer 1 latch	mm_PTM3 = 0x33							
25		Write: MC6840: Timer 2 Control	mm_PTM1 = 0x12	Timer 1/3 select: 3						
26		Write: MC6840: Main Control	mm_PTM0 = 0x12							
27		Write: MC6840: Timer 2 Control	mm_PTM1 = 0x13	Timer 1/3 select: 1						
28		Write: MC6840: Main Control	mm_PTM0 = 0x92							
29		Write: MC6840: Timer 2 Control	mm_PTM1 = 0xD3	Timer 1/3 select: 1						
30			interrupts()	Reenable interrupts						
31			RM45 = 0x00							
32			RM46 = 0x01							
33			RM37 = 0x04							
34			RM35 = 0x00							
35			RM36 = 0x00							
36			RM39 = 0x00							
37			RM40 = 0x9B							
38			MSG25 = 0x00							
39			RM01 = 0x00							
40			MSG23 = 0x00							
41			MSG24 = 0x01							
42			RM34 = 0x01							
43			RM11 = 0x00							
44		Write: LS138: Config Plug	mm_configPlug = 0x00							
45		Write: LS138: Relay Stat	mm_relayStatus = 0x00							
46		Write: LS138: DCT flag	mm_DCT = 0x00							
47		Write: LS138: 8279 IRQ	mm_kDispIRQ = 0x00							
48		Write: LS138: Config switch	mm_config_Switch = 0x00							
49			modeUUT = 0x00							
50			keyNum = 0x00							
51			RM32 = 0x00							

	C	E	F	G	H	I	J	K	L	M
52			RM10 = 0x00							
53			RM33 = 0x00							
54			RM55 = 0x00							
55			RM26 = 0x00							
56			RM61 = 0x00							
57			RM62 = 0x00							
58			RM63 = 0x00							
59			blankDisplay()							
60			refreshDisplay()							
61			RM11 = 0xAE							
62			Disp09 = 0x01	Test No. 01 - initial power-up						
63			Disp10 = 0x00							
64			refreshDisplay()							
65			RM15 = 0x04							
66			pause()							
67	LB_E0D7	Read: LS138: Config Plug	configPlug = (mm_configPlug & 0x03)	Read Configuration plug / determine test set to run						
68		Read: LS138: Config switch	configSwitch = (mm_configSwitch & 0x03)							
69			if (configPlug != configSwitch) goto LB_E1ED	Switch / plug mismatch!						
70			Disp16 = 0x0F	Status LED: Off						
71			refreshDisplay()							
72			if (configSwitch = 0x03) modeUUT = 0x04	We are testing a DCT						
73			else if (configSwitch = 0x02) modeUUT = 0x02	We are testing an LMT-2						
74			else modeUUT = 0x01	We are testing an LMT-100						
75			blankDisplay()							
76			Disp09 = 0x02	Test No. 02 - standby						
77			Disp10 = 0x00							
78			Disp01 = 0x00							
79			Disp02 = 0x00							
80			Disp03 = 0x00							
81			Disp04 = 0x00							
82			Disp05 = 0x00							
83			Disp06 = 0x00							
84			Disp07 = 0x00							
85			Disp08 = 0x00							
86			refreshDisplay()							
87	LB_E12B		getKeyData()	Possible start of main loop						
88			if ((RM11 & 0x80) == 0x00) goto LB_E15E							
89			if (keyDat != 0x0B) goto LB_E15E	Was Check key pressed?						
90			Disp01 = 0x00							
91			Disp02 = 0x00							
92			if ((RM11 & 0x06) == 0x06) goto LB_E159							
93			Disp03 = 0x00							
94			Disp04 = 0x0F							
95			if (RM11 == 0x02) goto LB_E159							
96			Disp03 = 0x00							
97			Disp04 = 0x00							
98			Disp05 = 0x00							
99			Disp06 = 0x00							
100			Disp07 = 0x00							
101			Disp08 = 0x00							
102	LB_E159		refreshDisplay()							

	C	E	F	G	H	I	J	K	L	M
103			goto LB_E12B							
104	LB_E15E		if ((RM11 & 0x40) == 0x00) goto LB_E167							
105			if (keyDat == 0x13) goto LB_E17C	Was Repeat key pressed?						
106	LB_E167		if ((RM11 & 0x20) == 0x00) goto LB_E173							
107			if (keyDat != 0x1A) goto LB_E173	Was Enter key pressed?						
108			goto LB_E279							
109	LB_E173		if ((RM11 & 0x10) == 0x00) goto LB_E193							
110			if (keyDat != 0x1B) goto LB_E193	Was Next key pressed?						
111	LB_E17C		RM11 = 0x00							
112			if ((modeUUT & 0x01) == 0x00) goto LB_E187	Check mode: LMT-2						
113			goto LB_E3AA							
114	LB_E187		if (RM49 != 0x08) goto LB_E5E6							
115			goto LB_EF21							
116	LB_E193		if ((RM11 & 0x08) == 0x00) goto LB_E199							
117			goto LB_E205							
118	LB_E199		RM15 = (RM11 << 6)							
119			getKeyNum()							
120			if (keyNum < 0) goto LB_E12B	No number was entered, so go back to main loop						
121			if ((RM15 & 0x80) == 0x80) goto LB_E1C4							
122			else if ((RM15 & 0x40) = 0x00) goto LB_E1BC							
123			Disp03 = Disp02							
124			Disp02 = Disp01							
125	LB_E1B1		Disp01 = keyNum							
126			refreshDisplay()							
127			goto LB_E12B							
128	LB_E1BC		Disp02 = Disp01							
129			goto LB_E1B1							
130	LB_E1C4		Disp08 = Disp07							
131			Disp07 = Disp06							
132			Disp06 = Disp05							
133			Disp05 = Disp04							
134			Disp04 = Disp03							
135			Disp03 = Disp02							
136			Disp02 = Disp01							
137			Disp01 = keyNum							
138			refreshDisplay()							
139			goto LB_E12B							
140	LB_E1ED		Disp16 = 0x04	Status LED: Fail						
141			refreshDisplay()							
142			RM15 = 0x06							
143			pause()							
144			goto LB_E0D7							
145	LB_E1FF		goto displayTest							
146	LB_E205		if (keyDat != 0x03) goto LB_E199							
147			If (Disp09 != 0x02) goto LB_E1FF	Is Test No. at 02?						
148			if (Disp10 != 0x00) goto LB_E1FF							
149					LDA A	#\$:00				
150					LDA B	#\$:08				
151					LDX	#\$:0C69				
152	LB_E21D				ADD A	\$:07,X				
153					DEX					

	C	E	F	G	H	I	J	K	L	M
154					DEC B					
155					BNE	LB_E21D				
156					CMP A	#\$:00				
157					BEQ	LB_E1FF				
158			RM25 = 0x06							
159					LDX	#\$:0C69				
160					LDA A	\$:07,X				
161	LB_E230				ADD A	\$:06,X				
162					DAA					
163					DEX					
164			RM25 = (RM25 - 1)							
165					BNE	LB_E230				
166					AND A	#\$:0F				
167					CMP A	Disp01				
168					BEQ	LB_E24B				
169			Disp16 = 0x04	Status LED: Fail						
170			refreshDisplay()							
171			goto LB_E12B							
172	LB_E24B		Disp16 = 0x07	Status LED: Pass						
173			refreshDisplay()							
174			RM15 = 0x04							
175			pause()							
176			RM49 = 0x08							
177			subroutine (LB_F950)							
178					LDA A	RM24				
179					LDA B	RM23				
180					ASL D					
181					ASL D					
182					STA A	RM63				
183					LDA A	RM23				
184					LDA B	RM22				
185					ASL D					
186					ASL D					
187					STA A	RM62				
188					STA B	RM61				
189			goto LB_EF21							
190	LB_E279		RM25 = 0x06							
191					LDX	#\$:0C69				
192					LDA A	\$:07,X				
193					AND A	#\$:0F				
194					CMP A	#\$:0F				
195					BNE	LB_E29F				
196					LDX	#\$:0C64				
197			RM25 = 0x01							
198					LDA A	\$:07,X				
199					AND A	#\$:0F				
200					CMP A	#\$:0F				
201					BNE	LB_E29F				
202					LDX	#\$:0C64				
203			RM25 = 0x01							
204					LDA A	#\$:00				

	C	E	F	G	H	I	J	K	L	M
205	LB_E29F				ADD A	\$.06,X				
206					DAA					
207					DEX					
208			RM25 = (RM25 - 1)							
209					BNE	LB_E29F				
210					AND A	#\$:0F				
211					CMP A	Disp01				
212					BEQ	LB_E2BA				
213			Disp16 = 0x04	Status LED: Fail						
214			refreshDisplay()							
215			goto LB_E12B							
216	LB_E2BA		if ((RM11 & 0x01) == 0x00) goto LB_E2C2							
217			goto LB_E352							
218	LB_E2C2		if ((RM11 & 0x02) == 0x00) goto LB_E2C2							
219			subroutine (LB_F950)							
220			blankDisplay()							
221			Disp16 = 0x07	Status LED: Pass						
222			Disp09 = 0x03	Test No. 03						
223			Disp10 = 0x00							
224			Disp01 = 0x00							
225			Disp02 = 0x00							
226			Disp03 = 0x00							
227			refreshDisplay()							
228			RM11 = 0xAD							
229			goto LB_E12B							
230	LB_E2EC		if (Disp01 != 0x01) goto LB_E320							
231			RM32 = 0x01							
232			RM49 = 0x01							
233			Disp15 = 0x05	Install LED: On						
234	LB_E2FB		RM11 = 0x14							
235			Disp16 = 0x07	Status LED: Pass						
236			Disp09 = 0x05	Change Test No. to 05						
237			Disp01 = 0x0F							
238			Disp02 = 0x0F							
239			Disp03 = 0x0F							
240			Disp04 = 0x0F							
241			refreshDisplay()							
242			goto LB_E12B							
243	LB_E320		if (Disp01 != 0x03) goto LB_E32F							
244			RM49 = 0x02							
245	LB_E328		RM32 = 0x00							
246			Disp15 = 0x0F	Install LED: Off						
247			goto LB_E2FB							
248	LB_E32F		if (Disp01 != 0x07) goto LB_E338							
249			RM49 = 0x00							
250			goto LB_E328							
251	LB_E338		if (Disp01 != 0x09) goto LB_E342							
252			RM49 = 0x03							
253			goto LB_E328							
254	LB_E342		Disp16 = 0x04	Status LED: Fail						
255			Disp09 = 0x04	Change test No. to 04						

	C	E	F	G	H	I	J	K	L	M
1072			goto LB_EE1C							
1073	LB_EE0E		Disp01 = 0x01							
1074			goto LB_EE1C							
1075	LB_EE12		Disp01 = 0x02							
1076			goto LB_EE1C							
1077	LB_EE16		Disp01 = 0x03							
1078			goto LB_EE1C							
1079	LB_EE1A		Disp01 = 0x04							
1080	LB_EE1C		Disp16 = 0x04	Status LED: Fail						
1081			refreshDisplay()							
1082			RM26 = 0x00							
1083			RM11 = 0x4C							
1084			goto LB_E12B							
1085	LB_EE31		Disp11 = 0x04	Relay A status: Fail						
1086			if ((relayTemp & 0x02) == 0x00) goto LB_EE67							
1087			Disp12 = 0x07	Relay B status: Pass						
1088	LB_EE3E		if ((relayTemp & 0x04) == 0x00) goto LB_EE6F							
1089			Disp13 = 0x07	Relay C status: Pass						
1090	LB_EE46		if ((relayTemp & 0x08) == 0x00) goto LB_EE77							
1091			Disp14 = 0x07	Relay D status: Pass						
1092	LB_EE4E		Disp16 = 0x04	Status LED: Fail						
1093			refreshDisplay()							
1094	LB_EE56		getKeyData()							
1095			if (keyDat != 0x13) goto LB_EE60	Was Repeat key pressed?						
1096			goto LB_EC67							
1097	LB_EE60		if (keyDat != 0x1B) goto LB_EE56	Was Next key pressed?						
1098			goto LB_ECFF							
1099	LB_EE67		Disp12 = 0x04	Relay B status: Fail						
1100			goto LB_EE3E							
1101	LB_EE6F		Disp13 = 0x04	Relay C status: Fail						
1102			goto LB_EE46							
1103	LB_EE77		Disp14 = 0x04	Relay D status: Fail						
1104			goto LB_EE4E							
1105	LB_EE7F		relayTemp = (RM20 ^ RM16)							
1106			if ((relayTemp & 0x01) == 0x00) goto LB_EE9A							
1107			Disp11 = 0x07	Relay A status: Pass						
1108			if ((relayTemp & 0x02) == 0x00) goto LB_EEB2							
1109			Disp12 = 0x07	Relay B status: Pass						
1110			if ((relayTemp & 0x04) == 0x00) goto LB_EEB9							
1111			Disp13 = 0x07	Relay C status: Pass						
1112			goto LB_ECF2							
1113	LB_EE9A		Disp11 = 0x04	Relay A status: Fail						
1114			if ((relayTemp & 0x02) == 0x00) goto LB_EEB2							
1115			Disp12 = 0x07	Relay B status: Pass						
1116	LB_EEA7		if ((relayTemp & 0x04) == 0x00) goto LB_EEB9							
1117			Disp13 = 0x07	Relay C status: Pass						
1118			goto LB_EE4E							
1119	LB_EEB2		Disp12 = 0x04	Relay B status: Fail						
1120			goto LB_EEA7							
1121	LB_EEB9		Disp13 = 0x04	Relay C status: Fail						
1122			goto LB_EE4E							

	C	E	F	G	H	I	J	K	L	M
1123	LB_EEC1		relayTemp = (RM20 ^ RM16)							
1124			if ((relayTemp & 0x01) == 0x00) goto LB_EED0							
1125			Disp11 = 0x07	Relay A status: Pass						
1126			goto LB_ECF2							
1127	LB_EED0		Disp11 = 0x04	Relay A status: Fail						
1128			goto LB_EE4E							
1129	LB_EED8		relayTemp = (RM20 ^ RM16)							
1130			if ((relayTemp & 0x01) == 0x00) goto LB_EEED							
1131			Disp11 = 0x07	Relay A status: Pass						
1132			if ((relayTemp & 0x02) == 0x00) goto LB_EEFD							
1133			Disp12 = 0x07	Relay B status: Pass						
1134			goto LB_ECF2							
1135	LB_EEED		Disp11 = 0x04	Relay A status: Fail						
1136			if ((relayTemp & 0x02) == 0x00) goto LB_EEFD							
1137			Disp12 = 0x07	Relay B status: Pass						
1138			goto LB_EE4E							
1139	LB_EEFD		Disp12 = 0x04	Relay B status: Fail						
1140			goto LB_EE4E							
1141	LB_EF05		Disp14 = 0x04	Relay D status: Fail						
1142			Disp16 = 0x04	Status LED: Fail						
1143			refreshDisplay()							
1144	LB_EF10		getKeyData()							
1145			if (keyDat != 0x13) goto LB_EF1A	Was Repeat key pressed?						
1146			goto LB_ECFF							
1147	LB_EF1A		if (keyDat != 0x1B) goto LB_EF10	Was Next key pressed?						
1148			goto LB_ED80							
1149	LB_EF21		RM26 = 0x00							
1150			blankDisplay()							
1151			Disp09 = 0x05	Test No. 25						
1152			Disp10 = 0x02							
1153			refreshDisplay()							
1154			if (modeUUT == 0x02) goto LB_EF39	Check mode: LMT-2						
1155			goto LB_E8A8							
1156	LB_EF39		subroutine (LB_FC9A)							
1157			MSG07 = 0x00							
1158			subroutine (LB_FC01)							
1159			if (MSG03 == RM22) goto LB_EF4B							
1160			goto LB_EE1A							
1161	LB_EF4B		zz99 = (MSG02 & 0x1F)							
1162			zz98 = (RM23 & 0x1F)							
1163			if (zz98 == zz99) goto LB_EF59							
1164			goto LB_EE1A							
1165	LB_EF59		RM54 = MSG04							
1166	LB_EF5E		if (RM54 == 0x01) goto LB_EF7D							
1167			else if (RM54 == 0x24) goto LB_EF7D							
1168			else if (RM54 == 0x4A) goto LB_EF81							
1169			else if (RM54 == 0x5D) goto LB_EF81							
1170			else if (RM54 == 0x58) goto LB_EF85							
1171			else if (RM54 == 0x5F) goto LB_EF85							
1172			goto LB_E8A8							
1173	LB_EF7D		RM55 = 0x02							

	C	E	F	G	H	I	J	K	L	M
1174			goto LB_EF87							
1175	LB_EF81		RM55 = 0x04							
1176			goto LB_EF87							
1177	LB_EF85		RM55 = 0x06							
1178	LB_EF87		blankDisplay()							
1179			Disp09 = 0x05	Test No. 45						
1180			Disp10 = 0x04							
1181			refreshDisplay()							
1182			RM15 = 0x08							
1183			pause()							
1184			subroutine (LB_FC9A)							
1185			if (RM55 != 0x06) goto LB_EFAB							
1186			MSG07 = 0x12							
1187			goto LB_EFAD							
1188	LB_EFAB		MSG07 = 0x89							
1189	LB_EFAD		subroutine (LB_FC01)							
1190			if (RM55 == 0x02) goto LB_EFBC							
1191			goto LB_F041							
1192	LB_EFBC				LDA A	MSG06				
1193					ASL A					
1194					ASL A					
1195					ASL A					
1196					STA A	RM16				
1197					LDA B	MSG06				
1198					LSR B					
1199					LSR B					
1200					LSR B					
1201					LSR B					
1202					LSR B					
1203					LDA A	MSG05				
1204					ASL A					
1205					ASL A					
1206					ASL A					
1207					ABA					
1208					STA A	RM17				
1209					LDA B	MSG05				
1210					LSR B					
1211					LSR B					
1212					LSR B					
1213					LSR B					
1214					LSR B					
1215					LDA A	MSG04				
1216					ASL A					
1217					ASL A					
1218					ASL A					
1219					ABA					
1220					STA A	RM18				
1221					LDA B	MSG04				
1222					LSR B					
1223					LSR B					
1224					LSR B					

	C	E	F	G	H	I	J	K	L	M
1225					LSR B					
1226					LSR B					
1227					STA B	RM19				
1228					LDA A	MSG06				
1229					ADD A	RM16				
1230					STA A	RM16				
1231					LDA A	MSG05				
1232					ADC A	RM17				
1233					STA A	RM17				
1234					LDA A	MSG04				
1235					ADC A	RM18				
1236					STA A	RM18				
1237					LDA A	#\$:00				
1238					ADC A	RM19				
1239					STA A	RM19				
1240					LDA A	#\$:00				
1241					STA A	RM57				
1242					STA A	RM58				
1243					STA A	RM59				
1244	LB_F00C				LDA D	#\$:FB1E				
1245					ADD B	RM16				
1246					STA B	RM16				
1247					ADC A	RM17				
1248					STA A	RM17				
1249					LDA D	#\$:FFFF				
1250					ADC A	RM18				
1251					STA A	RM18				
1252					ADC B	RM19				
1253					STA B	RM19				
1254					BCC	LB_F056				
1255					LDA A	#\$:01				
1256					ADD A	RM57				
1257					DAA					
1258					STA A	RM57				
1259					LDA A	#\$:00				
1260					ADC A	RM58				
1261					DAA					
1262					STA A	RM58				
1263					LDA A	#\$:00				
1264					ADC A	RM59				
1265					DAA					
1266					STA A	RM59				
1267			goto LB_F00C							
1268	LB_F041		subroutine (LB_FAF7)							
1269			subroutine (LB_F9B4)							
1270			RM57 = 0xFF							
1271			RM60 = (RM60 0xF0)							
1272			goto LB_F063							
1273	LB_F056		RM59 = (RM59 0xF0)							
1274			RM60 = 0xFF							
1275	LB_F063		subroutine (LB_FB15)							

	C	E	F	G	H	I	J	K	L	M
1276			Disp16 = 0x07	Status LED: Pass						
1277			refreshDisplay()							
1278	LB_F06E		getKeyData()							
1279			if (keyDat != 0x13) goto LB_F07B	Was Repeat key pressed?						
1280			goto LB_EF5E							
1281	LB_F07B		if (keyDat != 0x1B) goto LB_F06E	Was Next key pressed?						
1282			if (RM55 != 0x02) goto LB_F06E							
1283			blankDisplay()							
1284			Disp09 = 0x07	Test No. 47						
1285			Disp10 = 0x04							
1286			refreshDisplay()							
1287			RM15 = 0x08							
1288			pause()							
1289			subroutine (LB_FC9A)							
1290			MSG07 = 0x8F							
1291			subroutine (LB_FC01)							
1292			subroutine (LB_FAF7)							
1293			subroutine (LB_F9B4)							
1294			subroutine (LB_FB15)							
1295			refreshDisplay()							
1296	LB_F0AF		getKeyData()							
1297			if (keyDat != 0x13) goto LB_F0BC	Was Repeat key pressed?						
1298			goto LB_EF5E							
1299	LB_F0BC		if (keyDat != 0x1B) goto LB_F0AF	Was Next key pressed?						
1300			blankDisplay()							
1301			Disp09 = 0x08	Test No. 48						
1302			Disp10 = 0x04							
1303			refreshDisplay()							
1304			RM15 = 0x08							
1305			pause()							
1306			subroutine (LB_FC9A)							
1307			MSG07 = 0x92							
1308			subroutine (LB_FC01)							
1309			subroutine (LB_FAF7)							
1310			subroutine (LB_F9B4)							
1311			subroutine (LB_FB15)							
1312			refreshDisplay()							
1313	LB_F0E9		getKeyData()							
1314			if (keyDat != 0x13) goto LB_F0E9	Was Repeat key pressed?						
1315			goto LB_EF5E							
1316	refreshDisplay	Write: LS138: 8279 Cmd	mm_kDispCmd = 0xDC	8279 command 6: 1 1 1 (clear display RAM)						
1317				Delay - wait for 8279 to clear	LDA A	#\$:64				
1318	LB_F0FD				DEC A					
1319					BNE	LB_F0FD				
1320		Write: LS138: 8279 Cmd	mm_kDispCmd = 0xAA	8279 command 5: 1 0 1 0 (Port B only)						
1321		Write: LS138: 8279 Cmd	mm_kDispCmd = 0x18	8279 command 0: DD 1 1, KB 0 0 0						
1322		Write: LS138: 8279 Cmd	mm_kDispCmd = 0x2A	8279 command 1: Clock divisor 10 (1MHz / 10 = 100kHz)						
1323		Write: LS138: 8279 Cmd	mm_kDispCmd = 0x90	8279 command 4: Prepare to write to Display RAM						
1324					LDX	#\$:0C69				
1325			RM25 = 0x10							
1326					LDA A	\$:00,X				

	C	E	F	G	H	I	J	K	L	M
1327	LB_F11B	Write: LS138: 8279 Data		Send data to 8279	STA A	mm_kDispData				
1328					INX					
1329			RM25 = (RM25 - 1)							
1330					BNE	LB_F11B				
1331		Write: LS138: 8279 Cmd	mm_kDispCmd = 0x80	8279 command 4: Done writing to display RAM						
1332			return (subroutine)							
1333	pause	Write: MC6840: Main Control	mm_PTM0 = 0x22							
1334	LB_F130	Write: MC6840: Timer 1 MSB	mm_PTM2 = 0xFF							
1335		Write: MC6840: Timer 1 latch	mm_PTM3 = 0xFF							
1336	LB_F135	Read: MC6840: Get Status	timer = mm_PTM1							
1337			if ((timer & 0x01) == 0x01) goto LB_F135							
1338			RM15 = (RM15 - 0x01)							
1339			if (RM15 != 0x00) goto LB_F130							
1340			return (subroutine)							
1341	LB_F141		blankDisplay()							
1342			Disp09 = 0x01	Test No. 61						
1343			Disp10 = 0x06							
1344			refreshDisplay()							
1345			subroutine (LB_FC9A)							
1346			MSG07 = 0x67							
1347			subroutine (LB_FC01)							
1348			subroutine (LB_FA24)							
1349			keyCheck()							
1350			blankDisplay()							
1351			Disp09 = 0x02	Test No. 62						
1352			Disp10 = 0x06							
1353			refreshDisplay()							
1354			subroutine (LB_FC9A)							
1355			MSG07 = 0x69							
1356			subroutine (LB_FC01)							
1357			subroutine (LB_FA24)							
1358			keyCheck()							
1359			blankDisplay()							
1360			Disp09 = 0x03	Test No. 63						
1361			Disp10 = 0x06							
1362			refreshDisplay()							
1363			subroutine (LB_FC9A)							
1364			MSG07 = 0x6B							
1365			subroutine (LB_FC01)							
1366			subroutine (LB_FA24)							
1367			keyCheck()							
1368			blankDisplay()							
1369			Disp09 = 0x04	Test No. 64						
1370			Disp10 = 0x06							
1371			refreshDisplay()							
1372			subroutine (LB_FC9A)							
1373			MSG07 = 0x6D							
1374			subroutine (LB_FC01)							
1375			subroutine (LB_FA24)							
1376			keyCheck()							
1377	LB_F1B1		if ((RM48 & 0x02) == 0x02) goto LB_F1BA							

	C	E	F	G	H	I	J	K	L	M
1378			goto LB_F3C3							
1379	LB_F1BA		blankDisplay()							
1380			Disp09 = 0x00	Test No. 70						
1381			Disp10 = 0x07							
1382			refreshDisplay()							
1383			subroutine (LB_FC9A)							
1384			MSG07 = 0x3A							
1385			subroutine (LB_FC01)							
1386			RM72 = MSG04							
1387			RM15 = 0x02							
1388			pause()							
1389			subroutine (LB_FC9A)							
1390			MSG07 = 0x3C							
1391			subroutine (LB_FC01)							
1392			RM74 = MSG04							
1393			RM15 = 0x02							
1394			pause()							
1395			subroutine (LB_FC9A)							
1396			MSG07 = 0x42							
1397			subroutine (LB_FC01)							
1398			RM76 = MSG04							
1399			Disp16 = 0x07	Status LED: Pass						
1400			refreshDisplay()							
1401			RM15 = 0x10							
1402			pause()							
1403			blankDisplay()							
1404			Disp09 = 0x01	Test No. 71						
1405			Disp10 = 0x07							
1406			refreshDisplay()							
1407			subroutine (LB_FC9A)							
1408			MSG07 = 0x55							
1409			subroutine (LB_FC80)							
1410			RM15 = 0x02							
1411			pause()							
1412			subroutine (LB_FC9A)							
1413			MSG07 = 0x62							
1414			subroutine (LB_FC80)							
1415			RM15 = 0x02							
1416			pause()							
1417			subroutine (LB_FBC8_5)							
1418			subroutine (LB_FC57)							
1419			Disp16 = 0x07	Status LED: Pass						
1420			refreshDisplay()							
1421			RM15 = 0x10							
1422			pause()							
1423			blankDisplay()							
1424			Disp09 = 0x02	Test No. 72						
1425			Disp10 = 0x07							
1426			refreshDisplay()							
1427			subroutine (LB_FC9A)							
1428			MSG07 = 0x60							

	C	E	F	G	H	I	J	K	L	M
2092	displayTest		RM15 = 0x2E	Test - all LEDs and segments						
2093			Disp01 = 0x08							
2094			Disp02 = 0x08							
2095			Disp03 = 0x08							
2096			Disp04 = 0x08							
2097			Disp05 = 0x08							
2098			Disp06 = 0x08							
2099			Disp07 = 0x08							
2100			Disp08 = 0x08							
2101			Disp09 = 0x08							
2102			Disp10 = 0x08							
2103			Disp11 = 0x08	Relay A status LED						
2104			Disp12 = 0x08	Relay B status LED						
2105			Disp13 = 0x08	Relay C status LED						
2106			Disp14 = 0x08	Relay D status LED						
2107			Disp15 = 0x08	Install LED						
2108			Disp16 = 0x08	Status LED						
2109			refreshDisplay()							
2110			pause()							
2111			blankDisplay()							
2112			Disp03 = 0x07	Display firmware revision						
2113			Disp06 = 0x00							
2114			Disp07 = 0x06							
2115			Disp08 = 0x00							
2116			refreshDisplay()							
2117			RM15 = 0x24							
2118			pause()							
2119			goto RESET							
2120	blankDisplay		Disp01 = 0x0F	7447 LED display driver treats 0x0F as 'all off'						
2121			Disp02 = 0x0F							
2122			Disp03 = 0x0F							
2123			Disp04 = 0x0F							
2124			Disp05 = 0x0F							
2125			Disp06 = 0x0F							
2126			Disp07 = 0x0F							
2127			Disp08 = 0x0F							
2128			Disp09 = 0x0F							
2129			Disp10 = 0x0F							
2130			Disp11 = 0x0F	Relay A status LED						
2131			Disp12 = 0x0F	Relay B status LED						
2132			Disp13 = 0x0F	Relay C status LED						
2133			Disp14 = 0x0F	Relay D status LED						
2134			Disp15 = 0x0F	Install LED						
2135			Disp16 = 0x0F	Status LED						
2136			return (subroutine)							
2137	LB_F950				CLC					
2138			RM15 = 0x00							
2139			RM22 = 0x00							
2140			RM23 = 0x00							
2141			RM24 = 0x00							
2142			RM25 = 0x07							

	C	E	F	G	H	I	J	K	L	M
2143					LDX	#\$:0C69				
2144	LB_F961				LDA A	RM23				
2145					LDA B	RM22				
2146					ADD B	\$:07,X				
2147					STA B	RM22				
2148					BCS	LB_F971				
2149	LB_F96B		RM25 = (RM25 - 1)							
2150					BNE	LB_F985				
2151			return (subroutine)							
2152	LB_F971				ADD A	#\$:01				
2153					STA A	RM16				
2154					STA A	RM23				
2155					BCS	LB_F97B				
2156			goto LB_F96B							
2157	LB_F97B				LDA A	RM24				
2158					ADD A	#\$:01				
2159					STA A	RM24				
2160					LDA A	RM16				
2161			goto LB_F96B							
2162	LB_F985				ASL D					
2163					STA B	RM22				
2164					STA A	RM16				
2165					STA A	RM23				
2166					ROL	RM24				
2167					LDA A	RM24				
2168					STA A	RM17				
2169					LDA A	RM16				
2170					ASL D					
2171					ROL	RM17				
2172					ASL D					
2173					ROL	RM17				
2174					ADD B	RM22				
2175					STA B	RM22				
2176					ADC A	RM23				
2177					STA A	RM23				
2178					BCC	LB_F9AA				
2179					INC	RM24				
2180	LB_F9AA				LDA A	RM17				
2181					ADD A	RM24				
2182					STA A	RM24				
2183					DEX					
2184			goto LB_F961							
2185	LB_F9B4				LDX	RM67				
2186					LDA B	RM70				
2187					LDA A	#\$:00				
2188	LB_F9BB				STA A	\$:00,X				
2189					DEX					
2190					DEC B					
2191					BNE	LB_F9BB				
2192					INX					
2193					STX	RM67				

	C	E	F	G	H	I	J	K	L	M
2194					LDA A	#\$:80				
2195					STA A	RM69				
2196	LB_F9CA				LDX	RM65				
2197					LDA A	\$:00,X				
2198					PSH A					
2199					DEX					
2200					STX	RM65				
2201	LB_F9D4				PUL A					
2202					PSH A					
2203					AND A	RM69				
2204					BEQ	LB_F9F2				
2205					LDX	RM67				
2206					LDA B	RM70				
2207					LDA A	\$:00,X				
2208					ADD A	#\$:01				
2209	LB_F9E5				DAA					
2210					STA A	\$:00,X				
2211					DEC B					
2212					BEQ	LB_F9F2				
2213					INX					
2214					LDA A	\$:00,X				
2215					ADC A	#\$:00				
2216			goto LB_F9E5							
2217	LB_F9F2				LDA A	RM71				
2218					CMP A	#\$:01				
2219					BEQ	LB_FA10				
2220					LDX	RM67				
2221					LDA B	RM70				
2222					LDA A	\$:00,X				
2223					ADD A	\$:00,X				
2224	LB_FA03				DAA					
2225					STA A	\$:00,X				
2226					DEC B					
2227					BEQ	LB_FA10				
2228					INX					
2229					LDA A	\$:00,X				
2230					ADC A	\$:00,X				
2231			goto LB_FA03							
2232	LB_FA10		RM71 = (RM71 - 1)							
2233					BEQ	LB_FA22				
2234					LSR	RM69				
2235					BNE	LB_F9D4				
2236					PUL A					
2237					LDA A	#\$:80				
2238					STA A	RM69				
2239			goto LB_F9CA							
2240	LB_FA22				PUL A					
2241			return (subroutine)							
2242	LB_FA24		RM71 = 0x10							
2243			RM70 = 0x04							
2244			RM65 = 0x00							

	C	E	F	G	H	I	J	K	L	M
2245			RM66 = 0xA3							
2246			RM67 = 0x0C							
2247			RM68 = 0x0E							
2248					LDA A	MSG04				
2249					ROL A					
2250					ROL A					
2251					ROL A					
2252					AND A	#\$:03				
2253					STA A	RM47				
2254			MSG04 = (MSG04 & 0x3F)							
2255			MSG03 = MSG05							
2256			subroutine (LB_F9B4)							
2257			subroutine (LB_FB15)							
2258			Disp06 = 0x0F							
2259			Disp07 = RM47							
2260			Disp08 = 0x0F							
2261			refreshDisplay()							
2262			return (subroutine)							
2263	getKeyData	Read: LS138: 8279 IRQ	if ((mm_kDispIRQ & 0x01) == 0x01) goto getKeyData	Wait for key to be released						
2264		Write: LS138: 8279 Cmd	mm_kDispCmd = 0x50	8279 command 2: Read keyboard						
2265		Read: LS138: 8279 Data	keyDat = (mm_kDispData & 0x1F)	Read from 8279: Keyboard data						
2266			if (keyDat == 0x19) goto RESET	Was Reset key pressed?						
2267			return (subroutine)							
2268	keyCheck	Read: LS138: 8279 IRQ	if ((mm_kDispIRQ & 0x01) == 0x01) goto keyCheck	Wait for key to be released						
2269		Write: LS138: 8279 Cmd	mm_kDispCmd = 0x50	8279 command 2: Read keyboard						
2270		Read: LS138: 8279 Data	keyTemp = (mm_kDispData & 0x1F)	Read from 8279: Keyboard data						
2271			if (keyTemp != 0x13) goto LB_FA98	Was Repeat key pressed?						
2272					INS					
2273					INS					
2274			goto LB_F141							
2275	LB_FA98		if (keyTemp != 0x1B) goto LB_FA9D	Was Next key pressed?						
2276			return (subroutine)							
2277	LB_FA9D		if (keyTemp != 0x19) goto keyCheck	Was Reset key pressed?						
2278					INS					
2279					INS					
2280			goto RESET							
2281	getKeyNum		if (keyDat == 0x18) keyNum = 0	Key pressed: 0						
2282			else if (keyDat == 0x10) keyNum = 1	Key pressed: 1						
2283			else if (keyDat == 0x11) keyNum = 2	Key pressed: 2						
2284			else if (keyDat == 0x12) keyNum = 3	Key pressed: 3						
2285			else if (keyDat == 0x08) keyNum = 4	Key pressed: 4						
2286			else if (keyDat == 0x09) keyNum = 5	Key pressed: 5						
2287			else if (keyDat == 0x0A) keyNum = 6	Key pressed: 6						
2288			else if (keyDat == 0x00) keyNum = 7	Key pressed: 7						
2289			else if (keyDat == 0x01) keyNum = 8	Key pressed: 8						
2290			else if (keyDat == 0x02) keyNum = 9	Key pressed: 9						
2291			else keyNum = 0xFF							
2292			return (subroutine)							
2293	LB_FAE1				LDA A	RM24				
2294					LDA B	RM23				
2295					ASL D					

	C	E	F	G	H	I	J	K	L	M
2296					ASL D					
2297					STA A	RM63				
2298					LDA A	RM23				
2299					LDA B	RM22				
2300					ASL D					
2301					ASL D					
2302					STA A	RM62				
2303					STA B	RM61				
2304			return (subroutine)							
2305	LB_FAF7		RM71 = 0x18							
2306			RM70 = 0x04							
2307			RM65 = 0x00							
2308			RM66 = 0xA5							
2309			RM67 = 0x0C							
2310			RM68 = 0x0E							
2311			ZZ99 = MSG04							
2312			MSG04 = MSG06							
2313			MSG06 = ZZ99							
2314			return (subroutine)							
2315	LB_FB15		Disp01 = (RM57 & 0x0F)							
2316			Disp02 = (RM57 >> 4)							
2317			Disp03 = (RM58 & 0x0F)							
2318			Disp04 = (RM58 >> 4)							
2319			Disp05 = (RM59 & 0x0F)							
2320			Disp06 = (RM59 >> 4)							
2321			Disp07 = (RM60 & 0x0F)							
2322			Disp08 = (RM60 >> 4)							
2323			Disp16 = 0x07	Status LED: Pass						
2324			return (subroutine)							
2325	LB_FB63		RM71 = 0x08							
2326			RM70 = 0x02							
2327			RM65 = 0x0C							
2328			RM66 = 0x07							
2329			RM67 = 0x0C							
2330			RM68 = 0x0C							
2331			subroutine (LB_F9B4)							
2332			Disp06 = (RM57 & 0x0F)							
2333			Disp07 = (RM57 >> 4)							
2334			Disp08 = RM58							
2335			RM71 = 0x08							
2336			RM70 = 0x02							
2337			RM65 = 0x0C							
2338			RM66 = 0x09							
2339			RM67 = 0x0C							
2340			RM68 = 0x0C							
2341			subroutine (LB_F9B4)							
2342			Disp03 = (RM57 & 0x0F)							
2343			Disp04 = (RM57 >> 4)							
2344			refreshDisplay()							
2345			RM15 = 0x20							
2346			pause()							

	C	E	F	G	H	I	J	K	L	M
2347			return (subroutine)							
2348	LB_FC01		MSG06 = (MSG06 0x01)							
2349			MSG08 = 0x40							
2350			RM10 = 0x01							
2351			RM26 = 0x01							
2352			RM15 = 0x05							
2353			pause()							
2354					LDA A	MSG23				
2355					CMP A	#\$:41				
2356					BMI	LB_FC23				
2357					INS					
2358					INS					
2359			goto LB_EE0A							
2360	LB_FC23		if (MSG23 != 0x00) goto LB_FC23							
2361			MSG26 = 0x07							
2362			MSG24 = 0x01							
2363			RM15 = 0x07							
2364			pause()							
2365			if (MSG23 != 0x00) goto LB_FC41							
2366					INS					
2367					INS					
2368			goto LB_EE0E							
2369	LB_FC41		if (MSG23 != 0x34) goto LB_FC41							
2370			MSG23 = 0x00							
2371			if (RM01 == 0x00) goto LB_FC53							
2372					INS					
2373					INS					
2374			goto LB_EE16							
2375	LB_FC53		RM26 = 0x00							
2376			return (subroutine)							
2377	LB_FC57		MSG04 = RM63							
2378			MSG05 = RM62							
2379			MSG06 = (RM61 0x01)							
2380			RM10 = 0x01							
2381			RM26 = 0x01							
2382			RM15 = 0x24							
2383			pause()							
2384			if (RM10 <= 0x00) goto LB_FC7C							
2385					INS					
2386					INS					
2387			goto LB_EE0A							
2388	LB_FC7C		RM26 = 0x00							
2389			return (subroutine)							
2390	LB_FC80		RM10 = 0x01							
2391			RM26 = 0x01							
2392			RM15 = 0x10							
2393			pause()							
2394			if (RM10 <= 0x00) goto LB_FC96							
2395					INS					
2396					INS					
2397			goto LB_EE0A							