

Laboratory Exercise 11 – Report:

Nguyễn Hải Dương- 20194530

1. Assignment 4

- Mã nguồn:

```
1 # Assignment 4
2 .eqv IN_ADDRESS_HEXA_KEYBOARD 0xFFFF0012
3
4 .eqv OUT_ADDRESS_HEXA_KEYBOARD 0xFFFF0014
5
6 .eqv COUNTER 0xFFFF0013 # Time Counter
7
8 .eqv MASK_CAUSE_COUNTER 0x00000400 # Bit 10: Counter interrupt
9
10 .eqv MASK_CAUSE_KEYMATRIX 0x00000800 # Bit 11: Key matrix interrupt
11
12
13
14 .data
15
16     msg_keypress: .asciiz "Someone has pressed a key!\n"
17
18     msg_counter: .asciiz "Time interval "
19
20     msg_counter_endl: .asciiz "\n"
21
22 #####
23
24 # MAIN Procedure
25
26 #####
27
28 .text
29
30 li $s7, 1
31
32
33
34 main:
```

```
34 main:
35
36 #-----
37
38 # Enable interrupts you expect
39
40 #-----
41
42 # Enable the interrupt of Keyboard matrix 4x4 of Digital Lab Sim
43
44     li $t1, IN_ADDRESS_HEXA_KEYBOARD
45
46     li $t3, 0x80 # bit 7 = 1 to enable
47
48     sb $t3, 0($t1)
49
50 # Enable the interrupt of TimeCounter of Digital Lab Sim
51
52     li $t1, COUNTER
53
54     sb $t1, 0($t1)
55
56
57 #-----
58 # Loop and print sequence numbers
59
60 #-----
61
62 Loop:
63
64     nop
65
66
67
```

```
68         nop
69
70         nop
71
72 sleep:
73
74     addi $v0,$zero,32 # BUG: must sleep to wait for Time Counter
75
76     li $a0,200 # sleep 300 ms
77
78     syscall
79
80     nop # WARNING: nop is mandatory here.
81
82     b Loop
83
84 end_main:
85
86 ######
87
88 # GENERAL INTERRUPT SERVED ROUTINE for all interrupts
89
90 ######
91
92 .ktext 0x80000180
93
94 IntSR: #-----
95
96 # Temporary disable interrupt
97
```

```
97
98 #-----
99
100 dis_int:
101     li $t1, COUNTER # BUG: must disable with Time Counter
102     sb $zero, 0($t1)
103
104 # no need to disable keyboard matrix interrupt
105
106 #-----
107
108 # Processing
109
110
111 #
112 #-----
113
114 get_caus:
115
116     mfc0 $t1, $13 # $t1 = Coproc0.cause
117
118 IsCount:
119
120     li $t2, MASK_CAUSE_COUNTER# if Cause value confirm Counter..
121
122     and $at, $t1,$t2
123
124     beq $at,$t2, Counter_Intr
125
126 IsKeyMa:
127
128     li $t2, MASK_CAUSE_KEYMATRIX # if Cause value confirm Key..
129
130     and $at, $t1,$t2
```

```
130         and $at, $t1,$t2
131         beq $at,$t2, Keymatrix_Intr
132
133 others:
134
135
136         j end_process # other cases
137
138
139
140 Keymatrix_Intr:
141
142         li $t1, IN_ADDRESS_HEXA_KEYBOARD
143
144         li $t2, OUT_ADDRESS_HEXA_KEYBOARD
145
146
147
148 inter_1:
149
150         li $t3, 0x81 # check row 1 with key 0, 1, 2, 4
151
152         sb $t3, 0($t1) # must reassign expected row
153
154         jal inter
155
156
157
158 inter_2:
159
160         li $t3, 0x82 # check row 2 with key 4, 5, 6, 7
161
162         sb $t3, 0($t1) # must reassign expected row
163
```

```
163
164      jal inter
165
166
167
168 inter_3:
169
170      li $t3, 0x84 # check row 3 with key 8, 9, A, B
171
172      sb $t3, 0($t1) # must reassign expected row
173
174      jal inter
175
176
177
178 inter_4:
179
180      li $t3, 0x88 # check row 4 with key C, D, E, F
181
182      sb $t3, 0($t1) # must reassign expected row
183
184      jal inter
185
186
187
188 after_inter_4:
189
190      beq $a0, 0x0, prn_cod
191
192      j next_pc
193
194
195
196 inter:
```

```
196  inter:
197
198      lb $a0, 0($t2) # read scan code of key button
199
200      bne $a0, 0x0, prn_cod
201
202      jr $ra
203
204
205
206  prn_cod:
207
208      li $v0,34
209
210      syscall
211
212
213
214      li $v0,11
215
216      li $a0,'\\n' # print endofline
217
218      syscall
219
220
221
222  j end_process
223
224
225
226  Counter_Intr:
227
228      li $v0, 4 # Processing Counter Interrupt
229
```

```
229
230     la $a0, msg_counter
231
232     syscall
233
234
235
236     li $v0,1
237
238     add $a0, $s7, $zero
239
240     syscall
241
242
243
244     add $s7, $s7, 1
245
246
247
248     li $v0, 4 # Processing Counter Interrupt
249
250     la $a0, msg_counter_endl
251
252     syscall
253
254
255
256     j end_process
257
258 end_process:
259
260     mtc0 $zero, $13 # Must clear cause reg
261
262 en int: #-----
```

```

--- 262 en_int: #-----
263
264     # Re-enable interrupt
265
266     #
267
268         li $t1, COUNTER
269
270         sb $t1, 0($t1)
271
272     #
273
274     # Evaluate the return address of main routine
275
276     # epc <= epc + 4
277
278     #
279
280 next_pc:
281
282         mfc0 $at, $14 # $at <= Coproc0.$14 = Coproc0.epc
283
284         addi $at, $at, 4 # $at = $at + 4 (next instruction)
285
286         mtc0 $at, $14 # Coproc0.$14 = Coproc0.epc <= $at
287
288 return:
289
290         eret # Return from exception
291
292

```

- Kết quả chạy mô phỏng:



The screenshot shows the GDB debugger interface with three main panes:

- Text Segment:** Displays assembly code for the `bai4.asm` file. The code includes instructions like `addiu`, `lui`, `ori`, `sb`, `add`, and `nop`. A tooltip for the instruction at address 0x04040000 says: "addiu \$23,\$0,0x0000... 30: li \$7, 1".
- Labels:** Shows a list of labels and their addresses, such as `main` at 0x00400004, `Loop` at 0x00400020, and `msg_keypress` at 0x10010000.
- Data Segment:** Displays memory dump data from address 0x10010000 to 0x10010400. It shows values for registers and memory locations, with columns for Address, Value (+0), Value (+4), Value (+8), Value (+C), Value (+10), Value (+14), Value (+18), and Value (+1c).

- Giải thích:

Khi đang thực hiện vòng lặp mà có tín hiệu nhấn từ ma trận Lab Sim hoặc Khoảng thời gian lập tới giới hạn thì Chương trình sẽ thực hiện ngắn.

Đầu tiên chương trình thực hiện so sánh và tìm ra nguyên nhân ngắn, nếu là vượt quá thời gian lặp thì sẽ in ra message còn nếu là tín hiệu nhấn từ Lab Sim thì sẽ in ra ký tự được nhận (hệ cơ số 16)

MSSV: 20194530

0x41 là số 2;

0x11 là số 0;

0x21 là số 1;

0x24 là số 9;

0x12 là số 4;

0x22 là số 5;

0x41 là số 3;

0x11 là số 0;

MSSV: 20194530 → Kết quả như ảnh trên

2. Assignment 5

- Mã nguồn:

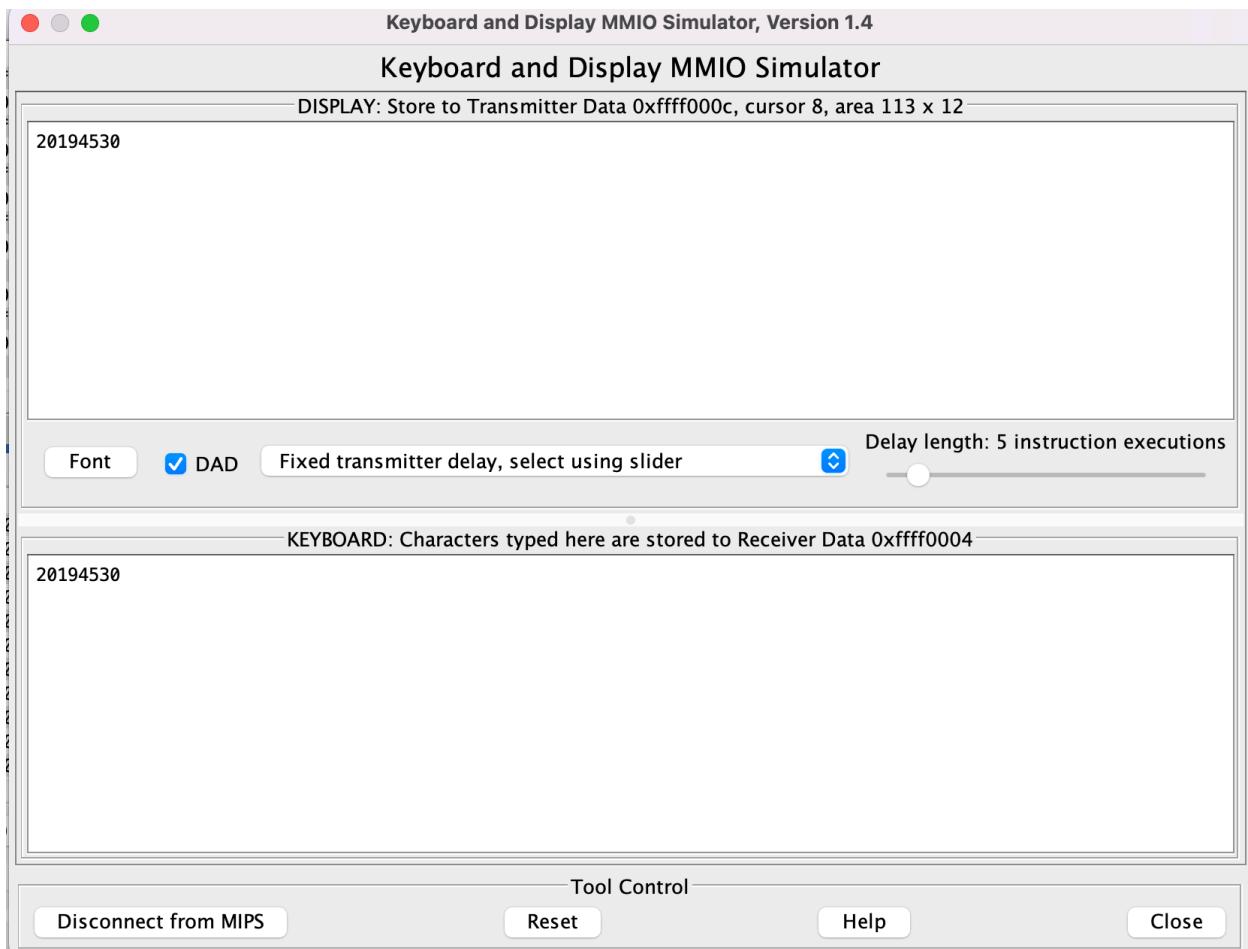
```
1 # Assignment 5
2 .eqv KEY_CODE 0xFFFF0004 # ASCII code from keyboard, 1 byte
3 .eqv KEY_READY 0xFFFF0000 # =1 if has a new keycode ?
4 # Auto clear after lw
5 .eqv DISPLAY_CODE 0xFFFF000C # ASCII code to show, 1 byte
6 .eqv DISPLAY_READY 0xFFFF0008 # =1 if the display has already to do
7 # Auto clear after sw
8 .eqv MASK_CAUSE_KEYBOARD 0x0000034 # Keyboard Cause
9
10 .text
11     li $k0, KEY_CODE
12     li $k1, KEY_READY
13
14     li $s0, DISPLAY_CODE
15     li $s1, DISPLAY_READY
16 loop:
17     nop
18 WaitForKey:
19     lw $t1, 0($k1) # $t1 = [$k1] = KEY_READY
20     beq $t1, $zero, WaitForKey # if $t1 == 0 then Polling
21 MakeIntR:
22     teqi $t1, 1 # if $t1 = 1 then raise an Interrupt
23     j loop
24 #
25 # Interrupt subroutine
26 #
27 .ktext 0x80000180
28 get_caus:
29     mfc0 $t1, $13 # $t1 = Coproc0.cause
30 IsCount:
31     li $t2, MASK_CAUSE_KEYBOARD# if Cause value confirm Keyboard..
32     and $at, $t1,$t2
33     beq $at,$t2, Counter_Keyboard
34     j end process
```

```

55
34         ~~~ +---+---, Counter_Keyboard
35         j end_process
36
36 Counter_Keyboard:
37 ReadKey:
38     lw $t0, 0($k0) # $t0 = [$k0] = KEY_CODE
39
40 WaitForDis:
41     lw $t2, 0($s1) # $t2 = [$s1] = DISPLAY_READY
42     beq $t2, $zero, WaitForDis # if $t2 == 0 then Polling
43
44
45 ShowKey:
46     sw $t0, 0($s0) # show key
47     nop
48
49 end_process:
50
51 next_pc:
52     mfc0 $at, $14 # $at <= Coproc0.$14 = Coproc0.epc
53     addi $at, $at, 4 # $at = $at + 4 (next instruction)
54     mtc0 $at, $14 # Coproc0.$14 = Coproc0.epc <= $at
55 return:
56     eret # Return from exception
57

```

- Kết quả chạy mô phỏng:



Tool Control

Disconnect from MIPS Reset Help Close

Text Segment

Bkpt	Address	Code	Basic	Source	11:	12:	13:	14:	15:	16:	17:	18:	19:	20:	21:	22:	23:	24:	25:	26:	27:	28:	29:	30:	31:	32:	33:	34:	35:	36:	37:	38:	39:	40:	41:	42:	43:	44:	45:	46:	47:	48:	49:	50:	51:	52:	53:	54:	55:	56:	57:	58:	59:	60:	61:	62:	63:	64:	65:	66:	67:	68:	69:	70:	71:	72:	73:	74:	75:	76:	77:	78:	79:	80:	81:	82:	83:	84:	85:	86:	87:	88:	89:	90:	91:	92:	93:	94:	95:	96:	97:	98:	99:	100:	101:	102:	103:	104:	105:	106:	107:	108:	109:	110:	111:	112:	113:	114:	115:	116:	117:	118:	119:	120:	121:	122:	123:	124:	125:	126:	127:	128:	129:	130:	131:	132:	133:	134:	135:	136:	137:	138:	139:	140:	141:	142:	143:	144:	145:	146:	147:	148:	149:	150:	151:	152:	153:	154:	155:	156:	157:	158:	159:	160:	161:	162:	163:	164:	165:	166:	167:	168:	169:	170:	171:	172:	173:	174:	175:	176:	177:	178:	179:	180:	181:	182:	183:	184:	185:	186:	187:	188:	189:	190:	191:	192:	193:	194:	195:	196:	197:	198:	199:	200:	201:	202:	203:	204:	205:	206:	207:	208:	209:	210:	211:	212:	213:	214:	215:	216:	217:	218:	219:	220:	221:	222:	223:	224:	225:	226:	227:	228:	229:	230:	231:	232:	233:	234:	235:	236:	237:	238:	239:	240:	241:	242:	243:	244:	245:	246:	247:	248:	249:	250:	251:	252:	253:	254:	255:	256:	257:	258:	259:	260:	261:	262:	263:	264:	265:	266:	267:	268:	269:	270:	271:	272:	273:	274:	275:	276:	277:	278:	279:	280:	281:	282:	283:	284:	285:	286:	287:	288:	289:	290:	291:	292:	293:	294:	295:	296:	297:	298:	299:	300:	301:	302:	303:	304:	305:	306:	307:	308:	309:	310:	311:	312:	313:	314:	315:	316:	317:	318:	319:	320:	321:	322:	323:	324:	325:	326:	327:	328:	329:	330:	331:	332:	333:	334:	335:	336:	337:	338:	339:	340:	341:	342:	343:	344:	345:	346:	347:	348:	349:	350:	351:	352:	353:	354:	355:	356:	357:	358:	359:	360:	361:	362:	363:	364:	365:	366:	367:	368:	369:	370:	371:	372:	373:	374:	375:	376:	377:	378:	379:	380:	381:	382:	383:	384:	385:	386:	387:	388:	389:	390:	391:	392:	393:	394:	395:	396:	397:	398:	399:	400:	401:	402:	403:	404:	405:	406:	407:	408:	409:	410:	411:	412:	413:	414:	415:	416:	417:	418:	419:	420:	421:	422:	423:	424:	425:	426:	427:	428:	429:	430:	431:	432:	433:	434:	435:	436:	437:	438:	439:	440:	441:	442:	443:	444:	445:	446:	447:	448:	449:	450:	451:	452:	453:	454:	455:	456:	457:	458:	459:	460:	461:	462:	463:	464:	465:	466:	467:	468:	469:	470:	471:	472:	473:	474:	475:	476:	477:	478:	479:	480:	481:	482:	483:	484:	485:	486:	487:	488:	489:	490:	491:	492:	493:	494:	495:	496:	497:	498:	499:	500:	501:	502:	503:	504:	505:	506:	507:	508:	509:	510:	511:	512:	513:	514:	515:	516:	517:	518:	519:	520:	521:	522:	523:	524:	525:	526:	527:	528:	529:	530:	531:	532:	533:	534:	535:	536:	537:	538:	539:	540:	541:	542:	543:	544:	545:	546:	547:	548:	549:	550:	551:	552:	553:	554:	555:	556:	557:	558:	559:	560:	561:	562:	563:	564:	565:	566:	567:	568:	569:	570:	571:	572:	573:	574:	575:	576:	577:	578:	579:	580:	581:	582:	583:	584:	585:	586:	587:	588:	589:	590:	591:	592:	593:	594:	595:	596:	597:	598:	599:	600:	601:	602:	603:	604:	605:	606:	607:	608:	609:	610:	611:	612:	613:	614:	615:	616:	617:	618:	619:	620:	621:	622:	623:	624:	625:	626:	627:	628:	629:	630:	631:	632:	633:	634:	635:	636:	637:	638:	639:	640:	641:	642:	643:	644:	645:	646:	647:	648:	649:	650:	651:	652:	653:	654:	655:	656:	657:	658:	659:	660:	661:	662:	663:	664:	665:	666:	667:	668:	669:	670:	671:	672:	673:	674:	675:	676:	677:	678:	679:	680:	681:	682:	683:	684:	685:	686:	687:	688:	689:	690:	691:	692:	693:	694:	695:	696:	697:	698:	699:	700:	701:	702:	703:	704:	705:	706:	707:	708:	709:	710:	711:	712:	713:	714:	715:	716:	717:	718:	719:	720:	721:	722:	723:	724:	725:	726:	727:	728:	729:	730:	731:	732:	733:	734:	735:	736:	737:	738:	739:	740:	741:	742:	743:	744:	745:	746:	747:	748:	749:	750:	751:	752:	753:	754:	755:	756:	757:	758:	759:	760:	761:	762:	763:	764:	765:	766:	767:	768:	769:	770:	771:	772:	773:	774:	775:	776:	777:	778:	779:	780:	781:	782:	783:	784:	785:	786:	787:	788:	789:	790:	791:	792:	793:	794:	795:	796:	797:	798:	799:	800:	801:	802:	803:	804:	805:	806:	807:	808:	809:	810:	811:	812:	813:	814:	815:	816:	817:	818:	819:	820:	821:	822:	823:	824:	825:	826:	827:	828:	829:	830:	831:	832:	833:	834:	835:	836:	837:	838:	839:	840:	841:	842:	843:	844:	845:	846:	847:	848:	849:	850:	851:	852:	853:	854:	855:	856:	857:	858:	859:	860:	861:	862:	863:	864:	865:	866:	867:	868:	869:	870:	871:	872:	873:	874:	875:	876:	877:	878:	879:	880:	881:	882:	883:	884:	885:	886:	887:	888:	889:	890:	891:	892:	893:	894:	895:	896:	897:	898:	899:	900:	901:	902:	903:	904:	905:	906:	907:	908:	909:	910:	911:	912:	913:	914:	915:	916:	917:	918:	919:	920:	921:	922:	923:	924:	925:	926:	927:	928:	929:	930:	931:	932:	933:	934:	935:	936:	937:	938:	939:	940:	941:	942:	943:	944:	945:	946:	947:	948:	949:	950:	951:	952:	953:	954:	955:	956:	957:	958:	959:	960:	961:	962:	963:	964:	965:	966:	967:	968:	969:	970:	971:	972:	973:	974:	975:	976:	977:	978:	979:	980:	981:	982:	983:	984:	985:	986:	987:	988:	989:	990:	991:	992:	993:	994:	995:	996:	997:	998:	999:	1000:	1001:	1002:	1003:	1004:	1005:	1006:	1007:	1008:	1009:	1010:	1011:	1012:	1013:	1014:	1015:	1016:	1017:	1018:	1019:	1020:	1021:	1022:	1023:	1024:	1025:	1026:	1027:	1028:	1029:	1030:	1031:	1032:	1033:	1034:	1035:	1036:	1037:	1038:	1039:	1040:	1041:	1042:	1043:	1044:	1045:	1046:	1047:	1048:	1049:	1050:	1051:	1052:	1053:	1054:	1055:	1056:	1057:	1058:	1059:	1060:	1061:	1062:	1063:	1064:	1065:	1066:	1067:	1068:	1069:	1070:	1071:	1072:	1073:	1074:	1075:	1076:	1077:	1078:	1079:	1080:	1081:	1082:	1083:	1084:	1085:	1086:	1087:	1088:	1089:	1090:	1091:	1092:	1093:	1094:	1095:	1096:	1097:	1098:	1099:	1100:	1101:	1102:	1103:	1104:	1105:	1106:	1107:	1108:	1109:	1110:	1111:	1112:	1113:	1114:	1115:	1116:	1117:	1118:	1119:	1120:	1121:	1122:	1123:	1124:	1125:	1126:	1127:	1128:	1129:	1130:	1131:	1132:	1133:	1134:	1135:	1136:	1137:	1138:	1139:	1140:	1141:	1142:	1143:	1144:	1145:	1146:	1147:	1148:	1149:	1150:	1151:	1152:	1153:	1154:	1155:	1156:	1157:	1158:	1159:	1160:	1161:	1162:	1163:	1164:	1165:	1166:	1167:	1168:	1169:	1170:	1171:	1172:	1173:	1174:	1175:	1176:	1177:	1178:	1179:	1180:	1181:	1182:	1183:	1184:	1185:	1186:	1187:	1188:	1189:	1190:	1191:	1192:	1193:	1194:	1195:	1196:	1197:	1198:	1199:	1200:	1201:	1202:	1203:	1204:	1205:	1206:	1207:	1208:	1209:	1210:	1211:	1212:	1213:	1214:	1215:	1216:	1217:	1218:	1219:	1220:	1221:	1222:	1223:	1224:	1225:	1226:	1227:	1228:	1229:	1230:	1231:	1232:	1233:	1234:	1235:	1236:	1237:	1238:	1239:	1240:	1241:	1242:	1243:	1244:	1245:	1246:	1247:	1248:	1249:	1250:	1251:	1252:	1253:	1254:	1255:	1256:	1257:	1258:	1259:	1260:	1261:	1262:	1263:	1264:	1265:	1266:	1267:	1268:	1269:	1270:	1271:	1272:	1273:	1274:	1275:	1276:	1277:	1278:	1279:	1280:	1281:	1282:	1283:	1284:	1285:	1286:	1287:	1288:	1289:	1290:	1291:	1292:	1293:	1294:	1295:	1296:	1297:	1298:	1299:	1300:	1301:	1302:	1303:	1304:	1305:	1306:	1307:	1308:	1309:	1310:	1311:	1312:	1313:	1314:	1315:	1316:	1317:	1318:	1319:	1320:	1321:	1322:	1323:	1324:	1325:	1326:	1327:	1328:	1329:	1330:	1331:	1332:	1333:	1334:	1335:	1336:	1337:	1338:	1339:	1340:	1341:	1342:	1343:	1344:	1345:	1346:	1347:	1348:	1349:	1350:	1351:	1352:	1353:	1354:	1355:	1356:	1357:	1358:	1359:	1360:	1361:	1362:	1363:	1364:	1365:	1366:	1367:</
------	---------	------	-------	--------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	---------

- Khi mà có một phím được nhập vào từ KEYBOARD thì biến t1 sẽ bằng 1 (KEY_READY đã sẵn sàng)
- Mà câu lệnh dòng 21 “ teqi \$t1, 1 ”: tức là khi biến t1 = 1 thì ngắt mềm sẽ được thực thi