PS2 Maus Emulator

Zuerst noch etwas grundsätzliches zum Keyboard: Jedes Keyboard sendet beim drücken (MAKE) einer Taste und beim loslassen einer Taste (Break)

jeweils einen code. Diese heissen scan codes. Heisst die Taste Shift einmal kurz gedrückt sendet den code &H12, &HF0, &H12. Hier sieht man schön das man manche Codes zerlegen muss, nur Shift drücken nutzt ja wie in Zeile 28 nichts. Also zerlegen in MAKE: &H12, dann den Buchstaben t senden, dann das BREAK für shift.

Der scancode muss an anfang die Menge der zu Übertragengen codes angeben, im fall von CTRL_ALT_entf sind es also 11

Hier eine auflistung der Scancodes, bitte beachten das diese für ein Englisches Tastaturlayout sind, z.b. kommt dann bei eingestelltem Duetschen Layout statt einem z ein Y auf dem Computer raus.

computer-engineering.org/ps2keyboard/scancodes2.html

Hier einmal als Beispiel:

Quellcode

```
1. $regfile = "m328pdef.dat" 'Der ATMEGA328P wird genutzt
2. $crystal = 8000000 'Es werden die internen 8MHz genutzt
3. $hwstack = 150
4. $swstack = 150
5. $framesize = 150
10. $lib "at_emulator.lib"
11. Config Atemu = Int1 , Data = Pind.3 , Clock = Pind.7
12. 'PS2 Tastatur sim config - Datapin MUSS ein INT pin sein, 2,2K pullups an Data und Clock leitung notwendig
13. Enable Interrupts
14. do
15. wait 5
16. 'schreibt nach 5 Sekunden= Test
17. Sendscankbd Taste_shift_up 'Shift make code senden
18. waitms 300
19. Sendscankbd Taste_t 'T senden
20. waitms 300
21. Sendscankbd Taste_shift_down 'Shift BREAK code senden
22. waitms 300
23. Sendscankbd Taste_e
24. waitms 300
25. Sendscankbd Taste_s
26. waitms 300
27. Sendscankbd Taste_shift 'Hier wird weiter klein geschrieben da der BREAK direkt nach dem MAKE code gesendet
wurde
28. waitms 300
29. Sendscankbd Taste_t
30. waitms 300
31. loop
32. ' *** Scancodes ***
33. Taste_e:
34. Data 3 , &H24 , &HF0 , &H24
35. Taste_s:
36. Data 3 , &H1B , &HF0 , &H1B
37. Taste_t:
38. Data 3 , &H2C , &HF0 , &H2C
39. Taste_shift:
40. Data 3 , &H12 , &HF0 , &H12
41. Taste_shift_up:
42. Data 1 , &H12
43. Taste_shift_down:
44. Data 2 , &HF0 , &H12
45. Taste_ctrl_alt_entf:
```

55. Data 11 , &H14 , &H11 , &HE0 , &H71 , &HE0 , &HF0 , &H71 , &HF0 , &H11 , &HF0 , &H14

Alles anzeigen

Aber es geht natürlich noch mehr, wir können auch die Maus mitbenutzen, angegeben wird die Verschiebung in Pixel:
Die Funktioniert mit dem Befehl PSMOUSEXY oder einzelne Pixel mit scancodes:

Quellcode

```
1. $regfile = "m328pdef.dat" 'Der ATMEGA328P wird genutzt
2. $crystal = 8000000 'Es werden die internen 8MHz genutzt
3. $hwstack = 150
4. $swstack = 150
5. $framesize = 150
10. Dim A as Byte
11. Enable Interrupts
12. $lib "at_emulator.lib"
13. Config Atemu = Int1 , Data = Pind.3 , Clock = Pind.7
14. 'PS2 Tastatur sim config - Datapin MUSS ein INT pin sein, 2,2K pullups an Data und Clock leitung notwendig
15. CONFIG PS2EMU= int0 , DATA = PIND.2, CLOCK=pind.6
16. 'PS2 Maus sim config - Datapin MUSS ein INT pin sein, 2,2K pullups an Data und Clock leitung notwendig
17. do
20. wait 5
21. 'schreibt nach 5 Sekunden= Test
22. Sendscankbd Taste_shift_up 'Shift make code senden
23. waitms 300
24. Sendscankbd Taste_t 'T senden
25. waitms 300
26. Sendscankbd Taste_shift_down 'Shift BREAK code senden
27. waitms 300
28. Sendscankbd Taste_e
29. waitms 300
30. Sendscankbd Taste_s
31. waitms 300
32. Sendscankbd Taste_shift 'Hier wird weiter klein geschrieben da der BREAK direkt nach dem MAKE code gesendet
wurde
33. waitms 300
34. Sendscankbd Taste_t
35. waitms 300
36. wait 5
37. PS2MOUSEXY 30 , 0, 0 '30 pixel nach rechts
40. waitms 500
41. for A = 1 to 30 'jetzt mal als Scancode, 30x ein pixel nach oben
42. Sendscan Mouseup
43. Waitms 10
44. next A
46. waitms 500
47. PS2MOUSEXY -30 , 0, 0 '30 pixel nach links
48. waitms 500
49. PS2MOUSEXY 0 , -30, 0 '30 pixel nach unten
```

```

50. waitms 500
51. sendscan R_close 'Maustasten funktionieren auch als Scancode, rechte Taste gedrückt
52. waitms 500
53. sendscan R_open 'Maustasten funktionieren auch als Scancode, rechte Taste losgelassen
54. waitms 500
55. PS2MOUSEXY 0 , 0, 1 'linke Maustaste gedrückt
56. waitms 500
57. PS2MOUSEXY 0 , 0, 0 'linke Maustaste losgelassent
58. waitms 500
60. loop
63. ' *** Scancodes ***
65. Taste_e:
66. Data 3 , &H24 , &HF0 , &H24
67. Taste_s:
69. Data 3 , &H1B , &HF0 , &H1B
70. Taste_t:
72. Data 3 , &H2C , &HF0 , &H2C
73. Taste_shift:
75. Data 3 , &H12, &HF0, &H12
76. Taste_shift_up:
78. Data 1 , &H12
80. Taste_shift_down:
81. Data 2 , &HF0, &H12
82. Taste_ctrl_alt_entf:
84. Data 11 , &H14 , &H11 , &HE0 , &H71 , &HE0 , &HF0 , &H71 , &HF0 , &H11 , &HF0 , &H14
86. Mouseup:
87. Data 3 , &H08 , &H00 , &H01
88. R_open:
90. Data 3 , &H08,&H00,&H00
92. R_close:
93. Data 3 , &H0a,&H00,&H00

```

Alles anzeigen