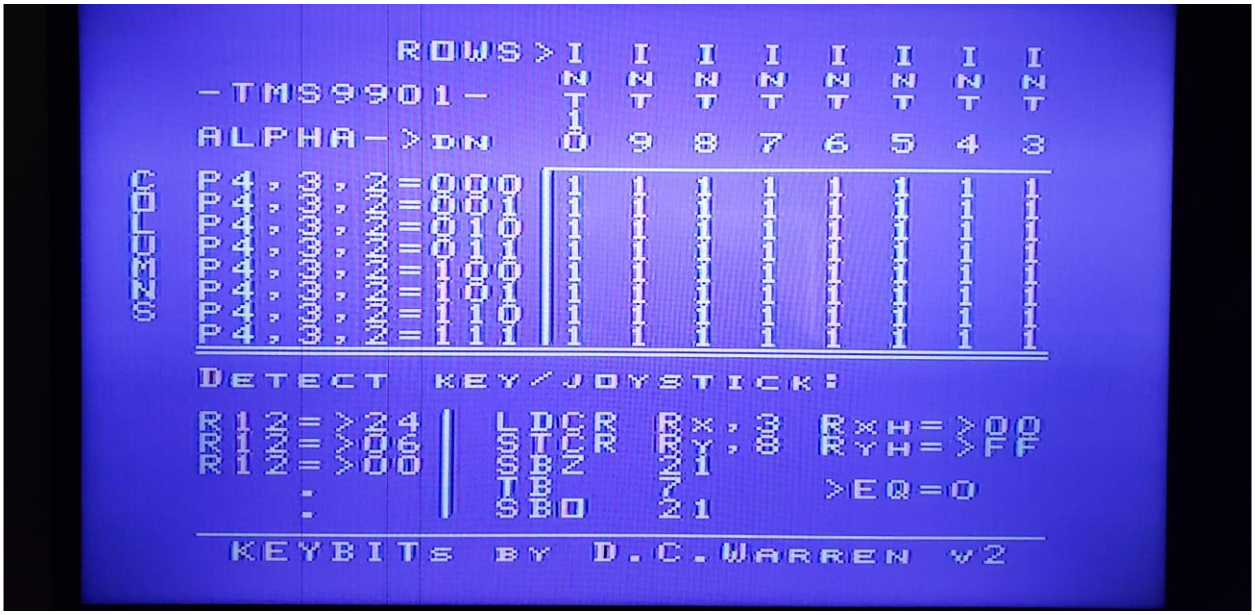
KEYBITs

KEYBITs is a utility that gives a visual on how the TMS9901 Programmable Systems Interface IC interacts with the TI-99/4A matrix keyboard. The TMS9901 is interfaced to the TMS9900 processor and provides interrupt detection for the VDP, external peripheral cards and the built in timer. It also provides an interface to the keyboard, cassette player and the sound chip audio gate through the Communications Register Unit (CRU) lines with the processor. (I highly recommend reading about the TI keyboard operation on Thierry's website nouspikel.com as an excellent informative source.)

So, why KEYBITs? The TI-99/4A has a built-in key scan routine in ROM that is available to the user for general use. That routine is the one to use if your application needs access to the entire keyboard. But, sometimes you may want to simply detect a key or two without going through the ROM routine with all of its overhead. You may want to detect a FCTN = (QUIT) directly, for example. KEYBITs can help you with that by displaying the ROW/COLUMN values to detect those two keys.

It's also fun to discover strange things the keyboard does such as the "phantom key" issue. Press FCTN, Shift, 1 all at the same time and you'll see that a forth key, the "A", is also detected even though it's not being pressed. This drove me crazy years ago when I wanted to use the FCTN-Shift-D keys in a specific application and a fourth key kept popping up. That was the genesis of the first version of KEYBITs that helped to visualize what was the keyboard was really doing at the hardware level.

Looking at the KEYBITs screen:



```
             ROWS>I  I  I  I  I  I  I  I
  -TMS9901-       N  N  N  N  N  N  N  N
                  T  T  T  T  T  T  T  T
                  1
  ALPHA->DN       0  9  8  7  6  5  4  3
C P4,3,2=000 |1  1  1  1  1  1  1  1
O P4,3,2=001 |1  1  1  1  1  1  1  1
L P4,3,2=010 |1  1  1  1  1  1  1  1
U P4,3,2=011 |1  1  1  1  1  1  1  1
M P4,3,2=100 |1  1  1  1  1  1  1  1
N P4,3,2=101 |1  1  1  1  1  1  1  1
S P4,3,2=110 |1  1  1  1  1  1  1  1
  P4,3,2=111 |1  1  1  1  1  1  1  1
DETECT   KEY/JOYSTICK:
R12=>24 | LDCR  RX,3  RXH=>00
R12=>06 | STCR  RY,8  RYH=>FF
R12=>00 | SBZ   21
   .    | TB    7     >EQ=0
   .    | SBO   21
  KEYBITs BY D.C.WARREN v2
```

ROWS at the top are the pin designations for the TMS9901 that provide the INPUTS from the keyboard matrix. They are INTERRUPTs 3 through 10. In this case they are used as inputs, not interrupts. The COLUMNS are pins P2 through P4 on the TMS9901 and are programmed as OUTPUTS by the console going to a 3x8 multiplexer which is connected to the keyboard matrix. So, to access a key or joystick you must turn on a column and then read the rows. KEYBITs tells you which column and which row is used for any particular key.

The bottom third of the screen displays what assembly code you might use to detect a key. Going back to the FCTN = example. If you press FCTN = you will see which column and rows both the FCTN key and the = key activate in the top matrix. The “Detect key/joystick:” section will only show one key at a time. So, now press FCTN only to show the column/row values; >00 for the column and >EF for the row. Code to detect FCTN could be:

```
FCTN   DATA   >EF00      ROW code that KEYBITs displays for FCTN key
       LI     R12,>24    Set the CRU base to point to the COLUMNs (P4-P2)
       CLR    R1         Load the >00 value for COLUMN 0
       LDCR   R1,3       Write it to the TMS9901 activating COLUMN 0
       LI     R12,>06    Set the CRU base to point to the ROWs (INT10-INT3)
       STCR   R2,8       Read in all eight rows into the MSByte of R2
       CB     R2,@FCTN   If FCTN is detected the EQ bit will be set here
```

For the “=” key:

```
EQ     DATA   >FE00      ROW code that KEYBITs displays for = key
       LI     R12,>24    Set the CRU base to point to the COLUMNs (P4-P2)
       CLR    R1         Load the zero value for COLUMN 0
       LDCR   R1,3       Write it to the TMS9901 activating COLUMN 0
       LI     R12,>06    Set the CRU base to point to the ROWs (INT10-INT3)
       STCR   R2,8       Read in all eight rows into the MSByte of R2
       CB     R2,@EQ     If = is detected the EQ bit will be set here
```

But, since the FCTN and = keys are in the same COLUMN you can detect them both at the same time if you’d prefer! Just check for a ROW value of >EE on COLUMN 0.

If you want to detect the Alpha key:

```
       CLR    R12        Set CRU base to zero for the TMS9901
       SBZ    21         Set the P5 line on the TMS9901 to zero (low)
       TB     7          Test ROW INT7 for the Alpha key.  EQ bit 0 if down, 1 if up
       SBO    21         Be sure to turn off the P5 line on the way out!
                           (SBO does not affect EQ bit)
```

I hope this proves useful, at least fun! Just another programmer’s tool for the tool kit.