

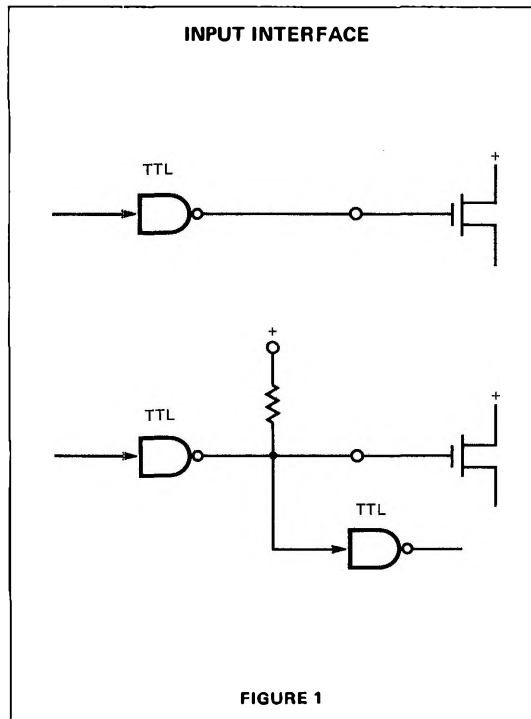
READ-ONLY MEMORY

INTRODUCTION

The 2526 is a high speed 5,184-bit Static Read-Only Memory. It may be organized as 64 x 9 x 9 for use as a character generator in dot matrix displays, or as a 512 x 9 ROM for general purpose use. It features TTL compatible inputs, three-state TTL compatible outputs, two standard supply voltages (+5, -12), output data latches, and less than 700ns access time. The 2526 is fabricated using Signetics P-MOS silicon gate process and is packaged in a 24-pin silicone dual in-line package.

INPUT CIRCUITS

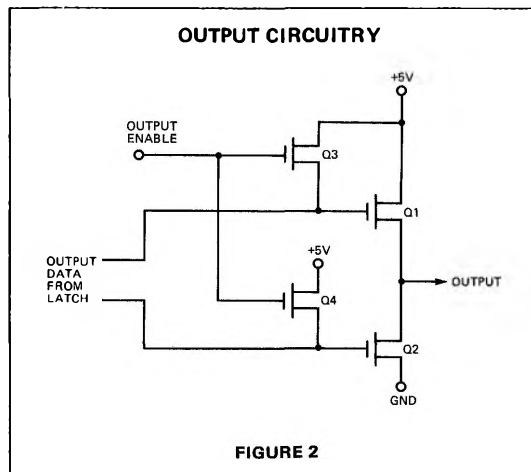
The inputs to the 2526 use a configuration similar to that used in most of the other 2500 series products. Interface requirements with TTL circuitry are described in detail in the Signetics MOS Handbook. In general, a standard TTL gate driving only the 2526 does not require any interfacing resistors. See Figure 1. If another TTL gate is driven in addition to the 2526, a 10k ohm pull-up resistor will improve the input noise margins.

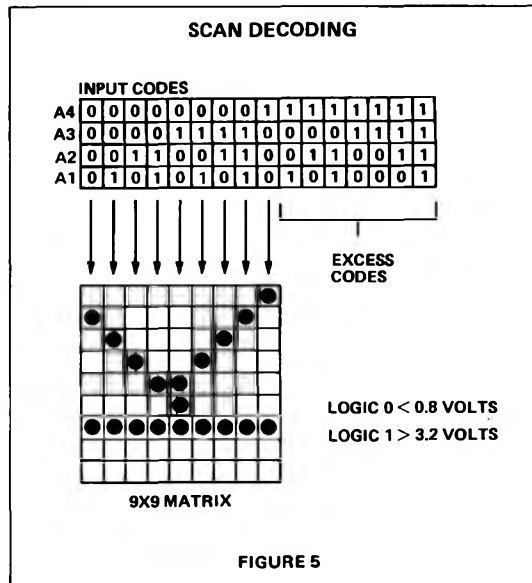
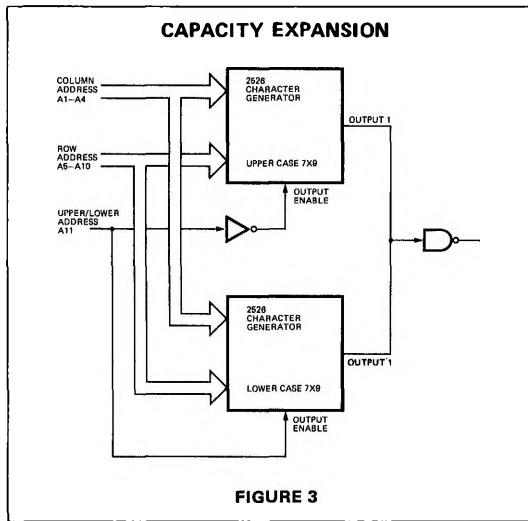


OUTPUT CIRCUITS

The outputs from the 2526 use a three-state push-pull configuration that allows wired-OR connection of several circuits for expanded capacity. The push-pull circuitry provides low impedance outputs for both high and low output voltages. See Figure 2. For a low output voltage Q₂ is turned ON and Q₁ is turned OFF with Q₃ and Q₄ kept OFF by a high Output Enable voltage. For a high output voltage Q₁ is turned ON and Q₂ is turned OFF while Q₃ and Q₄ are OFF. When the Output Enable voltage goes low, however, both Q₃ and Q₄ are turned ON, keeping Q₁ and Q₂ both OFF for any condition the output latch assumes. In this state the output of the 2526 is essentially floating, allowing other circuits to dominate the output line.

Figure 3 shows one way to make use of this three-state output. Two 2526 Character Generators are tied together at their outputs and fed to the receiving logic circuitry, e.g., a parallel to serial converter. One 2526 can contain the dot matrix information for upper case characters and the other can contain the lower case information, thus providing a full 128 character set. All inputs for the two generators are tied in parallel except the Output Enables which serve as A11 address inputs. One 2526 receives the A11 signal and the other receives $\overline{A11}$. In this way only one set of outputs at a time will activate the output lines. If the system configuration requires periods where neither output is active, that fact can be gated with A11 to turn off both Output Enable signals. To reduce power dissipation, the A11 information can be gated with the READ signals to avoid turning on the unused 2526.



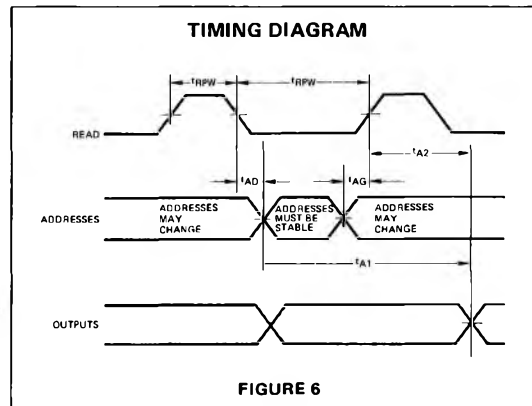
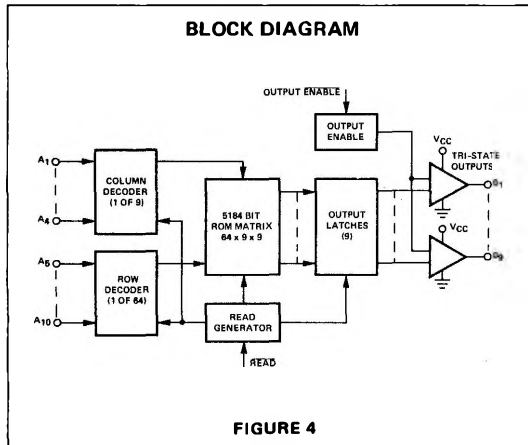


ADDRESS DECODING

The Signetics 2526 Character Generator is organized to provide 64 character locations with each location described by a 9 x 9 matrix of bits. The block diagram in Figure 4 shows the address assignments for the character and scan functions. The six address inputs A5 through A10 are decoded directly to provide a 1-of-64 character selection. The four address inputs A1 through A4 are decoded to provide a 1-of-9 selection of scans within each character. Since four address lines can generate 16 scan selections instead of only 9, there are seven excess codes. See Figure 5. The 1-of-9 scan decoder forces the excess input codes to generate all logic "1's" at the output latches. The address decoding circuits are only activated during a READ operation in order to save power when the memory is not being used.

TIMING

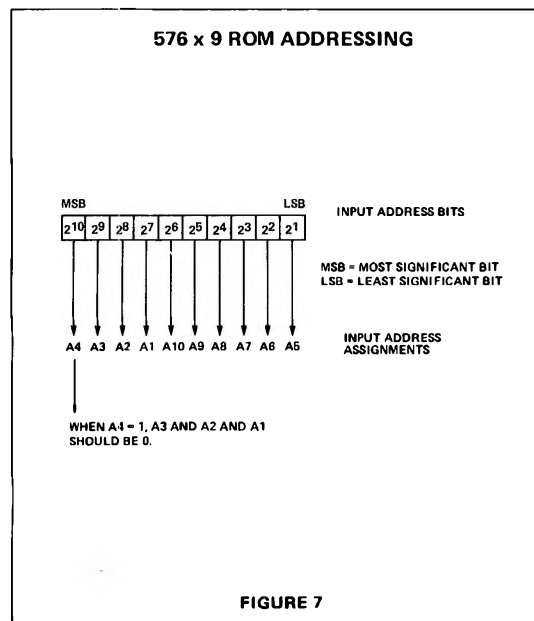
The timing diagram in Figure 6 shows how the READ signal controls the operation of the memory. The address inputs propagate through the decoders and the bit matrix when READ goes low. The output data are strobed into the latches when READ goes high. The state of the OUTPUT ENABLE signal determines whether or not the latched data are transferred to the outputs. With OUTPUT ENABLE high, the worst case access time from stable addresses to valid output data is 700ns. Notice that addresses must be stable for only a short period of time so that address changes may be made in parallel with the access operations. Once the data are set into the latches, they remain stable for a full READ cycle until the next cycle's data are available.



MEMORY ORGANIZATION

The 2526 is intended primarily for use as a 7 x 9 dot matrix character generator, and the address decoding scheme reflects this purpose. Address lines A1 through A4 (see Figure 5) generate the required nine scan selects plus seven excess codes. Thus, an attempt to use the 2526 with all ten address lines in a 1024 x 9 configuration would fail because there are only 5,184 bits in the memory and the excess input address codes would not be able to generate relevant output data. However, if address input A4 is tied to a logic "0", the excess codes are eliminated. The remaining nine address lines may then be used to address a 512 x 9 ROM. The ninth scan in each character is ignored along with the excess codes and a subset of 4,608 bits is used to provide the 512 x 9 capacity.

Other organizations are possible, of course, as long as the total memory capacity is not exceeded. 576 x 9 is the real capacity of the memory (576 x 9 = 5,184). The extra 64 x 9 (576 x 9 - 512 x 9 = 64 x 9) can be accessed by careful use of address A4. See Figure 7. The critical condition is stated in the figure: when A4 is logical "1", A0, A2, and A1 should be logical "0's". When A4 is logical "0", any code is allowed in the remaining nine bits. When some form of counter is used to generate the address inputs, it will often be convenient to assign the most significant bit (MSB) to A4 and the six least significant bits to A5 through A10. In this way, the highest allowed bit configuration will correspond to a binary count of 575 and the forced zero states of A3, A2, and A1 will be easier to implement.



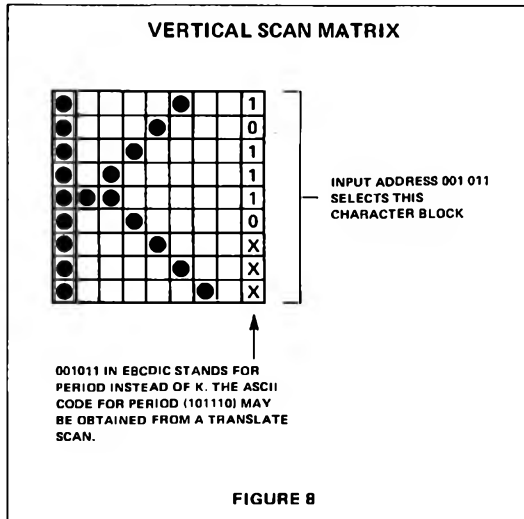
CHARACTER ORGANIZATION

When used as a 7 x 9 dot matrix character generator, the 9 x 9 dot configuration of each character allows the 2526 to be used with either vertical or horizontal scanning techniques. Figure 5 shows a 7 x 9 configuration for the letter K that is oriented for use with a horizontal scan. As each horizontal slice through the character is extracted from the ROM, the two extra bits may be ignored and the seven remaining bits serially shifted to control the dot formation.

Figure 8 shows the letter K oriented within the 9 x 9 matrix for use with a vertical scan. Each vertical slice through the character is extracted from the ROM and then serially shifted to control the dot formation. Two complete scans are not used for dots and may supply blank spaces between characters or may be ignored. Alternatively, those extra scan positions may be put to good use for translating character codes. When a code translation is desired, the column address (A1 through A4) is set to the appropriate translate scan instead of one of the dot matrix scans, and the code to be translated forms the row address (A5 through A10). The dot matrix contents of that character location are not related to the input code, but the output from the translate scan provides the desired new code.

Assume that the dot matrix letter K in Figure 8 is placed in the character array at an address corresponding to the ASCII-6 code for K (001011). Then the dots for K can only be retrieved by using the proper ASCII code as an address. The same code pattern in EBCDIC, however, stands for the period. To perform an EBCDIC to ASCII translation it is only necessary to insert the ASCII code for the period (101110) in the translate scan of the K character position. This code can then be used directly for any purpose or it can, in turn, be applied as an input to select the dot matrix for the period.

The spare bits in the 9 x 9 matrix of each character are most convenient to use for translations when the matrix is arranged for vertical scans. In that way a single read operation can perform the translation. A 7 x 9 vertical matrix leaves two spare scans for translations so that a two-way translation between two codes is possible, or two source codes can be translated into a single target code. The spare bits in the horizontal scan case are only available two at a time, so are more awkward to use for translations. In either case, the spare bits can be used to expand the character dot matrix from 7 x 9 to 9 x 9. Several special characters can be constructed (e.g., arrows) and some augmented standard characters (e.g., %) can be more legible.



APPLICATION AREAS

There are many places where a nine bit wide ROM can be useful. A nine bit output from a function look-up table can provide an extra degree of accuracy. A sine function table, for example, could supply an added bit of resolution in the result. For arithmetic tables with 8-bit operands, the ninth output bit can be used as a sign bit or a carry bit for increased flexibility. Many byte-plus-parity systems are organized around 9-bit data paths and some of their memory requirements can only be satisfied with a 9-bit ROM. 9-bit or 18-bit minicomputers often need Read Only instruction storage for bootstrap loaders and other non-volatile routines.

The added sophistication of new CRT terminal designs is making 7 x 9 characters more and more popular. Increased legibility, decreased errors and better lower case characters are the immediate advantages. The 2526 provides an economical, easy-to-use approach for implementing 7 x 9 character graphics.