TMS320 DSP
Number 17
DESIGNER'S NOTEBOOK
TEXAS INSTRUMENTS

Designing with TMS320C40 Comm Ports: Part 1

Contributed by Keith Larson and Jim Patterson

Design Problem What are some design issues/tips when designing with 'C40 comm ports?

Solution The 'C40 comm port is a very-high-speed data transmission circuit. Its speed and the close proximity of multiple data lines create special challenges. The following caveats are intended to help you past some of the potential problem areas.

First a question and answer. In the 'C40 User's Guide (pp. 14-33–14-35), it says that CSTRB is an output before CREQ goes high, but it also says CSTRB is an input after CREQ goes high. The timing data implies that two outputs are connected together for 0.5 p. The answer is that while both 'C40s are driving these lines for a period of time, they are both driving in the same direction (V_{oh}). As a result, there is no current from one device to the other.

Signal Quality

The transmission line aspects of the comm port circuit make it sensitive to signal quality. General design rules that would be applicable to high-speed (<10 ns) memory interface design would be appropriate for 'C40 comm port interconnections.

Further points to keep in mind include:

1) An overlap feature is built into $\overline{\text{CREQ}}$, $\overline{\text{CSTRB}}$, and $\overline{\text{CRDY}}$ when a token is transferred. This overlap will cause these signals to all drive high (at both ends), ensuring that neither end is susceptible to floating or low-noise signals.

   It is important to match the clocks or else the original driver may not give up his end soon enough, which causes bus contention.

2) When the token exchange occurs, the falling edge of $\overline{\text{CACK}}$ indicates that there are no transfers in progress, so it is ok to drive both ends of $\overline{\text{CSTRB}}$ high (1) Figure 14-23.

3) The requester then acknowledges the receipt of $\overline{\text{CACK}}$ low by driving $\overline{\text{CREQ}}$ high and staying active high (3) Figure 14-23.

4) $\overline{\text{CREQ}}$ going high is interesting because of (5) Figure 14-24. In this case, the rising edge of $\overline{\text{CREQ}}$ causes the $\overline{\text{CREQ}}$ input to switch over to an active output high. At this time, both devices are driving $\overline{\text{CREQ}}$ high. The rising edge of $\overline{\text{CREQ}}$ also causes $\overline{\text{CACK}}$ and $\overline{\text{CSTRB}}$ to change to inputs, also with only a couple of gate delays.

5) Finally $\overline{\text{CACK}}$, which is now floating, is driven active high (4) Figure 14-23.

VERY IMPORTANT

6) The clocks of the two 'C40's connected together must be within a 2:1 ratio. If this is not adhered to, the overlap will last too long and the new master (the one with the faster clock) may start driving low before the old master has relinquished that line. This will cause signal contention and possibly a lot of current.

Design Hints

- Use series resistors in all lines. This helps match the output buffer impedance to line impedance, protects against signal contention, and has low power dissipation. If the line length is small (6"), a single resistor in the middle can be used. The resistor value, plus buffer output impedance, should match the line impedance. The buffer output impedance is in the range of 20 to 70 ohms. A resistor value of 27 – 33 ohms may be a reasonable start. Some experimentation may be needed.

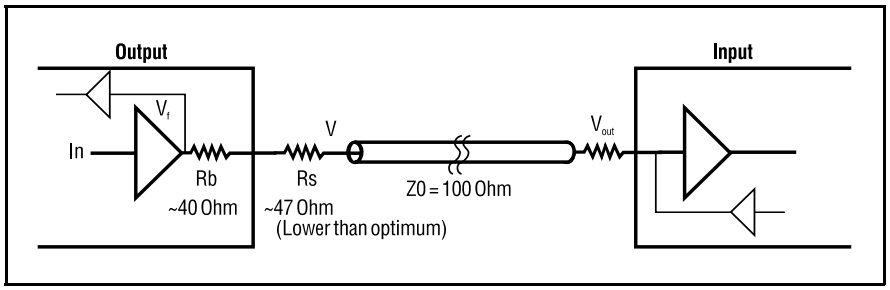

Figure 1.

- Try to keep the line impedance as high as possible. Routing signals on a top layer without a shield above them will help yield both clean signals and high impedance. Do not route signals on top of each other. When it is necessary to cross traces on adjacent layers, they should cross as right angles to reduce coupling. High line impedance will reduce the sensitivity of the circuit to changes in the output buffer impedance, and will be a benefit when interfacing to external cables, as typical ribbon cable is about 100 ohms.
- Because it is sometimes difficult to route high-impedance lines (especially long ones) in a circuit board, an external ribbon cable can be used to jump over the length of the board. In this case, only two headers need to be installed in the circuit board. Use an alternating signal and ground scheme. For quality signals a (4 control + 8 data + 1 shield) $\times$ 2=26-wire ribbon is needed. The shield is needed for the signal that is otherwise on the edge.
- The driver output consists of three transistors, one pullup to V_{CC} and two pulldowns. The DV_{ss} transistor (Q2) is on above 1.8 volts and the CV_{ss} transistor (Q3) below. The advantage of a two-transistor pulldown is two-fold. First, a major portion of the switching noise in Q2 is dumped into DV_{SS} and does not corrupt the clean logic CV_{SS}. Second, the ratio of Q2/Q3 and the 1.8-V switching threshold provides a nearly-ideal driving signal for a wide range of transmission line impedances. Note the Ron values. They are quite low, and if a fault occurs something will get HOT!
- If long lengths are needed or jumps to other boards are needed, then a unidirectional data flow should be considered. As there is currently no preferred

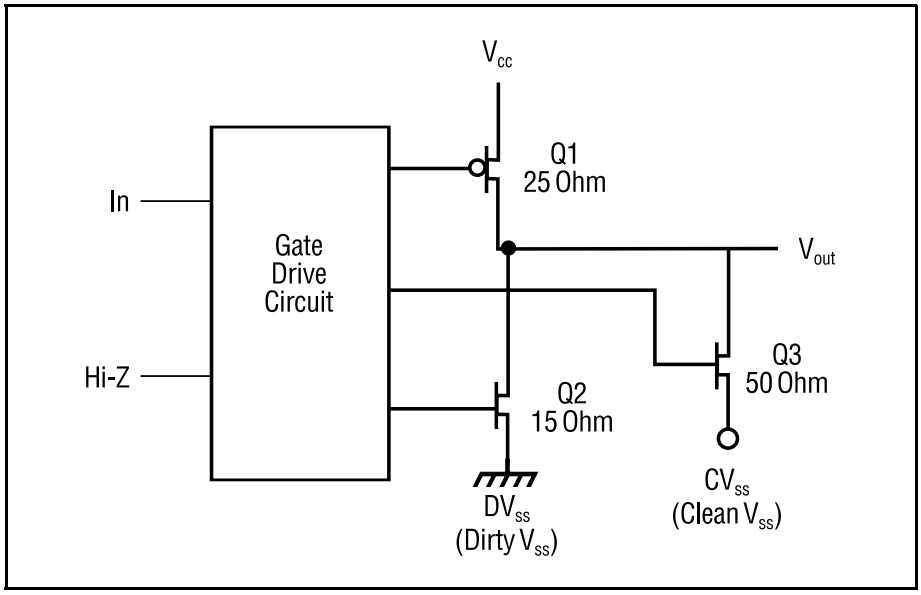

Figure 2.

method of buffering the token for bi-directional buffers. The best method is to use slow buffers with hysterisis for $\overline{\text{CSTRB}}$ and $\overline{\text{CRDY}}$. This has two advantages. It cleans up the signals and helps eliminate glitches which can be erroneously perceived as valid control. It also allows the data bits to settle before the receiver sees $\overline{\text{CSTRB}}$.

CAUTION:
The width of $\overline{\text{CSTRB}}$ low should not exceed 1.0 H at the receiving end. If it does, the byte sequencer, which has looped back to byte zero, will see this low and recognize $\overline{\text{CSTRB}}$ as the next valid byte. This is not a problem unless you are working with very long distances. In this case, use flip-flops to locally shorten $\overline{\text{CSTRB}}$ at the receiver while returning a valid $\overline{\text{CRDY}}$ width to the sender. Wide widths at the sender are not a problem. See Figure 3 for an example.

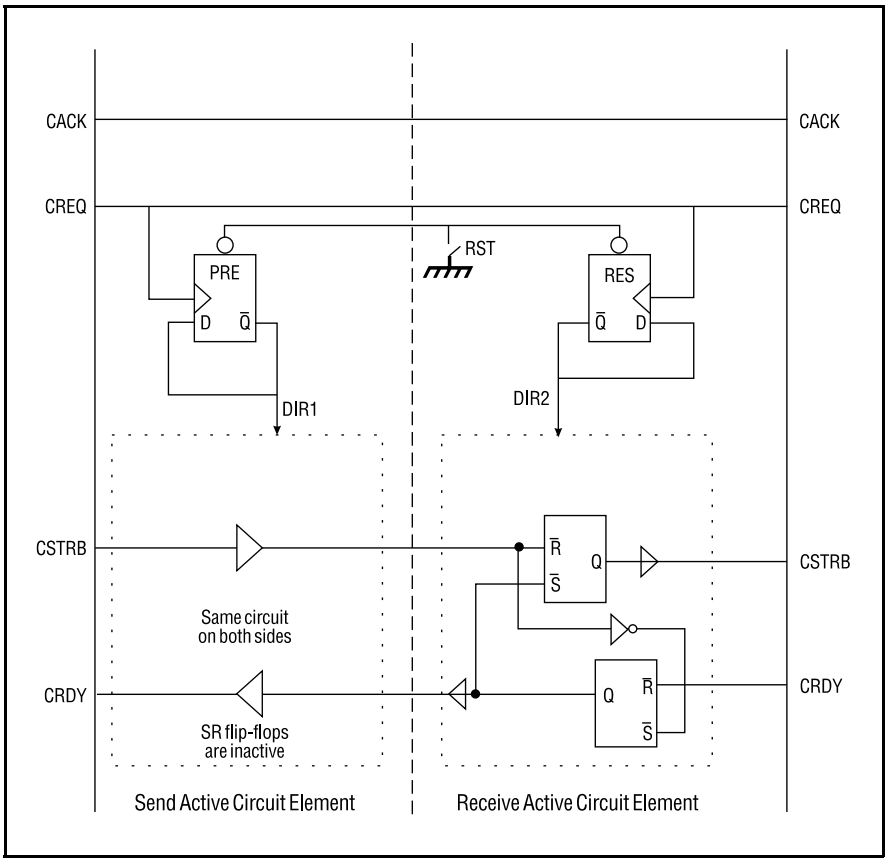

Figure 3.

CSTRB Circuit With Token Direction Detection

Because all signals are bi-directional, it is difficult to determine the direction of data transfer. A method which has been shown to work is given in Figure 3. In this case, the rising edge of $\overline{\text{CREQ}}$ is used to toggle the previous value in a flip-flop giving direction. The inital state is determined by reset at power up.

Once direction is known, controlling the width of $\overline{\text{CSTRB}}$ is straightforward. Looking at the circuit you will notice that in one direction only CSTRB/CRDY buffering is done at one end and a pair of SR flips are in the circuit at the other.

For the data receive end (with SRs), a low incoming CSTRB will cause the 'C40's pin to go low and stay low until the 'C40 responds with $\overline{\text{CRDY}}$ low. When $\overline{\text{CRDY}}$ falls, the 'C40's $\overline{\text{CSTRB}}$ (local) will go high, satisfying the 1-H criteria. When $\overline{\text{CSTRB}}$ (incoming) goes back high, the SR flip pair is ready to receive another $\overline{\text{CSTRB}}$.

Conclusion

For distances less than 12", series resistor matching is reliable so long as the design guidelines described above are adhered to. For distances greater than 12", a uni-directional transfer has been shown to be reliable when all signals are properly buffered. The width of $\overline{\text{CSTRB}}$ is important and for very long distances may need to be controlled by external logic.