TC MODULE CONCEPTS

The TC module is used to merge the address and data which is sent to the memory storage modules during a write and provides the path to send the address to the storage modules on a memory read reference. It is important to keep in mind that the TC module, like all the T series modules, is a processor orientated module. The Q and S series modules are memory quadrant orientated. The TC receives a 24 bit address packet from the processor TA module which is followed by three 24 bit data packets in the next three clock periods. The address packet can arrive at the TC in any clock period and can be intended for any memory quadrant. The TC module guides the address and data to the intended memory quadrant and sends it off in the proper phase time for the TC's associated processor.

The TC module receives control from the TA module that tells the TC which memory quadrant the reference is for. This allows the TC to direct the incoming address and data along a path to the desired quadrant. Each processor has a specific clock period, or phase time, in which it can access a quadrant. The TC places incoming address and data packets into a delay network at a point such that the first packet will leave for the memory quadrant in the proper phase time. The amount of delay on the TC depends on the phase time in which the address packet arrives and on the memory quadrant that the reference is to.

If there is a conflict in the quadrant associated with an address packet entering the TC module, the TC module will save the address and data packets until the conflict is resolved. The TC can buffer the address and data for two references to each quadrant. The address and data are saved in two recirculating buffer chains called primary and secondary buffers. The buffers each hold four 24 bit packets. The primary buffer is used for the first conflict and the secondary is used in the case of a second conflict. The primary buffer is also used as a delay network for a reference if there are no quadrant conflicts.