7 Series FPGAs Memory Resources

Block Rams
- The block RAM in the FPGA can store 36 kBits of data as 2x 16KB RAMS or one 36kB ram
  - Reads and writes are synchronous operations
  - Two adjacent block RAMS can be combined to one deeper 64K x1 memory without any extra logic
- Blocks rams are placed in columns and can be cascaded for enable deeper and wider memory implementations
- Has two independent access ports and data can be written or read from either port, each is synchronous and has its own address, data in data out, clock and write enable. These are synchronous writes and require a clock edge
  - Conflicting writes can’t cause any hardware damage when timing two blocks because no monitor for arbitration for identical addresses on both ports.
- Read/write both one block edge. When using an output register read takes one extra cycle for latency
- Conflict Avoidance: Both ports can access any memory location at any time. V7 is a true dual port RAM.
  - Two ports can have a sync or async clock (frequency/phase different for two ports)
- Can powergate unused block rams at 18Kb granularity, not initialized during configuration, works by disabling internal operations.
- Block RAM can be constrained for placement
- Applications: Larger Ram Structures can be created with special routing provided by the column RAMS.

Built-in FIFO Support
- Building FIFOs in using block RAMs is a common technique done, so V7 has logic to enable the creation of sync/async FIFOs. This way we don’t waste any CLB space in creating a counter, comparator, etc.
- The capacity of a 18kB FIFO is 4k entries by 4 bits and double for 36kb FOFO but 8k entries by 4 bits.
- Using FWFT (first-word fall-through) you get +1 entries by 4 bits
- Implementation can be done with dual clock FIFO which is free running write and read clocks and has the advantage of avoiding any ambiguity when two frequencies are unrelated
- Can also be implemented with synchronous FIFO the behavior of the flags is not predictable after the first write. Need to synchronize the reset to try and mitigate some of them.
- FIFOs can also be cascaded to increase the depth
- When placing block RAM and FIFO primitives in the same location, the FIFO must occupy the lower port.
Built-in Error Correction

- There is a built-in Hamming code error correction that uses the extra 8-bits in a 72-bit wide RAm and is done transparently.
- The error bits are generated during the write operation and placed with the 64-bit data into memory.
- During read, 72 bits of data (64 bits of data and 8 for parity) are read from memory into the ECC decoder which generates status bits to indicate no error, single bit error, and multi-bit error.
- They are used during each read to correct single-bit error.
- The parity bits are written into memory and write out to the FPGA at each rising edge.
- ECC is built into the 36kB RAM, and is supported for the FIFO as well but does not output to the FPGA the address location being read.
- To use the ECC mode, you just need to set it to be true with parity supported/not.