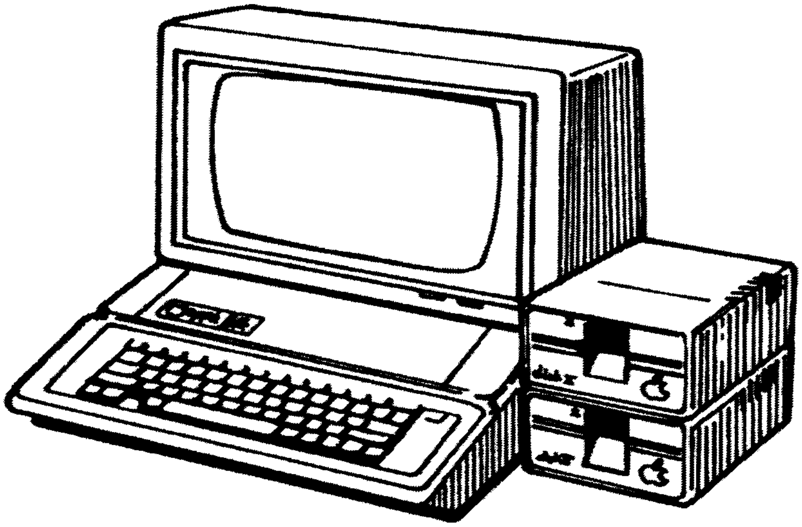
Apple][Computer Information

IWM Floppy Disk I/O Controller Info

Misc. IWM Operation Notes with handwritten Corrections

Bob Bailey -- Apple Computer, Inc. -- November 29, 1983



SOURCE
Brutal Deluxe Software web site -- www.brutal-deluxe.fr
31 December 2008

Commands

Commands generate control line pulses which are only as wide as DEV. These control reading and writing of registers. They are decoded by L6, L7, and DMOTOR-ON. DMOTOR-ON includes the 1 second timer when enabled. When L6 or L7 are changed the decoder uses the new state. Thus, during a DEV the address A0-A3 is asserted, ~~L6~~ L6 or L7 is changed, the decode is made, a command is generated, data is read into the Data Bus D0-D7 or written from the Data Bus to a register. A0 must be 0 to enable a read command. This corresponds to clearing a state. A0 must be 1 to enable a write command. This corresponds to setting a state.

The DMOTOR-ON input to the decoder comes from the same line that enables the selected drive motor (ENABLE 1, ENABLE 2) but it has some added delay so it changes when the device is de-selected. This insures that DMOTOR-ON is always stable during DEV, only one command will occur. For example, if L4 is set MOTOR-ON becomes 1, but DMOTOR-ON will remain 0 during that DEV and be 1 throughout the next DEV. Do not use the transition of L4 to make a command occur during that DEV, but consider the motor to be either on or off and L6, L7 to the transitioning inputs.

Bob Bailey 29 November 1983

Reading

Reads occur with MOTOR-ON=1 and L6=0, L7=0. The read logic is not reset and therefore, one byte must be read in before ~~it~~ random data is flushed out.

The input signal from RDDATA is synchronized to FCLOCK. Negative transitions are detected and generate a pulse of 1 FCLOCK period. These pulses generate 1's which are fed into the read data shift register. If another negative transition does not occur within a specified period, a 0 is fed into the read data shift register. However, if a negative transition does occur it shifts in a 1 and resets the timer. Group code has 2 important rules: 1) the leading bit (MSB) is 1 and, 2) there is a maximum of 2 consecutive zeroes. Therefore, all legal bit sequences can be analyzed by the patterns 1, 01, and 001 where the leading bit (MSB) is shown on the right.

After a negative transition a 0 is inserted into the shift register if the next negative transition is 24 FCLOCKS or more later. If it is less than 24 FCLOCKS another 1 is ~~shifted~~ inserted and the timer reset. If a 0 is inserted another 0 will be inserted if the next negative transition does not occur for 16 more FCLOCKS. Thus, if negative transitions of RDDATA are spaced by

0 - 23	FCLOCKS	DATA = 11
24 - 39	FCLOCKS	01
40 - 55	FCLOCKS	001

The read data shift register shifts each time a 0 or 1 is fed to it. Once the MSB has shifted through to the far end the shift register resets to all zeroes in order to receive the next byte. It will begin shifting in new bits when it is fed a 1.

In asynchronous mode the read data shift register parallel loads the read data register once the MSB=1 reaches the end of the shift register. The full byte is now available for reading. Port mode should always be used with asynchronous mode. In port mode the MSB read on D7 will be stable when read because the MSB is latched at the beginning of the read data command. This insures D7 will meet the set-up and hold requirements of the processor. Furthermore, if D7 is read as 1 then D0-D6 will be valid and stable. Once D7 is read as 1 by the processor the MSB of the read data register will reset 14 FCLOCKS (1.75 usec) later. This time is not affected by fast/slow mode.

In synchronous mode the read data register is loaded each time the read data shift register shifts. However, once the MSB=1 arrives at the far end of the shift register the read data register will not re-load for 2 data shifts + 4 FCLOCKS (8 FCLOCKS in slow mode). This is to ensure that the correct byte is in the read data register long enough to be seen by the processor but not long enough so as not to be seen as a valid byte twice. The rising edge of D7 is delayed so that if the processor sees D7=1 it is guaranteed that D0-D6 will have been correctly written into a processor register.

reset Asynchronous writes are entered with motor-on=1 going through the sequence set L6, set L7. This causes the pre-write pre-set pulse followed by the release of all write logic. The write data command is generated which loads data from the Data Bus to the write data buffer. This buffer in turn is parallel loaded into the write data ~~shift register.~~ Eight +/- 1/2 FCLOCKS after set L7 the parallel load ends. After 8 more FCLOCKS the MSB of the byte to be written (which is 1) will cause the WRDATA output to toggle from 1 to 0. Subsequently the write data shift register shifts and the WRDATA output toggles if the shifted bit is 1. The shift and toggle are separated by 8 FCLOCKS. After all 8 bits are shifted the write data shift register is loaded. This parallel load ends 8 x 16 FCLOCKS after the last parallel load ended. ¶ The handshake bit is set by the end of the parallel load and reset by the write data command. If it is 1 it means the write data buffer is available since its contents have been loaded into the write data shift register. If the handshake bit is 0 the write data buffer is not available since data has been written into the buffer but has not yet completed loading into the write data shift register.

no ¶ The handshake is read by the read status command which is prompted by CLR L6. This is part of a polling loop so the processor can poll until the handshake is 1 indicating the write data buffer is available. Then SET L7 will cause the write data command ~~and~~ to write a byte to the write data buffer. ¶ The underrun bit indicates ~~that~~ whether or not a new byte has been written into the write data buffer before the end the parallel load. If no new data is available an underrun occurs and the /WRREQ output will go to 1 before the next transition ~~or~~ of WRDATA occurs. This disables the write head before the old byte is rewritten. The underrun bit is read by the read status command. It is cleared while exiting the write mode by CLR L7.

Synchronous writes are entered with motor-on=1 through the sequence set L6, set ~~L6~~ L7. Unlike asynchronous write which uses FCLOCK to generate the time base, synchronous writes are done using Q3. Thus to have 2us bit cells instead of 4us requires asynchronous mode. The data to be written must be brought in with ~~the~~ set L7 which generates the write data command or during the next Q3 as is the case in false reads. The data will be parallel loaded from the write data buffer to the write data shift register. This will end 4-5 Q3 periods after the first time L7 is set. Two Q3 periods later the MSB of the data byte (which is required to be 1) will cause the WRDATA output to toggle from 1 to 0. The shifting and toggling are each updated every 8 Q3 periods. Each time a new byte is written the write data shift register goes into the load mode but the shift and toggle time bases are unaffected. It is required that a write device occurs every 8 Q3 periods. If a write data command does not occur, zeroes will be shifted out and not transitions of WRDATA will occur. Exit by CLR L7, CLR L6.

3/3

Bob Bailey

29 November 1983