1802/Mini Pixie Video Card

Theory of Operation

David S. Madole
February 5, 2023

Introduction

The following describes the theory of operation of the 1802/Mini Pixie video card, which is an alternative implementation of the CDP1861 video generator chip made by RCA in the 1970's but implemented with 4000-series CMOS logic.

While the 1802/Mini Pixie video card is designed to be software-compatible with the original CDP1861 chip, it interfaces with the processor in a different manner to overcome some disadvantages, and also has some other enhancements. In order to understand the 1802/Mini Pixie video card, it would help to first have an understanding of how there CDP1861 works from it's datasheet:

http://www.cosmacelf.com/publications/data-sheets/cdp1861.pdf

(Link courtesy of cosmacelf.com, operated by Dave Ruske).

The full design data of the 1802/Mini Pixie card can be found in the author's Gihub repository, including a PDF version of the schematic under the Releases section:

https://github.com/dmadole/1802-Mini-Pixie-Video

The circuit descriptions that follow will be broken down into eight areas:

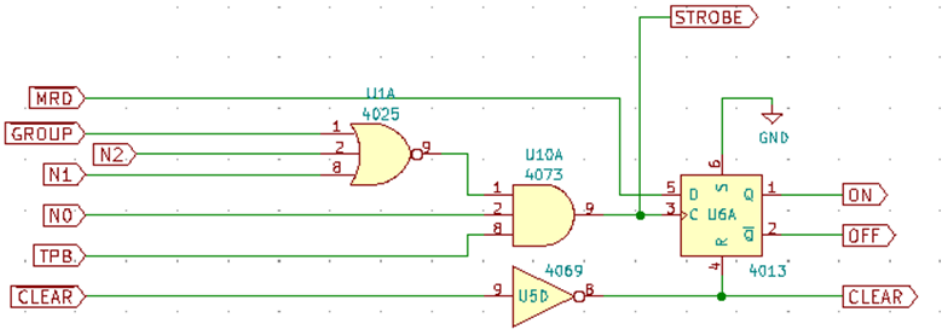
1. Control
2. Clock
3. Timing Generation
4. Processor Control Signals
5. Processor Synchronization
6. Video Generation
7. Synchronization Generation
8. Composite Video Output

Each will include an excerpt of the schematic, but it is probably helpful to also have the whole schematic at hand to be able to understand the context and interconnections.

Control

The control function of the Pixie card is very simple, as there is only one operation, which is to turn on or off the generation of video. The card is hardwired to detect I/O to port 1 with a read operation (opcode 69) enabling video, and a write operation (opcode 61) disabling video. This creates the ON and OFF signals which are used to gate certain operations of the card.

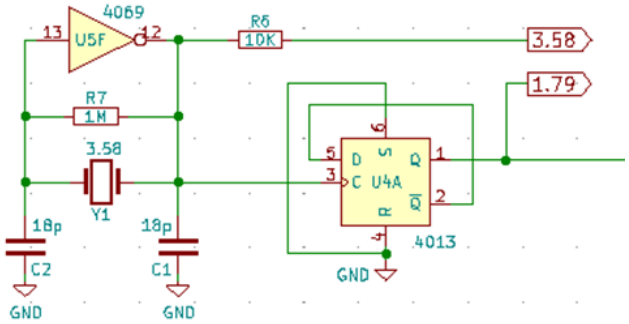
It is important that the card initializes in a known state, so the CLEAR- signal is negated and used to reset the counters in the circuit to all zeros.



Control Circuit

Clock

Although it is not the ideal frequency, an NTSC colorburst crystal is used as the clock source because it is close enough and commonly available and low-cost. This crystal frequency of 3.58 MHz is divided by two to get the 1.79 MHz dot-clock frequency which drives the whole circuit. The 3.58 MHz raw crystal frequency is not used but is connected to the color expansion connector.



Clock Circuit

Timing Generation

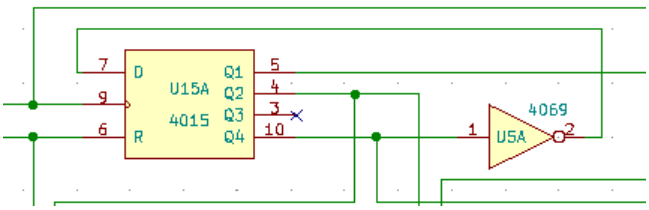
The core of the circuit is a series of counters which correspond to the scanning of the electron beam in the display tube and control all aspects of signal generation. There are four counters cascaded, all of which are Johnson-type counters which are made of shift registers with the last output complemented and fed back to the input. Each counter requires a number of shift register bits equal to half the maximum count.

The basic counter operation generates a sequence starting at all zero, then shifts in one bits one at a time, and when the register is all ones, shifts in zeros until back to the all-zero state, then repeats. Some counters in the circuit deviate from this basic model and will be addressed case-by-case.

Johnson counters are used because they are inherently synchronous (all output bits change at the same time) and minimal logic is needed to decode arbitrary ranges of count outputs. Any output or range of output can be detected with at most one AND or OR gate and one inverter.

The first counter counts the dot clock to produce a byte clock which all other timing is derived from. This is a simple divide-by-8 counter which counts 0000, 1000, 1100, 1110, 1111, 0111, 0011, 0001… which gives it a period of 1.79 MHz divided by 8, or about 224 KHz.

Besides overall timing, this counter drives the latching of each DMA byte from memory into the output shift register, as well as driving processor machine cycle synchronization.

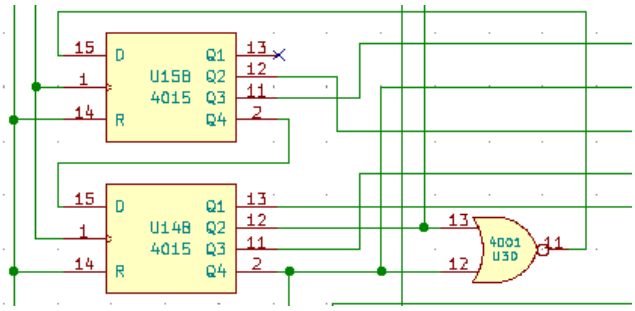


Byte Counter

Next in the chain is the horizontal timing counter, which implements a 14-count sequence that counts once per dot counter cycle. This cycles at the horizontal synchronization frequency which is the 224 KHz byte counter output divided by 14, or about 15,980 Hz.

Rather than a basic Johnson counter, this is implemented with asymmetrical feedback using a NOR gate so that it counts through a sequence of 8 zeroes and 6 ones rather than 7 of each. This simplifies the circuit because 8 is the number of bytes displayed in each horizontal line and so each output bit of the counter goes low for 8 counts with no further decoding needed.

The output of this counter is used to generate the horizontal sync pulse, to generate the DMA request when in the active display field, and to enable latching of DMA data during the display field.

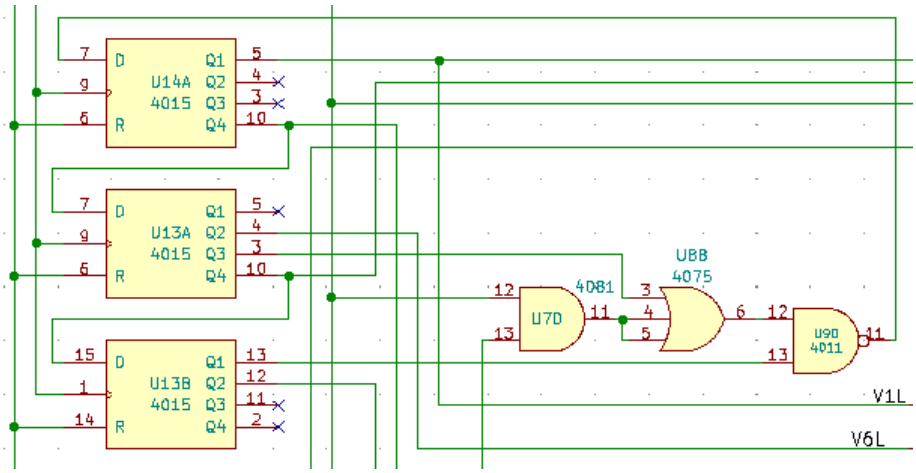


Horizontal Counter

The next two counters together form the vertical counter which counts each of the 262 horizontal scan lines. This is broken into two cascades counters to simplify the circuit and decoding.

The first counter counts a cycle of either 16 or 18 counts based on what lines are being scanned, and the second counter counts 16 groups of the first counter, with the first counter counting 16 cycles for the first 13 groups and 18 cycles for the last 3 groups. This gives a total number of lines equal to 13x16 + 3x18 which is 262 lines of output per each frame.

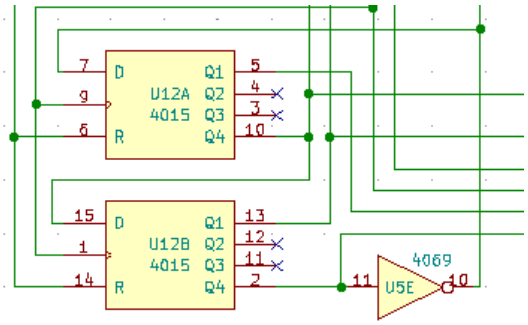
The first counter counts a sequence of 7 zeros and 9 ones for 16-cycle counts, and 9 zeroes and 9 ones for 18-cycle counts. This is controlled by a set of feedback gates which also take input from the second vertical counter to change count depending on the group of scan lines.



First Vertical Counter

The second vertical counter is a basic counter that counts the output of the first vertical counter and counts a sequence of 8 zeroes and 8 ones. Each output is low for 8 counts, which when multiplied by the 16 counts in the first vertical counter for those cycles is equal to the 128-line display area. The cycle

period of this counter is the 15,980 Hz horizontal counter rate divided by 262, giving the approximately 61 Hz vertical synchronization rate.

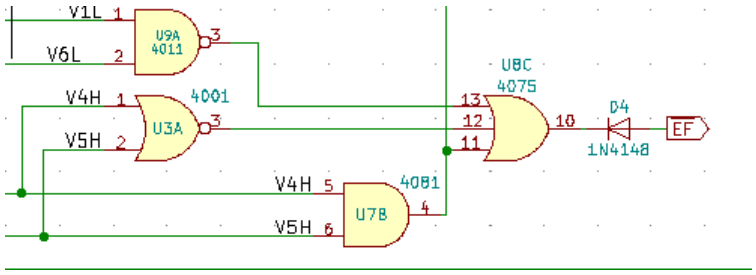


Second Vertical Counter

Processor Control Signals

In a Pixie video system, the processor is integral to the video circuitry as its DMA function is used to drive the retrieval of display data from the main system memory. There are three primary signals used to interface with the processor: EF, INT, and DMAOUT.

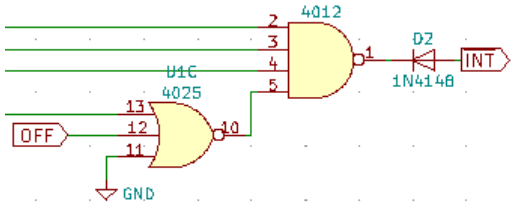
The EF signal goes low during the four horizontal lines preceding both the start and the end of the active display area. This allows the processor to antipate and detect these events to control program flow and timing. The signal is generated using three gates, which detect the last four counts of the first vertical counter, and the third and eleventh counts of the second vertical counter. These are combined by a fourth gate to generate the needed timing.



EF Signal Generation

The INT signal goes low for the two horizontal scan lines preceding the start of the active display area. This is used to drive the processor interrupt so that the interrupt service routine can be triggered to setup register R0 and other tasks needed for the DMA transfers to follow. This is generated by detecting the last two cycles of the first vertical counter and the third cycle of the second vertical counter.

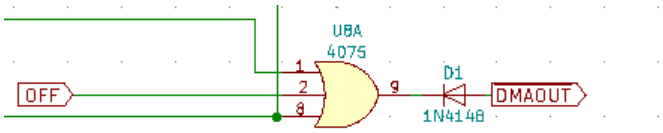
While the counters and many other outputs of the card run at all times, the INT signal is only generated when the display is enabled, so it is also gated with the OFF signal from the control circuit.



INT Signal Generation

The DMAOUT signal causes the processor to execute DMA cycles to transfer the display data and goes low for 8 bytes during each of the 128 lines of the display area. It is simply the combination of one of the horizontal counter outputs, which go low for 8 bytes, with one of the second vertical counter outputs,

which go low for 128 horizontal lines. This signal is also only generated while the display is enabled and so is also gated with the OFF control signal.



DMAOUT Signal Generation

Processor Synchronization

Because the processor DMA is used to drive display data retrieval from memory, these cycles need to happen at exactly the right time when the data is needed for output. On the CDP1861, this is done partially by using the video dot clock for the processor clock, which causes processor machine cycles to be the same length as display byte cycles. This has the disadvantage of fixing the processor clock to a relatively slow speed.

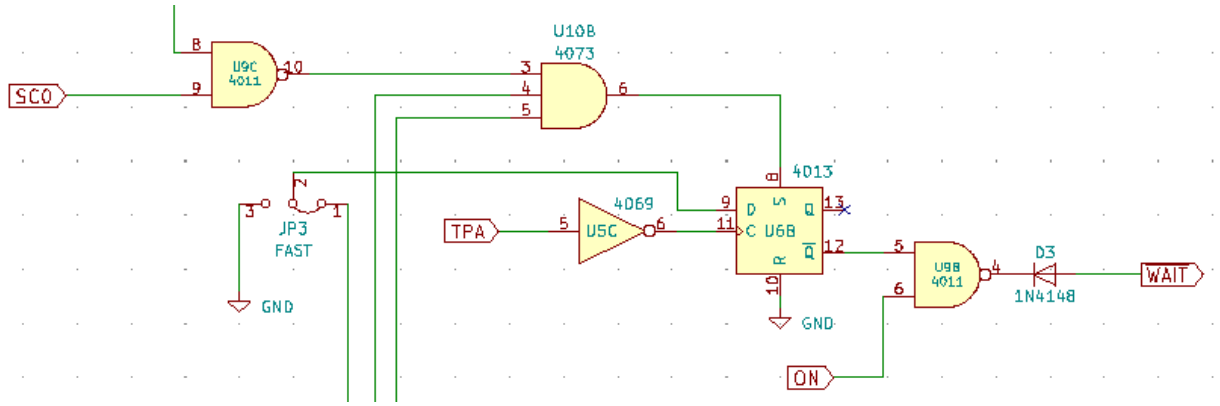
There is a second part also needed to align the processor cycle state with video generation since the DMAOUT signal is only sampled by the processor during S1 (execute) cycles. In the CDP1861, this is done by skipping a count of its horizonal counter. This has the disadvantage of causing a short horizonal scan line in the video output, which will create a loss of synchronization in the monitor often lasting about the next 100 scan lines.

In the 1802/Mini Pixie circuit, the needed synchronization is instead accomplished by running the processor run at an arbitrary clock rate higher than the dot rate and pausing the processor periodically by asserting the WAIT- signal. The processor is paused at the falling edge of its TPA signal as this is a time when all address lines and memory control signals are stable during the memory access part of the machine cycle. This timing works correctly on both the 1802 and 1804/5/6 processors which sample WAIT- at different times.

Pausing the processor at this time during a DMA cycle means that the display data will be present on the bus and can be latched while the processor is paused. Once the display data is latched, the processor is resumed by raising WAIT- again.

The second needed part of the synchronization, aligning to the execute machine cycle, is done by suppressing the wake-up of the processor by one machine at the start of each scan line when it is out of alignment. This is determined by looking at the SC0 line at the edge of the horizonal sync pulse.

The WAIT- signal is only ever asserted when the display is enabled so that the processor runs unimpeded at its full clock rate when the display is off. There is a jumper-selectable option that additionally determined whether the processor is paused during all cycles when the display is enabled, or only during and just before the display period. The latter setting will improve processing speed since the processor will run at its full clock rate approximately 7/16 of the time even when video is enabled.

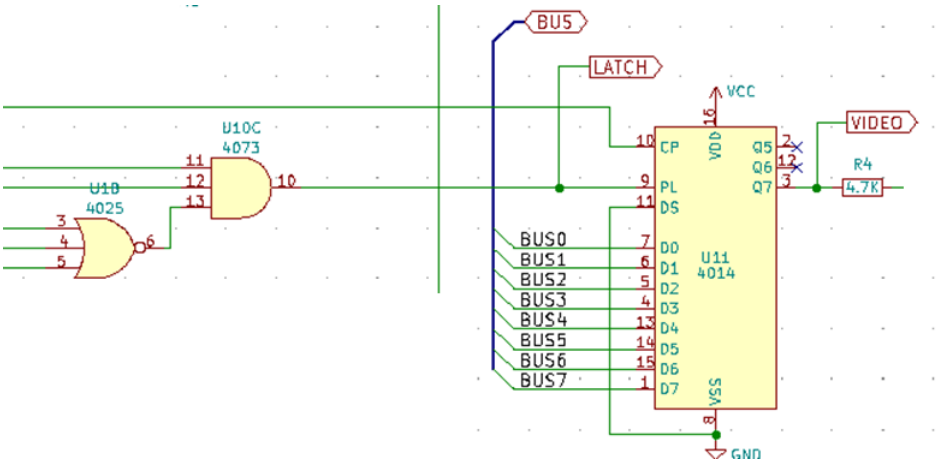


Processor Syncronization Circuit

Video Generation

The video output is generated by latching each byte of DMA output data at the start of each dot clock cycle. Due to the synchronization circuit, this will happen while the processor is paused and the data bus is stable. The timing of the latch signal is derived in a similar way to the DMAOUT signal, except one byte counter cycle later, since it takes the processor one cycle to respond to DMAOUT.

Once the byte data is latched, it is shifted out one bit per dot clock, which forms the video display data. The shift register is clocked continuously, but because it's serial input is tied low, it only actually outputs data for 8 cycles following each byte latch.



Video Generation Circuit

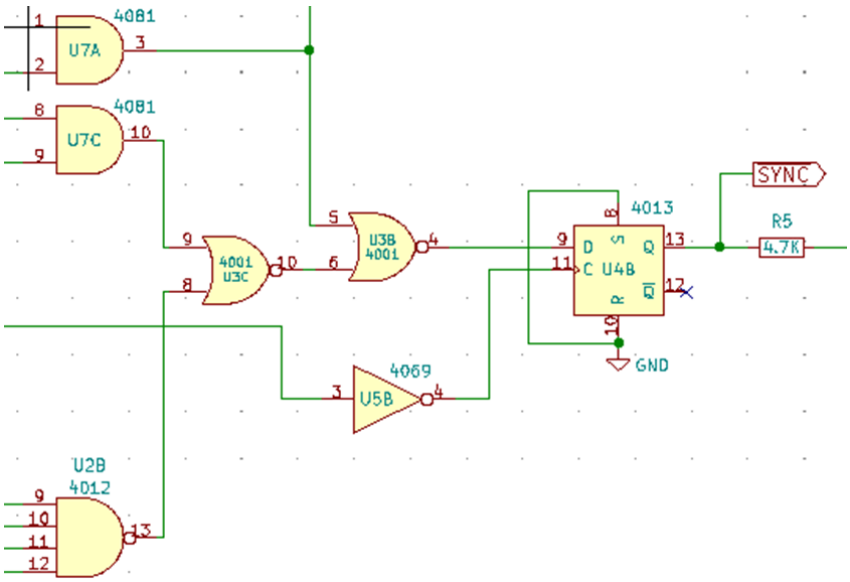
Synchronization Generation

The monitor needs a synchronization signal to keep its internal scanning in alignment with the display data and counters in the video card. This consists of two pulses that are combined with the video data. At the start of each scan line, there is a negative pulse which lasts for one byte clock cycle, and at the top of each frame there is a negative pulse lasting three scan lines.

During the time that these signals overlap, positive pulses need to be generated instead to mark the horizontal sync since the signal is low for three lines due to the vertical sync signal. In the CDP1861, this is accomplished by using XOR to combine the horizontal and vertical sync pulses, however this is not the optimal approach as the synchronization signal needs to go low at the start of each line and this results in it going high at the start of the lines during the vertical sync interval. This can cause a minor loss of synchronization on some monitors as the negative edge moves out of place.

Rather than using an XOR, in addition to the normal negative sync pulse, the 1802/Mini Pixie card also generates a positive pulse just prior to the horizontal sync time during the vertical sync time, which results in a falling edge exactly where it should be.

The synchronization signals, once generated and combined are latched with a flip-flop approximately three dot times into each byte cycle. This serves two purposes, one is to remove any timing glitches caused by mas-matched propagation delays in the sync generation gates, and the other is to shift the image slightly to the left since the sync is now delayed relative to the image data. This corrects the right shift that is normally seen on the CDP1861.

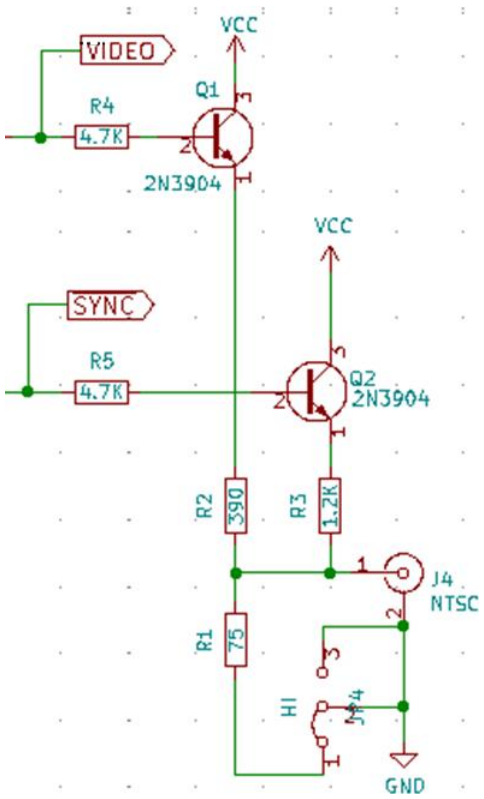


Synchronization Circuit

Composite Video Output

The video and synchronization signals are combined to form the final composite video output. The output has three possible voltage levels, which are approximately 0.4 volts for black pixels and non-display time, 0 volts during sync pulses, and 1.2 volts when a white pixel is displayed. A 4000-series CMOS gate cannot drive the 75 ohm load that a standard monitor produces, so an amplifier is needed.

The 1802/Mini Pixie card uses two common-collector transistor amplifiers to drive and combine the outputs, as these have a voltage gain that is close to unity independent of the actual transistors gain (beta). This gives a predicable signal level based only on the voltage divider resistors without any tuning needed regardless of the gain of the transistor used.



Composite Video Output