**Pico Elliott 920M Emulator**

**==========================**

**Andrew Herbert – Last Revised 19/11/2023**

**Aim**

**----**

**To build an emulator for the Elliott 920M avionics computer using a Raspberry Pi Pico.**

**Principles of Operation**

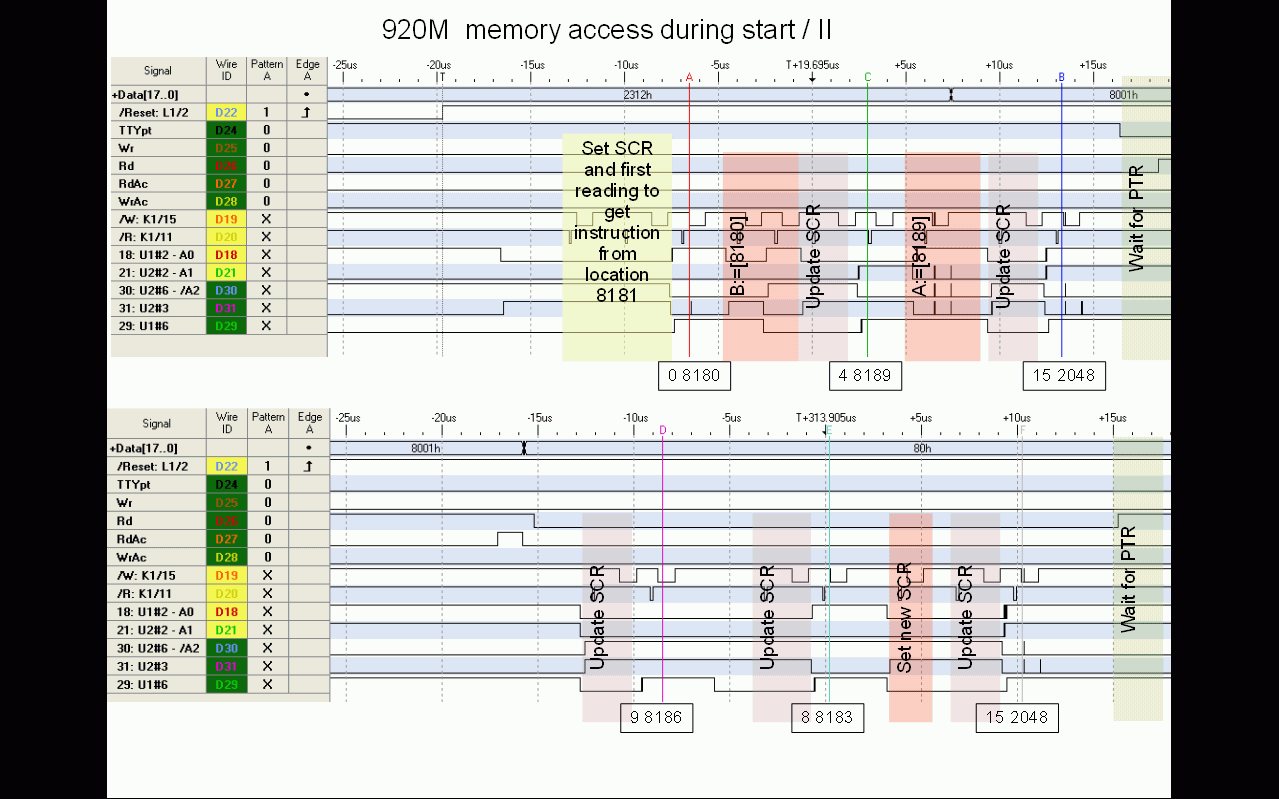
**-----------------------**

**The software provides an emulation of the Elliott 920M minicomputer with 8K of store. Electrically the 920M is assumed to be connected to a paper tape station with paper tape reader, paper tape punch and optional teletype.**

**The emulation of the 900 series instruction set is taken from my generic 900 series emulation. It does not at present implement the “undefined” effects distinct to the 920M.**

**The 920M emulation uses the Pico GPIO pins to emulate the electronic interface of the 920M. (See hardware section for definition of signals and associated GPIO pins.)**

**920M to Paper Tape Station Interface**

****

**To make a read request (15 2048 or 15 2052 instruction), the 920M raises the signal RDRREQ and awaits a corresponding ACK signal from the PTS. If the request is for data to be read from the teletype (15 2052), the signal TTYSEL is also asserted. The PTS holds up ACK for about 2uS. Within this time the 920M is required to copy 8 bits of data from the eight reader data lines (RDR+1 to RDR+128) and then remove the RDRREQ and TTYSEL signals. The 920M should not assert a further read (or punch) request until after the paper tape station has cleared the ACK signal.**

**To make a punch request (15 6144 or 15 6148 instruction), the 920M first sets 8 bits of data to the eight punch data lines (PUN+1 to PUN+128) and then asserts the signal PUNREQ and awaits a corresponding ACK signal from the PTS. If the request is for data to be punched on the teletype (15 6148 instruction, the signal TTYSEL is also asserted. The PTS holds up ACK for about 4uS. Within this time the 920M is required to remove the RDRREQ and TTYSEL signals. The 920M should not assert a further punch (or read) request until after the paper tape station has cleared the ACK signal.**

**At the start of execution, in emulation mode, the emulator waits for power on (NOPOWER low) to be maintained for about 10ms and reads the II\_AUTO pin to determine whether to autostart (II\_AUTO low) or to run initial instructions (II\_AUTO high).**

**In emulation mode, LOG high signals that diagnostic messages should be sent to the serial USB port. LOG low suppresses these messages.**

**If NOPOWER high is detected at any point, including during input/output operations, the emulation is halted and the system halts. The system can be restarted by pressing the RESET button.**

**The status LED is used to signal emulation status. When running the LED flashes every 1.0 seconds. When an error condition arises, it flashes every 0.5 seconds. During an I/O transfer it flashes every 0.25 seconds**

**Logging reports and diagnostic messages are output to the USB serial channel.**

**Hardware**

**--------**

**The external case of the 920M emulator provides two LEDs, two switches and a push button. The red LED indicates power is supplied to the emulator (via a USB cable). The green LED is the status indicator as described above. One switch is labelled "POWER" and delivers the NOPOWER signal (high when switched off, LOW when switched on). The second is labelled LOG and controls the LOG signal (on for logging enabled, off for logging disabled). The push button is labelled RESET and restarts the emulator in the Pico.**

**Electronics**

**-----------**

**The system consists of a pair of Pico processors on a protoboard accompanied by three TXS0108E 8-bit bi-directional level-shifters. One Pico runs PicoProbe to provide a means to download binary programs into the second Pico which runs the emulation. The PicoProbe also provides a USB serial connection for logging output.**

**As the Picos use 3V3 logic, the GPIO pins are connected to the level-shifters for conversion to and from the TTL 5V level of the 920M.**

**The ACK and II\_AUTO lines are pulled down on start up to prevent false signals when the PTS is inactive or not connected. The NOPOWER and LOG lines are pulled up. (If not connected to a powered-on PTS, the level shifter asserts ACK).**

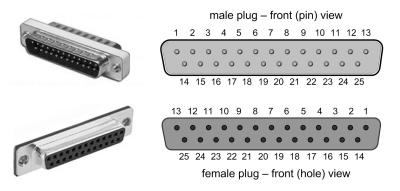
**The pins of the Pico are connected as follows:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Pin** | **GPIO** | **Signal** | **Level Shifter** | **Pin** | **GPIO** | **Signal** | **Level Shifter** |
| **1** | **GPIO0** | **Serial link to PicoProbe** |  | **40** | **VBUS** |  |  |
| **2** | **GPIO1** | **Serial link to PicoProbe** | **Re** | **39** | **VSYS** |  |  |
| **3** | **GND** |  |  | **38** | **GND** |  |  |
| **4** | **GPIO2** | **RDR +1** | **C1** | **37** | **3V3\_EN** |  |  |
| **5** | **GPIO3** | **RDR +2** | **C2** | **36** | **3V3 OUT** |  |  |
| **6** | **GPIO4** | **RDR +4** | **C3** | **35** | **ADC REF** |  |  |
| **7** | **GPIO5** | **RDR +8** | **C4** | **34** | **GPIO28** | **STATUS** |  |
| **8** | **GND** |  |  | **33** | **GND** |  |  |
| **9** | **GPIO6** | **RDR +16** | **C5** | **32** | **GPIO27** | **LOG** |  |
| **10** | **GPIO7** | **RDR +32** | **C6** | **31** | **GPIO26** | **PUNREQ** | **A5** |
| **11** | **GPIO8** | **RDR +64** | **C7** | **30** | **RUN** |  |  |
| **12** | **GPIO9** | **RDR+128** | **C8** | **29** | **GPIO22** | **RDRREQ** | **A4** |
| **13** | **GND** |  |  | **28** | **GND** |  |  |
| **14** | **GPIO10** | **PUN +1** | **B1** | **27** | **GPIO21** | **TTYSEL** | **A3** |
| **15** | **GPIO11** | **PUN +2** | **B2** | **26** | **GPIO20** | **II/AUTO** | **A2** |
| **16** | **GPIO12** | **PUN +4** | **B3** | **25** | **GPIO19** | **ACK** | **A1** |
| **17** | **GPIO13** | **PUN +8** | **B4** | **24** | **GPIO18** | **NOPOWER** |  |
| **18** | **GND** |  |  | **23** | **GND** |  |  |
| **19** | **GPIO14** | **PUN +16** | **B5** | **22** | **GPIO17** | **PUN+128** | **B8** |
| **20** | **GPIO15** | **PUN +32** | **B6** | **21** | **GPIO16** | **PUN +64** | **B7** |

**Pins 1 and 40 are at the USB connector end of the Pico card.**

****

**The external interface is delivered at a female DSub 25 connector, with pins as follows:**

****

**Colour coding of DSub pins:**

**1 ACK Brown 14 RDR +1 Brown**

**2 PUN +1 Brown 15 RDR +2 Red**

**3 PUN +2 Red 16 RDR +4 Orange**

**4 PUN +4 Orange 17 RDR +8 Yellow**

**5 PUN +8 Yellow 18 RDR +16 Green**

**6 PUN +16 Green 19 RDR +32 Blue**

**7 PUN +32 Blue 20 RDR +64 Violet**

**8 PUN +64 Violet 21 RDR +128 Grey**

**9 PUN +128 Grey 22 GND Black**

**10 II/AUTO Red 23 GND Black**

**11 TTYSel Orange 24 GND Black**

**12 RDRReq Green 25 GND Black**

**13 PUNReq Yellow**

**Note: 12, 13 colour code incorrect, wired as listed.**

**Loopback Test**

**-------------**

**Commented out in the source code for the emulator are a set of routines that provide a loopback test, if a suitable loopback plug is connected. For the 920M the loop back connects the reader data pins to the punch data pins, TTY\_SELECT to II\_AUTO and RDRREQ to ACK.**

**Notes on emulator program**

**=========================**

**The emulator is programmed in C.**

**Two cores are used. One runs the emulation, the second runs a parallel “blinker” routine that flashes the status light according to a global variable blink, which can be set to NO\_BLINK, SLOW\_BLINK or FAST\_BLINK.**

**For some reason probably to do with initialization of the USB system it is necessary to include a short (250ms) wait before performing serial i/o and launching the blinker core.**

**It is necessary to set the GPIO slew rate for the PTS interface output pins to GPIO\_SLEW\_RATE\_SLOW so that the level shifters respond correctly – the Pico has a very fast rise time.**

**It is important to be careful with the use of the gpio\_put\_masked function – its is not guaranteed that all the selected pins will be set simultaneously, and therefore the receiving PTS may see garbled data. For this reason PUNREQ is signalled by following the loading of punch data and TTYSEL using gpio\_put\_masked with a call to gpio\_put for the PUNREQ.**

**Notes**

**=====**

**Pico GPIO operation timings:**

**Function gpio\_put time = 0.040075 uS, 25 MHz**

**Function gpio\_put\_masked time = 0.056100 uS, 18 MHz**

**Function gpio\_get time = 0.040076 uS, 25 MHz**

**Function gpio\_get\_all time = 0.040075 uS, 25 MHz**

**920M Instruction Timing**

**=======================**

**The timings below are for 902M MCM7 5343, and will no doubt vary slightly with temperature and voltage.**

**Each value was obtained by obeying the instruction one million times, using a loop of 8000 instructions run 125 times.**

**The figures below should all be reduced by 0.01 usec to allow for the loop overhead of 8 words per 8000.**

**Function  Unmodified  Modified  
 0:       10.38 usec  13.51 usec  
 1:       10.60 usec  13.73 usec  
 2:       12.86 usec  15.98 usec  
 3:       11.70 usec  
 4:       10.60 usec  13.74 usec  
 5:       11.70 usec  
 6:       10.60 usec  13.73 usec  
 7 <0:    10.82 usec  
 7 =0:    12.86 usec  
 7 >0:    11.98 usec  
 8:       10.60 usec  
 9 <0:    10.60 usec  
 9 =0:     9.72 usec  
 9 >0:     9.72 usec  
10:       11.75 usec  14.88 usec  
11:       13.73 usec         <== See below  
12:       31.91 usec  35.04 usec  
13:       33.01 usec  36.19 usec  
14 0:     10.83 usec  
14 36:    51.19 usec  
14 8156:  51.19 usec**

**15 Not timed**

**Thus, a simple instruction takes around 11us.**

**B-modification seems to add 3.13usec.**

**Emulator Instruction Timing**

**===========================**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Instruction** |  | **uS** |  | **Notes** |
| **Simple loop** |  | **0.36** |  | **8 ;+1 8 ;-1** |
|  |  |  |  |  |
| **0** |  | **0.18** |  | **0 0 8 ;-1** |
| **1** |  | **0.19** |  | **etc** |
| **2** |  | **0.19** |  |  |
| **3** |  | **0.19** |  |  |
| **4** |  | **0.19** |  |  |
| **5** |  | **0.21** |  |  |
| **6** |  | **0.19** |  |  |
| **7** |  | **0.19** |  | **(<>0)** |
| **7** |  | **0.17** |  | **(=0)** |
| **8** |  | **0.18** |  | **(half simple loop)** |
| **9** |  | **0.17** |  | **(>=0)** |
| **9** |  | **0.19** |  | **(<0)** |
| **10** |  | **0.20** |  |  |
| **11** |  | **0.20** |  |  |
| **12** |  | **0.32** |  |  |
| **13** |  | **0.40** |  |  |
| **14** |  | **0.30** |  | **14 0** |
| **14** |  | **0.30** |  | **14 48** |
| **15** |  | **9.41** |  | **includes 4uS ACK and 2uS  of pulse spacing intervals** |

**B-modification adds about 0.01uS.**