H8 LED Activity Indicator Controller

OVERVIEW

A small controller circuit board plugs into any full or half card slot on the H8 backplane. The controller monitors A1-A7, IORD, IOWR, IORW, and M1 bus signals and has outputs for up to eight LEDs. The LEDs are wired to a tiny display module board that connects to the controller with a ten-conductor ribbon cable. This two-board arrangement allows the activity LEDS to be mounted away from the backplane or controller card with a minimum of cabling. A GAL is programmed to turn on one or more LEDs when a specified combination of address and I/O control lines exists. 74HCT123 one-shots stretch the pulse for good visibility.

By using independent GAL logic to monitor I/O activity, any H8 I/O function can be monitored without additional wires between boards and with no requirement for an existing activity LED on the board. For example, there is no board-mounted LED that monitors APU (arithmetic processor) activity. But such a monitor is easy to implement using the GAL logic. The GAL logic approach also allows you to combine several monitors onto a single LED or to easily change what is being monitored without modifying any wiring.

GAL LOGIC

The GAL is programmed using WinCUPL. The WinCUPL builds the LED control signals in four steps.

1) Identify ports which are to be monitored for I/O activity
2) Match a port with an I/O control signal (READ, WRITE, or R/W)
3) Combine ports into functional groupings (SERIAL, USB, DISK)
4) Link a port or port group to an LED

Identifying a port to monitor:

The commonly used H8 ports have been identified in the WinCUPL source. If you want to monitor a port not already identified in the source, you can add a line like this:

```
154Q = !A7 & A6 & A5 & !A4 & A3 & A2 & !A1;
```

This tells the GAL to look for the bit pattern “0110110x” on the address bus. A0 is always a “don’t care” – this device cannot distinguish between “odd” or “even” port numbers.

Match a port and I/O control signal:

Decide whether you want to monitor READ (IORDA), WRITE (IOWRA), or both (IOACT) with this port, and give this monitor a simple mnemonic. For example, to monitor port 154/155Q for READ activity, you could write:

```
MYACT = 154Q & IORDA;
```

In the simplest case, you would assign this monitor to one of the eight LEDs. To use LED5 to monitor MYACT activity, you would code:

```
LED5 = MYACT;
```

Combine ports into functional groupings:

Your new monitor may be one of several that have some logical relationship. You can group these related monitors together using simple OR logic:

```
BIGACT = MYACT + YOURACT + THEIRACT;
```

Assign monitors or monitor groups to LEDS:

Finally, assign any combination of monitors you like to an LED:

```
LED5 = MYACT;
```

If you want a single LED to display multiple monitors, just use OR logic:

```
LED5 = MYACT # USB;
```

This example would display either our new monitor (MYACT) or the pre-defined USB activity monitor on LED5.

LED ASSIGNMENTS

LED1-4	Four additional activity LEDS in the new (SEBHC 2023) case
LED5	The topmost “fifth” LED on either the Heath or Collado front panel boards
LED6-8	Not implemented in the new case hardware

GAL SOURCE

```
Name     LED_CONTROL ;
PartNo   00 ;
Date     9/8/2022 ;
Revision 01 ;
Designer Engineer ;
Company  Terry Smedley ;
Assembly None ;
Location  ;
Device   g16v8a;

/* *************** INPUT PINS *********************/
/* UNBUFFERED RAW BUS SIGNALS */
PIN 1   = M1                        ; /*                                 */
PIN 2   = !A1                       ; /*                                 */
PIN 3   = !A2                       ; /*                                 */
PIN 4   = !A3                       ; /*                                 */
PIN 5   = !A4                       ; /*                                 */
PIN 6   = !A5                       ; /*                                 */
PIN 7   = !A6                       ; /*                                 */
PIN 8   = !A7                       ; /*                                 */
PIN 9   = IORD                      ; /*                                 */
PIN 11  = IOWR                      ; /*                                 */

/* *************** OUTPUT PINS *********************/
PIN 19  = !LED1                     ; /*                                 */
PIN 18  = !LED2                     ; /*                                 */
PIN 17  = !LED3                     ; /*                                 */
PIN 16  = !LED4                     ; /*                                 */
PIN 15  = !LED5                     ; /*                                 */
PIN 14  = !LED6                     ; /*                                 */
PIN 13  = !LED7                     ; /*                                 */
```

```
PIN 12  = !LED8                     ; /*                                */

/* ********* INTERMEDIATE VALUES *********/
/* DECODE RAW PORT ADDRESS RANGES */
170Q = !A7 & A6 & A5 & A4 & A3 & !A2;          /* DISK CONTROLLER @ 170Q-173Q   */
174Q = !A7 & A6 & A5 & A4 & A3 & A2 & !A1;     /* DISK CONTROLLER @ 174Q-175Q   */
200Q = A7 & !A6 & !A5 & !A4 & !A3 & !A2;       /* APU @ 200Q-203Q               */
260Q = A7 & !A6 & A5 & A4 & !A3 & !A2;         /* VDIP1 @ 260Q-263Q             */
270Q = A7 & !A6 & A5 & A4 & A3 & !A2 & !A1;    /* HA-8-3 VDP @ 270Q-271Q        */
272Q = A7 & !A6 & A5 & A4 & A3 & !A2 & A1;     /* HA-8-3 PSG @ 272Q-273Q        */
274Q = A7 & !A6 & A5 & A4 & A3 & A2 & !A1;     /* DISK CONTROLLER @ 274Q-275Q   */
320Q = A7 & A6 & !A5 & A4 & !A3 & !A2 & !A1;   /* 8250 SER OR Z89-11 PAR @ 320Q-321Q*/
330Q = A7 & A6 & !A5 & A4 & A3 & !A2 & !A1;    /* 8250 SERIAL OR VDIP1 @ 330Q-331Q */
340Q = A7 & A6 & A5 & !A4 & !A3 & !A2 & !A1;   /* 8250 SERIAL PORT @ 340Q-341Q  */
350Q = A7 & A6 & A5 & !A4 & A3 & !A2 & !A1;    /* 8250 SERIAL PORT @ 350Q-351Q  */
364Q = A7 & A6 & A5 & A4 & !A3 & A2 & !A1;     /* HA-8-3 APU @ 364Q-365Q        */
366Q = A7 & A6 & A5 & A4 & !A3 & A2 & A1;      /* HA-8-3 ADC @ 366Q-367Q        */
040Q = !A7 & !A6 & A5 & !A4 & !A3 & !A2 & !A1; /* SS1 ADC @ 040Q-041Q           */
064Q = !A7 & !A6 & A5 & A4 & !A3 & A2 & !A1;   /* SS1 APU @ 064Q-065Q           */
014Q = !A7 & !A6 & !A5 & !A4 & A3 & A2;        /* 8255 @ 014Q-017Q              */
020Q = !A7 & !A6 & !A5 & A4 & !A3 & !A2;       /* 8255 @ 020Q-023Q              */
024Q = !A7 & !A6 & !A5 & A4 & !A3 & A2;        /* SS1 DACS @ 024Q, 026Q         */
030Q = !A7 & !A6 & !A5 & A4 & A3 & !A2;        /* 8255/PCA9665 @ 030Q-033Q      */
034Q = !A7 & !A6 & !A5 & A4 & A3 & A2;         /* 8255 @ 034Q-037Q              */
054Q = !A7 & !A6 & A5 & !A4 & A3 & A2;         /* 8255 @ 054Q-057Q              */

/* DECODE I/O CONTROL SIGNALS */
IORW = IOWR # IORD;         /* READ/WRITE */
IOACT = IORW & !M1;         /* MASK WITH !M1 FOR CLEAN I/O */
IORDA = IORD & !M1;         /* READ ONLY */
IOWRA = IOWR & !M1;         /* WRITE ONLY */

/* ASSIGN DEVICE NAMES TO PORT ACTIVITY */
H37 = 170Q & IOACT;         /* H37 DISK CONTROLLER R/W                    */
H17 = 174Q & IOACT;         /* H17 DISK CONTROLLER R/W                    */
H67 = 274Q & IOACT;         /* H67 DISK CONTROLLER R/W                    */
APUX = 200Q & IOACT;        /* AM9511 STANDALONE R/W                      */
USB1 = 260Q & IOACT;        /* OLD USB VDIP1 R/W                          */
VDP = 270Q & IOWRA;         /* TMS99X8 VDP WRITE                          */
PSG = 272Q & IOWRA;         /* AY-3-8910 PSG WRITE                        */
PAR1 = 320Q & IOWRA;        /* SS1 PARALLEL PORT WRITE                    */
SER1 = 320Q & IOACT;        /* H8-4 SERIAL PORT R/W                       */
SER2 = 330Q & IOACT;        /* H8-4 SERIAL PORT R/W                       */
USB2 = 330Q & IOACT;        /* NEW USB VDIP1 R/W                          */
SER3 = 340Q & IOACT;        /* H8-4 OR DUART SERIAL PORT R/W              */
CON = 350Q & IOWRA;         /* H8-4 OR DUART SERIAL PORT WRITE            */
HAAPU = 364Q & IOACT;       /* HA-8-3 APU R/W                             */
HAADC = 366Q & IOACT;       /* HA-8-3 ADC R/W                             */
SSADC = 040Q & IOACT;       /* SS1 ADC R/W                                */
SSAPU = 064Q & IOACT;       /* SS1 APU R/W                                */
SSDAC = 024Q & IOACT;       /* SS1 DACS R/W                               */
I2CBB = 034Q & IOACT;       /* I2C BITBANG R/W                            */
I2CPCA = 030Q & IOACT;      /* I2C PCA9665 R/W                            */
PPIO1 = 014Q & IOACT;       /* PPIO 1 R/W                                 */
PPIO2 = 020Q & IOACT;       /* PPIO 2 R/W                                 */
PPIO3 = 054Q & IOACT;       /* PPIO 3 R/W                                 */

/* COMBINE DEVICES TO LOGICAL ACTIVITY GROUPINGS */
```

```
SER = SER3 # CON;                /* SERIAL = SER3 (340Q) + CONSOLE (350Q)*/
USB = USB1 # USB2;               /* USB = USB1 (260Q) + USB2 (330Q)      */
APU = SSAPU # HAAPU # APUX;      /* APU = SS1 (064Q) + HA83 (364Q) + AM9511 (200Q) */
ADC = SSADC # HAADC;             /* ADC = SS1 (040Q) + HA83 (366Q)       */
HA83 = VDP # PSG # HAAPU # HAADC; /* ANY I/O ON HA83                     */
SS1 = SSAPU # SSADC # SSDAC;     /* ANY I/O ON SS1                       */
I2C = I2CBB # I2CPCA;            /* I2C = BITBANG (034Q) + PCA9665 (030Q)*/
PPIO = PPIO1 # PPIO2 # PPIO3;    /* 8255 PPIO ACTIVITY                   */

/* FINALLY! ASSIGN DEVICES/GROUPS TO LEDS */

/* LED1-4 ARE FOUR LEDS ON NEW CASE (TOP TO BOTTOM) */
LED1 = USB # PPIO;         /* USB ACTIVITY AND PPIO                      */
LED2 = SER # PAR1 # I2C;   /* I/O DEVICES - PRINTER, CONSOLE, I2C        */
LED3 = HA83;               /* ANY HA-8-3, POLLING MAKES THIS CONTINUOUS  */
LED4 = SS1;                /* ANY HA-8-2, POLLING MAKES THIS CONTINUOUS  */

/* LED5 IS "FIFTH LED" ON HEATH OR COLLADO FRONT PANELS */
LED5 = H17 # H37 # H67;    /* ANY DISK ACCESS                            */

/* LED6-8 ARE NOT IMPLEMENTED IN HARDWARE AT THIS TIME */
LED6 = H17;                /* JUST H17                                   */
LED7 = APU;                /* APU (HA-8-2, HA-8-3, STANDALONE APU)       */
LED8 = VDP # SSDAC;        /* I/O DEVICES - VDP (HA-8-3), DAC (HA-8-2)   */
```

TAS 10/26/22