

of the logical "OR" gate 318. A "5" output of the de-multiplexer 306 is coupled to a third input of the logical "OR" gate 320. The "0", "6" and "7" outputs of the de-multiplexer 306 have no connection.

A "1" output of the de-multiplexer 308 is coupled to a fourth input of the logical "OR" gate 312. A "2" output of the de-multiplexer 308 is coupled to a fourth input of the logical "OR" gate 314. A "3" output of the de-multiplexer 308 is coupled to a fourth input of the logical "OR" gate 316. A "4" output of the de-multiplexer 308 is coupled to a fourth input of the logical "OR" gate 318. A "5" output of the de-multiplexer 308 is coupled to a fourth input of the logical "OR" gate 320. The "0", "6" and "7" outputs of the de-multiplexer 308 have no connection.

A "1" output of the de-multiplexer 310 is coupled to a fifth input of the logical "OR" gate 312. A "2" output of the de-multiplexer 310 is coupled to a fifth input of the logical "OR" gate 314. A "3" output of the de-multiplexer 310 is coupled to a fifth input of the logical "OR" gate 316. A "4" output of the de-multiplexer 310 is coupled to a fifth input of the logical "OR" gate 318. A "5" output of the de-multiplexer 310 is coupled to a fifth input of the logical "OR" gate 320. The "0", "6" and "7" outputs of the de-multiplexer 310 have no connection.

An output of the logical "OR" gate 312 is coupled to a first, non-inverted, input of a logical "AND" gate 322. An output of the logical "OR" gate 314 is coupled to a second, inverted, input of the logical "AND" gate 322 and to a first, non-inverted, input of a logical "AND" gate 324. An output of the logical "OR" gate 316 is coupled to a third, inverted, input of the logical "AND" gate 322, to a second, inverted, input of the logical "AND" gate 324 and to a first, non-inverted, input of a logical "AND" gate 326. An output of the logical "OR" gate 318 is coupled to a fourth, inverted, input of the logical "AND" gate 322, to a third, inverted, input of the logical "AND" gate 324, to a second, inverted input of the logical "AND" gate 328 and to a first, non-inverted, input to a logical "AND" gate 328.

An output of the logical "OR" gate 320 is coupled to a fifth, inverted, input of the logical "AND" gate 322, to a fourth, inverted, input of the logical "AND" gate 324, to a third, inverted, logical "AND" gate 326, to a second, inverted input of the logical "AND" gate 328 and coupled to enable a vector address register 212. An output of the logical "AND" gate 322 is coupled to enable a vector address register 202. An output of the logical "AND" gate 324 is coupled to enable a vector address register 204. An output of the logical "AND" gate 326 is coupled to enable a vector address register 208. An output of the logical "AND" gate 328 is coupled to enable a vector address register 210.

The contents of the vector address register 202 are selectively coupled to a vector address bus 330 according to a level of the output of the logical "AND" gate 322. The width of the vector address bus 330 preferably corresponds to the number of bits used by the microprocessor for addressing memory (e.g. 32 bits), though other addressing schemes can be utilized. The contents of the vector address register 204 are selectively coupled to the vector address bus 330 according to a level of the output of the logical "AND" gate 324. The contents of the vector address register 208 are selectively coupled to the vector address bus 330 according to a level of the output of the logical "AND" gate 326.

The contents of the vector address register 210 are selectively coupled to the vector address bus 330 according to a level of the output of the logical "AND" gate 328. The contents of the vector address register 212 are selectively

coupled to the vector address bus 330 according to a level of the output of the logical "OR" gate 320. The vector address bus 330 is coupled to a microprocessor 332 for the system to provide appropriate interrupt vector addresses to logic within the microprocessor 332 for implementing a program counter for the microprocessor 332.

For operation of the circuit 300, the configuration registers 102-110 and vector address registers 202-212 are appropriately configured. Preferably, the registers 102-110 and 202-212 are accessible to the microprocessor 332 and are initially configured upon start-up of the microprocessor-based system, such as by a start-up software routine. Once the system is operating, the contents of the registers 102-110 and 202-212 can be altered by software commands or by user input. Alternately, the registers 102-110 and 202-212 are configured by logic circuits external to the microprocessor 332.

For example, assume the registers 102-110 and 202-212 are configured according to the example illustrated in FIG. 2B. Therefore, the interrupt configuration register 102 contains the bit pattern 011; the interrupt configuration register 104 contains the bit pattern 001; the interrupt configuration register 106 contains the bit pattern 101, the interrupt configuration register 108 contains the bit pattern 010; and the interrupt configuration register 110 contains the bit pattern 000. In addition, the vector address register 202 contains the vector address B'; the vector address register 204 contains the vector address D'; the vector address register 208 contains the vector address A'; the contents of the vector address register 210 are a don't care since none of the interrupt signals are mapped to the register 210; and the vector address register 212 is configured with the vector address C'.

Therefore, whenever the interrupt signal INT. A is active, the "3" output of the de-multiplexer 302 is a logical high voltage, resulting in the output of the logical "OR" gate 316 also being a logical high voltage. Whenever the interrupt signal INT. B is active, the "1" output of the de-multiplexer 304 is a logical high voltage, resulting in the output of the logical "OR" gate 312 also being a logical high voltage. Whenever the interrupt signal INT. C is active, the "5" output of the de-multiplexer 306 is a logical high voltage, resulting in the output of the logical "OR" gate 320 being a logical high voltage. Whenever the interrupt signal INT. D is active, the "2" output of the de-multiplexer 308 is a logical high voltage, resulting in the output of the logical "OR" gate 314 being a logical high voltage. Whenever the interrupt signal INT. E is active, the "0" output of the de-multiplexer 310 is a logical high voltage. Because the "0" output has no connection, the interrupt signal INT. E is masked. Accordingly, the configuration of the de-multiplexers 302-310 and logical "OR" gates 312-320 functions as a mapping circuit for appropriately mapping the interrupt signals INT. A-E to the outputs of the logical "OR" gates 312-320 according to the contents of the configuration registers 102-110.

The above example assumes the interrupt signals INT. A-E are active high, level sensitive, signals. If they are active low signals, an inverter is preferably included for each interrupt signal to convert the active low signals to active high signals. If the interrupt signals are edge sensitive, a latch is preferably included for each interrupt signal to convert the edge sensitive signals to level sensitive signals.

The configuration of the logical "AND" gates 322-328 function as a priority encoder circuit to prioritize the interrupt signals. The interrupt mapped to the vector address