Hamming Codes

Recall the basic structure of a Hamming code. Given bits 1...,m, the bit at position 2^n (starting at n=0, the first bit) is parity for all the bits with a 1 in position n. For example, the first bit is chosen such that the sum of all odd-numbered bits is even.

Q1. Suppose you had the bits 0011 and we want to add some bits to allow single error correction.

a) How many bits do we have to add?

b) Which bits are parity bits?

c) Which bits does each parity bit cover?

d) Write the completed coded representation for this bit string.

e) What do we need to do to make this into a SEC-DED code?

Q2. Find the original bits given the following SEC Hamming Code.

a) 0110111

b) 1001000

Q3. If we didn't need SEC but wanted SED, how many bits would we need to add for 4 bits? For 16 bits?

Interrupt

Definition: "An unscheduled event that disrupts program execution"

- Hardware (asynchronous): I/O service, timer expiration, hardware failure, etc.
- Software (synchronous): Exceptions/ traps.
  Examples: divide-by-zero, overflow, invalid memory address, page fault.

Handling an interrupt:

- Flush pipeline in case of exception (bubble out the following instructions)
- Disable interrupts (setting interrupt enable bit to 0)
- Save PC of the offending instruction in "Exception Program Counter" (EPC)
- Jump to interrupt service routine/ interrupt handler
- After exception/ interrupt is handled, jump to EPC and resume execution

Q4. What would be an alternative to using interrupts for interacting with a (much slower) device? Keeping this in mind, why is an interrupt-based approach efficient?

Q5. For each of the 5 MIPS pipeline stages, list exceptions that could occur.

Q6. An *inter-processor interrupt* (IPI) is a type of interrupt where one processor will interrupt another. What are some situations where this could be used?

Q7. An *interrupt storm* is when a processor is bogged down by a huge number of rapid interrupts (usually due to a faulty device or configuration). How could one mitigate this problem?