

## Discussion 14: Parity, ECC, RAID

### Hamming ECC

Recall the basic structure of a Hamming code. Given bits  $1, \dots, m$ , the bit at position  $2n$  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.

- i. How many bits do we need to add to  $0011_2$  to allow single error correction?

Parity Bits: 3

- ii. Which locations in  $0011_2$  would parity bits be included?

Using P for parity bits: PP0P011<sub>2</sub>

- iii. Which bits does each parity bit cover in  $0011_2$ ?

Parity bit #1: 1, 3, 5, 7

Parity bit #2: 2, 3, 6, 7

Parity bit #3: 4, 5, 6, 7

- iv. Write the completed coded representation for  $0011_2$  to enable single error correction.

1000011<sub>2</sub>

- v. How can we enable an additional double error detection on top of this?

Add an additional parity bit over the entire sequence.

- vi. Find the original bits given the following SEC Hamming Code: 0110111<sub>2</sub>

Parity group 1: error

Parity group 2: okay

Parity group 4: error

Incorrect bit:  $1 + 4 = 5$ , change bit 5 from 1 to 0: 0110011<sub>2</sub>

0110011<sub>2</sub> → 1011<sub>2</sub>

- vii. Find the original bits given the following SEC Hamming Code: 1001000<sub>2</sub>

Parity group 1: error

Parity group 2: okay

Parity group 4: error

Incorrect bit:  $1 + 4 = 5$ , change bit 5 from 1 to 0: 1001100<sub>2</sub>

1001100<sub>2</sub> → 0100<sub>2</sub>

### RAID

Fill out the following table:

	Configuration	Pro / Good for...	Con / Bad for...
RAID 0	Data disks without check information	No overhead Fast read / write	Reliability
RAID 1	Mirrored Disks: Extra copy of disks	Fast read / write Fast recovery	High overhead → Expensive
RAID 4	Transfer units = a sector within a single disk. Errors are detected within a single transfer unit Can handle independent reads/writes per disks	Higher throughput of small reads	Still slow small writes (A single check disk is a bottleneck)
RAID 5	Check information is distributed across all disks in a group.	Higher throughput of small writes	

Small accesses = an access to a single disk in a group