CSSE2010

CSSE2010/CSSE7201

Semester 2, 2021

Lecture 1

Course Introduction
Bits, Bytes and Binary

School of Information Technology and Electrical Engineering
The University of Queensland

Today's Outline

- Course Intro, Organisation and Admin
- Assessment details
- Course Blackboard – quick walkthrough
- Technical Content – Binary Numbers

Welcome !

CSSE2010/CSSE7201 – Introduction to Computer Systems

- CSSE2010-IN
- CSSE2010-EX
- CSSE7201-IN
- CSSE7201-EX

READ THE ECP

Course Staff

- Course Coordinator and Lecturer: Dr Chamith Wijenayake
  - Office: St Lucia Campus 78-537
  - Email: c.wijenayake@uq.edu.au
  - Office Hours: TBC
- A large group of tutors

Study formula: 1L-2P-1L-2P-3S-1E

What's This Course All About?

We will learn about Computers in different aspects

Source: Google Images

Software

Hardware

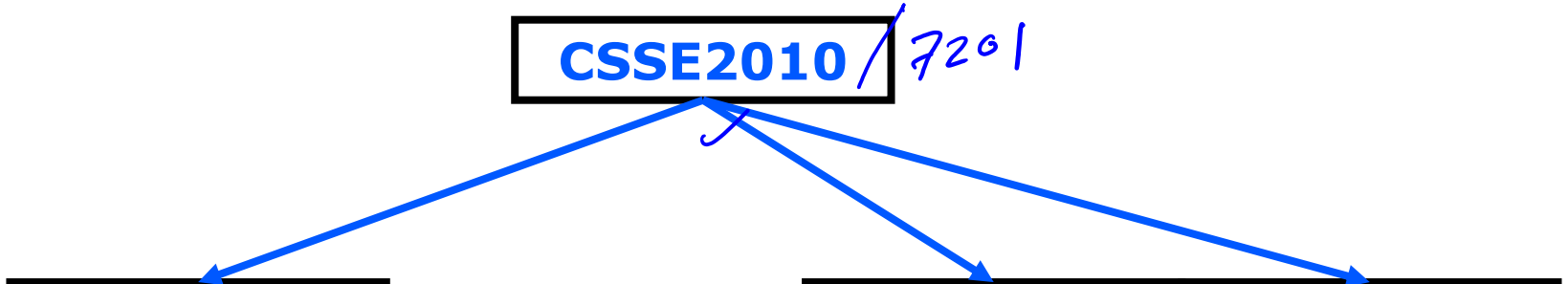
W13
W12
W11
W10
W09

W08
W07
W06

W05
W04
W03
W02
W01

Where does this course lead?

- Follow on courses



Computer Systems Principles & Programming

Engineering Team Project 1

→ COMS3200 – Computer Networks 1
→ COMP3301 – Operating Systems Architecture
→ CSSE3010 – Embedded Systems Design & Interfacing

Course Learning Activities (ECP Sections 3 and 4)

- **Run in two modes: Internal (IN) and External (EX)**
- **Lectures: Monday 6-7pm** and **Wednesday 9am-10am**. Conducted via zoom and recorded. Zoom meeting ID on Blackboard
- **Learning Labs (Prac sessions):**
  - **Internal (IN) mode: COVID-19 guidelines must be followed at all times**
    - **In-person at 47-104, two sessions per week**
    - Sign up for a pair of sessions PRA-FD-x-P1, PRA-FD-x-P2, 5 pairs to select from
    - **Students will be loaned a lab kit which must be returned or purchased by the end**
    - **If you drop the course or change to EX mode you must return/purchase the kit**
  - **External (EX) mode:**
    - **Online over zoom, two sessions per week**
    - Sign up for a pair of sessions PRA-EX-x-P1, PRA-EX-x-P2, 2 pairs to select from
    - **Students need to acquire an Arduino Uno and some other components by week 6**
    - If you are an EX student from Brisbane, you still need to acquire your own hardware
- Practice exercise

Contact eait.mytimetable@uq.edu.au for any sign on issues

Staying Healthy @ UQ

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

Stay home if you are unwell

Cover your mouth and nose when you sneeze or cough

Avoid touching your face

Wash your hands thoroughly

Don't share personal items

Clean surfaces

Maintain space between each other

Put used tissues in the bin

Call your General Practitioner (doctor) or UQ Health Care and explain your symptoms

CRICOS 00025B

Need the facts? **about.uq.edu.au/coronavirus-advice-uq-community**

Assessment Details – ECPs Section 5

CSSE2010-IN/EX

- **Weekly quizzes (10%=1.25% * 8)**
- **Assignment 1 (20%)**
  - Part 1 – 1-hr timed MCQ quiz (10%)
  - Part 2 - Digital logic design (10%)
- **Assignment 2 (20%)**
  - AVR microcontroller programming
  - 10% pass hurdle
- **Final Exam (50%)**
  - 40% pass hurdle
  - IN: invigilated written exam on campus
  - EX: invigilated (ProctorU) online exam

CSSE7201-IN/EX

- **Weekly quizzes (5%=0.625% * 8)**
- **Assignment 1 (20%)**
  - Part 1 – 1-hr timed MCQ quiz (10%)
  - Part 2 - Digital logic design (10%)
- **Assignment 2 (25%)**
  - AVR programming + ISA questions
  - 10% pass hurdle
- **Final Exam (50%)**
  - 40% pass hurdle
  - IN: invigilated written exam on campus
  - EX: invigilated (ProctorU) online exam

Resources

- Course Blackboard site
  - Announcements
  - Readings
  - Notes
  - Software downloads
  - Discussion Board (Edstem) for technical matters
  - Other admin matters and also course content – make use of consultation hours and email me at c.wijenayake@uq.edu.au to book an appointment
- No textbook required
  - Some references listed in course profile

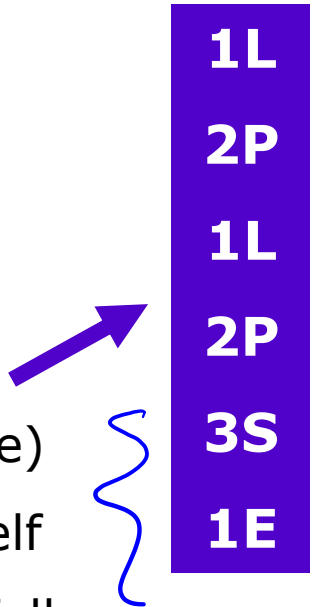
TurningPoint Polls (App or Web)

- App:
  - Select "East Asia" region
- Web:
  - http://responsewaresg.net/
- To respond
  - Join as guest – no sign-in required
  - Enter the session ID (usually csse2010s2)
  - **Enter your student number as the user ID**

These polling questions are not assessed

Role of Teaching Staff and Students

- What is Teaching?
  - One definition:
    "*The purposeful creation of situations from which motivated learners should not be able to escape without learning or developing*" [Cowan 1998]



What should I be doing to be successful in this course:

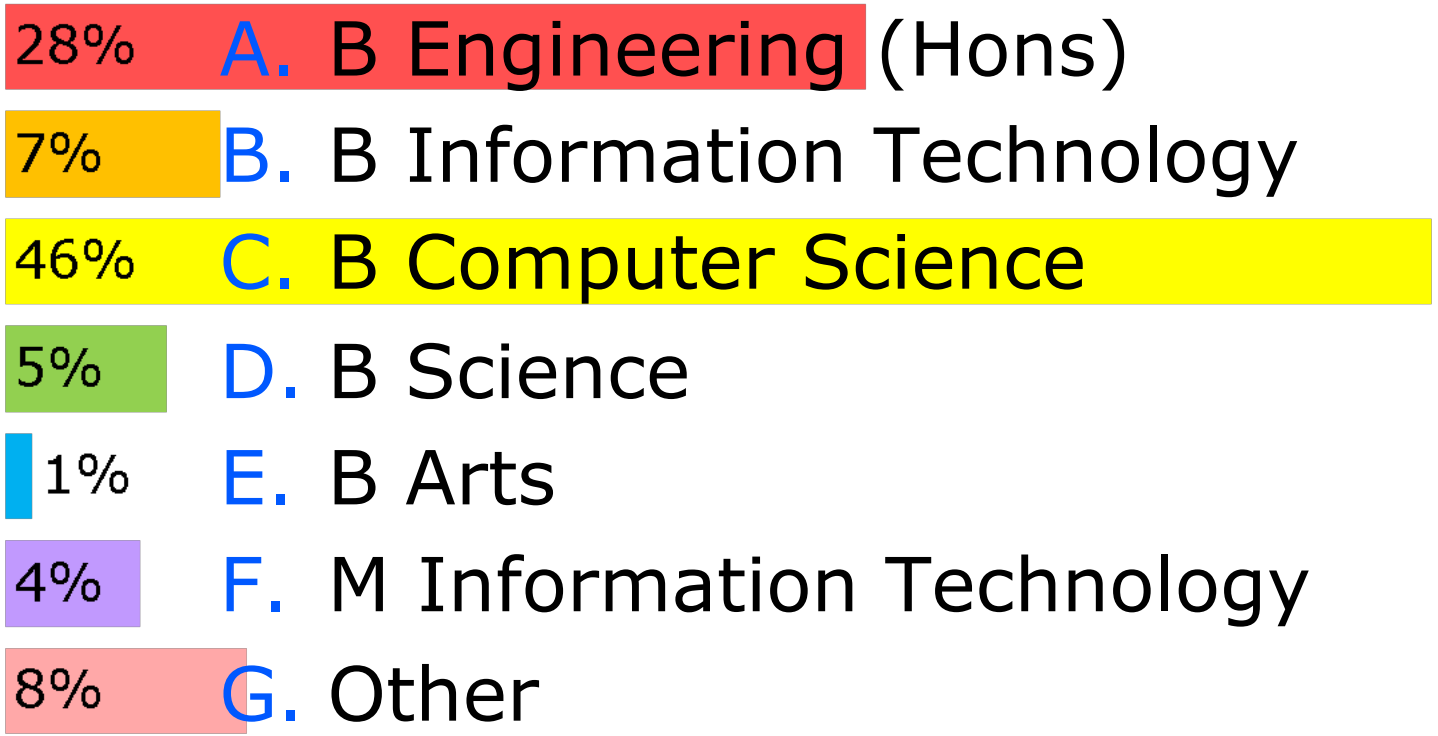
- ✓ Attend the two weekly lectures (watch the recording otherwise)
- ✓ Spend some time going through the lecture content by yourself
- ✓ Attend the two lab sessions weekly – complete the lab tasks fully
- ✓ Revise the weekly content and do the quiz
- ✓ Complete the additional exercises given

Some Questions for You…
First – make sure response setup works

TurningPoint

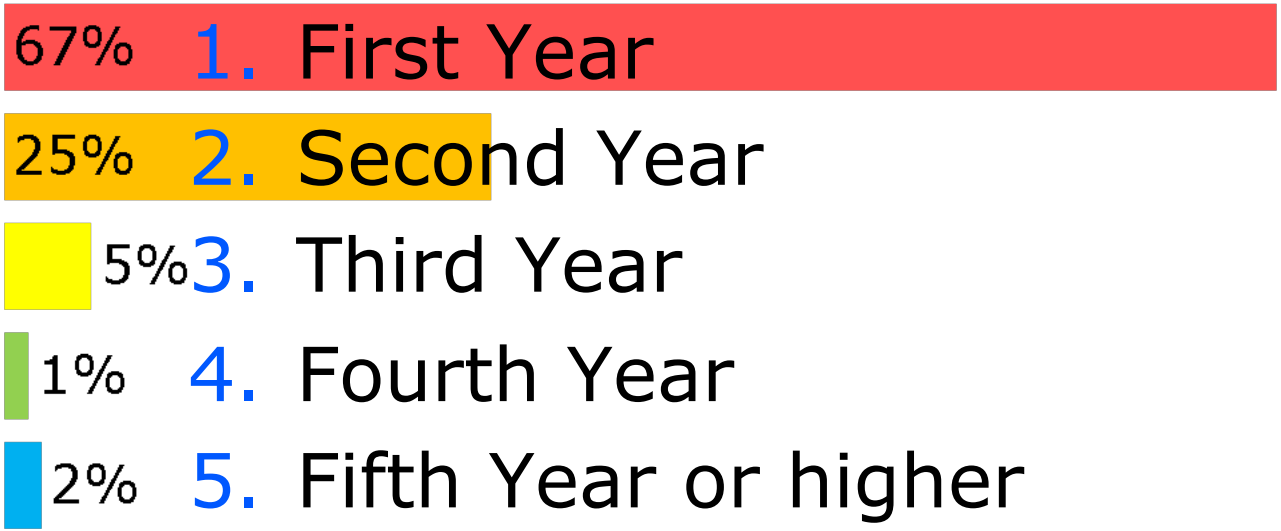
- App region is "East Asia"
- Web address is http://responsewaresg.net/
- Guest (not signin)
- Enter Session ID: **csse2010s2**
- Enter student number (as User ID)

What degree program are you studying?



0

What year level are you in?



0

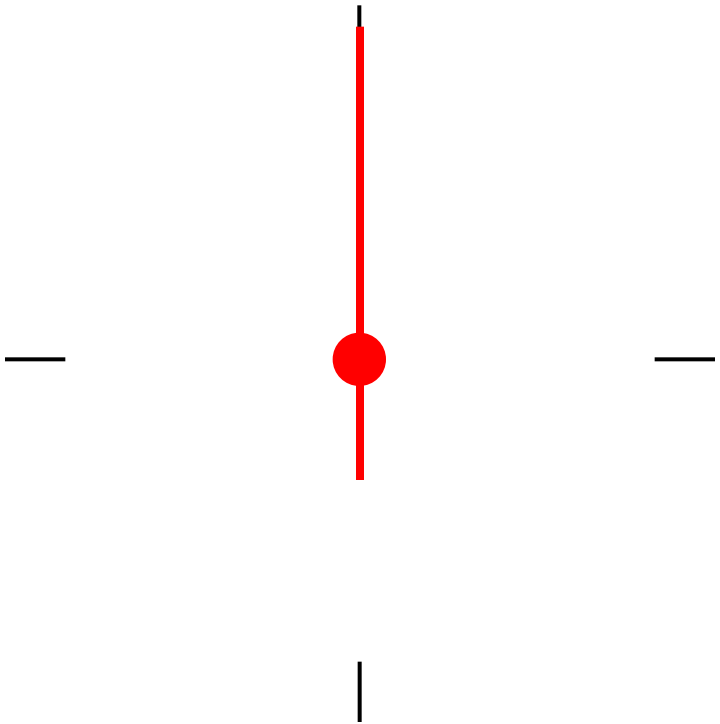
Are you in IN or EX mode?

1. IN
2. EX

0

Short Break

- Stand up and stretch



Computers and Binary Numbers

INSTRUCTIONS

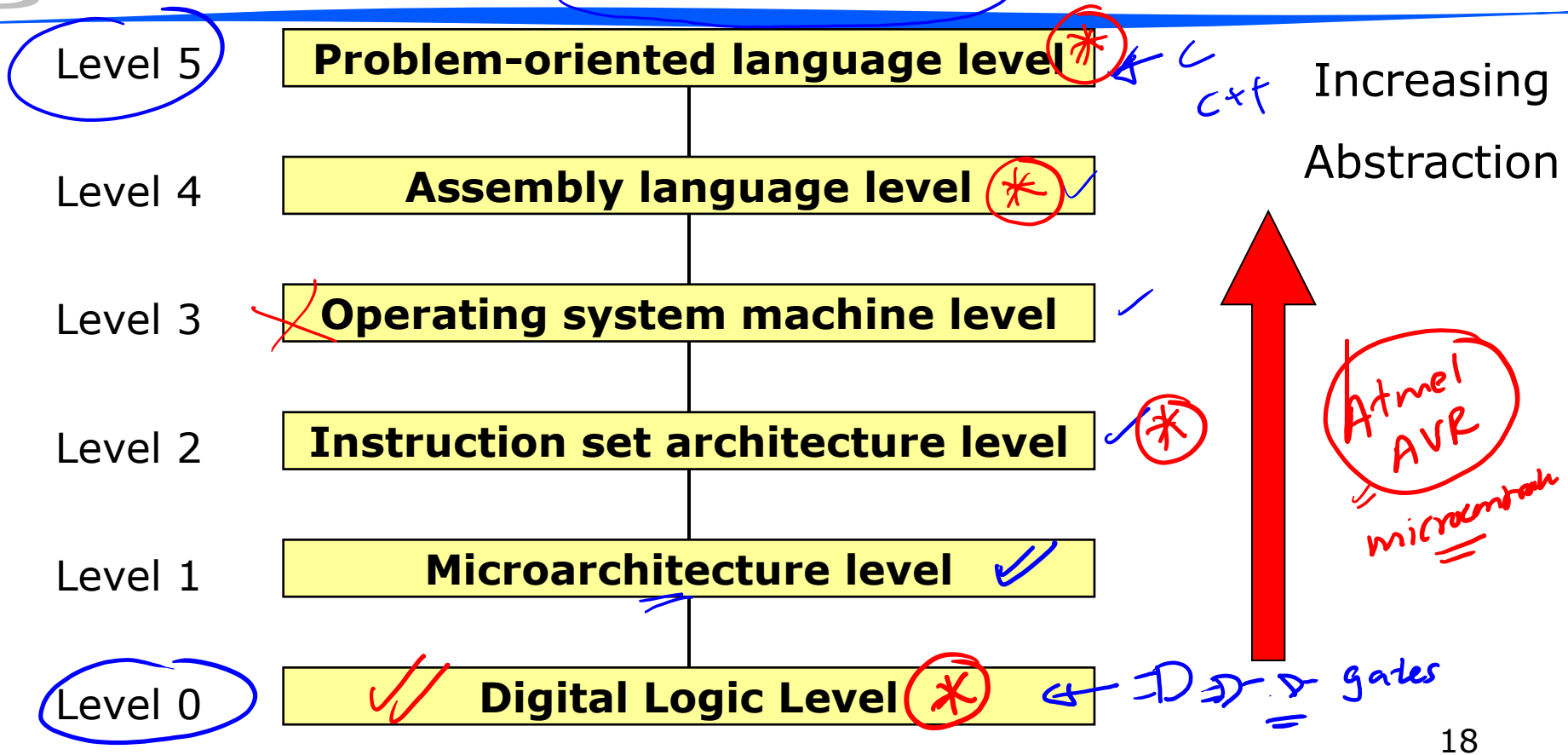
INPUTS ⇨ **Computer**

General purpose or embedded computers ⇨ OUTPUTS

10100101011010100111101010100101111000110101010

Binary representation of digital data

A Computer at Different Levels of Abstraction



Bits, Bytes and Binary Numbers

- Computers represent everything in binary
- Bit = binary digit (0 or 1)
- Byte = 8 bits
  - e.g.

0	1	0	1	0	1	1	1

- Modern computers deal with words which are usually a power of 2 number of bytes, e.g.
  - 1, 2, 4, or 8 bytes = 8, 16, 32, 64 bits

Representing Whole (Unsigned) Numbers in Binary

- Each bit position has a value:

	9	8	7	6	5	4	3	2	1	0
	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
....	512	256	128	64	32	16	8	4	2	1

Bit position

Value (Powers of 2)

base 2

- Converting binary to decimal:

  Add values of each position where bit is 1

7	6	5	4	3	2	1	0
1	0	0	1	0	0	0	1

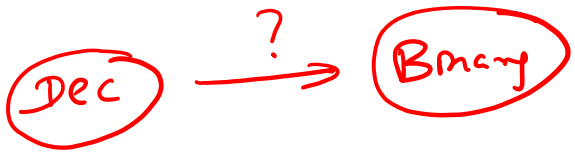
128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 145

10100111 ?

1001001

Converting Decimal to Binary

- **Example**
  - Convert 53 (decimal) to binary ?



- **Method 1**
  - rewrite n as sum of powers of 2 (by repeatedly subtracting largest power of 2 not greater than n)
  - Assemble binary number from 1's in bit positions corresponding to those powers of 2, 0's elsewhere

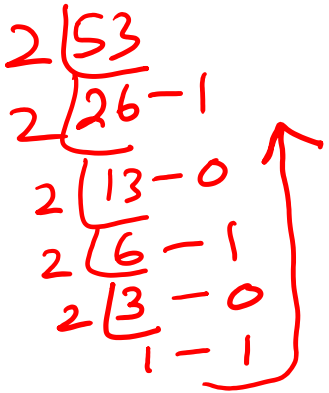
$53 = 32 + 16 + 4 + 1$

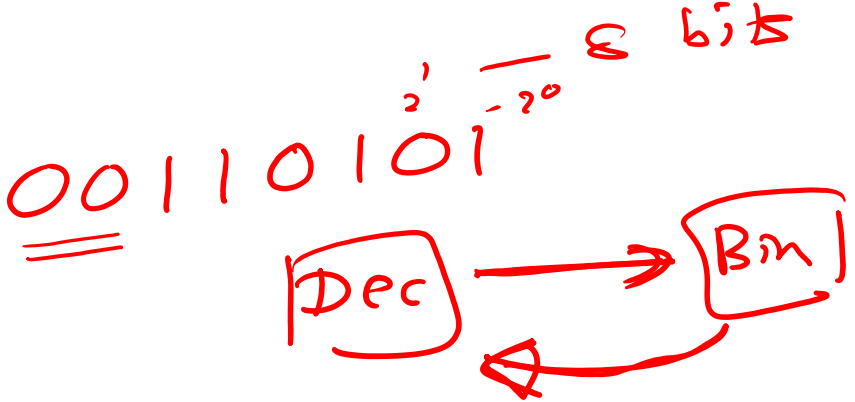
110101

$2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

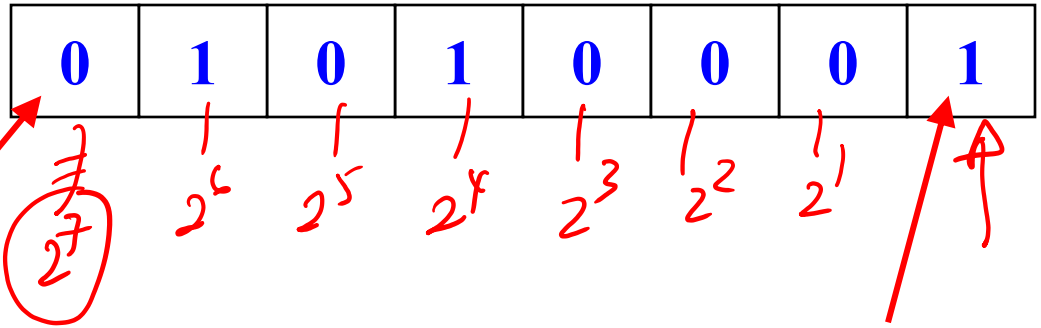
Converting Decimal to Binary (cont.)

- **Example**
  - Convert 53 (decimal) to binary
- **Method 2** – building up bits from the right (least significant)
  - Divide n by 2
  - Remainder of division (0 or 1) is next bit
  - Repeat with n = quotient





Least and Most Significant Bits



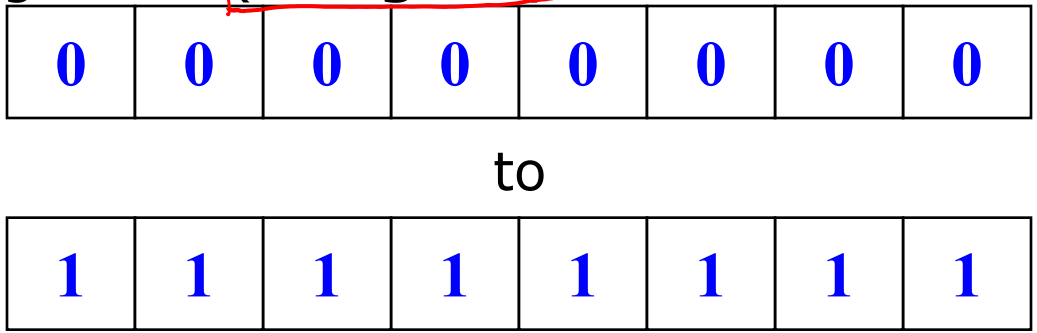
Most significant bit (**MSB**) – the bit position that's "worth" the most ($2^7 = 128$, in this case)

Least significant bit (**LSB**) – the bit position that's "worth" the least ($2^0 = 1$)

For an n-bit unsigned word, the MSB is worth $2^{(n-1)}$

Number Range

- Assuming whole (**unsigned**) numbers…



Smallest number that can be represented: all 0's

Value is 0.

$0 - 2^n - 1$ (255)

Largest number that can be represented: all 1's

For an n-bit word, the largest representable number is $(2^n)-1$, e.g. 255 in this case

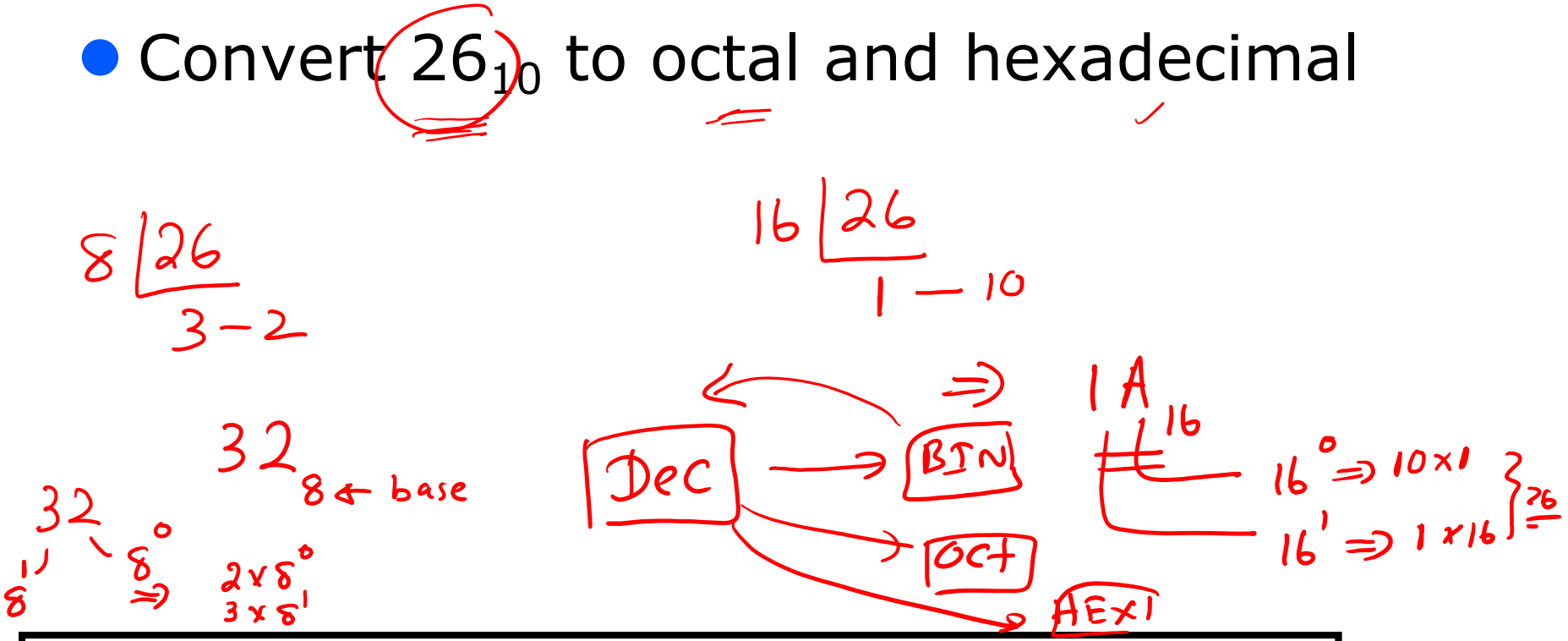
Other Radices

- **Radix** = number system base
- A radix-k number system
  - k different symbols to represent digits 0 to k-1
  - Value of each digit is (from the right) k^0, k^1, k^2, k^3, …
- Often convenient to deal with
  - **Octal** (radix-8) - Symbols: 0 1 2 3 4 5 6 7
    - One octal digit corresponds to 3 bits
  - **Hexadecimal** (radix-16) - Symbols: 0 1 2 3 4 5 6 7 8 9 A B C D E F
    - One hexadecimal digit corresponds to 4 bits (useful!)

Dec	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Oct	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	20	21
Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11

Exercise

- Convert 26_{10} to octal and hexadecimal



You have 1½ minutes

How Do We Know the Radix?

- What's the value of 101 or 747? How do we know the radix?
- Can use a subscript, e.g. 101_2 or 101_{16}

Other options …

- **Hexadecimal**
  - Leading 0x, e.g. 0x101, 0x747 — C language, Atmel AVR
  - Trailing h, e.g. 101h, 747h — Some assembly languages
  - Leading $, e.g. $747 — Atmel AVR Assembly
- **Octal**
  - Leading 0, e.g. 0101, 0747 — C language, Atmel AVR
  - Trailing q, e.g. 101q, 747q — Some assembly languages
  - Leading @, e.g. @747 — Some assembly languages
- **Binary**
  - Leading 0b, e.g. 0b101 — Atmel AVR Assembly, Some C
  - Trailing b, e.g. 101b — Some assembly languages
  - Leading %, e.g. %101 — Some assembly languages

We'll mostly use these

Reminders

- No labs on Mon-Wed of week 1 – i.e. only one session for week 1. From week 2, you will have 2 lab sessions per week.
- Lab 1 – you will learn more about different binary representation formats for signed numbers
- Sign up for a pair of lab sessions
- Any issues, email eait.mytimetable@uq.edu.au