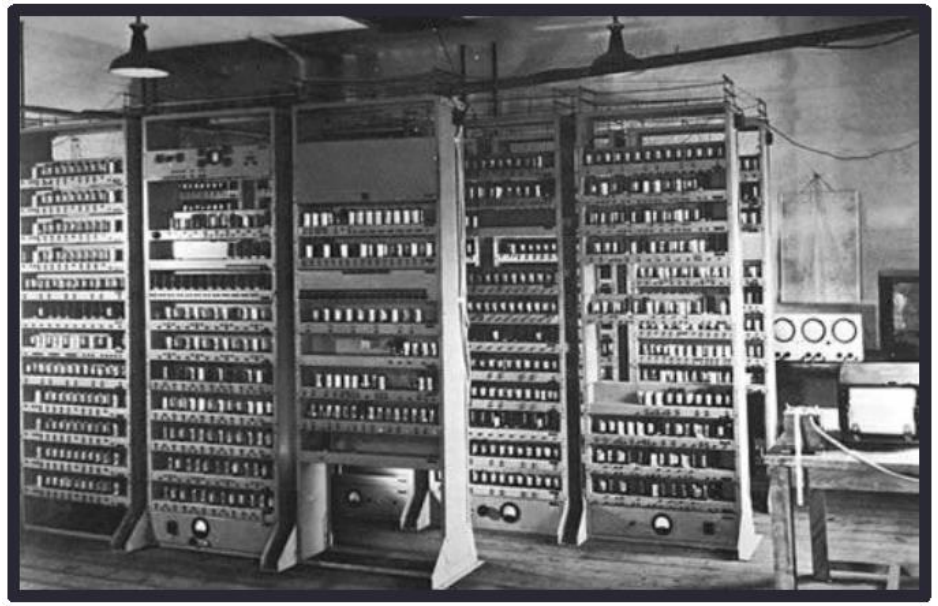
EEE 216 Microprocessor Systems (Part 1)



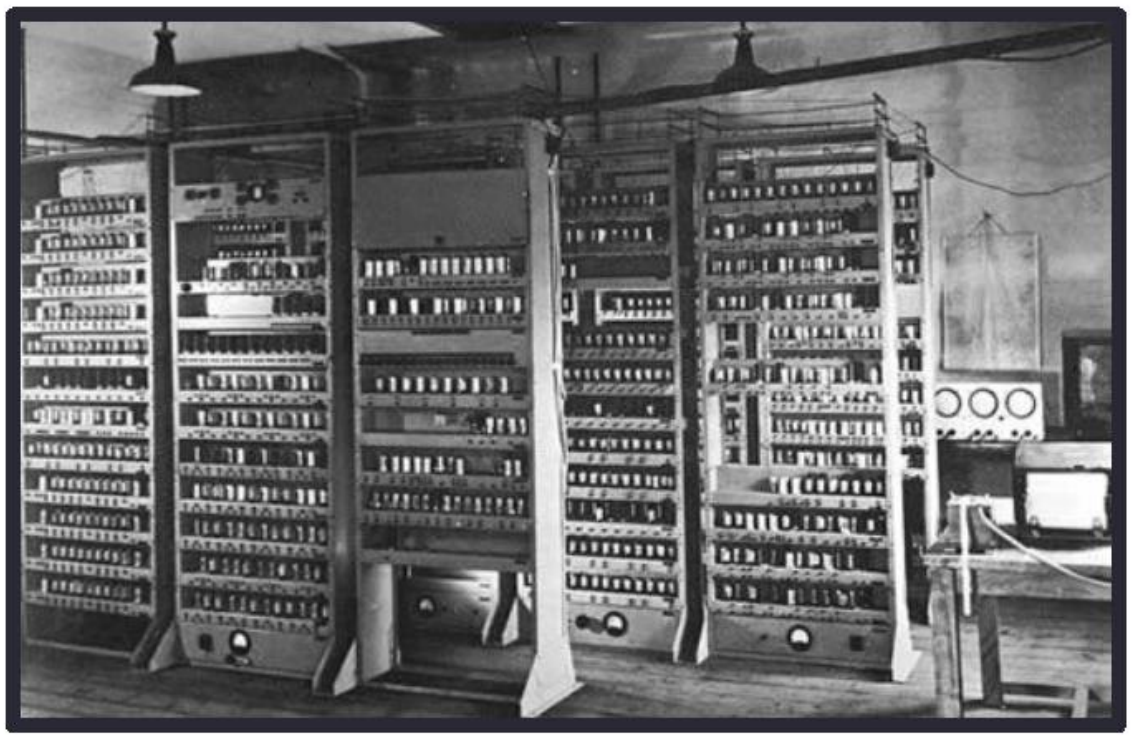
60 years

Who Am I?

- Dr Tony Centeno,
  - Assoc. Prof Department of Electrical and Electronic Engineering
  - Room: EB316
  - Email: Anthony.Centeno@xjtlu.edu.cn
  - Office Hours: Please see ICE
    - I will try and keep these hours free for students to come to my office to discuss any problems.
    - You can come at other times but please email first, to make sure I am available.

- EDSAC – first stored program computer
- Cambridge 1949
- 650 instructions per second.
- 1024 17-bit words of memory in mercury ultrasonic delay lines.
- 3000 valves, 12 kW power consumption, occupied a room 5m by 4m.
- Early use to solve problems in meteorology, genetics and X-ray crystallography.

60 years ago...



Electronic Delay Storage Automatic Computer

... and Now

- ARM7 core – up to 130 million instructions per second. 1995-2011.
- ARM7 core in many variations is most successful embedded processor today.
- Picture shows LPC2124 microcontroller which includes ARM7 core + RAM, ROM integrated peripherals.
  - The complete microcontroller is the square chip in the middle
  - 128K X 32 bit words flash RAM
  - 10mW/Mhz clock
- Original ARM design:
  - Steve Furber, Acorn Risc Machines, Cambridge, 1985

ARM7 CPU – LPC-2124 microcontroller

Why study this module?

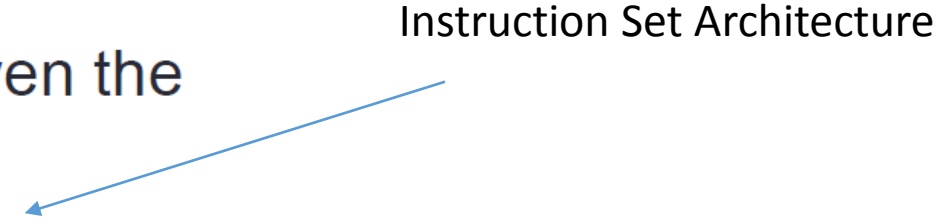
- https://www.intel.com/content/www/us/en/education/k12/the-journey-inside/explore-the-curriculum/microprocessors.html

Introduction:

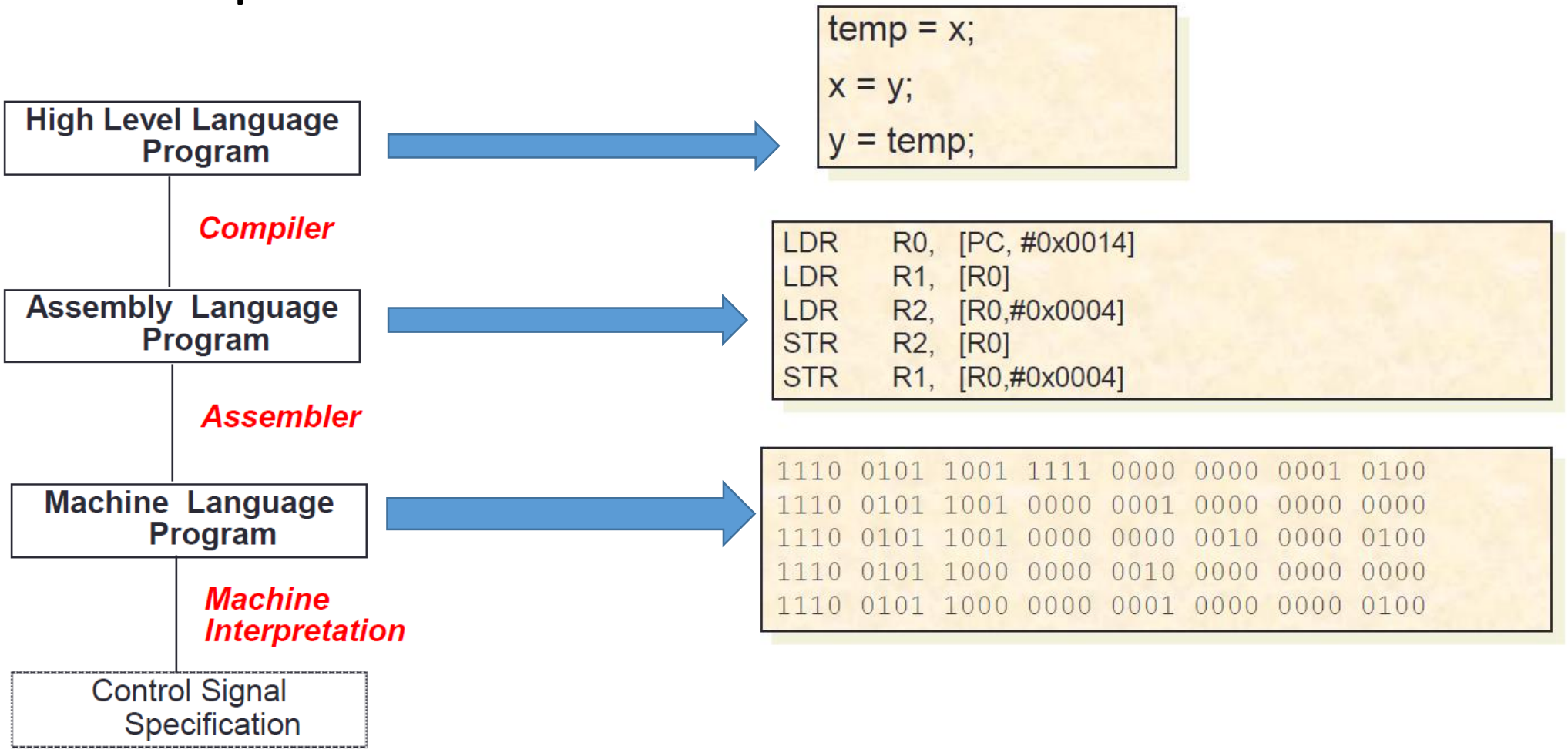
- This module is a “bridge” between digital electronics (hardware) and high level languages (software).
- You will learn basic concepts required to understand how computers work
  - What programs are.
  - How they are executed in hardware.

Part 1: Introduce the topic

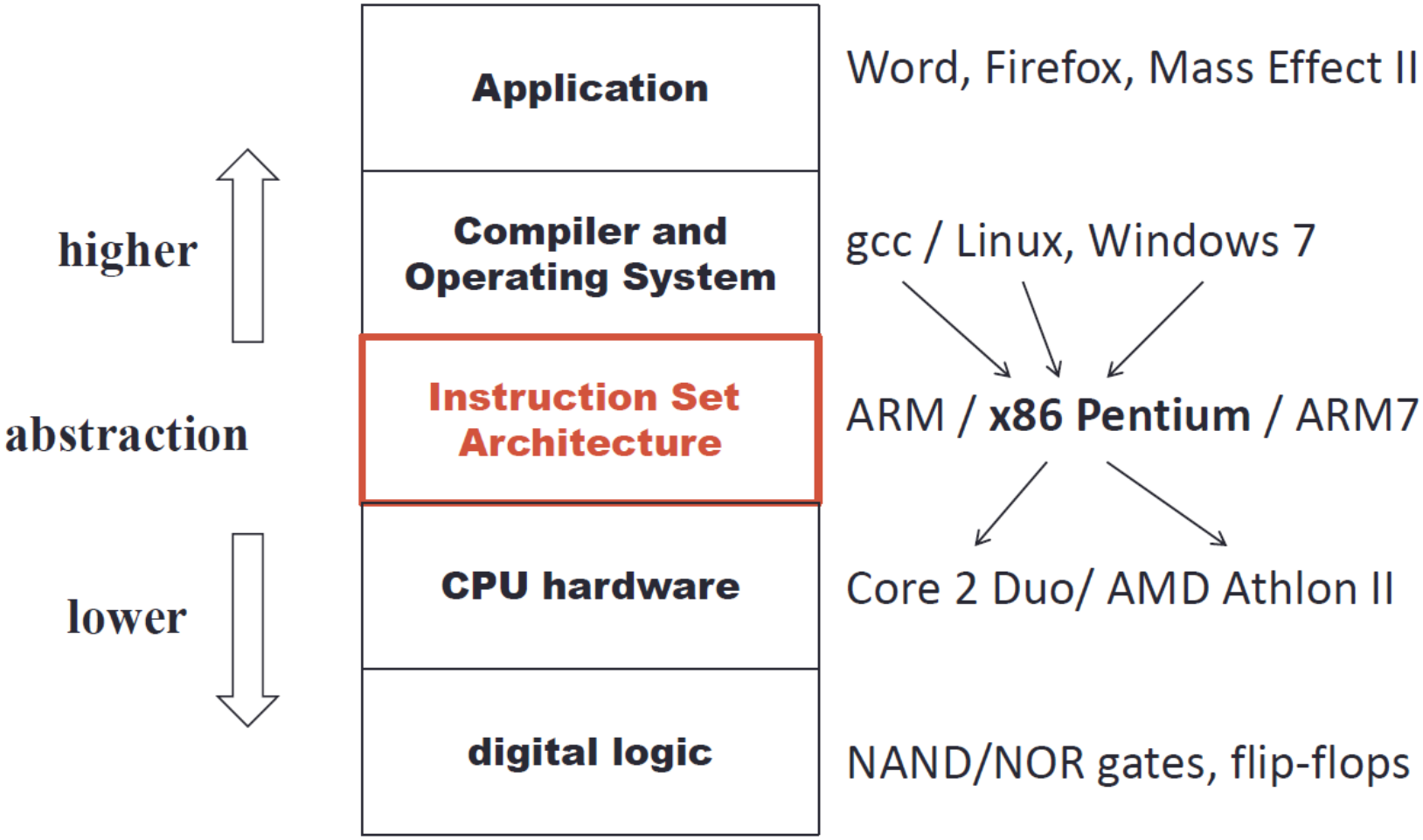
- What is computer architecture? What is a program? How do hardware and software relate?
- How has technological development driven the development of computers?
- What are the key characteristics of an ISA?
- How do assembler languages relate to machine code as representations of instructions?
- How are numbers represented in computers? How does arithmetic work (two's complement)?



Levels of representation in computers



What is a microprocessor system?



Instruction Set Architecture (ISA)

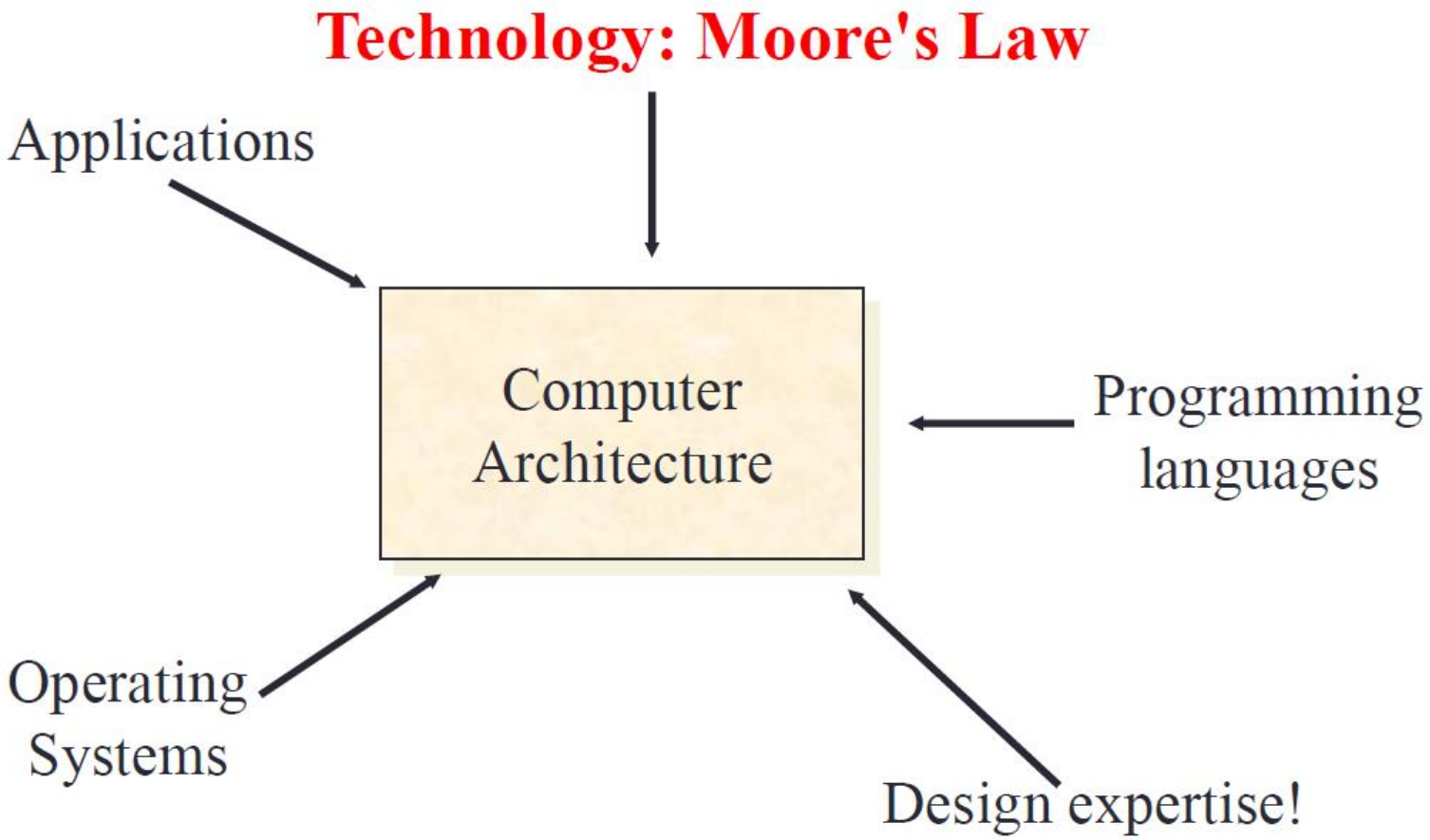
- What the computer does (not how it does it).

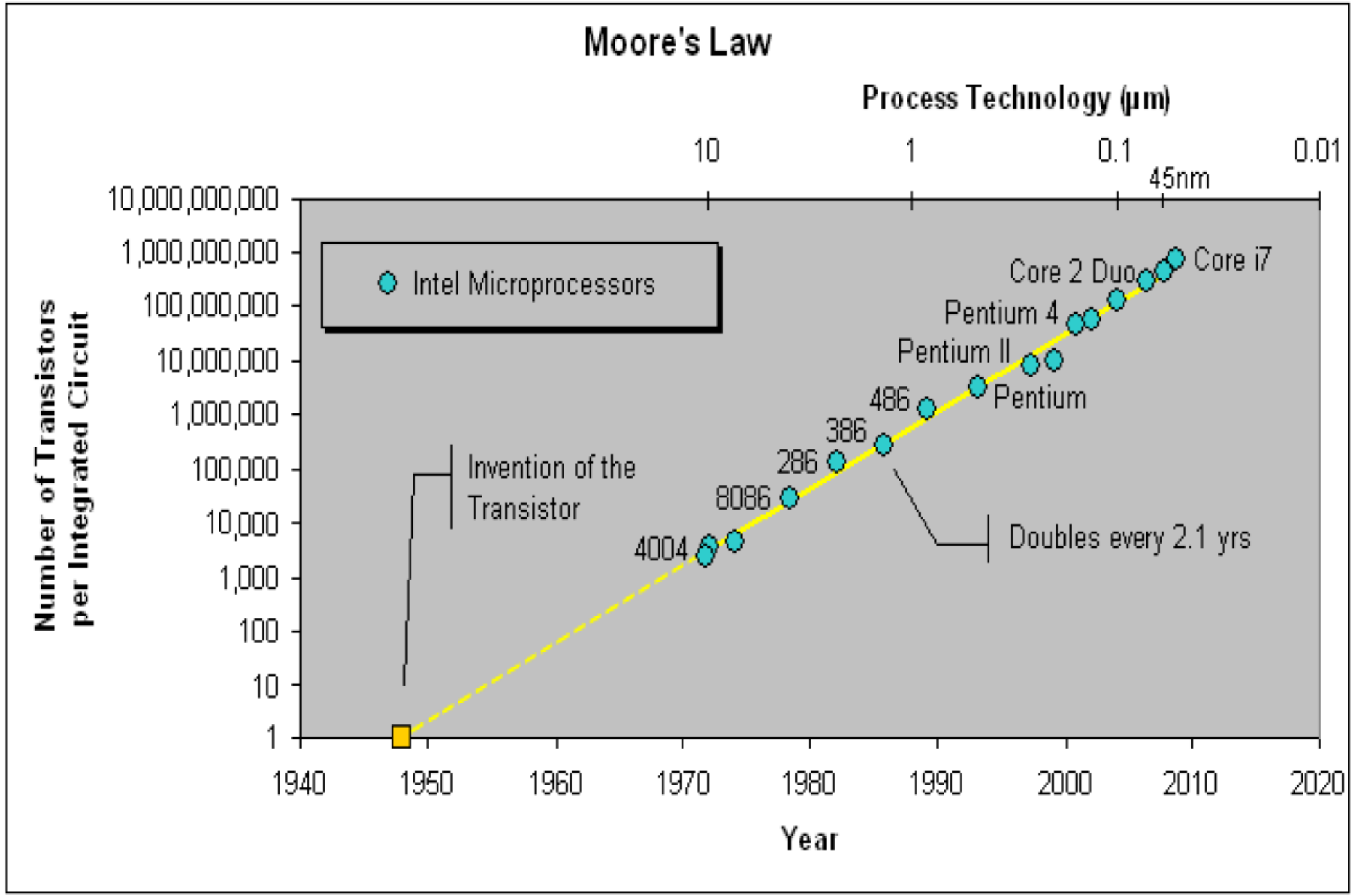
ISA includes:-

- **Instruction (or Operation Code) Set**
  - Data Types & Data Structures: Encodings & Representations
  - Instruction Formats
- Organization of Programmable Storage (main memory etc)
- Modes of Addressing and Accessing **Data Items** and **Instructions**
- Behaviour on Exceptional Conditions (e.g. hardware divide by 0)

- 8086/pentium ISA
  - Allows operating systems and applications to work seamlessly across different computer architectures:
    - Pentium
    - Core 2 duo
    - Athlon
    - Phenom
- ARM ISA
  - Supports highly optimising compiler & operating system software for embedded applications
  - ARM cores can be licensed & sold by different vendors
- Key advantage: **different implementations of the same ISA can all run identical software**

What factors influence computer architecture?

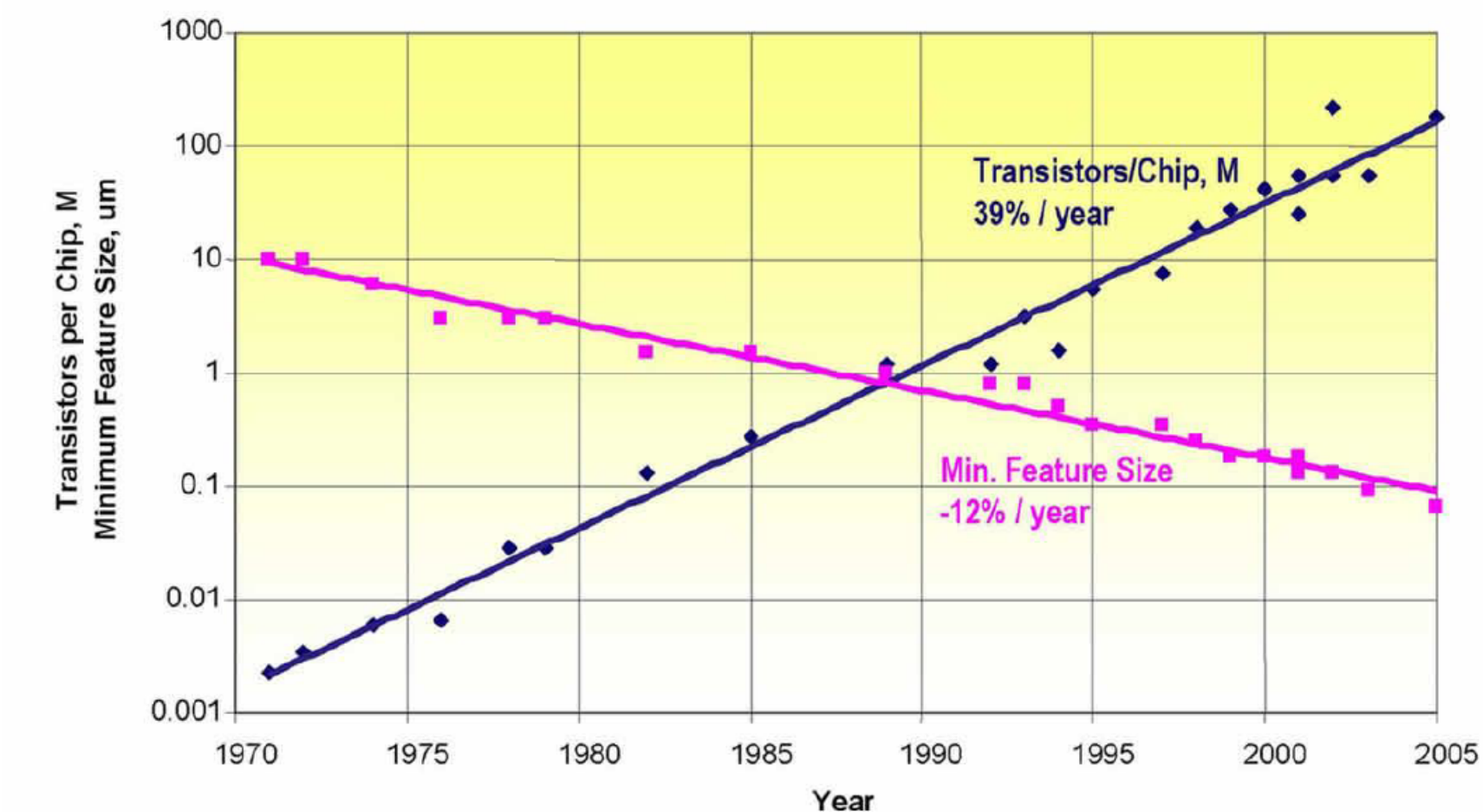



Moore's Law
Process Technology (µm)
10
1
0.1
0.01
45nm
Number of Transistors per Integrated Circuit
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
100
10
1
Intel Microprocessors
Invention of the Transistor
4004
8086
286
386
486
Pentium
Pentium II
Pentium 4
Core 2 Duo
Core i7
Doubles every 2.1 yrs
1940
1950
1960
1970
1980
1990
2000
2010
2020
Year

Semiconductor feature size drives technology

- **Transistors/cm^2** scales as (feature size)$^{-2}$
  - 2005 100nm
  - 2011 32nm
- **Speed** scales as (feature size)$^{-1/2}$
  - Other factors limit speed at very small sizes

Note:
Chips get bigger with time
10%/year of transistor increase is from larger chip size



When will Moore's law end?

- **Technology**
  - Gordon Moore coined Moore's law in 1965
  - Number of transistors per chip roughly doubles every 2-3 years
  - In 2007 Moore said it could last for another 10-15 years
  - The technological limits on making transistors smaller are the size of atoms - in 2007 transistor gates were 5 atoms thick!
  - Use of high-k Hafnium in gates has pushed capability beyond previous 45nm limits to 22nm now. Intel is now (2013) developing tools for 14nm, 10nm is predicted for 2015!
- **Economics**
  - Current generation CPUs (2013): 22nm
  - Globalfoundries' "Fab 8" in New York, 2012, 28nm, 5 years to build,
  - Intel's "Fab 42" Arizona, 2013, 14nm
  - Len Jelinek (chief analyst for at isuppli): 2014, around 20nm, will be end of economic return from new fabs. Moore's Law will stop.
  - Semiconductor manufacturers expect continued technological improvement in specific markets, e.g. non-volatile memory to replace hard disks

Semiconductor manufacturing processes

10 µm – 1971
6 µm – 1974
3 µm – 1977
1.5 µm – 1982
1 µm – 1985
800 nm – 1989
600 nm – 1994
350 nm – 1995
250 nm – 1997
180 nm – 1999
130 nm – 2001
90 nm – 2004
65 nm – 2006
45 nm – 2008
32 nm – 2010
22 nm – 2012
14 nm – 2014
10 nm – 2017
7 nm – 2018
5 nm – *~2020*

Half-nodes

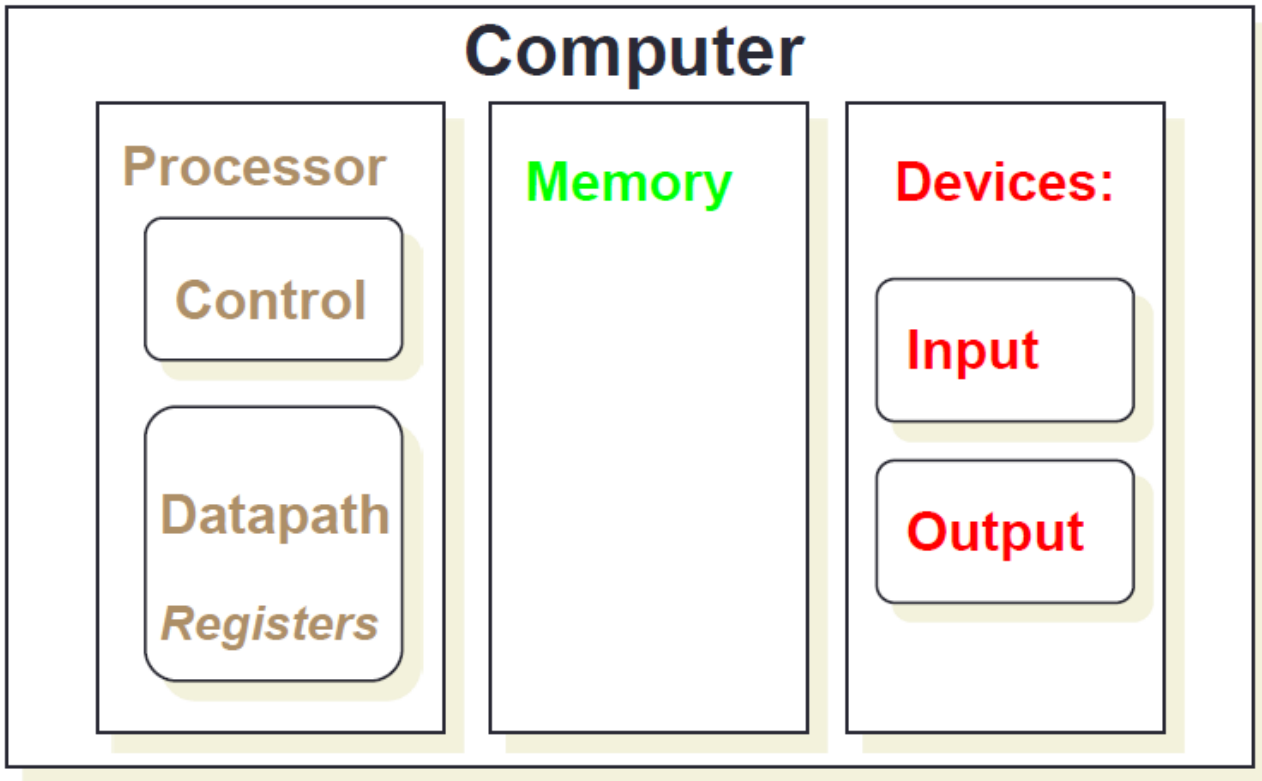
V·T·E

Summary:

- VLSI (very large scale integration) has increased speed and density of CPUs by shrinking dimensions
  - This is limited by size of atoms (Quantum effects), so will stop.
- Moore’s Law predicted a doubling of computing power every two years.
  - This has held true but will stop soon without a change in technology or architecture (Quantum computing).

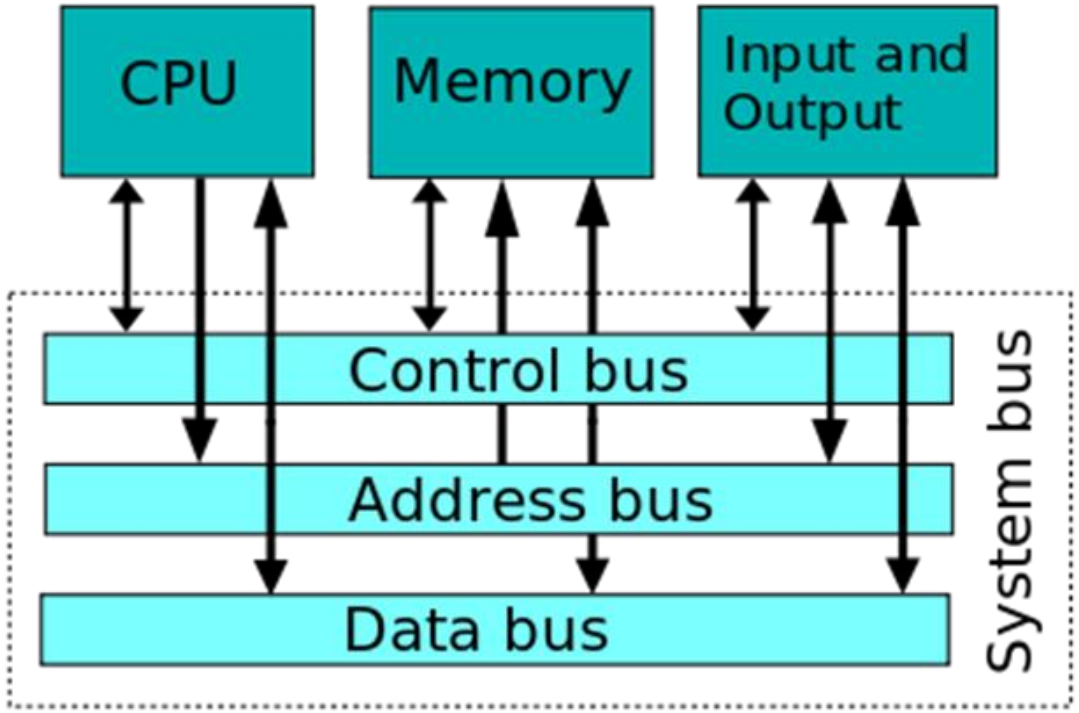
Internal Organisation of Computer

Processor aka CPU (Central Processing Unit)



- Major components of Typical Computer System
- Data is mostly stored in the computer memory separate from the Processor, however **registers** in the processor datapath can also store small amounts of data

A bit more detail:



Summary:

- Every computer consists of 5 components

  Central Processor Unit (CPU)

  1. Datapath
  2. Control

  3. Memory
  4. Input Devices
  5. Output Devices

In this module:

- We will focus on Instruction Set Architectures and their use through ***Assembly Language Programming***
- We will use the ARM processor as the main example.

Lecture capture

- From next week lectures will be videoed using the Lecture capture system.
- The videos will be made available on ICE.
- You should note that the lecture powerpoint presentations will be captured on the Video and will not be separately uploaded onto ICE.
- Research shows that students who make there own notes perform much better in exams:
  - You should make notes based on the lectures, captured video and independent study.
  - Your use of the lecture capture video will be monitored and I can see what you have accessed. If you ask for help I will check that you have accessed it and you must show me notes based on your independent study.