

Lecture 1:

Introduction

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

John Owens

Introduction to Computer Architecture

UC Davis EEC 170, Fall 2019

Credits

- These apply to the rest of the slides in this course!
- Thanks to the following for lecture material: *Computer Organization and Design*, Patterson & Hennessy, © 2005–2018, © Mary Jane Irwin / Penn State 2005, © David Patterson / UCB 2003, © John Kubiakowicz / UCB 2002, © Krste Asinovic/Arvind / MIT 2002, © Morgan Kaufmann Publishers 1998.

WeChat: cstutorcs

Agenda

Administrivia

History

What is Computer Architecture?

Technology

Assignment Project Exam Help

Course Outline

<https://tutorcs.com>

WeChat: cstutorcs

Teaching Staff

- Instructor
 - John Owens | jowens@ece | www.ece/~jowens/
 - Office hour: Coffee House, W 9–9:55 am (unless they move the class, in which case Silo)
Assignment Project Exam Help
- TA
 - Jason Mak | jwmak@ucdavis
WeChat: cstutorcs
 - Office hours: TBA
 - TBA
 - We've been promised a second TA

Electronic Resources

- **<https://canvas.ucdavis.edu/courses/367542>**
- **Lecture notes, other course material posted on Canvas**
- **All electronic communication will be handled through Slack**
 - Link on Canvas syllabus page
 - https://join.slack.com/t/eec170-f19/shared_invite/enQtNzQ4Nzc5NDgyNjU2LWVIMjkzZjkzZGUzMzU20WNIODU3Nzg20WU4M2M3ZDUwODMxYTE2NDZlMGM4ZTk2MzdmMThkYTY1MWRkNjc5NTI
Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs
- **Do not send email to instructional staff**
- **Use assignment/project/other public forums to ask public questions**
- **Group-message the instructional staff for private questions**

Classroom Time

- **Class is MW 10–11:50**
- **10–11:20: “Lecture”**
- **11:20–11:50: “Discussion”**
- **In reality: merged together**
Assignment Project Exam Help
 - Discussion mostly will be used for lecture
<https://tutorcs.com>
 - Also for project descriptions, etc.
- **Administrative announcements: Generally, in middle of class**

Lecture Style

- Students prefer blackboard
- It's hard to show complex diagrams on a blackboard though
 - Plus I like slides better
- I generally teach ideas more than examples
Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

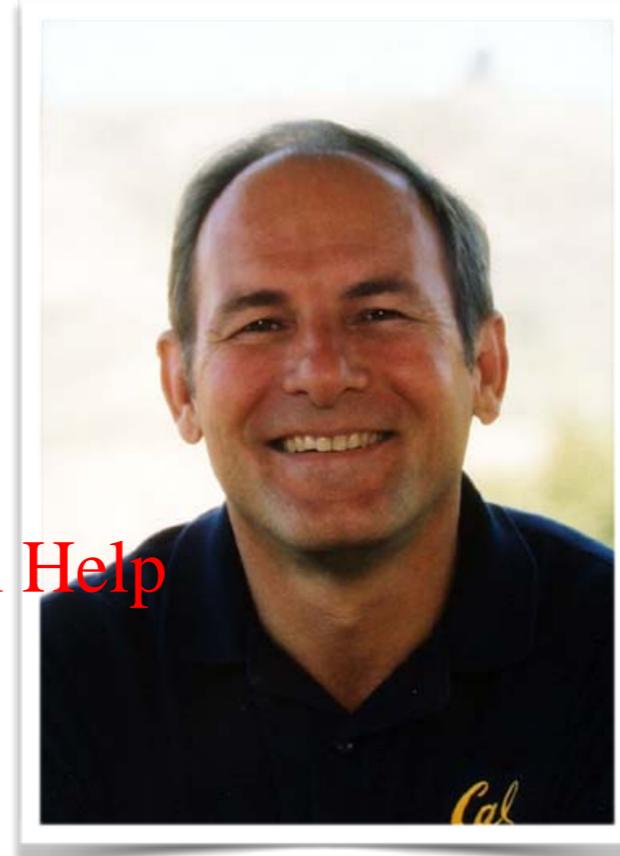
Textbook



Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs



- **David Patterson and John Hennessy, “Computer Organization and Design”, 5th/RISC-V edition (Morgan Kaufmann), 2017**
 - Don't get other versions
 - Syllabus has links to how you can get a 14-day eval electronic version

Grading

- Homework: 10% [minimally graded]
- Projects: 15% (expected: 3 total)
- Midterms: 2 @ 20% each
- Final: 35%
 - I will hand out a sample exam soon, on paper only
<https://tutorcs.com>

Assignment Project Exam Help

WeChat: cstutorcs

Important Dates

- Midterms
 - M 21 Oct
 - M 18 Nov
 - 2 hours in class
Assignment Project Exam Help
 - TAs will likely administer this
<https://tutorcs.com>
- Final WeChat: cstutorcs
 - M 9 Dec (10:30a–12:30p)
 - TAs will definitely administer this
- Makeups on midterm or final are oral only

Homework Turn-In

- Homework goes in 2131 Kemper
- Homework is due before class (10 am)
 - Do not bring hw to class
- Written homework must be done individually
Assignment Project Exam Help
- Homework and exam solutions will be on paper (in class)
<https://tutorcs.com>
 - We'll try to make ~~WeChat~~ solutions ASAP after due date
 - Please do not pass these solutions on to other students
 - Use of online/printed solutions to homework/projects is cheating
 - Passed out sample quizzes/midterms today; that's all you get

Things You Should Do

- Ask questions!
 - Especially when things aren't clear
- Give feedback!
 - Email or face-to-face
 - Assignment Project Exam Help
 - Tell me what I'm doing poorly
 - <https://tutorcs.com>
 - Tell me what I'm doing well
 - WeChat: cstutorcs
 - Tell the TAs too

Things You Shouldn't Do

- **Cheat**
- **Skip class**
- **Be late for class**
- **Read the paper in class**
Assignment Project Exam Help
- **Allow your cell phone to ring in class**
<https://tutorcs.com>
- **Ask OH questions without preparing**
 - Make sure you do the reading!
 - Identifying what you have trouble with helps me

Getting the Grade You Want

- *Do this*
 - Come to class!
 - Ask questions in class when you don't understand
 - Assignment
 - Project
 - Exam
 - Help
 - Come to office hours (mine and TA's)
 - Start the hw early
 - Understand the course material
 - *Don't do this*
 - Cheat
 - Skip class (I'll know when you're not there!)
 - Be late for class
 - Read the paper in class
 - Open your laptop in class unless there's a good reason
 - Allow your cell phone to ring in class
- <https://tutorcs.com>
WeChat: cstutorcs

My Goals

- This class was hard
- I learned a lot
- The work I did in this class was worthwhile
- The instructor was fair
Assignment Project Exam Help
- The instructor was effective
<https://tutorcs.com>
- The instructor cared about my learning
WeChat: daturcs

The Computer Revolution

- Progress in computer technology
 - Underpinned by Moore's Law
- Makes novel applications feasible
 - Computers in automobiles
Assignment Project Exam Help
 - Cell phones
<https://tutorcs.com>
 - Human genome project
WeChat: cstutorcs
 - World Wide Web
 - Search engines
- Computers are pervasive

Classes of Computers

■ Personal computers

- General purpose, variety of software
- Subject to cost/performance tradeoff

■ Server computers

- Network based
- High capacity, performance, reliability
- Range from small servers to building sized

■ Supercomputers

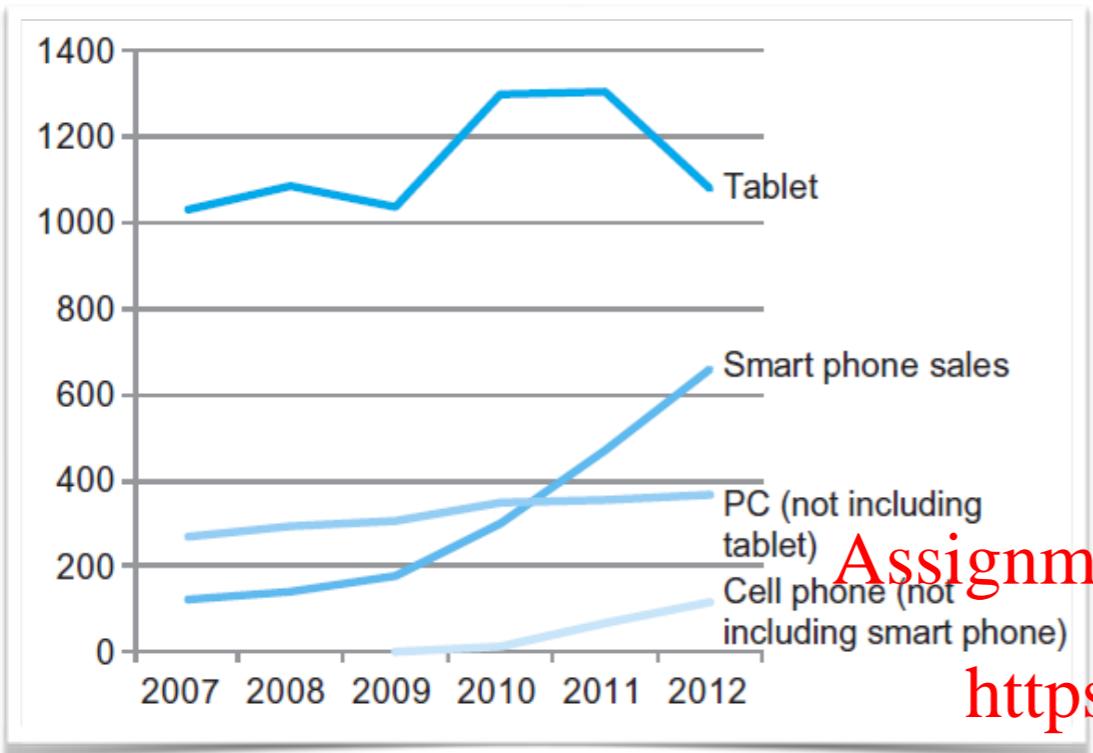
WeChat: cstutorcs

- High-end scientific and engineering calculations
- Highest capability but represent a small fraction of the overall computer market

■ Embedded computers

- Hidden as components of systems
- Stringent power/performance/cost constraints

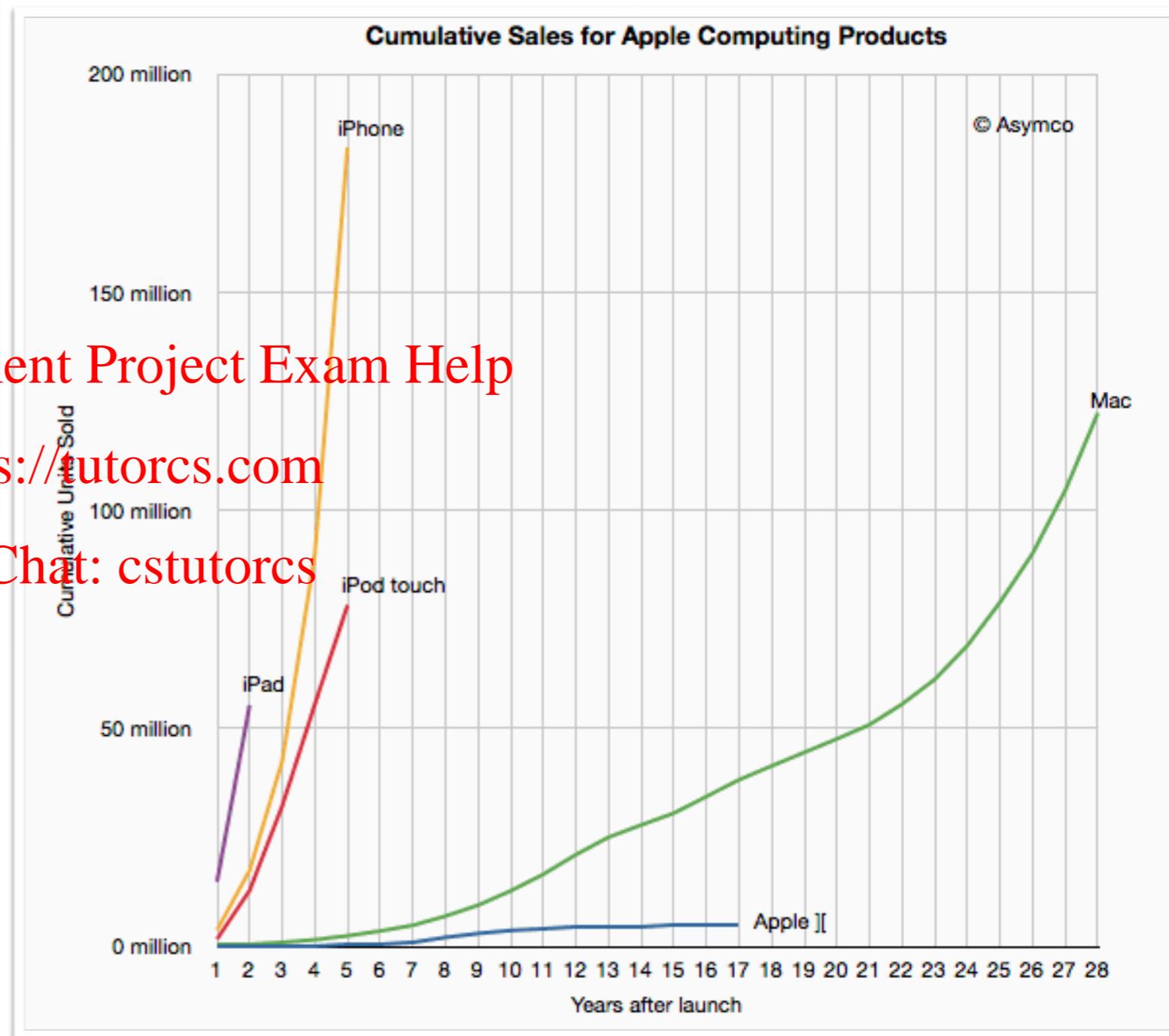
The PostPC Era



Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs



<https://blog.codinghorror.com/welcome-to-the-post-pc-era/>

The PostPC Era

- **Personal Mobile Device (PMD)**
 - **Battery operated**
 - **Connects to the Internet**
 - **Hundreds of dollars**
 - **Smart phones, tablets, electronic glasses**
Assignment Project Exam Help
<https://tutorcs.com>
- **Cloud computing**
 - **Warehouse Scale Computers (WSC)**
 - **Software as a Service (SaaS)**
 - **Portion of software run on a PMD and a portion run in the Cloud**
 - **Amazon and Google**

What You Will Learn

- How programs are translated into the machine language
 - And how the hardware executes them
- The hardware/software interface
- What determines program performance
Assignment Project Exam Help
 - And how it can be improved
<https://tutorcs.com>
- How hardware design We can improve performance improves performance
- What is parallel processing (but only a little of this)

Understanding Performance

■ Algorithm

- **Determines number of operations executed**

■ Programming language, compiler, architecture

- **Determine number of machine instructions executed per operation**
Assignment Project Exam Help

■ Processor and memory system

<https://tutorcs.com>

- **Determine how fast instructions are executed**
WeChat: cstutorcs

■ I/O system (including OS)

- **Determines how fast I/O operations are executed**

Eight Great Ideas

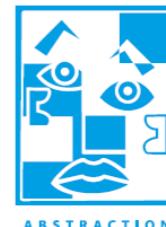
- Design for *Moore's Law*
- Use *abstraction* to simplify design
- Make the *common case fast*
- Performance via *parallelism*
Assignment Project Exam Help
<https://tutorcs.com>
- Performance via *pipelining*
WeChat: cstutorcs
- Performance via *prediction*
- *Hierarchy* of memories
- *Dependability* via redundancy



MOORE'S LAW



COMMON CASE FAST



ABSTRACTION



PARALLELISM



PIPELINING



PREDICTION



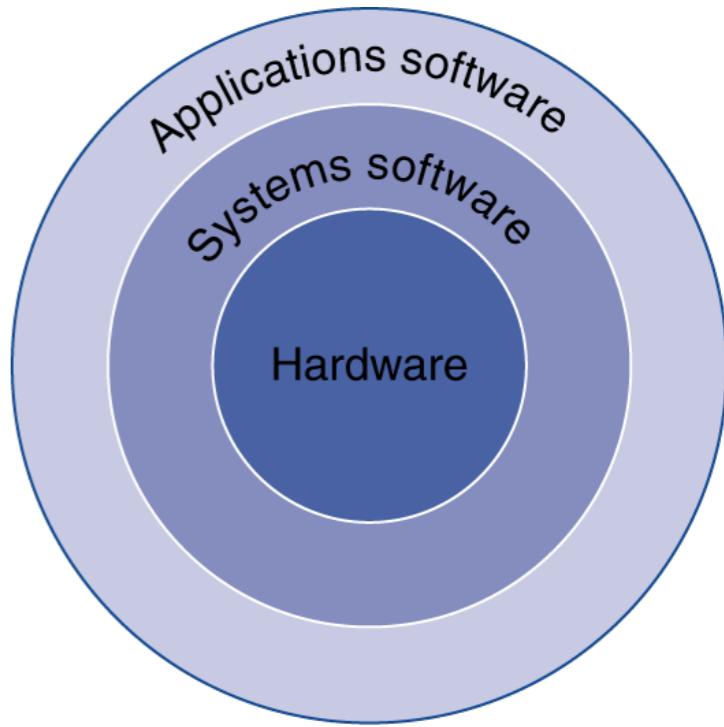
HIERARCHY



DEPENDABILITY

Below Your Program

- Application software
 - Written in high-level language
- System software
 - Compiler: translates HLL code to machine code
Assignment Project Exam Help
 - Operating System: service code
<https://tutorcs.com>
 - Handling input/output
WeChat: cstutorcs
 - Managing memory and storage
 - Scheduling tasks & sharing resources
- Hardware
 - Processor, memory, I/O controllers



Levels of Program Code

■ High-level language

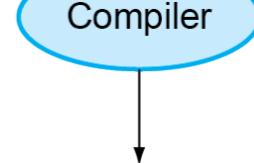
- Level of abstraction closer to problem domain
- Provides for productivity and portability

Assignment Project Exam Help

<https://tutorcs.com>

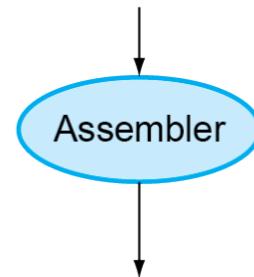
High-level
language
program
(in C)

```
swap(int v[], int k)
{int temp;
 temp = v[k];
 v[k] = v[k+1];
 v[k+1] = temp;
}
```



Assembly
language
program
(for RISC-V)

```
swap:
slli x6, x11, 3
add x6, x10, x6
ld x5, 0(x6)
ld x7, 8(x6)
sd x7, 0(x6)
sd x5, 8(x6)
jalr x0, 0(x1)
```



Binary machine
language
program
(for RISC-V)

```
00000000001101011001001100010011
00000000011001010000001100110011
000000000000000110011001010000011
00000000100000110011001110000011
00000000011100110011000000100011
00000000010100110011010000100011
00000000000000000000100000001100111
```

■ Assembly language

- Textual representation of instructions

WeChat: cstutorcs

■ Hardware representation

- Binary digits (bits)
- Encoded instructions and data

Components of a Computer

The BIG Picture

- Same components for all kinds of computer

- Desktop, server, embedded

- Input/output includes

- User-interface devices

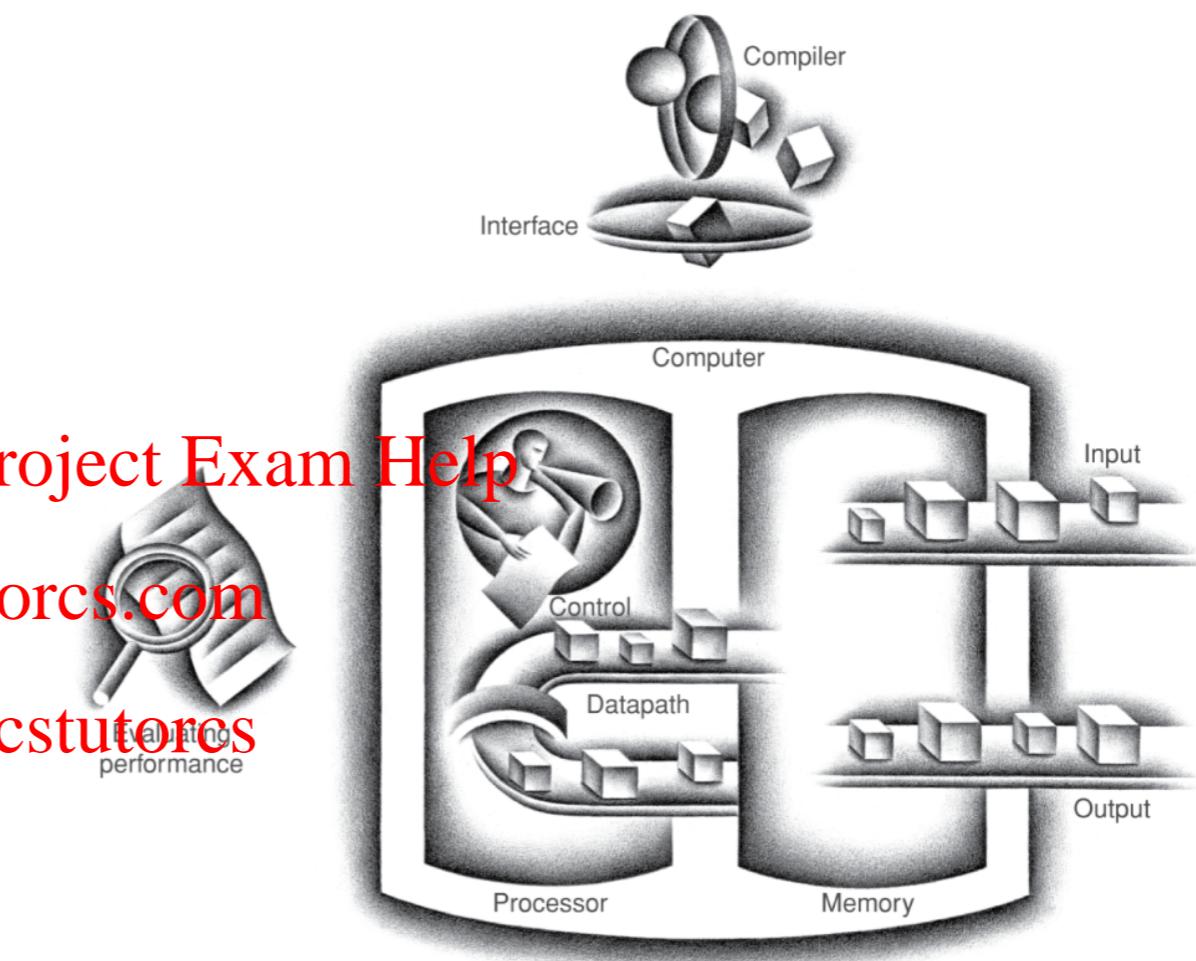
- Display, keyboard, mouse

- Storage devices

- Hard disk, CD/DVD, flash

- Network adapters

- For communicating with other computers



Touchscreen

- PostPC device
- Supersedes keyboard and mouse
- Resistive and Capacitive types
 - Most tablets, smart phones use capacitive
 - Capacitive allows multiple touches simultaneously

Assignment Project Exam Help

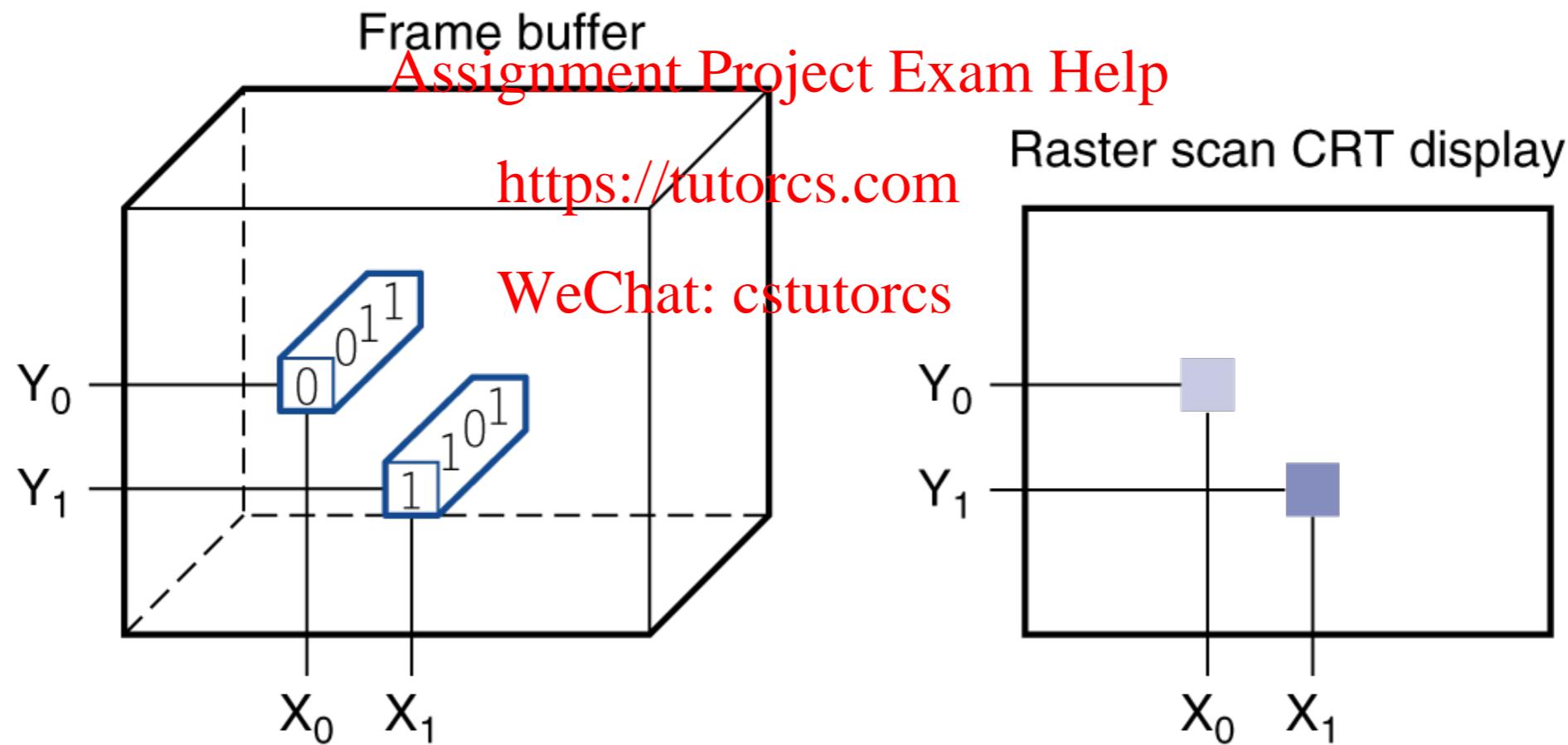
<https://tutorcs.com>

WeChat: cstutorcs

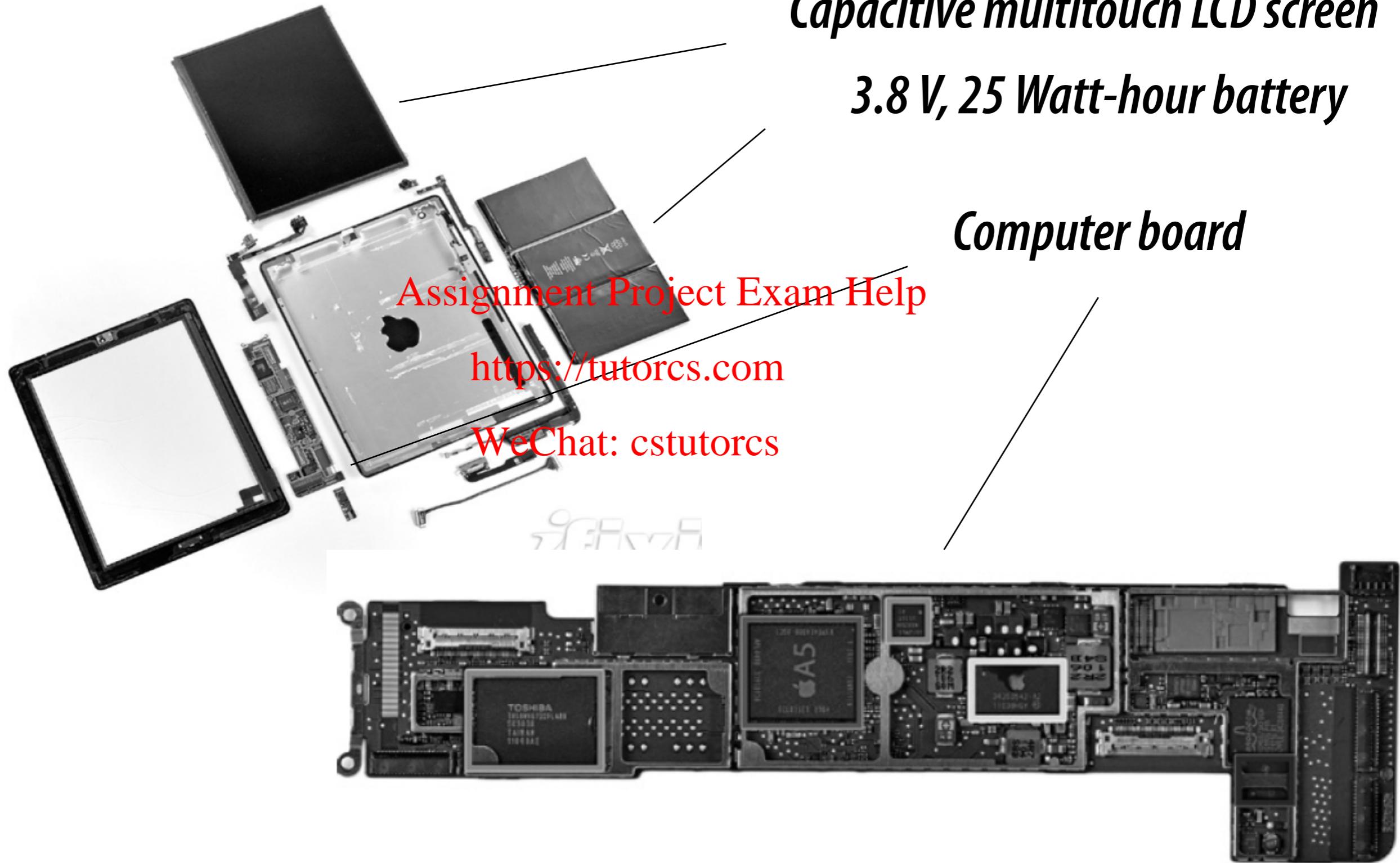


Through the Looking Glass

- LCD screen: picture elements (pixels)
 - Mirrors content of frame buffer memory



Opening the Box



Inside the Processor (CPU)

- **Datapath: performs operations on data**
- **Control: sequences datapath, memory, ...**
- **Cache memory**
 - **Small fast SRAM memory for immediate access to data**
Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Inside the Processor

■ Apple A5



Abstractions

The BIG Picture

- Abstraction helps us deal with complexity
 - Hide lower-level detail
- Instruction set architecture (ISA)
 - The hardware/software interface
Assignment Project Exam Help
- Application binary interface <https://tutorcs.com>
 - The ISA plus system software interface
- Implementation
 - The details underlying and interface

A Safe Place for Data

■ Volatile main memory

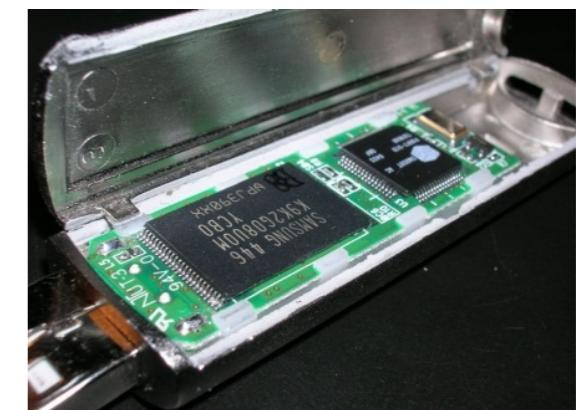
- Loses instructions and data when power off

■ Non-volatile secondary memory

- Magnetic disk
- Flash memory
- Optical disk (CDROM, DVD)

Assignment Project Exam Help

WeChat: cstutorcs



Networks

- **Communication, resource sharing, nonlocal access**
- **Local area network (LAN): Ethernet**
- **Wide area network (WAN): the Internet**
- **Wireless network: WiFi, Bluetooth**
Assignment Project Exam Help

<https://tutorcs.com>

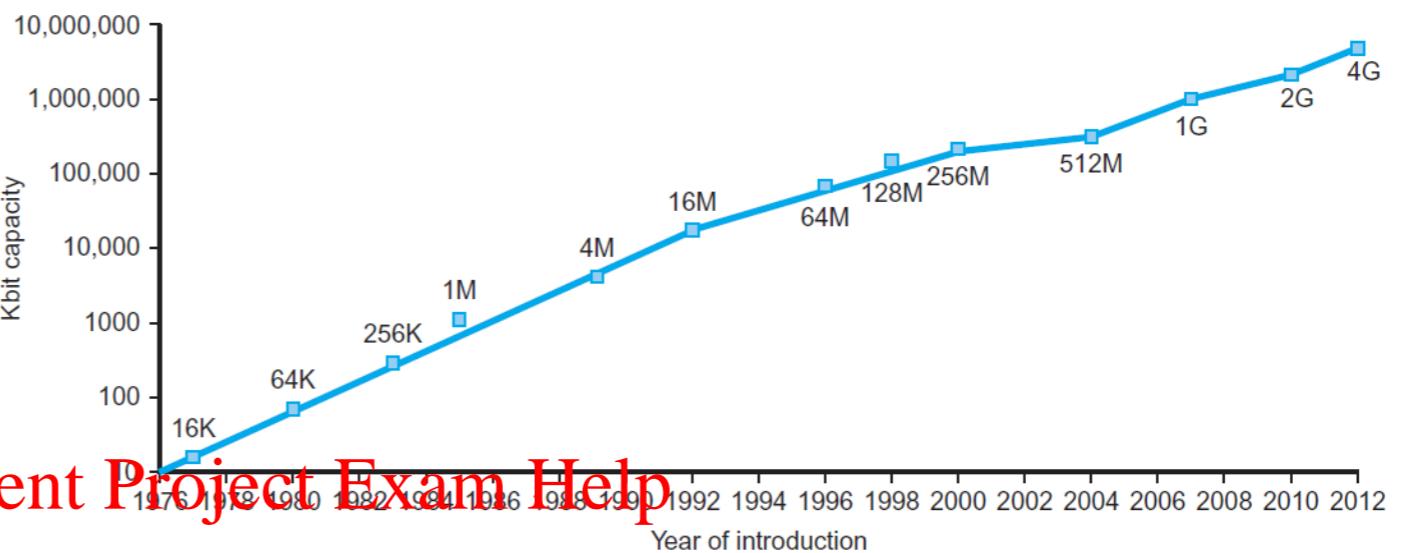
WeChat: cstutorcs



Technology Trends

■ Electronics technology continues to evolve

- Increased capacity and performance
- Reduced cost



Assignment Project Exam Help

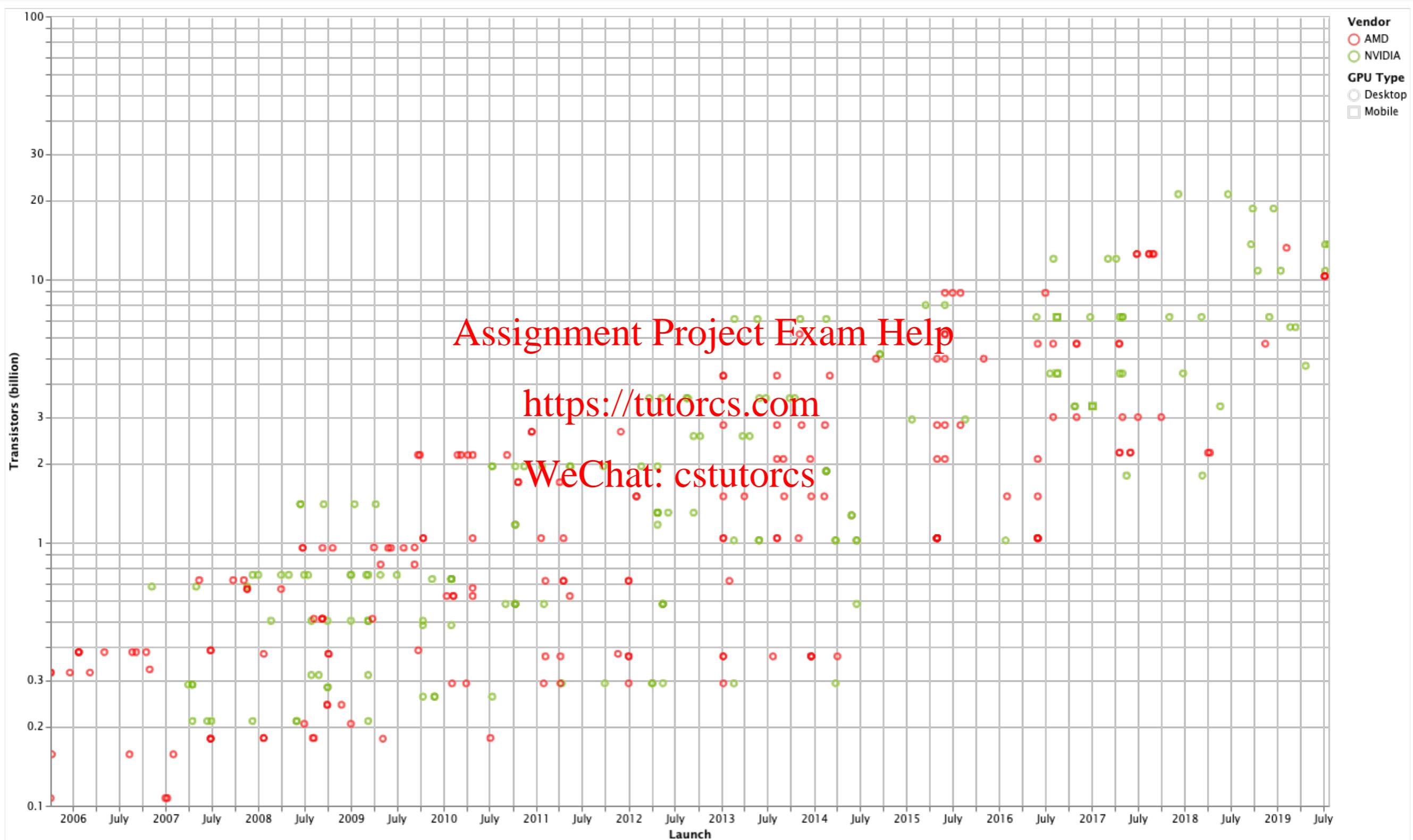
<https://tutorcs.com>

DRAM capacity

WeChat: cstutorcs

Year	Technology	Relative performance/cost
1951	Vacuum tube	1
1965	Transistor	35
1975	Integrated circuit (IC)	900
1995	Very large scale IC (VLSI)	2,400,000
2013	Ultra large scale IC	250,000,000,000

GPUs: Transistors vs. Time



Semiconductor Technology

- **Silicon: semiconductor**
- **Add materials to transform properties:**

- **Conductors**
- **Insulators**
- **Switch**

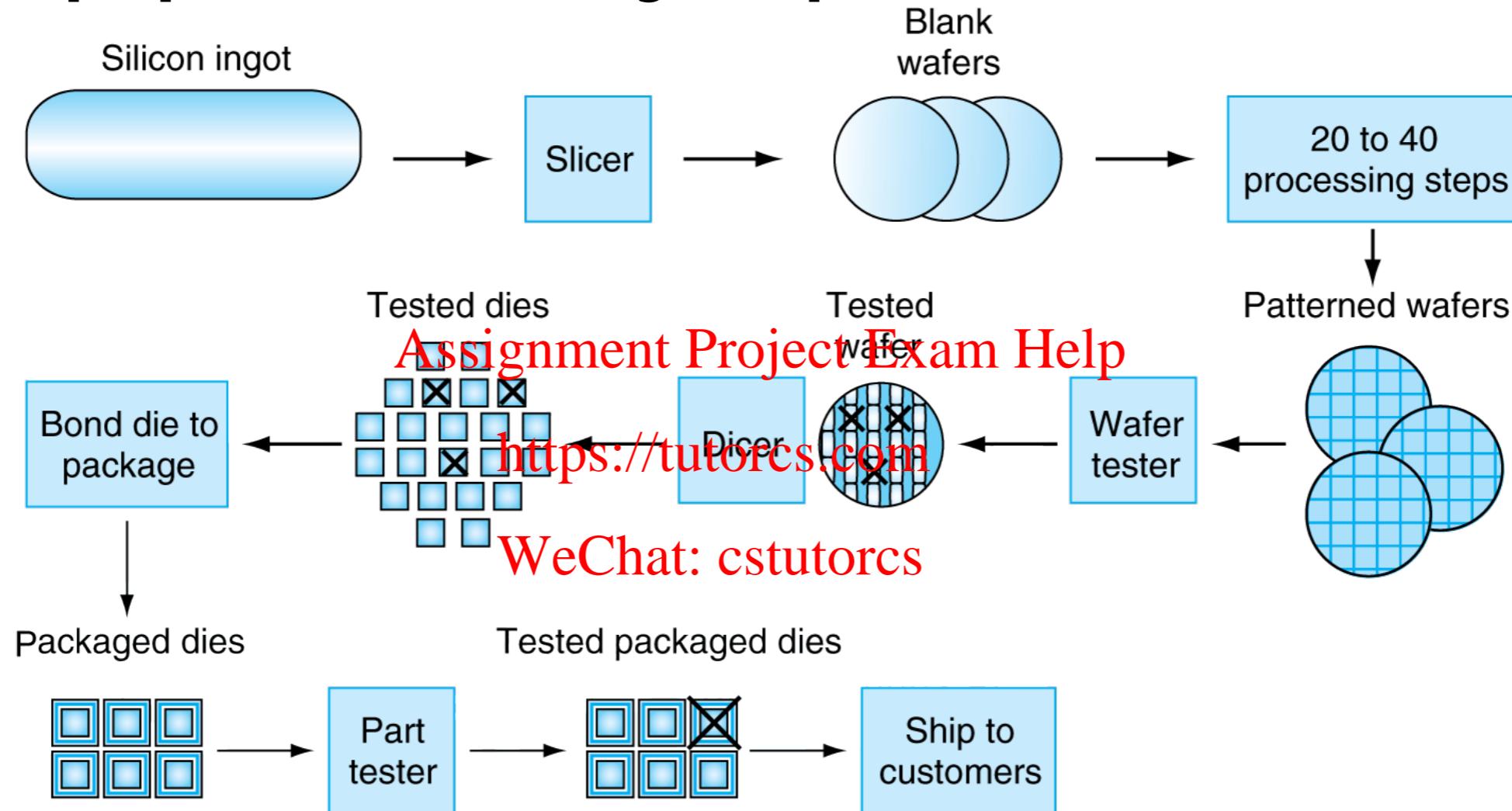
Assignment Project Exam Help

<https://tutorcs.com>

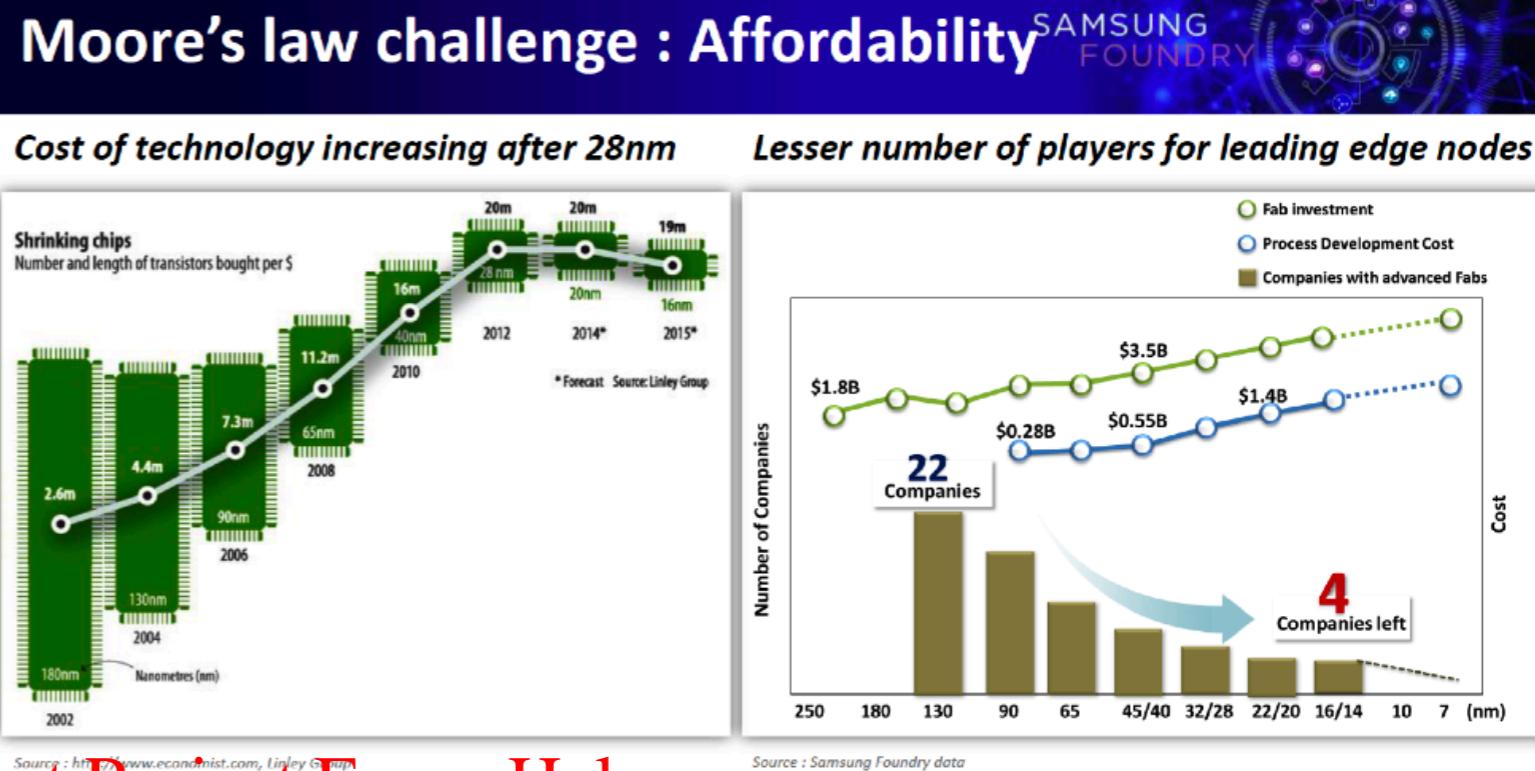
WeChat: cstutorcs

Manufacturing ICs

■ Yield: proportion of working dies per wafer



Cost of Fabs



Assignment Project Exam Help

SEMICON West 2016 - Node 10 to node 5

SAMSUNG Samsung Semiconductor USA

<https://tutorcs.com>

- Rock's Law: Cost of fabs double every 4 years
- \$14B for Samsung's "18 Line" Pyeongtaek fab (2017 opening)
 - Rise of fabless design houses
 - Rise of for-hire fabs (TSMC, etc.)
 - \$14B fab means \$28B in revenue is required

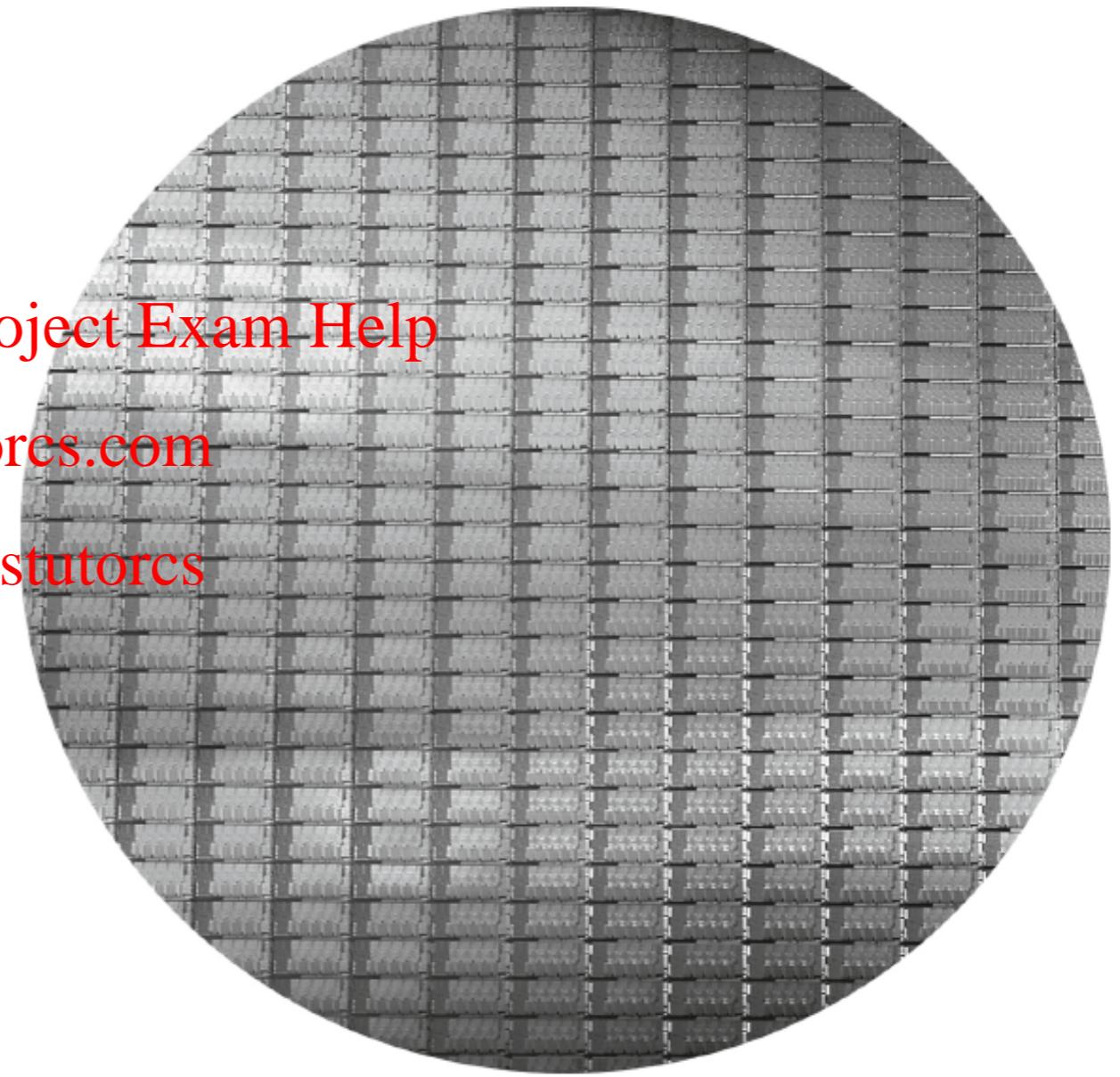
Intel Core i7 Wafer

- **300mm wafer, 280 chips, 32nm technology**
- **Each chip is 20.7 x 10.5 mm**

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs



Integrated Circuit Cost

■ Nonlinear relation to area and defect rate

- Wafer cost and area are fixed
- Defect rate determined by manufacturing process
- Die area determined by architecture and circuit design

Assignment Project Exam Help

<https://tutorcs.com>

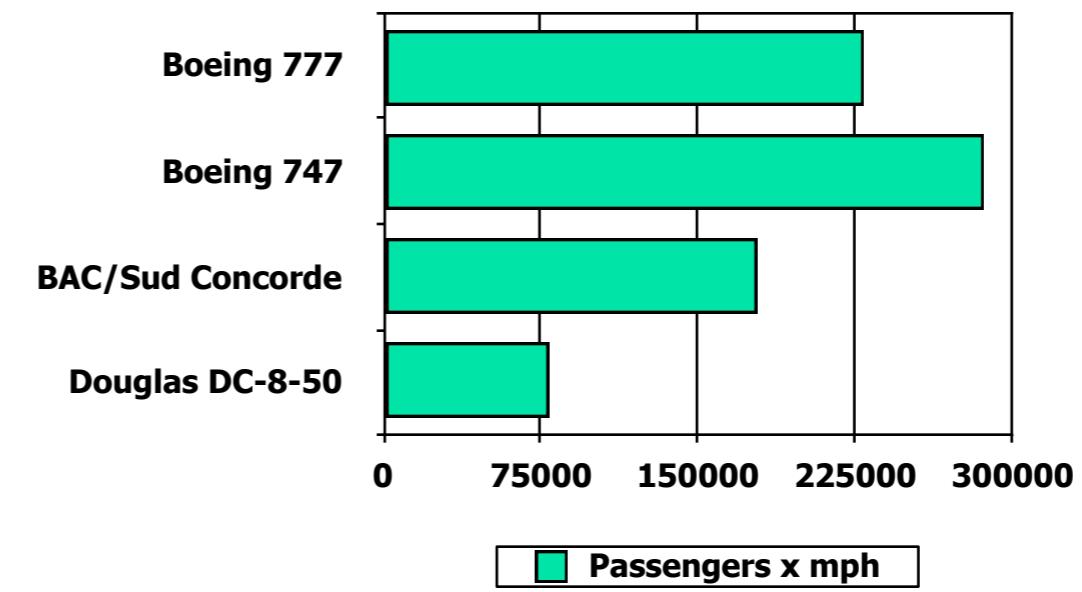
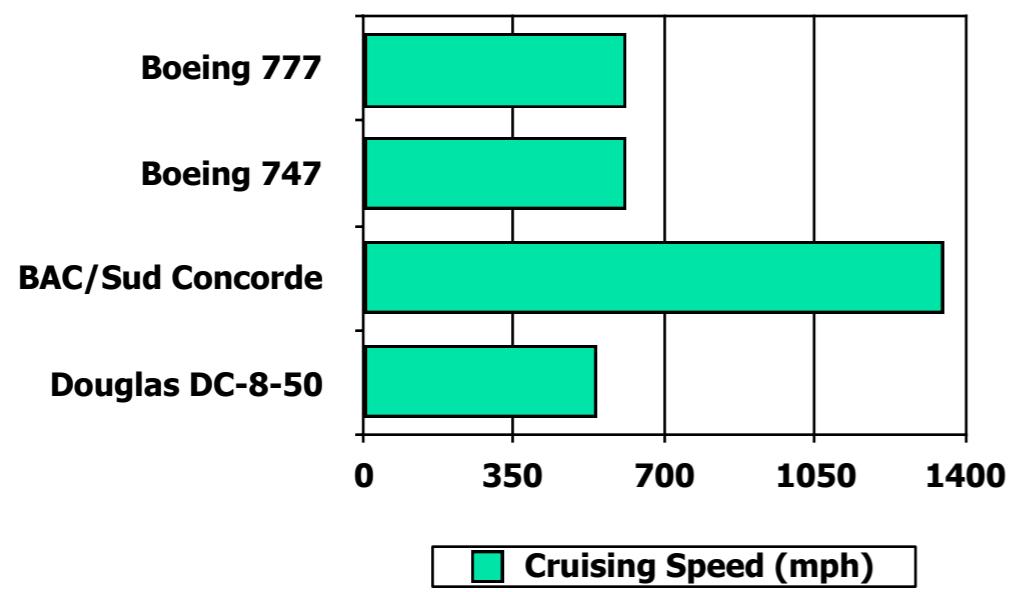
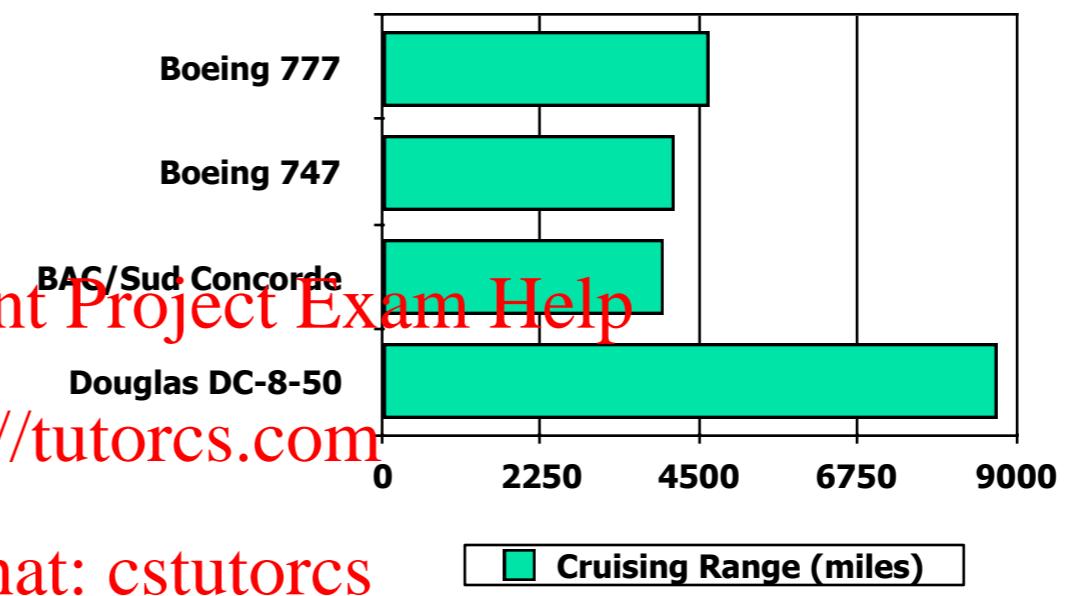
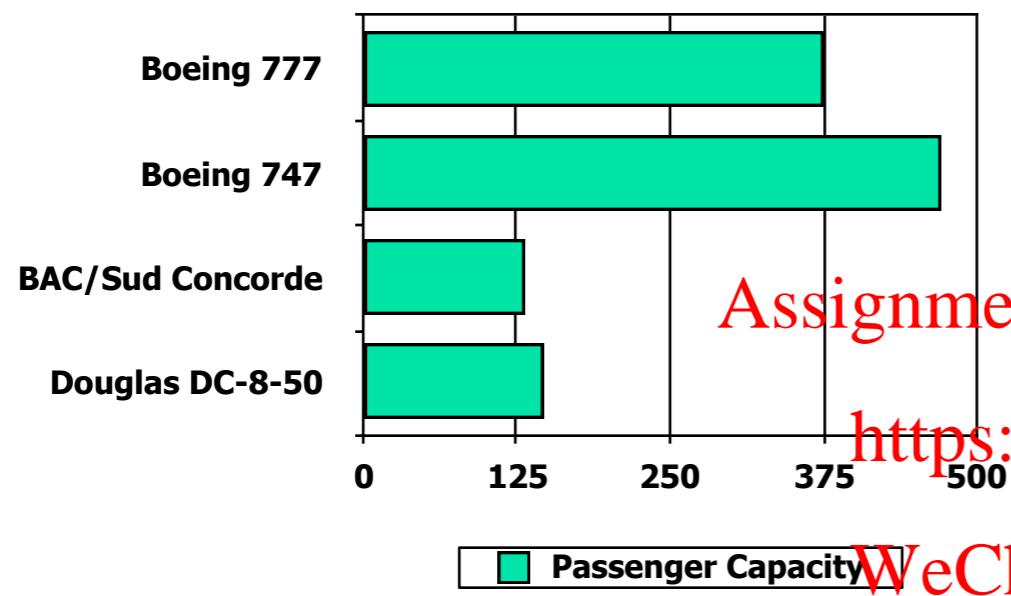
$$\text{Cost per die} = \frac{\text{Cost per wafer}}{\text{Dies per wafer} \times \text{Yield}}$$

$$\text{Dies per wafer} \approx \text{Wafer area}/\text{Die area}$$

$$\text{Yield} = \frac{1}{(1 + (\text{Defects per area} \times \text{Die area}/2))^2}$$

Defining Performance

- Which airplane has the best performance?



Response Time and Throughput

- Response time
 - How long it takes to do a task
- Throughput
 - Total work done per unit time
 - e.g., tasks/transactions/... per hour
- How are response time and throughput affected by
 - Replacing the processor with a faster version?
 - Adding more processors?
- We'll focus on response time for now...

Relative Performance

- Define Performance = 1/Execution Time
- “X is n time faster than Y”

$$\frac{\text{Performance}_x}{\text{Performance}_y} = \frac{\text{Execution time}_y}{\text{Execution time}_x} = n$$

<https://tutorcs.com>

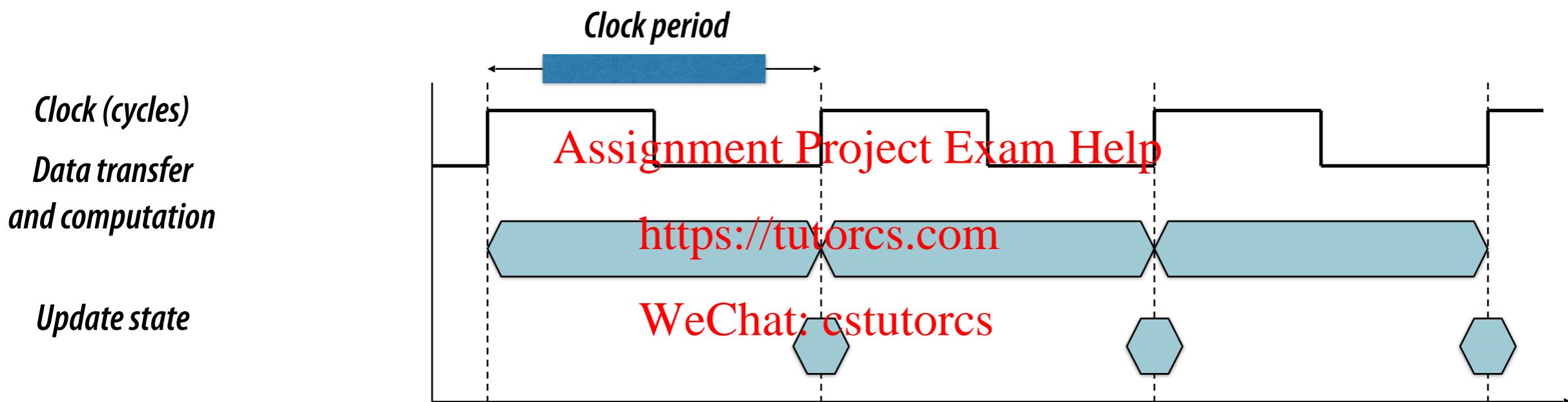
- Example: time taken to run a program
 - 10s on A, 15s on B
 - $\frac{\text{Execution Time}_B}{\text{Execution Time}_A} = 15s / 10s = 1.5$
 - So A is 1.5 times faster than B

Measuring Execution Time

- Elapsed time
 - Total response time, including all aspects
 - Processing, I/O, OS overhead, idle time
 - Determines system performance
Assignment Project Exam Help
- CPU time
 - Time spent processing We can't control what other jobs do a given job
 - Discounts I/O time, other jobs' shares
 - Comprises user CPU time and system CPU time
 - Different programs are affected differently by CPU and system performance

CPU Clocking

- Operation of digital hardware governed by a constant-rate clock



-

Clock period: duration of a clock cycle

$$\text{e.g., } 250\text{ps} = 0.25\text{ns} = 250 \times 10^{-12}\text{s}$$

-

Clock frequency (rate): cycles per second

$$\text{e.g., } 4.0\text{GHz} = 4000\text{MHz} = 4.0 \times 10^9\text{Hz}$$

CPU Time

- Performance improved by
 - Reducing number of clock cycles
 - Increasing clock rate
 - Hardware designer must often trade off clock rate against cycle count

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

CPU Time = CPU Clock Cycles \times Clock Cycle Time

$$= \frac{\text{CPU Clock Cycles}}{\text{Clock Rate}}$$

CPU Time Example

- Computer A: 2GHz clock, 10s CPU time
- Designing Computer B
 - Aim for 6s CPU time
 - Can do faster clock, but causes $1.2 \times$ clock cycles
- How fast must Computer B clock be? Assignment Project Exam Help

$$\text{Clock Rate}_B = \frac{\text{Clock Cycles}_B}{\text{CPU Time}_B} = \frac{1.2 \times \text{Clock Cycles}_A}{6s}$$

$$\begin{aligned}\text{Clock Cycles}_A &= \text{CPU Time}_A \times \text{Clock Rate}_A \\ &= 10s \times 2\text{GHz} = 20 \times 10^9\end{aligned}$$

$$\text{Clock Rate}_B = \frac{1.2 \times 20 \times 10^9}{6s} = \frac{24 \times 10^9}{6s} = 4\text{GHz}$$

Instruction Count and CPI

- **Instruction Count for a program**
 - Determined by program, ISA and compiler
- **Average cycles per instruction**
 - Determined by CPU hardware
Assignment Project Exam Help
<https://tutorcs.com>
 - If different instructions have different CPI
<https://tutorcs.com>
WeChat: cstutorcs
 - Average CPI affected by instruction mix
WeChat: cstutorcs

Clock Cycles = Instruction Count × Cycles per Instruction

CPU Time = Instruction Count × CPI × Clock Cycle Time

$$= \frac{\text{Instruction Count} \times \text{CPI}}{\text{Clock Rate}}$$

CPI Example

- Computer A: Cycle Time = 250ps, CPI = 2.0
- Computer B: Cycle Time = 500ps, CPI = 1.2
- Same ISA
- Which is faster, and by how much?

Assignment Project Exam Help

$$\begin{aligned}\text{CPU Time}_A &= \text{Instruction Count} \times \text{CPI}_A \times \text{Cycle Time}_A \\ &= I \times 2.0 \times 250\text{ps} = I \times 500\text{ps}\end{aligned}$$

A is faster...

$$\begin{aligned}\text{CPU Time}_B &= \text{Instruction Count} \times \text{CPI}_B \times \text{Cycle Time}_B \\ &= I \times 1.2 \times 500\text{ps} = I \times 600\text{ps}\end{aligned}$$

$$\frac{\text{CPU Time}_B}{\text{CPU Time}_A} = \frac{I \times 600\text{ps}}{I \times 500\text{ps}} = 1.2$$

...by this much

CPI in More Detail

- If different instruction classes take different numbers of cycles

$$\text{Clock Cycles} = \sum_{i=1}^n (\text{CPI}_i \times \text{Instruction Count}_i)$$

<https://tutorcs.com>

Weighted average CPI
WeChat: estutorcs

$$\text{CPI} = \frac{\text{Clock Cycles}}{\text{Instruction Count}} = \sum_{i=1}^n \left(\text{CPI}_i \times \frac{\text{Instruction Count}_i}{\text{Instruction Count}} \right)$$

Relative frequency

CPI Example

- Alternative compiled code sequences using instructions in classes A, B, C

Class	A	B	C
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

- *Sequence 1: IC = 5*

- *Clock Cycles*
 $= 2 \times 1 + 1 \times 2 + 2 \times 3$
 $= 10$

- *Sequence 2: IC = 6*

- *Clock Cycles*
 $= 4 \times 1 + 1 \times 2 + 1 \times 3$
 $= 9$

- *Avg. CPI = 10/5 = 2.0*

- *Avg. CPI = 9/6 = 1.5*

Performance Summary

The BIG Picture

- Performance depends on

- Algorithm: affects IC, possibly CPI

$$\text{CPU Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock cycle}}$$

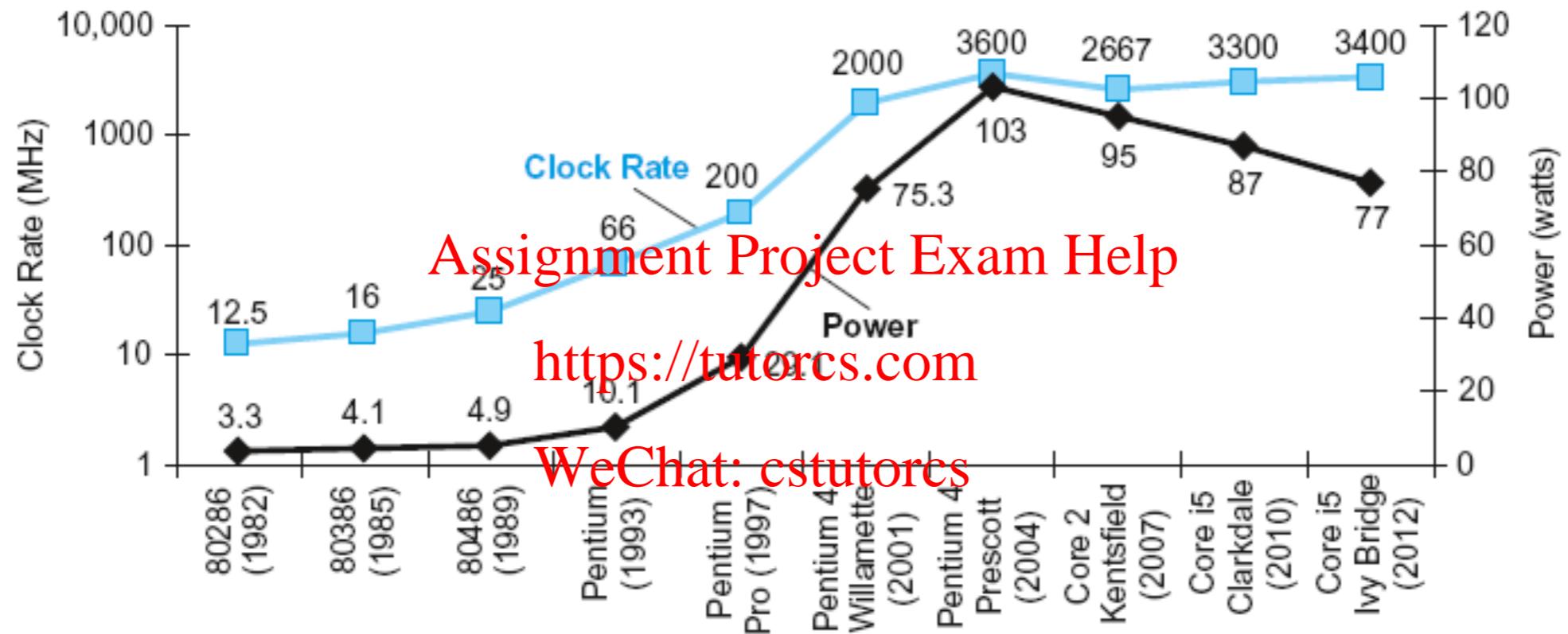
Assignment Project Exam Help

- Instruction set architecture: affects IC, CPI, T_c
<https://tutorcs.com>

WeChat: cstutorcs

Power Trends

■ In CMOS IC technology



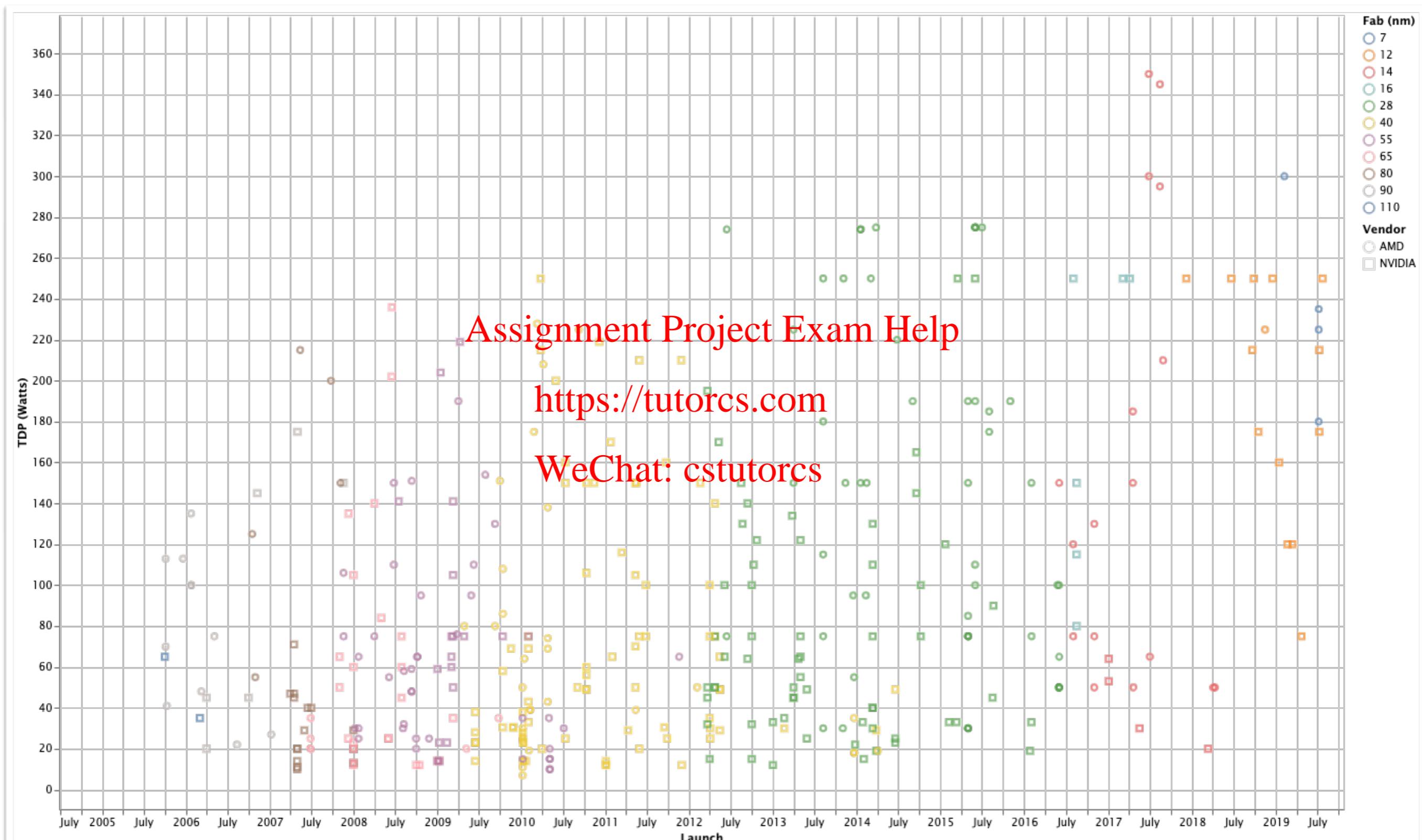
Power = Capacitive load × Voltage² × Frequency

×30

5V → 1V

×1000

GPU Power vs. Time



Reducing Power

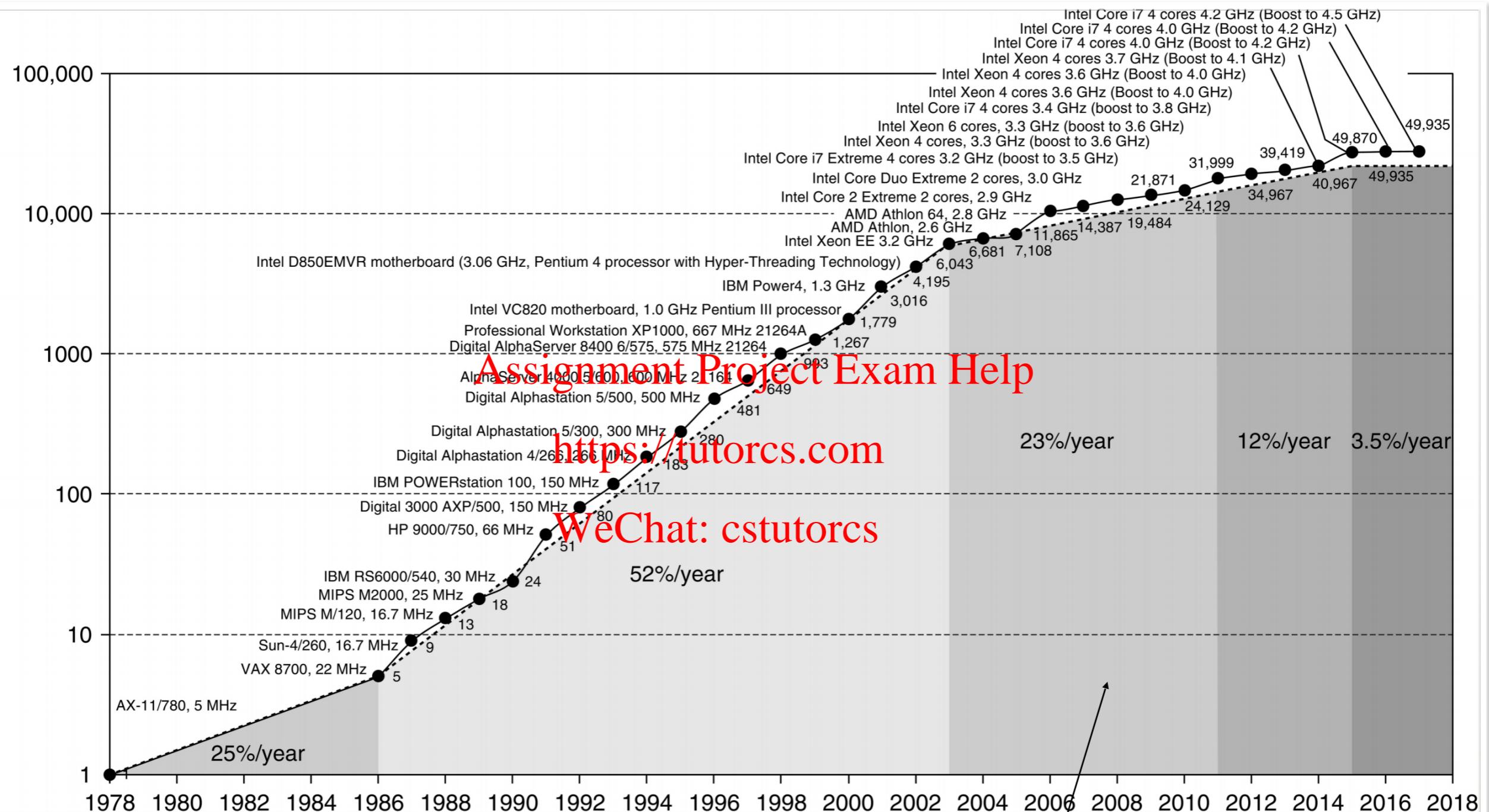
- Suppose a new CPU has
 - 85% of capacitive load of old CPU
 - 15% voltage and 15% frequency reduction

Assignment Project Exam Help
<https://tutorcs.com>
WeChat: cstutorcs

$$\frac{P_{\text{new}}}{P_{\text{old}}} = \frac{C_{\text{old}} \times 0.85 \times (V_{\text{old}} \times 0.85)^2 \times F_{\text{old}} \times 0.85}{C_{\text{old}} \times V_{\text{old}}^2 \times F_{\text{old}}} = 0.85^4 = 0.52$$

- The power wall
 - We can't reduce voltage further
 - We can't remove more heat
- How else can we improve performance?

Uniprocessor Performance



Constrained by power, instruction-level parallelism, memory latency

Multiprocessors

- Multicore microprocessors
 - More than one processor per chip
- Requires explicitly parallel programming
 - Compare with instruction level parallelism
Assignment Project Exam Help
 - Hardware executes multiple instructions at once
<https://tutorcs.com>
 - Hidden from the programmer
WeChat: estutorcs
- Hard to do
 - Programming for performance
 - Load balancing
 - Optimizing communication and synchronization

SPEC CPU Benchmark

- Programs used to measure performance
 - Supposedly typical of actual workload
- Standard Performance Evaluation Corp (SPEC)
 - Develops benchmarks for CPU, I/O, Web, ...
- SPEC CPU2006
 - Elapsed time to execute a selection of programs
 - Negligible I/O, so focuses on CPU performance
 - Normalize relative to reference machine
 - Summarize as geometric mean of performance ratios
 - CINT2006 (integer) and CFP2006 (floating-point)

Assignment Project Exam Help

<https://tutoros.com>

WeChat: CPU tutoros

$$\sqrt[n]{\prod_{i=1}^n \text{Execution time ratio}_i}$$

CINT2006 for Intel Core i7 920

Description	Name	Instruction Count x 10 ⁹	CPI	Clock cycle time (seconds x 10 ⁻⁹)	Execution Time (seconds)	Reference Time (seconds)	SPECratio
Interpreted string processing	perl	2252	0.60	0.376	508	9770	19.2
Block-sorting compression	bzip2	2390	0.70	0.376	629	9650	15.4
GNU C compiler	gcc	794	1.20	0.376	358	8050	22.5
Combinatorial optimization	mcf	211	1.00	0.376	221	9120	41.2
Go game (AI)	go	1274	1.10	0.376	527	10490	19.9
Search gene sequence	hmmer	2618	0.60	0.376	590	9330	15.8
Chess game (AI)	sjeng	1948	0.80	0.376	586	12100	20.7
Quantum computer simulation	libquantum	659	0.44	0.376	109	20720	190.0
Video compression	h264avc	3793	0.50	0.376	713	22130	31.0
Discrete event simulation library	omnetpp	367	2.10	0.376	290	6250	21.5
Games/path finding	astar	1250	1.00	0.376	470	7020	14.9
XML parsing	xalancbmk	1045	0.70	0.376	275	6900	25.1
Geometric mean	-	-	-	-	-	-	25.7

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

SPEC Power Benchmark

- Power consumption of server at different workload levels
 - Performance: ssj_ops/sec
 - Power: Watts (Joules/sec)

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

$$\text{Overall ssj_ops per Watt} = \left(\sum_{i=0}^{10} \text{ssj_ops}_i \right) / \left(\sum_{i=0}^{10} \text{power}_i \right)$$

SPECpower_ssj2008 for Xeon X5650

Target Load %	Performance (ssj_ops)	Average Power (Watts)
100%	865,618	258
90%	786,688	242
80%	698,051	224
70%	607,826	204
60%	521,391	185
50%	436,157	170
40%	345,919	157
30%	262,071	146
20%	176,061	135
10%	86,784	121
0%	0	80
Overall Sum	4,787,166	1,922
$\Sigma ssj_ops / \Sigma power =$		2,490

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Pitfall: Amdahl's Law

- Improving an aspect of a computer and expecting a proportional improvement in overall performance

$$T_{\text{improved}} = \frac{T_{\text{affected}}}{\text{improvement factor}} + T_{\text{unaffected}}$$

Assignment Project Exam Help
https://tutorcs.com

- *Example: multiply accounts for 80s/100s*
WeChat: cstutorcs
- *How much improvement in multiply performance to get 5× overall?*

$$20 = \frac{80}{n} + 20 \quad \blacksquare \quad \text{Can't be done!}$$

- *Corollary: make the common case fast*

Fallacy: Low Power at Idle

- Look back at i7 power benchmark
 - At 100% load: 258W
 - At 50% load: 170W (66%)
 - At 10% load: 121W (47%)
Assignment Project Exam Help
- Google data center <https://tutorcs.com>
 - Mostly operates at 10–50% load
 - At 100% load less than 1% of the time
- Consider designing processors to make power proportional to load

Pitfall: MIPS as a Performance Metric

■ MIPS: Millions of Instructions Per Second

- Doesn't account for
 - Differences in ISAs between computers
 - Differences in complexity between instructions
Assignment Project Exam Help

<https://tutorcs.com>
WeChat: cstutorcs

$$\begin{aligned} \text{MIPS} &= \frac{\text{Instruction count}}{\text{Execution time} \times 10^6} \\ &= \frac{\text{Instruction count}}{\frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}} \times 10^6} = \frac{\text{Clock rate}}{\text{CPI} \times 10^6} \end{aligned}$$

- *CPI varies between programs on a given CPU*

Concluding Remarks

- **Cost/performance is improving**
 - Due to underlying technology development
- **Hierarchical layers of abstraction**
 - In both hardware and software
- **Instruction set architecture**
 - The hardware/software interface
- **Execution time: the best performance measure**
- **Power is a limiting factor**
 - Use parallelism to improve performance

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Class Topics

- **Ch. 1: Computer Topics and Technology**
- **Ch. 2: Instructions: Language of the Computer**
- **Ch. 3: Arithmetic for Computers**
- **Ch. 4: The Processor** Assignment Project Exam Help
- **Ch. 5: Large and Fast: Exploiting Memory Hierarchy** <https://tutorcs.com>
WeChat: cstutorcs
- **Note: New material in this class since the last time I taught it (Fall '10):**
 - We now cover assembly/ISAs + arithmetic in 170
 - RISC V vs. MIPS

Rough Syllabus

- In your book, most of Chapters 1–5
 - Chapter 6 is on parallel computing—EEC 171
- Likely will not cover some details in:
 - Chapter 2 (recursion, linking/loading Java, atomics, other instruction sets)
Assignment Project Exam Help
https://tutorcs.com
 - Chapter 3 (rounding methods, accurate arithmetic, SIMD/x86)
WeChat: cstutorcs
 - Chapter 5 (dependable memory hierarchy, FSM for cache, coherence)