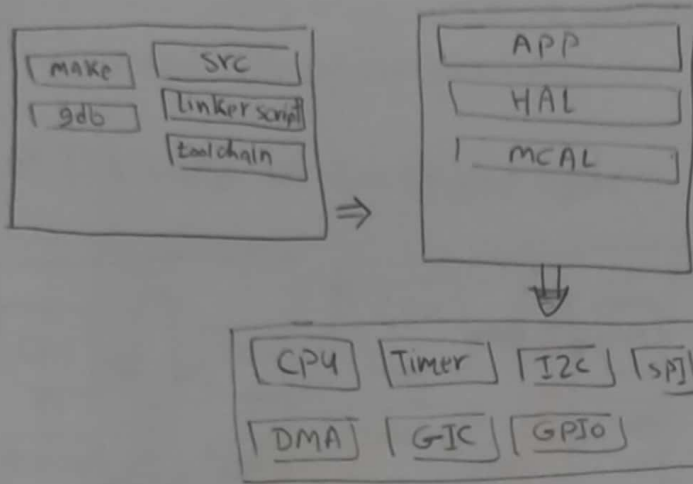


# unit 6 Lesson 1

## Big Picture



CPU, MCU, SOC,

دورة بيتم من تصنيع IC's كيتي بعدا

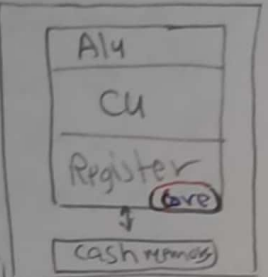
RAM, Rom are made of IC's

electronic componant اي كيتي IC تعيب IC

## Microprocessors (MPU)

It's an Integrated circuit made to arithmetic and logical operation

fetch → decode → execute



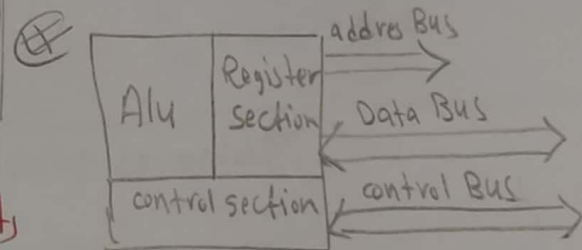
MP, uP or MPU, CPU

Processor = brain

CPU (Central processing unit)

Primary Core ⇒ CPU

Co-processor (second) حابطة مساعدة  
Ex: GPU Graphics coprocessor  
DSP

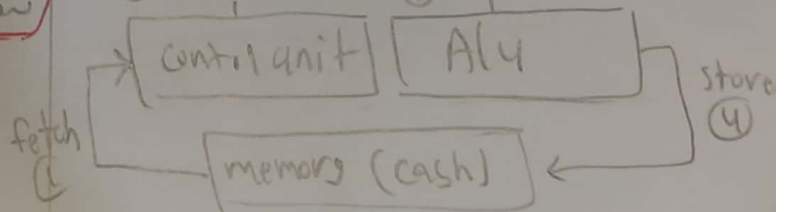


Bus = many lines not one line

## Instruction life cycle

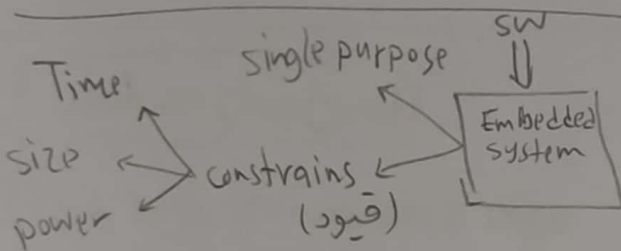
1 decode

3 execute



## What is Embedded system??

It is a special purpose computer system designed to perform one or a few function



## IC Integrated Circuits:-

It's is a single chip which has a set of electronic circuits

VLSI very large scale Integrated circuit

componant دي نفس ال فوق وادي تصوى من الطبيعة كثر الاكثر الطبيعية

## Moore's Law

transistor

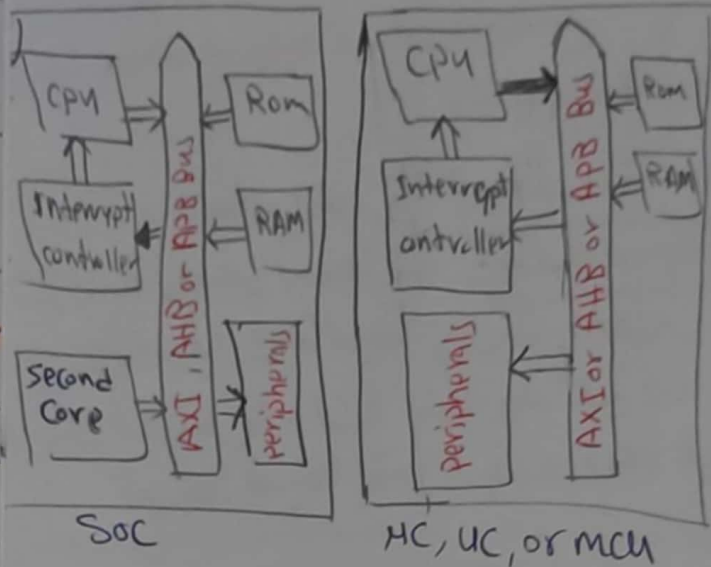
تضاعف كل عامين دورة من طريق نقل

transistor size

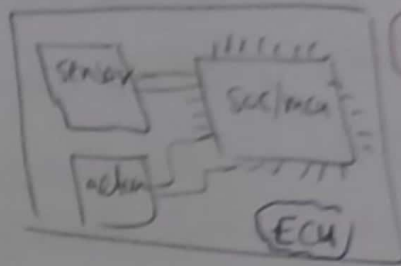
## Micro Controller (MCU)

It's an Integrated Circuits consist of CPU and other components

It's used in Embedded system



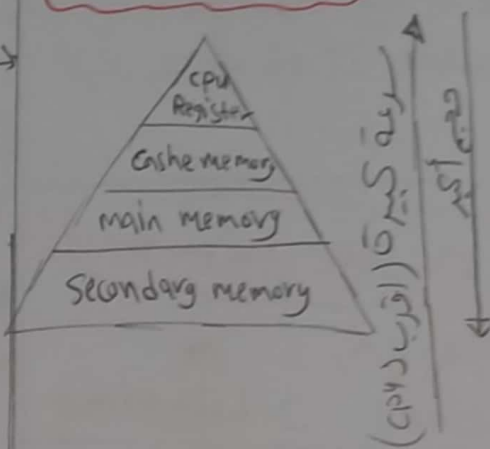
## Electronic Control Unit (ECU)



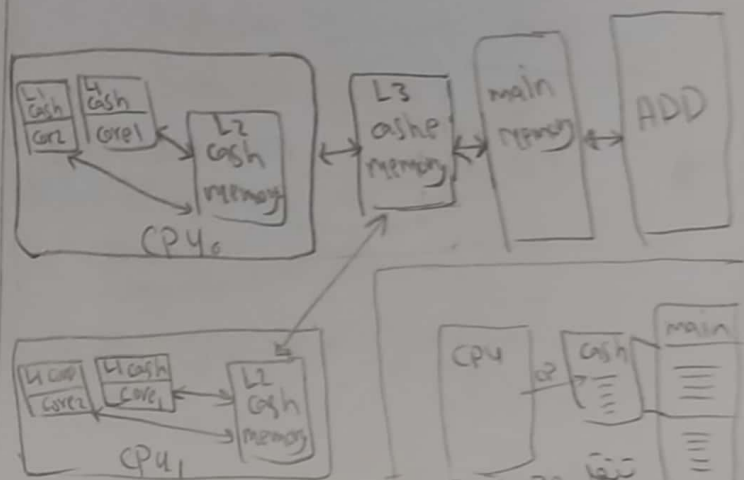
الغالبية مثل يكون فيها من ECU  
مثلا على اسي يكون لكن جزء ECU يتا  
على اسي مثلا لو حصل عطل في واحة  
مثلا لو حصل عطل في الباقا

Conclusion 37 (الانصاف)

## Cache memory



## Cache designed



## System on chip (SoC)

It's a high performance MCU

هو MCU عاااa

Second core or CPU2

و ممكن يكون بقة جابات sound, Graphic  
وطبعا طالا في Core جابو اكي  
ليكون بقة peripherals جابو  
و ممكن يتغل بيل او يعن  
يفتحها، لكون جابو جابو جابو

تصميم MCU او SoC مقصود  
يتاكد يعن التصميم من تايت والا مكانا كذا

من الرسمة السابقة

افترض انك كنت تفعل 2 CPU من MCU  
وحصل ان كل CPU غير قيمة variable  
في نفس الوقت دة معناة ان اول واحد  
ليغيرها من cache ياتسنة وبالتالي نستغير  
في main اننا الثاني كانه يحملها من cache  
في memory ياتسنة فيل ما تتغير اي  
ال هيد هل؟؟

الحاقه دي معروفة من Embedded بآكم  
(cash coherence) لهذا بعد ما يحصل تغيير في  
main لغيره انوصاتيل يفيد في cache  
الثانية على طول عنانه لو CPU (استخدمه)  
يلاقيها متغيرة من القيمة القديمة

## Floating-point unit (FPU)

دي IC معموله عنانه تتعامل مع  
(floating value)

في CPU MCU تكون جوة CPU دي  
arm cortex m4 ويمكن تكون في MCU  
بيرة لو مد لها دة على حسب APP دي ما قلته

## memory protection unit (MPU)

دي عبارة عن IC يتقوم بتقسيم memory  
وتبدا تفرض قيود عليه على الكائنات فيها  
في اماكن معينة مثلا لو انت تفعل OS  
كل OS لي memory ياتسنة اي ان يستغ  
OS معين يكتب على memory (الثاني هي  
(mpu) ان يستغ دة يحصل

## Memory management unit (MMU)

دي عبارة عن IC المرفق منها مثلا  
لو عند App يتفعل من 8GB RAM انك  
مقال (mpu) 1GB RAM فانا بقوله من  
خلال IC دي انت عنك 8GB RAM  
تارن فابدا استغ

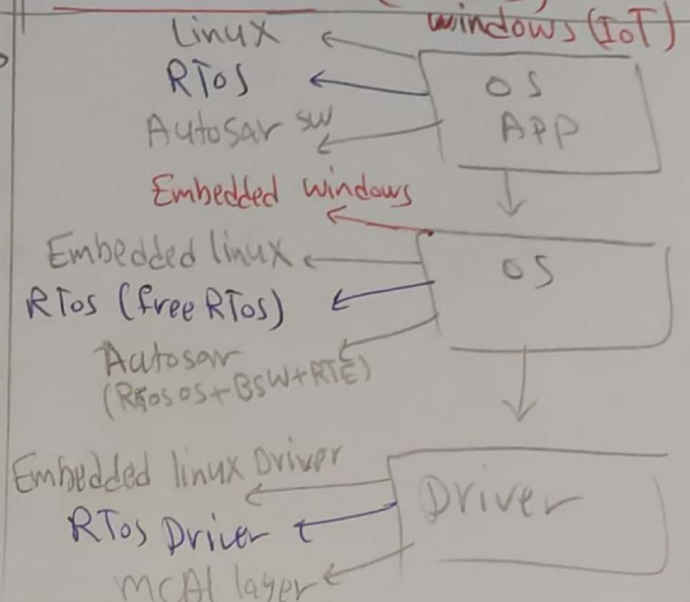
خطيب بيحصل دة ازان هو ييقسم الذاكرة  
ال عنده ويكتب على جزي ولما يتفعل  
يستخدمها عنانه ياتسنة اليزن الكورال بيرة  
يتغير ويسمح وهكذا

## FP4 أو MP4 أو MM4

مممكن يكونوا مع CPU جوة  
أوبيرة في MCU كل دة على حسب  
التصميم

## Baremetal SW vs OS

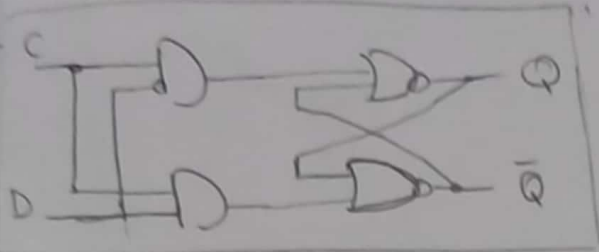
مشرحة بالتفصيل في  
Embedded C (unit 3)





## Basic Memory Element

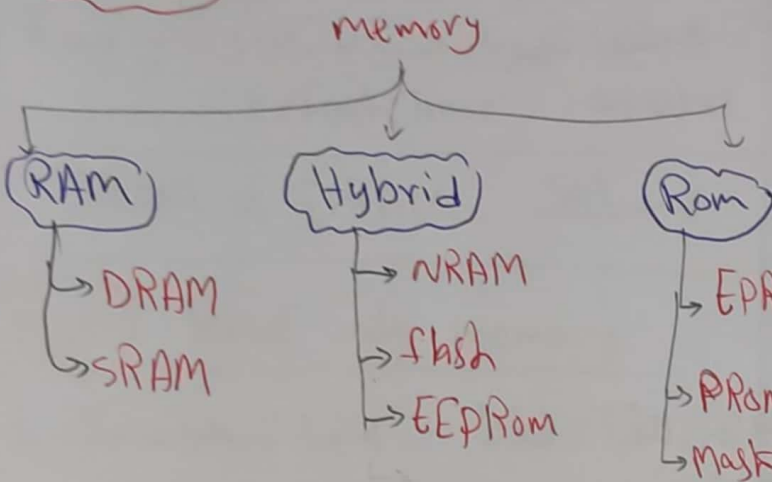
### D Flip Flop



Register:- consist of 4 flip flop

Serial  
Parallel

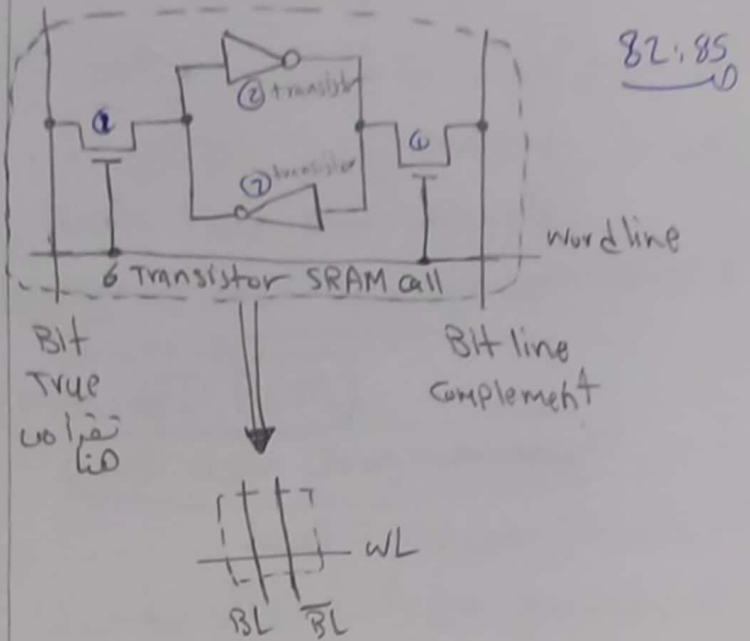
## memory



## RAM = Random Access Memory

memory من تصكب وتقرأ من عليها  
Volatile memory يعني لا الكبريا  
Data لا يتحفظ  
Based on Mosfet

## SRAM Static RAM



Bit unit = 6 Transistor

Static لا تحتاج Refreshing Time  
إنت بتحتاج القصة مرة واحدة وقتها لا موجودة طولها الكبريا موجودة

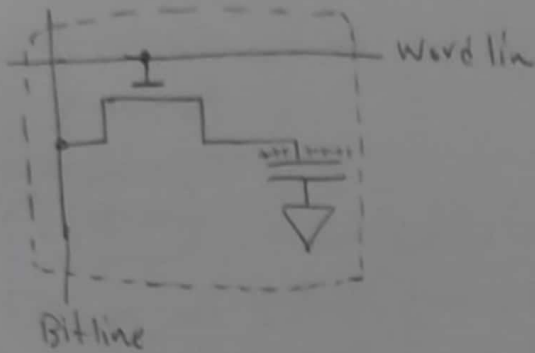
SPEED SRAM > DRAM

Refresh Time لا تحتاج

الوظيفة الأساسية هي لا cache  
من يتحكم power كبر

# DRAM

تتكون من 1 transistor + 1 capacitor



1- انت الوقت على 1 على wordline يعني  
كما انت متصل transistor وبالتالي سيبقى  
يقرا من Bitline فلا يمكن على capacitor  
يكون واحد ولكن مع الوقت اتصالات  
في لتقل ف انت تحتاج كل ثانية  
تعطيلها كهرباء ف تقرأ وترجع قيمة  
capacitor (Refresh Time)

SRAM vs DRAM 101

## Rom Read only memory

هي للقراءة فقط CPU يعرف يكتب  
فيها

انها مش يتم مسح عند فصل الكهرباء

Rom \* اقل سرعة من RAM

Based on Floating Gate Mosfet

## type of Rom

### PRom

هي تيرمج مرة واحدة فقط

masked يعني (OTP)

(only Time programmable)

ويم برمجتها مرة واحدة عن طريق burner

### MASKED ROM

not user programmable

هي تيرمج مرة واحدة فقط

هي تيرمج مرة واحدة - والي بيعل كذا

IC manufacturing (المصنع للمنتج)

### PRom Vs Masked

→ PRom

تصنع وتعطيلك انت ف ارغبة وانت

تيرمجها مرة واحدة فقط

→ masked

تصنع وتيرمج من خلال الشركة

المصنعة لعلاقها من board على PC

### EProm

هي Rom مدمج تكب

عليها وتمسحها وترجع تكب تاني بس

المشكلة في المسح مش من خلال الكهرباء

لازم (UV-EProm)

# HyBRID

هنا RAM ممكن تكتب وتقرأ بارز  
 ورن Rom ان الحثون non volatile  
 متى تمسح

## EEPROM

هنا Rom ولكن انت  
 حاليا تقدر تمسح! ان عليها ان EProm  
 ولكن هنا باي استفاد ان الكبر بار بار

## Flash

هو الافضل والا حسب حاليا  
 واني من EEPROM + فيه ان مميزات EEPROM

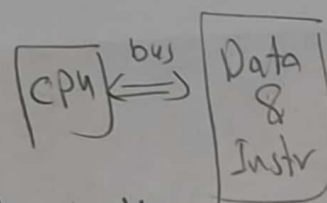
## NVRAM

هي عبارة عن SRAM  
 ومكان عليها Battery backup

$$NVRAM = SRAM + Battery$$

\* وممكن في روفر الاحياء بيحفظ NVRAM  
 2. EEPROM هنا لو انك ما بي قطع  
 البطارية تـ عمل NVRAM لحد ما ينقل  
 EEPROM to NVRAM Data

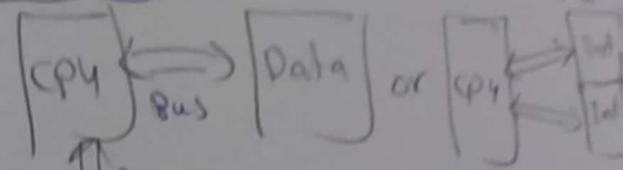
## Von-Neumann



the same memory  
 and bus used to stor both Data  
 and Instruction

\* unable to access program memory and  
 memory simultaneously  
 فيا هو Bus واما في هخره Data يـ هخره

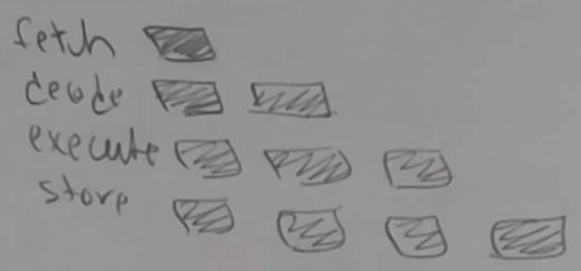
## Harvard



هناك 2 Bus  
 واحد لـ address  
 واحد لـ Data

نورم يتفعل  
 Pipeline  
 parallel

## Pipeline



1- يعمل Fetch في أول CLK وثاني CLK يعمل  
 Fetch ويعمل Decode والثالث CLK يعمل  
 Fetch ويعمل Decode ويعمل Execute  
 وهكذا ----

ليس رة فية عيوب

1- لو فية Instruction معتمدة على ان قبلها  
 وانا تفعل من لاه تاخو الاشارة الاول  
 execute 4 CLK Instruction يكون حله

## CISC vs RISC

Complex Instr set

CISC هي عملية يتعمل فيها مجموعة  
 عمليات كثيرة في نفس الوقت  
 SW → سهل  
 HW → معقد  
 not support pipeline

RISC هي عملية يتم في اربعة مراحل كثيرة  
 SW → معقد  
 HW → سهل  
 Embedded

	Type	volatile	writable	erase size	Max erase cycles	Cost per byte	Speed
RAM	SRAM	✓	✓	Byte	unlimited	expensive	fast
	DRAM	✓	✓	Byte	unlimited	moderate	moderate
ROM	RRAM	X	✓ one-time only	We Can not erase data	We Can not erase data	moderate	fast
	Masked ROM	X	X	We Can not erase data	We Can not erase data	Cheaper	fast
	EPROM	X	✓ with burner	all the data	limited (see the data sheet)	moderate	fast
Hybrid	EEPROM	X	✓	Byte	limited (see the data sheet)	expensive	fast to read slow to erase write
	Flash	X	✓	by the block	limited (see the data sheet)	moderate	fast to read slow to erase write
	NVRAM	X	✓	Byte	unlimited	expensive	fast