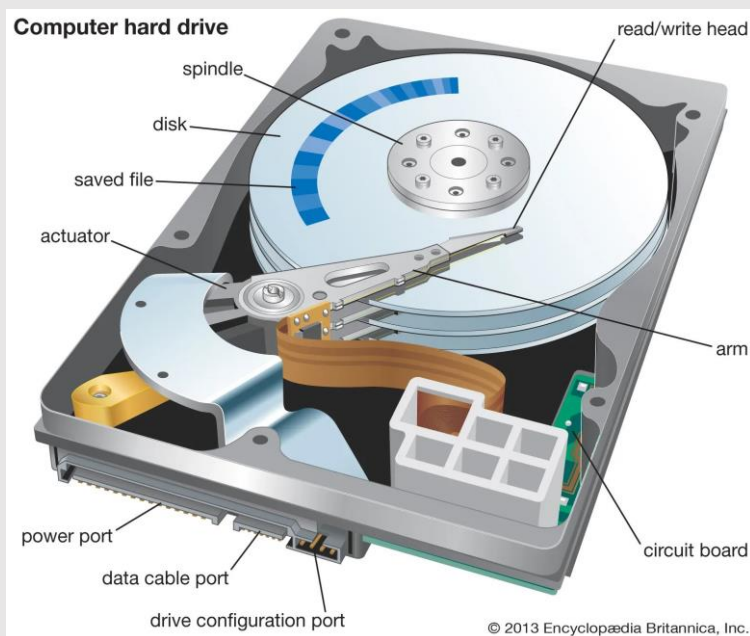


1-computer structure

-Hard drive (HDD , SSD) :

هارد درایو یک قطعه حیاتی در هر سیستم کامپیوتری است که وظیفه ثبت دائمی اطلاعات را دارد

هارد HDD : این نوع هارد از نوع مکانیکال هست یعنی اینکه از تکنولوژی مکانیکی برای خواندن و نوشتن اطلاعات استفاده میکند.



در ای نوع هارد اعداد صفر و یک به صورت فیزیکی روی لای های دیسک در حال چرخش با یک لنز بسیار قوی هک میشود و همان لنز در صورت نیاز همان صفر و یک هارو میخواند و در آن تغییراتی ایجاد میکند .

هارد SSD : در این نوع هارد برخلاف نوع قبلی اطلاعات به صوت فیزیکی و مکانیکی ذخیره نمیشود و تمام صفر و یک ها در واحد های بی شماری از ترانس و آی سی های مختلفی از جمله ترانس های NAND ذخیره میشوند و ماحیت آنها الکتریکی است.



مزایا و معایب این دو هارد :

- بازیابی اطلاعات از روی هارد های اس اس دی کاری نشدنی و بسیار دشوار هست چون صفر و یک ها ماهیت الکتریکی دارند . ولی در هارد های اچ دی دی چون صفر و یک ها ماهیت فیزیکی دارند بازیابی اطلاعات بسیار راحت تر از هارد های اس اس دی هستند.

- هارد های اچ دی دی کم هزینه تر از هارد های اس اس دی هستند و برای مصارف معمولی و آرکایو (archive) به صرفه تر هستند

- هارد های اس اس دی سرعت انتقال بیشتری را در اختیار کاربر قرار میدهند که برای مصارفی مثل : سرور , نصب سیستم عامل و ... مناسب هستند در صورتی که هارد های اچ دی دی از سرعت عمل کمتری برخوردار هستند.

- عمر هارد های اچ دی دی بلند تر از هارد های اس اس دی هست

-Ram(Random Access Memory) :

رم یکی قطعه حیاتی در سیستم های کامپیوتری هست که وظیفه ثبت **موقت** اطلاعات برای دسترسی سریع سی پی یو به اطلاعات را دارد.

اطلاعات صفر و یک در رم ها درواحد های میکروچیپ که شامل تعداد ترانزیستور و آی سی هستند ذخیره میشوند و هرکدام از این واحد ها را یک منبع ذخیره یا یک مموری آدرس مینامند که با آدرس هایی رندوم مثل 00xx120 مشخص میشوند.

رم ها به صورت **مستقیم** به سی پی یو متصل میشوند و سی پی یو هیچ ارتباطی با دیگر قطعات مثل هارد ها ندارد.

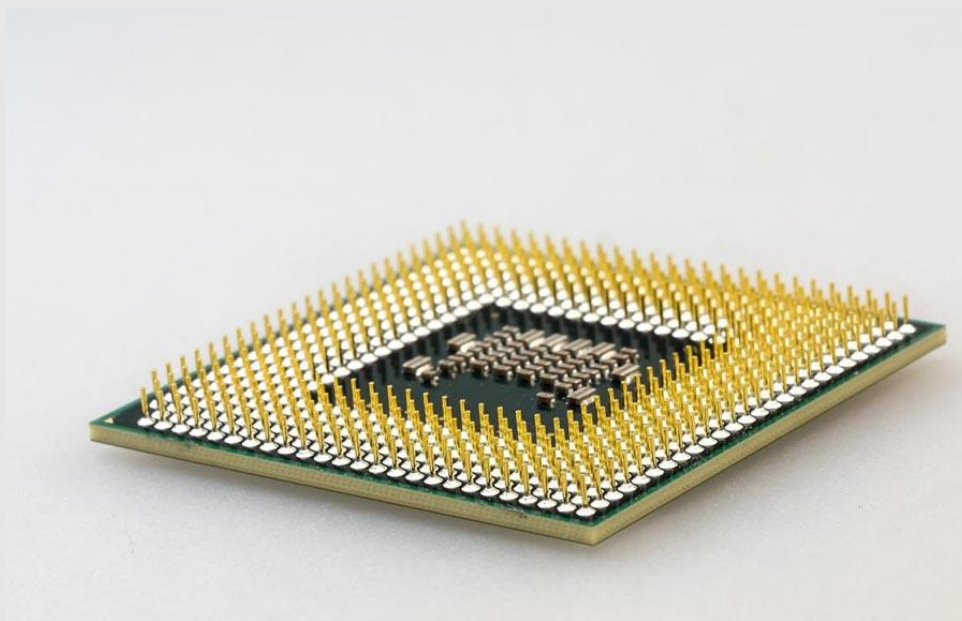
رم ها از سرعت بسیار بالایی حدود 20 گیگابایت بر ثانیه و بیشتر را دارند و دلیل همین هست که فضایی که رم ها دارند به 512 مگابایت تا 128 گیگابایت محدود میشوند.



وقتی اطلاعات به پایه های رم میرسند وضعت مموری آدرس ها رو که صفر هستند رو به یک و دیگر اطلاعات تغییر میدهند و این ها همان اطلاعات ذخیره شده در میکروکنترلر های رم ها هستند.

CPU (Central Processor Unit):

سی پی یو پردازنده مرکزی یک سیستم هست که وظیفه پردازش اطلاعات آمده از طرف **رم** را که در قالب صفر و یک ها هستند را دارد. صفر و یک ها وارد سی پی یو میشوند و به هم تبدیل میشوند و این اطلاعات از سی پی یو خارج و به رم باز میگردند تا به دیگر بخش ها بروند تا دستورات را انجام بدهند. سی پی یو از میلیارد ها ترانزیستور تشکیل شده اند که هر کدام وظیفه پردازش یک صفر و یک را دارند.

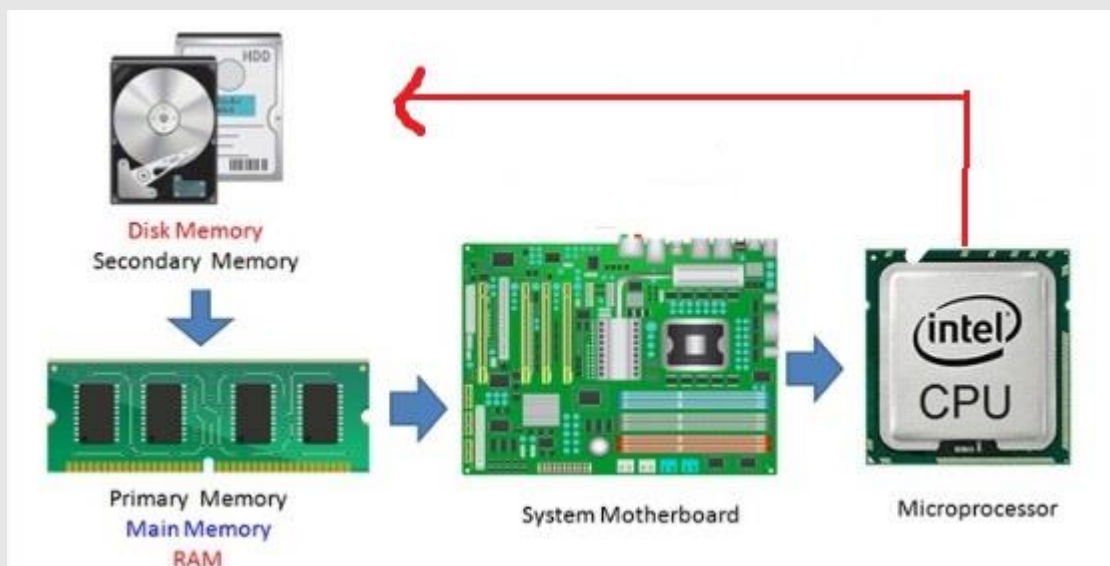


- Integration of CRD :

حالا تمام این قطعات با هم ارتباط برقرار میکنند که نتیجه این ارتباط انجام محاسبات کامپیوتری میشود.

حالا سازکار ارتباطی این ها چگونه است ؟

وقتی که یک تسک در سیستم اجرا میشود اول اطلاعات مورد نیاز برای اجرای تسک از هارد به رم منتقل میشود یعنی صفر و یک هایی که در روی دیسک هارد نوشته شده اند یا صفر و یک هایی که در هارد های اس اس دی به صورت الکتریکی نوشته شده اند به رم منتقل میشود و وضعیت صفر و یک میکروکنترلر های پایه های رم تغییر پیدا میکنند و در آنجا پردازش های مورد نیاز برای انتقال آنها به سی پی یو انجام میشوند و اطلاعات آماده ارسال به سی پی یو میشوند . بعد اطلاعات با فرمان سی پی یو به آن منتقل میشوند و سی پی یو پردازش های ریاضی و .. را روی آنها با استفاده از ترانسیستور های داخل آن و تغییر ماهیت صفر و یک ها انجام میدهد و بعد از پایان یافتن تسک پیام کامل شدن تسک از همان مسیری که آمده بر میگردد به رم و هارد تا وضعیت را ثابت کند.



مثال باز کردن یک فیلم :

- 1- اطلاعات دائمی از هارد به رم منتقل میشود
- 2- اطلاعات فیلم در رم پردازش میشوند تا بتواند وارد سی پی یو شود
- 3- اطلاعات فیلم د سی پی یو پردازش میشود و به دیگر قطعات باز میگردد
- 4- اطلاعات بازگشتی به نمایش گذاشته میشوند.

در برنامه نویسی :

کد هایی که مینویسیم همه وقتی سیو میشوند در هارد ما ذخیره میشوند

موقعی که اجرا میکنیم کد را این کد ها به رم منتقل میشود تا به سی پی یو برود مثلا فلان متغیر در آدرس مموری 0xx001 ذخیره میشود و به همین ترتیب بقیه المان های دیگر.

سی پی یو این هارا از ممور آدرس ها دریافت میکند و پاسخ مناسب را برمیگرداند مثلا دستور چاپ کردن در خروجی.

2-Programming languages summary

هر زبان برنامه نویسی کاربرد های خود را دارند :

طراحی وب : پایتون پی اچ پی سی شارپ جاوا اسکریپت رابی و ...

طراحی اپلیکیشن اندروید : جاوا دارت/فلاتر کاتلین

طراحی اپلیکیشن آی او اس : سوئیفت دارت/فلاتر

هوش مصنوعی : پایتون سی پلاس پلاس

ماشین لرنینگ : پایتون



زبان های برنامه نویسی به ترتیب به لول های مختلفی دسته بندی میشوند :

1- زبان های تفسیر شده: به دسته زبان هایی میگویند که از تکنولوژی **شعی گرایی** بهره مند هستند که از های لول ترین زبان ها هستند مثل زبان پایتون یا شل

2- زبان های غیر شعی گرا های/مید لول : زبان هایی که یک لول از زبان های شعی گرا پایین تر هستند و نسبت به آنها از سرعت **بیشتری** برخوردار هستند ولی یادگیری آنها دشوار تر هست مثل زبان سی پاسکال و ...

3- زبان های های اسمبلی : یه اولین لایه از زبان های برنامه نویسی میگویند که توسط انسان قابل درک هستند یعنی انسان با خواندن کد ها میتواند ساز و کار آنها را از روی کلمات آنها درک کنند مثل زبان اینتل ایکس 86 و ...

4- زبان های ماشینی : زبان هایی هستند که دیگر کلمات در آنها وجود ندارد و از لحاظ املائی توسط انسان غیر قابل درک هستند و فقط مفسر های ماشین ها میتوانند آنها را درک کنند.

5- زبان باینری : تمامی زبان ها در انتها به زبان باینری تبدیل میشوند . این زبان متشکل از همان صفر و یک ها هستند که زبانی هست که توسط ترانسیستور ها یا دیگر قطعات سخت افزاری قابل درک هستند . این زبان آخرین لایه هر کد برنامه نویسی هست .

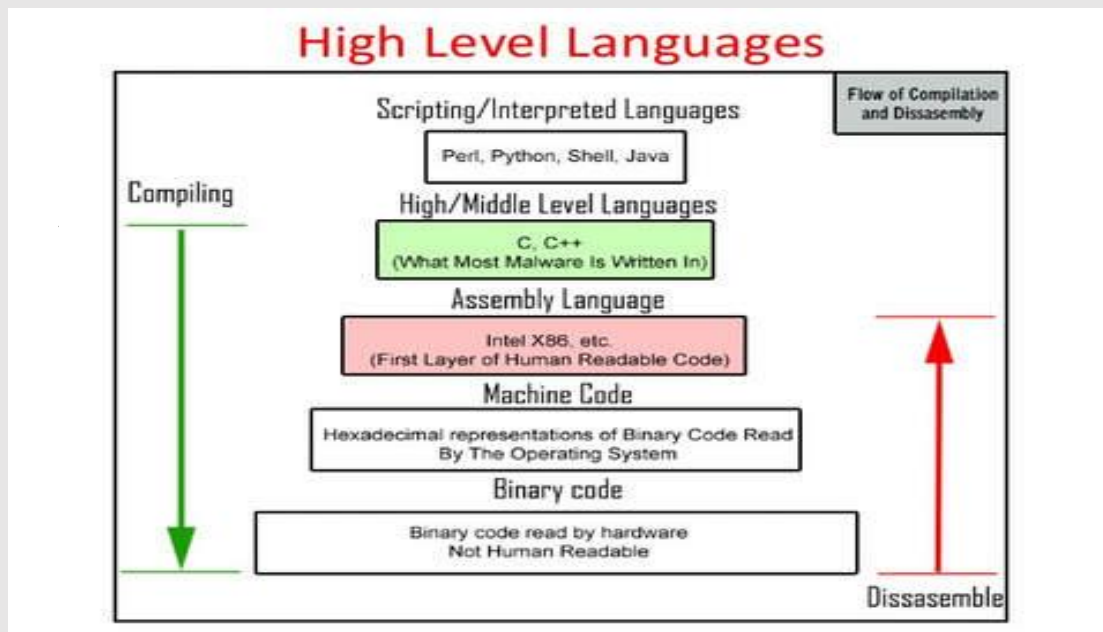
.

.

.

.

.



هر چه از سمت بالا به پایین حرکت میکنیم زبان ها سخت تر و سریع تر میشوند و برعکس

زبان هایی بالایی موقع اجرا شدن به زبان های پایینی ترجمه میشوند این وظیفه مفسر و مترجم زبان ها هست مثلا مفسر زبان پایتون و ...

سازکار زبان مثلا پایتون : وقتی یک کد پایتونی اجرا میشود اول مفسر آنرا به زبان لول پایینتر ترجمه میکند مثلا زبان سی . بعد همان زبان سی به زبان اسمبلی ترجمه میشود و آن به زبان ماشین و بعد به زبان باینری که تبدیل به صفر و یک میشود و سخت افزار میتواند آن صفر و یک هارا بخواند و پاسخ مناسب را بدهد. بعد پاسخ هم به همین صورت به باینری زبان ماشین زبان اسمبلی زبان های/مید لول و ... (از پایین به بالا) ترجمه میشود تا پاسخ دریافت شود.