

## سوال ۱:

### الف:

هر دو دریافت کننده جایزه تورینگ سال ۲۰۱۷ برای تغییر انقلابی در پردازنده‌های ریسک هستند. کتاب معماری کامپیوتر آنها نسل جدید را مورد تأثیر قرار داده و تدریس میشود. تا کار و ایده آنها بتواند برای نسل بعد مورد استفاده باشد.

هنسی: رییس قبلی دانشگاه استنفورد. میپس رو تاسیس کرد تا کار تیم‌های استنفورد رو تجاری کنه.  
پترسن: استاد بازنشسته دانشگاه برکلی و رییس قبلی ACM. پردازنده‌های ریسک در کمپانی سان.

### ب:

در مورد دوران طلایی پیش رو در صنعت کامپیوتر از زاویه دید خودشون قراره صحبت بشه. اول یه تاریخچه‌ای گفته میشه که دوران طلایی قبلی چی بود و چطور جلو رفت داستان. و اینکه الان چه چالش‌هایی داریم و عصر جدید چطور انقلابی خواهد بود احتمالاً.

### ج:

پترسن:

در دهه ۶۰ آی بی ام مشکلی داشت که ۴ مدل کامپیوتر مختلف برای ۴ کاربرد مختلف داشت. در تلاش بود که اینها را یکی کند. اول با رام جلو رفتن که میکروپروگرامینگ نام داشت. ارزونتر بود. اما باگ‌هاش رو نمیشد رفع کرد. بعداً رم اومد و بیشتر مورد استفاده بود.

در مورد تاریخچه و مشکلات سیسک گفته شد و اینکه مقدار زیادی از پیاده‌سازی‌ها مورد استفاده قرار نمی‌گیرند. در نتیجه پردازنده‌های ریسک وارد شدند که حاصل کار مشترک برکلی و استنفورد بود. بعدها پردازنده‌های MIPS توسط استنفورد معرفی شدند.

در مورد معایب و کندی CISC گفته شد و ویدیویی از آن زمان‌ها که توضیح میداد چرا و چگونه RISC ساخته شد.

هنسی:

در مورد پرفورمنس پردازنده‌ها در یک برنامه و توان پردازشی گفت. این گفته شد که همیشه این نمودارها مثل قانون مور از جایی به بعد کار نمی‌کنند و ما نیاز به نگرش جدیدی خواهیم داشت. مثل RISC که یک زمان آمد...

بیان شد که سخت‌افزار باید به امنیت کمک کند وگرنه کسی اعتماد نمیکند. می‌گه که باید باید مقادیری از کانسرن‌های سکيوریتی کرنل به سخت‌افزار منتقل بشه. چون توی سخت‌افزار دقیق‌تر بررسی میشه و رخداد خطا کمتر هست. مثلاً timing based مشکل‌ها توسط سخت‌افزار دیده شوند بجای کرنل.

در مورد SW-centric و HW-centric گفت و اینکه تلاش زیادی میشه که سخت‌افزار افیشنت باشه اما کد بد روی آن ران میشه.

در مورد شکل‌گیری VLIW توسط اینتل و اچ پی گفت و اینکه کامپایلر هندل میکند. اما پاره ای از مشکلات از جمله کش میس و آنپردیکتبل بودن باعث کندی و فیل شدن پروژه شد. جان در مورد آینده و رنسانس و موقعیت‌های معماری گفت. اینکه بتوانیم معماری ای داشته باشیم که بر اساس دومین کار کند. یعنی یک CPU general purpose نباشد و بر اساس نوع کارکرد تغییر کند. پترسن:

در پایان هم پترسن در مورد یک آرکیtekچر اپن سورس گفت. RISC-V یک اینستراکشن ست اپن سورس است.

## د:

تاریخ تکرار می‌شود. همیشه نیازمند پویایی و بروز کردن امکانات خود در همه ابعاد هستیم. مثل گذشته که رشد متوقف شده بود و نسل ما تغییری ایجاد کردیم، الان نوبت شماست.

## ث:

در مورد پیام بلی. در مورد ایده هنسی که آینده سخت‌افزار اینتگریت کردن سکیوریتی و جنرال پریس نبودن سی‌پی‌یو ها هست،

این طور میبینم که هر دوی این ایده ها به هم گره خورده هستن. و ایده اصلی این هست که سخت افزار بتونه از دومین نرم‌افزاری که در لحظه داره روش اجرا میشه آگاه باشه.

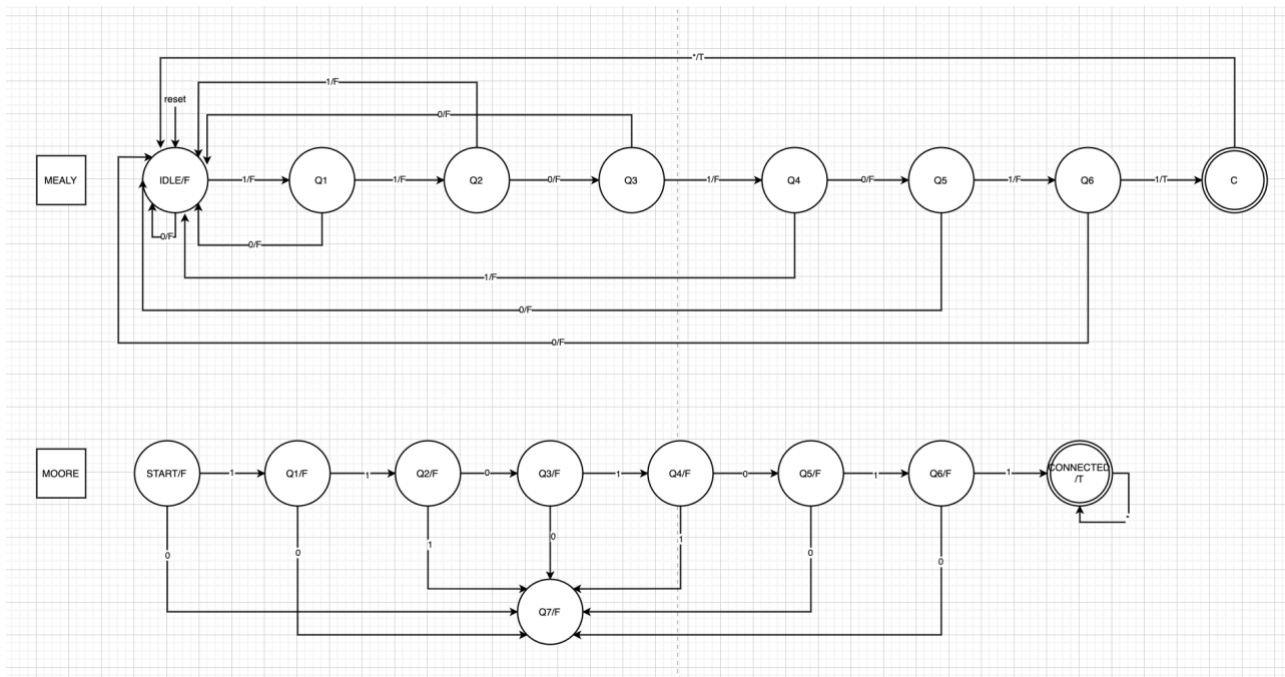
بعد این به نظرم میرسه که شاید CPU هامون یکم شبیه FPGA هایی بشن از دو سو بهشون نگاه بشه. نگاه ساختار CPU و نگاه Instruction در حال اجرا.

و مثلاً این FPGA ها در حین ران تایم، بتونن سنتر بشن دوباره. مثلاً یه سکتی ای داشته باشیم سرکانکتست سویچ پراسس، در در این لحظه CPU برای پراسس آتی دوباره سنتر بشه. حالا یه واحد خارجی هم باشه که این سنتر شدن رو هندل کنه...

حرف هم منطقی یه. راستش با گذر این همه سال از لینوکس، دیگه به نظر نمی‌یاد که قرار باشه یه تغییر خیلی بنیادی در نوع نگرش ما به OS رخ بده. با این اوصاف چرا ما سرویس و اینتگریشن بالاتری بین کرنل و سخت افزار نداشته باشیم؟ مثل همین آگاهی از دومین یا سکیوریتی.

فقط نگرانی این هست که یه CPU سنتر شده قطعاً پرفورمنس یه ASIC رو نداره و توی این ترید آف کی برنده میشه؟ چیزی هست که شدنش انقلابی یه اما نشدن پر از هزینه.

سوال ۲:  
الف و ب:



کد قسمت‌های ج و د را به فایل با نام‌های Q2iii.v و Q2iv.v ضمیمه شده است.

### سوال ۳:

به فایل با نام smarthHouse.v ضمیمه شده است.

