

پروژه درس طراحی سیستم‌های بی‌درنگ

حسین آقائی - ۹۸۱۰۵۶۱۹

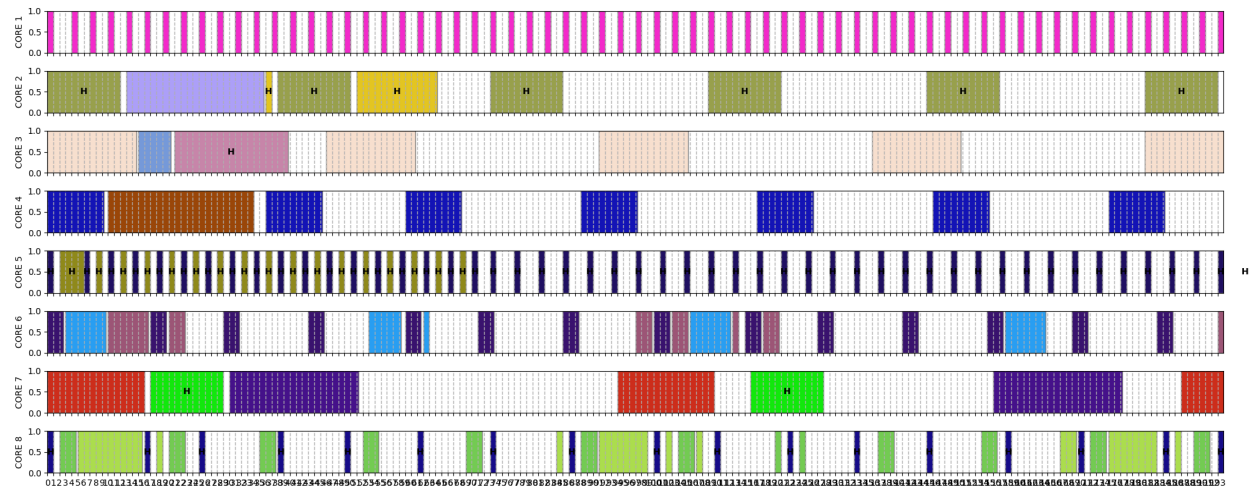
آرمین دلگسار - ۹۸۱۰۱۵۴۴

ابتدا فرضیاتی را که در طول پروژه داشتیم مطرح می‌کنیم.

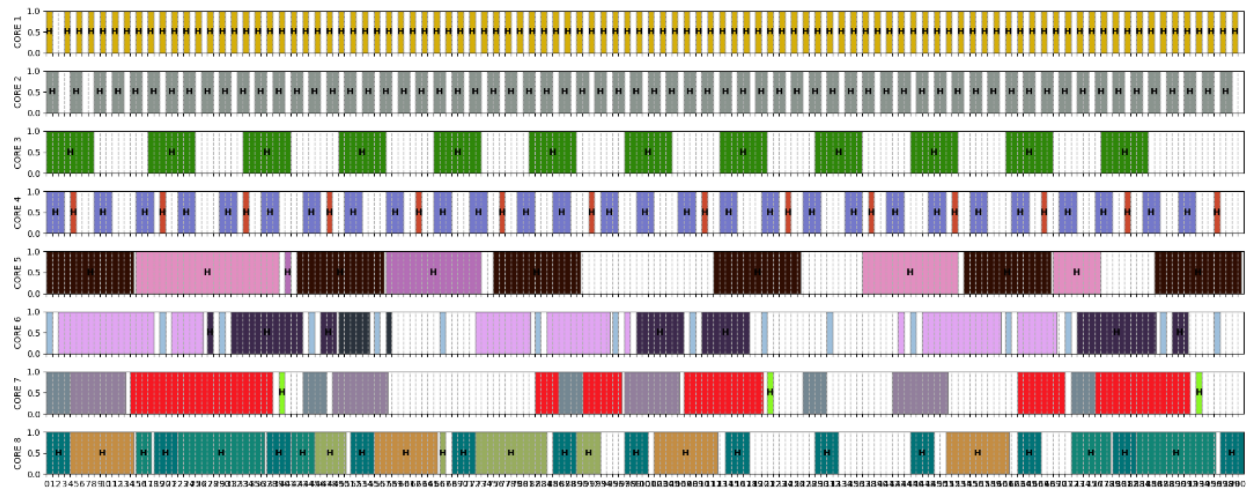
- فرکانس هر processor بین ۱.۶ تا ۱.۲ گیگاهرتز مشابه cortex-A7 خواهد بود. این یعنی وظایف با زمان اجرایی که از الگوریتم uunifast بدست می‌آید نیاز به فرکانس بیشینه ۱.۶ دارند و ما در صورت نیاز و امکان، فرکانس را تا پایین‌ترین حد ممکن کاهش خواهیم داد.
- الگوریتم uunifast تنها utilization تسک‌ها را به ما می‌دهد، در نتیجه ما مقدار زمان اجرا را یک عدد رندوم بین ۱ و ۲۰ انتخاب می‌کنیم و براساس پریود وظیفه را از تقسیم زمان اجرا به utilization بدست می‌آوریم.
- برای درک بالاتر نتایج، ما تعداد وظایف را در خروجی، ۲۰ و تا زمان ۲۰۰ رسم می‌کنیم.
- ما همه وظایف از جمله سخت و نرم را از نوع periodic فرض کردیم.
- چون با الگوریتم uunifast وظایفی که تولید می‌کنیم، utilization های ۰.۳ و ۰.۵ و ۰.۷ دارند، پس هیچ‌وقت وظیفه‌ای ددلاین خود را از دست نخواهد داد اگر از الگوریتم EDF برای هر processor به صورت مستقل استفاده کنیم.
- طبق فرض اول، تمام processorها در صورت روشن بودن، با فرکانس حداقلی ۱.۲ گیگاهرتز کار خواهند کرد. در نتیجه بهترین حالتی که می‌توانیم داشته باشیم این است که وظایف را تا حد امکان با این فرکانس اجرا کنیم تا توان مصرفی به حداقل برسد. برای این کار ما مقدار زمانی را که حداکثر پردازنده idle است بدست می‌آوریم و به طور مساوی در زمان‌های مختلف پخش می‌کنیم و تسک‌ها را با فرکانس کمتر اجرا می‌کنیم تا زمان بیشتری مصرف کنند. در بهترین حالت، پردازنده در تمام زمان خود، با فرکانس ۱.۲ اجرا خواهد شد.
- برای اینکه فرض بالا برای همه processor ها ممکن باشد، ما با استفاده از الگوریتم binpacking، وظایف را روی هسته‌ها به طوری که همه آنها کمترین اختلاف را از utilization داشته باشند پخش می‌کنیم.

حال برای ۳ حالت مختلف ۰.۳ و ۰.۵ و ۰.۷، هرکدام ۲۰ تسک می‌سازیم و تا ثانیه ۲۰۰ نمودار زمان‌بندی آن‌ها را رسم می‌کنیم. طبق فرضیاتی که مطرح شد، هیچ وقت وظیفه‌ای ددلاین خود را از دست نخواهد داد، در نتیجه نرخ miss deadline صفر خواهد بود و نموداری برایش نخواهیم داشت.

Utilization 0.3:



Utilization 0.5:



Utilization 0.7:

