

تمرین اول

موضوع: پیاده‌سازی مرتب‌سازی درجی با زبان اسمبلی

محیط Keil

میکروکنترلر ARM Cortex-M4

برد STM32F401 Nucleo-64

پوشه پروژه در کنار گزارش قرار داده شده است.

## توضیحات کلی برنامه

در ابتدا ما رجیسترهای مورد نیاز را نامگذاری کردیم تا در ادامه کدنویسی ساده تر شود.

سپس فضا ها را تعریف کردیم. طبق صورت سوال یک فضا به صورت فقط خواندی داریم که در آن دو مقدار NUMELEMS و INITELEMS داریم. با استفاده از دستور DCD مقداردهی اولیه کردیم. همچنین فضای ELEMS را به صورت خواندی نوشتنی تعریف کردیم تا مقادیر اولیه را در آن بارگذاری و مرتب کنیم. آرایه ELEMS با مقدار 0 مقداردهی اولیه شده است. و در اخر یک فضای فقط خواندی برای کد تعریف کردیم.

دستورات bit16 فقط به رجیسترهای r0-r7 و bit32 به رجیسترهای r8-r15 دسترسی دارند. برای این فضا از دستور align استفاده کردیم تا بتوان آن ها را در هر آدرسی قرار داد.

در ابتدا آدرس آرایه INITELEMS و آرایه ELEMS و مقدار NUMELEMS را درون رجیسترها بارگذاری کردیم تا از آن در ادامه استفاده کنیم. در ابتدا مقادیر اولیه را از آرایه INITELEMS به ELEMS با یک حلقه منتقل کردیم. و در ادامه روتین INSSORT را فراخوانی کردیم تا آرایه ELEMS را مرتب کند.

روتین INSSORT از یک حلقه تشکیل شده که بر روی تمامی خانه های آرایه ELEMS حرکت میکند. در داخل آن ساب روتینی به اسم WHILE\_LOOP فراخوانی میشود تا اگر قسمت مرتب شده، بهم ریخته باشد، آن را مرتب کند. قبل از ورود به ساب روتین LR درون پشته ذخیره میشود تا بعد از BL کردن به ساب روتین آدرس بازگشت از بین نرود.

در نهایت بعد از مرتب شدن آرایه LR را از پشته میخوانیم و مقدار آن را درون PC قرار میدهیم تا به لیبل پایانی برسد و برنامه خاتمه یابد.

## INITELEMS خواندنی نوشتنی

با تعریف INITELEMS به صورت خواندنی نوشتنی، کاربر میتواند مقادیر آرایه تغییر داده و از آن استفاده کند. در سناریو دوم، چون مرتب سازی درجی به صورت درجا کار میکند، میتواند در همین آرایه انجام شود و دیگر نیاز به فضای دوم نیست.

اسکرین شات هایی از ارایه ELEMS که حالت آن را قیل، چین و بعد از اجرا رویتن مرتب سازی نشان می دهد.

[illegible]

[illegible][illegible]