

마이크로프로세서 프로젝트보고서

1. 프로젝트 개요

a.

2. 32-bit RGBA 이미지를 메모리의 연속된 주소공간에 저장하는 함수 ARM code

3. 3가지 이미지 변환 함수의 동작 방식 및 block diagram

a. Function1 : 32-bit RGBA image → 8-bit RGB image

b. Function2 : 32-bit RGBA image → grayscale image

c. Function3 : 32-bit RGBA image → binary (black/white) image

4. 3가지 이미지 변환 함수의 C언어 코드 및 ARM code

a. Function1

- C코드 및 ARM code 결과물
- 시뮬레이션 결과 및 설명

b. Function2

- C코드 및 ARM code 결과물
- 시뮬레이션 결과 및 설명

c. Function3

- C코드 및 ARM code 결과물
- 시뮬레이션 결과 및 설명

5. ARM code 최적화를 위한 C언어 코드 및 ARM code

a. Function4: Memory relocation (32-bit RGBA pixels → {8-bit R}, {8-bit G}, {8-bit B})

- 특이점

b. (Function4 적용) 수정된 Function1

- C코드 및 ARM code 결과물
- 시뮬레이션 결과 및 설명

c. (Function4 적용) 수정된 Function2

- C코드 및 ARM code 결과물
 - 시뮬레이션 결과 및 설명
 - d. (Function4 적용) 수정된 Function3
 - C코드 및 ARM code 결과물
 - 시뮬레이션 결과 및 설명
 - e. 결과 비교
6. ARM code 추가 최적화
- 성능 비교
 - 결과 비교
7. ④의 ARM code와 ⑤의 ARM code 사이의 성능 비교
- 성능 비교
8. 프로젝트 수행일지
9. Trouble shooting