



EDDI

Electronic Design
Development Institute

에디로봇아카데미

임베디드 마스터 Lv1 과정

제 3기

2022. 01. 31

김원석

CONTENTS

- #define과 const
- Calling Convention
- 함수를 작성하는 방법
- 배열을 만드는 방법

#define과 const

- C언어에서 상수를 선언하는 방법은 const를 사용하여 상수를 만드는 방법이 있고 #define을 활용하여 매크로 상수를 만들어 값을 치환하는 방법이 있다.

차이점

- #define으로 선언한 상수는 메모리를 할당 받지 않는다. 전처리시에 단순 치환된다. const는 메모리를 할당 받는 변수다.
- const는 변수이므로 디버깅시에 쉽게 값을 확인할 수 있고, 데이터 타입을 선언할 수 있어 관련 오류를 쉽게 감지할 수 있다.
- 수학적식을 사용해 상수 선언을 했을 때, 상수를 포함한 계산식 사용에 유의해야 한다. #define은 수학적식이 그대로 치환되고, const는 선언 시 사용한 수학적식을 계산한 값이 메모리에 저장되어 있어 있다.

Calling Convention

- 하드웨어 자원은 유한하다.
- 메모리 계층 구조(Memory Hierarchy)를 살펴보면 레지스터는 매우 빠르지만 개수가 유한하다.
- 따라서 레지스터를 효율적으로 사용해야 한다.
- 레지스터를 효율적으로 사용하기 위해 함수 호출 시 입력 몇 개까지는 레지스터로 입력 받고 입력이 일정 개수 이상이 되면 Stack에 저장한다.
- 이러한 함수 호출 규약을 Calling Convention 이라고 한다.

함수를 작성하는 방법

1. 리턴(반환)타입을 작성한다.
2. 함수의 이름을 작성한다.
3. 소괄호에 입력(파라미터)을 작성한다.
4. 중괄호 내부에 이 함수가 할 일을 작성한다.

```
int test_func(int num)
{
    return num * 2;
}
```

<Example>

배열을 만드는 방법

1. 데이터 타입을 적는다.
2. 변수 이름을 적듯이 배열의 이름을 적는다.
3. 대괄호를 열고 몇 개를 만들지 개수를 작성한다.
4. 초기화 코드인 `= { 0 }` 을 작성한다.

옵션 : 초기값을 별도로 주려면 `{ 1, 3, 7 }`과 같은 형식으로 작성한다.

※ 배열의 이름은 배열의 시작 주소와 같다.

```
int arr[THE_OTHER_MAX] = { 0 };
```

<Example>