AVR 프로그래밍

김상원

https://youtu.be/E8Ovm7f_xmo

한성대학교 HANSUNG UNIVERSITY

CryptoCraft LAB

AVR이란?

기초 어셈블리 구현

코드 분석

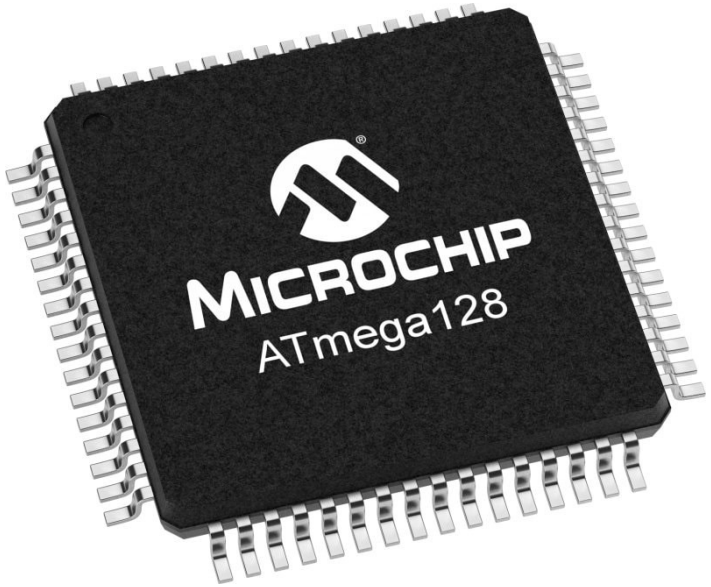
Q & A

AVR이란?

- 1996년 미국 Atmel에서 개발
- RISC(Reduced Instruction Set Computing) 아키텍처
- **8-bit 프로세서**
- 저전력 운영으로 배터리 작동 장치 및 이동형 장치에 적합함
- **ATmega128**, UC3, XMEGA, megaAVR 등 다양한 모델 포함
- **ATmega128**이 교육용으로 가장 널리 사용됨

AVR이란?

- ATmega128 스펙
  - 133개의 RISC 명령어
  - 32개의 8-bit 범용 레지스터
  - 16MHz CPU 클록
  - 128KB 플래시 메모리
  - 4KB EEPROM
  - 4KB SRAM



기초 어셈블리 구현

- 사용할 명령어 모음

명령어	동작
ST	레지스터에서 메모리로 저장
LD	메모리에서 레지스터로 로드
MOVW	레지스터 2개(워드) 단위로 이동(복사)
ADD	레지스터 덧셈
RET	함수 반환

기초 어셈블리 구현

- adder.s

```
MOVW R26, R24
LD R18, X
MOVW R26, R22
LD R19, X
MOVW R30, R20
ADD R18, R19
ST Z, R18
RET
```

```c
#include <avr/io.h>

extern void adder(char *x, char *y, char *z);

int main(void)
{
    char a = 2;
    char b = 8;
    char c = 0;

    adder(&a, &b, &c);
}
```

```
.global adder //adder가 전역변수
.type adder, @function //adder는 함수다

adder :
MOVW R26, R24
LD R18, X
MOVW R26, R22
LD R19, X
MOVW R30, R20

ADD R18, R19

ST Z, R18

RET
```

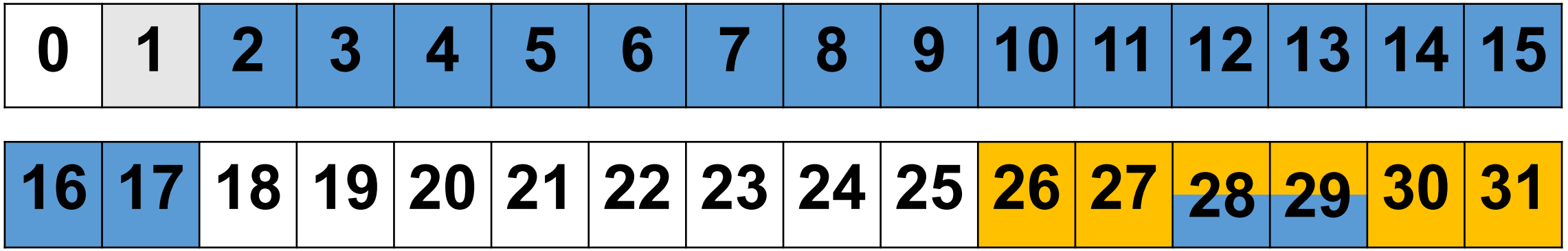
기초 어셈블리 구현

- MOVW Rd, Rr
- ADD Rd, Rr
- LD Rd, X
- ST Z, Rr
  - Rd : 목적지 레지스터
  - Rr : 출발 레지스터
  - X, Z : 포인터 레지스터

```
MOVW R26, R24
LD R18, X
MOVW R26, R22
LD R19, X
MOVW R30, R20
ADD R18, R19
ST Z, R18
RET
```

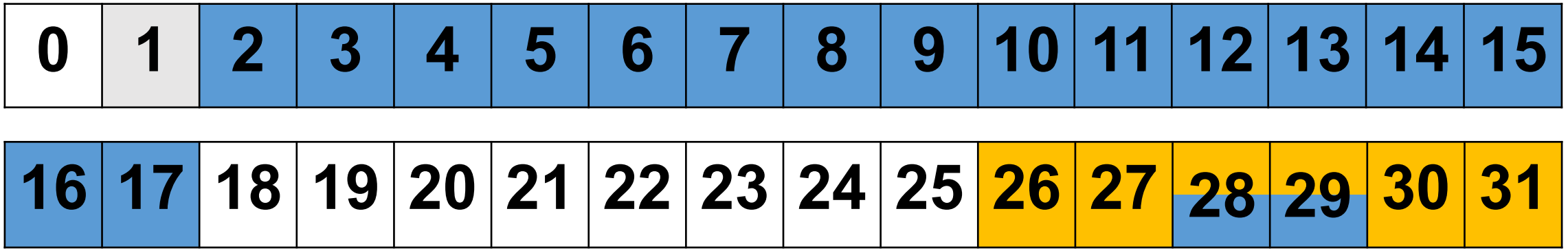
- 즉, MOVW R26, R24는
  R24, R25의 내용을 R26, R27로 이동(복사)

코드 분석



- R1 : ZERO, 종료 시에는 항상 0으로 유지
- R2~R17, R28, R29 : Callee Saved, 사용 이전의 값 보존이 필요
- R26, R27 : X pointer
- R28, R29 : Y pointer
- R30, R31 : Z pointer

코드 분석



- 매개변수는 (R24,R25), (R22, R23), (R20, R21) ... 순서로 입력
  - 변수의 주소값이 16-bit 형태로 저장되어 있음
- 포인터로 활용하기 위해서는 X, Y, Z를 통해야 함
- 따라서 MOVW 명령어를 통해 X, Y, Z 포인터로 사용 가능한 레지스터로 값을 이동

코드 분석

- R20 ~ R25에 매개변수 주소 값이 저장
- 이 상태에서는 값을 불러올 수 없으므로, 각각을 X 또는 Z로 이동
  - Y는 callee saved 이므로 사용하지 않음
- LD를 통해서 메모리에 위치한 값을 가져올 수 있음

```
extern void adder(char *x, char *y, char *z);

int main(void)
{
    char a = 2;
    char b = 8;
    char c = 0;

    adder(&a, &b, &c);
}
```

adder Unknown identifier

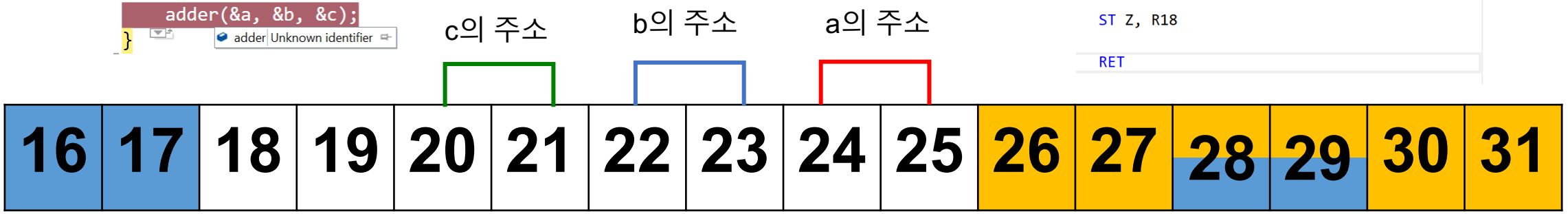
```
.global adder //adder가 전역변수
.type aader, @function //adder는 함수다

adder :
MOVW R26, R24
LD R18, X
MOVW R26, R22
LD R19, X
MOVW R30, R20

ADD R18, R19

ST Z, R18

RET
```



Q & A