
Chapter 130. 퓨즈 비트

퓨즈 비트는 마이크로컨트롤러 프로그래밍에서 중요한 일부분이지만 한 번 설정한 이후에는 변경하는 경우가 거의 없어 퓨즈 비트를 다루는 경우는 거의 없다. 하지만 퓨즈 비트를 잘못 설정하면 마이크로컨트롤러가 동작하지 않을 수도 있으므로 주의해야 한다. ATmega328에는 플래시 메모리, EEPROM, DRAM의 3종류 메모리가 포함되어 있다는 점은 앞에서 설명하였다. 여기에 3바이트의 비휘발성 메모리가 숨겨져 있으며 이 메모리가 이 장에서 이야기 할 퓨즈 비트이다. 3바이트의 퓨즈는 각각 확장 퓨즈(extended fuse), 하이 퓨즈(high fuse), 로 퓨즈(low fuse)라고 불린다. 퓨즈 설정에서 주의할 점은 '0'의 값을 가지는 비트가 설정된 비트이며 '1'의 값을 가지는 비트는 설정되지 않은 비트를 나타낸다는 점이다.

1. 하이 퓨즈

먼저 하이 퓨즈 바이트를 살펴보자. 표 1은 ATmega328의 하이 퓨즈 바이트를 나타낸 것으로 공장 출하시의 디폴트값과 아두이노 우노에서 설정된 값을 함께 나타내었다.

비트 이름	비트 번호	설명	디폴트값	Arduino UNO 설정값
RSTDISBL	7	외부 리셋 금지	1 (외부 리셋 가능)	1
DWEN	6	디버그 가능	1 (디버그 불가능)	1
SPIEN	5	ISP 시리얼 프로그래밍 가능	0 (ISP 시리얼 프로그래밍 가능)	0
WDTON	4	와치독 타이머 항상 켜기	1 (와치독 타이머 끄)	1
EESAVE	3	칩 내용을 지울 때 EEPROM 내용 보존	1 (EEPROM 내용도 지움)	1
BOOTSZ1	2	부트로더 크기	00 (4Kbyte)	11
BOOTSZ0	1			
BOOTRST	0	리셋 시 시작 시점	1 (0번지부터 시작)	0 (부트로더에서 시작)

표 1. 하이 퓨즈 바이트

(1) RSTDISBL (Reset Disable)

외부 리셋을 금지시킨다. 외부 리셋은 ATmega328의 1번 핀에 의한 리셋을 가리키며 1번 핀을 리셋으로 사용하지 않는 경우에는 포트 C의 6번 핀으로 일반적인 입출력 핀으로 사용할 수 있다. 하지만 리셋 핀은 ISP를 통한 프로그램 다운로드에도 사용되므로 외부 리셋이 금지되면 ISP 역시 사용할 수 없다. 리셋 핀에 의한 리셋을 사용할 수 없는 경우에도 전원이 켜질 때의 리셋(power-on reset)과 브라운 아웃에 의한 리셋(brown out reset) 등은 사용할 수 있다.

(2) DWEN (Debug Wire Enable)

디버그 기능을 활성화시킨다. 디버그 기능이 활성화되면 칩 내 디버깅(on-chip debugging) 기능을 사용할 수 있다. 디버깅은 디버그 와이어 인터페이스(debug wire interface)를 통해 이루어지며 전용 장비가 있어야만 사용할 수 있다.

SPIEN (SPI Serial Programming Enable) : 시리얼 프로그래밍이 가능하도록 설정한다. 시

리얼 프로그래밍이 금지되면 ISP 방식으로 프로그래밍이 불가능해지므로 주의하여야 한다.

(3) WDTON (Watchdog Timer Always On)

와치독 타이머 기능을 활성화시킨다. 와치독 타이머는 시스템의 자동 리셋을 위한 타이머이다. 와치독 타이머를 일정 시간 간격으로 설정하면 와치독 타이머가 만료되기 이전에 갱신해 주어야 하며 시스템의 처리 지연 등 비정상적인 동작으로 인해 갱신에 실패하면 시스템이 자동으로 리셋된다. 와치독 타이머를 갱신해 주는 작업을 ‘와치독 타이머 리셋’이라 하고, 와치독 타이머 만료에 따른 시스템의 리셋을 ‘와치독 리셋’이라고 부르므로 혼돈하지 않도록 한다.

(4) EESAVE (EEPROM Save)

마이크로컨트롤러의 플래시 메모리에 프로그램을 다운로드 할 때 EEPROM은 지우지 않는다. 프로그램 다운로드 과정을 살펴보면 프로그램을 다운로드 하기 전에 플래시 메모리의 내용을 먼저 지우고 프로그램이 다운로드 된 후에는 기록된 내용을 확인하고 있음을 발견할 수 있다. 이 때 EESAVE 비트가 설정되어 있으면 플래시 메모리에 프로그램을 다운로드 하는 과정에서 EEPROM의 내용은 그대로 보존한다.

(5) BOOTSZn (n = 0, 1) (Bootloader Size)

부트로더의 크기를 지정한다. 부트로더는 플래시 메모리의 가장 뒷부분에 위치하며 부트로더의 크기는 최소 256 워드에서 최대 2048 워드까지 4단계로 설정할 수 있다. 그림 1은 플래시 메모리에서의 부트로더 위치 영역을 나타내며, 비트 설정에 따른 부트로더의 크기는 표 2와 같다.

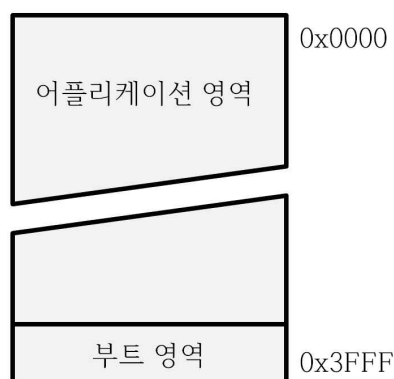


그림 1. ATmega328의 플래시 메모리 구조

BOOTSZ 퓨즈 비트	부트 영역 크기 (word)	프로그램 저장 영역 주소	부트 영역 주소
11	256	0x0000 - 0x3EFF	0x3F00 - 0x3FFF
10	512	0x0000 - 0x3DFF	0x3E00 - 0x3FFF
01	1024	0x0000 - 0x3BFF	0x3C00 - 0x3FFF
00	2048	0x0000 - 0x37FF	0x3800 - 0x3FFF

표 2. BOOTSZn (n = 0, 1) 비트 설정에 따른 부트로더 크기

(6) BOOTrST (Select Reset Vector)

컨트롤러가 리셋되는 경우 부트로더부터 실행이 시작되는지의 여부를 나타낸다. 디폴트값은 1로 부트로더를 사용하지 않고 0번지부터 프로그램이 시작되지만, 아두이노의 경우 부트로더를 사용하므로 BOOTrST 비트는 0으로 설정된다.

2. 로 퓨즈

표 3은 ATmega328의 로 퓨즈 바이트를 나타낸 것으로 공장 출하시의 디폴트값과 아두이노 우노에서 설정된 값을 함께 나타내었다.

비트 이름	비트 번호	설명	디폴트값	Arduino UNO 설정값
CKDIV8	7	클록을 내부적으로 8로 나눔	0 (클록을 낮춤)	1 (클록을 낮추지 않음)
CKOUT	6	클록을 포트 B 0번 핀으로 출력	1 (클록 출력 없음)	1
SUT1	5	초기 구동 시간	1	1
SUT0	4		0	1
CKSEL3	3	클록 소스	0	1
CKSEL2	2		0	1
CKSEL1	1		1	1
CKSEL0	0		0	1

표 3. 로 퓨즈 바이트

(1) CKDIV8 (Divide Clock by 8)

클록을 8로 나누어 사용할 지의 여부를 나타낸다. ATmega328은 공장 출하 시에 내부 RC 오실레이터를 사용하도록 설정되어 있으며, 클록을 8로 나누어 사용하도록 되어 있다. 따라서 기본적으로 1MHz 클록으로 동작하도록 설정되어 있다.

(2) CKOUT (Clock Output)

ATmega328에서 사용하는 클록의 외부 출력 여부를 나타낸다. ATmega328과 동기화가 필요한 외부 회로에 클록을 공급하는 용도로 사용할 수 있다. 클록 출력이 설정되면 해당 핀은 입출력 핀으로 사용할 수 없다.

(3) CKSELn (n = 0, ... , 3) (Select Clock Source)

클록 소스를 선택한다. CKSELn 비트 설정에 따른 클록 소스는 표 4와 같다.

클록 소스	CKSEL3...0
저전력 크리스탈 오실레이터	1111 - 1000
풀 스윙 (full swing) 크리스탈 오실레이터	0111 - 0110
저주파 크리스탈 오실레이터	0101 - 0100
내부 128KHz RC 오실레이터	0011
조정된 (calibrated) 내부 RC 오실레이터	0010
외부 클록	0000
예약됨 (reserved)	0001

표 4. 클록 옵션

표 4에서 알 수 있듯이 ATmega328과 함께 사용할 수 있는 클록의 종류는 다양하다. ATmega328의 공장 출하 시 설정은 00102으로 8MHz로 조정된 내부 RC 오실레이터를 사용한다. 데이터시트에 의하면 ATmega328은 최대 20MHz 클록을 사용할 수 있지만 일반적으로 16MHz 이하의 클록 주파수를 사용하며 아두이노 우노의 경우에도 16MHz를 사용한다. 아두이노 우노의 클록 선택 설정값은 1111₂로 저전력 크리스탈 오실레이터를 사용하도록 설정되어 있다. 풀스윙 크리스탈 오실레이터는 저전력 크리스탈 오실레이터에 비해 전력 소모가 많지만 보다 정확한 클록을 생성할 수 있다. 저주파 크리스탈 오실레이터는 KHz 단위의 클록을 사용할 경우 선택하는 옵션으로 8MHz 또는 16MHz 클록을 사용하는 경우에는 사용되지 않는다. 외부 클록이 사용되는 경우에는 PB6 (XTAL1) 핀에 연결하여 사용하며 PB7 (XTAL2) 핀

은 입출력 핀으로 사용이 가능하다.

(4) SUTn (n = 0, 1) (Select Start-Up Time)

저전력 모드에서 깨어나거나 리셋 이후의 초기 구동 시간을 선택한다. 초기 구동 시간은 사용하는 클록에 따라 달라진다. 8MHz로 조정된 내부 RC 오실레이터가 사용되는 경우 초기 구동 시간은 표 5와 같다.

SUT1...0	파워 다운 모드나 파워 절약(save) 모드에서 구동 시간	리셋 이후 추가 지연 시간 (VCC = 5V)	전원 상황
00	6CK	14CK	BOD enabled
01	6CK	14CK + 4.1ms	fast rising power
10	6CK	14CK + 65ms	slowly rising power
11	-	-	-

표 5. 조정된 내부 RC 오실레이터가 사용되는 경우의 초기 구동 시간

16MHz의 저전력 크리스탈 오실레이터가 사용되는 경우의 초기 구동 시간은 표 6과 같다.

SUT1...0	파워 다운 모드나 파워 절약(save) 모드에서 구동 시간	리셋 이후 추가 지연 시간 (VCC = 5V)	전원 상황
00	-	-	-
01	16K CK	14CK	BOD enabled
10	16K CK	14CK + 4.1ms	fast rising power
11	16K CK	14CK + 65ms	slowly rising power

표 6. 저전력 크리스탈 오실레이터가 사용되는 경우의 초기 구동 시간

8MHz 내부 오실레이터와 16MHz 외부 크리스탈이 사용되는 모든 경우에 리셋 후 가장 긴 초기 구동 시간이 설정되어 있음을 알 수 있다. fast rising power와 slowly rising power는 전원부에서 ATmega328의 동작에 충분한 전력을 공급하기까지의 시간을 말하며 안정적인 동작을 위해 slowly rising power를 흔히 선택한다. 브라운 아웃 감지(BOD, Brown Out Detection) 회로는 마이크로컨트롤러의 VCC를 계속 검사하여 지정한 전압 이하로 전압이 떨어

어지는 경우 마이크로컨트롤러를 리셋하는 역할을 하며, 브라운 아웃 감지가 설정되면 리셋 후 동작을 시작하기 이전에 충분한 VCC가 공급되는 것을 보장하므로 추가 지연 시간이 가장 짧다.

3. 확장 퓨즈

비트 이름	비트 번호	설명	디폴트값	Arduino UNO 설정값
-	7	-	1	1
-	6	-	1	1
-	5	-	1	1
-	4	-	1	1
-	3	-	1	1
BODLEVEL2	2	브라운 아웃 감지 레벨	1	1
BODLEVEL1	1		1	1
BODLEVEL0	0		1	1

표 7. 확장 퓨즈 바이트

표 7은 ATmega328의 확장 퓨즈 바이트를 나타낸 것이다. 확장 퓨즈에는 브라운 아웃 트리거 레벨을 설정하는 3개의 비트만 존재한다. 비트 설정에 따른 브라운 아웃 트리거 레벨은 표 8과 같다.

BODLEVEL 2...0	V _{BOT}
111	BOD 금지
110	1.8
101	2.7
100	4.3
011	예약됨
010	
001	
000	

표 8. 브라운 아웃 트리거 전압

마이크로컨트롤러는 공급 전압 VCC가 브라운 아웃 리셋 임계치 (Brown-out Reset Threshold) 전압인 V_{BOT} 이하로 떨어지는 경우 리셋된다. 이 때 잡음에 의한 리셋을 방지하기 위해 히스테리시스 전압인 V_{HYST} 가 50mV로 정의되어 있어서 실제 리셋이 발생하는 전압은 다음과 같이 정의된다.

$$V_{BOT-} = V_{BOT} + V_{HYST} / 2$$