

1. ¿Cuál es el propósito de los bits de configuración? Explique cada uno de ellos con sus propias palabras.

Existen 3 configuraciones, Extended Fuse Byte, Fuse High Byte y Fuse Low Byte, las cuales permiten controlar el comportamiento del microcontrolador.

**Extended Fuse Byte:** Configuran el nivel de voltaje en el que trabajará el microcontrolador. DE por sí se encuentra en configuración 111 donde el BOD está desactivado.

Table 29-12. BODLEVEL Fuse Coding<sup>(1)(2)</sup>

BODLEVEL 2:0 Fuses	Min. $V_{BOT}$	Typ $V_{BOT}$	Max $V_{BOT}$	Units
111	BOD Disabled			
110	1.7	1.8	2.0	V
101	2.5	2.7	2.9	
100	4.1	4.3	4.5	
011	Reserved			
010				
001				
000				

**Fuse High Byte:** Esta configuración contiene varias combinaciones. Una de ellas es el tamaño del boot, donde se trabaja con el bit 1 y bit 2. Luego existe la combinación WDTON, donde se trabaja con el bit 4, que permite reiniciar el microcontrolador en un tiempo específico. Luego esta el bit RSTDISBL que permite poner en 0 o desactivar el reset físico, que por lo general se encuentra establecido en 1 (desactivado).

High Fuse Byte	Bit No	Description	Default Value
RSTDISBL <sup>(1)</sup>	7	External Reset Disable	1 (unprogrammed)
DWEN	6	debugWIRE Enable	1 (unprogrammed)
SPIEN <sup>(2)</sup>	5	Enable Serial Program and Data Downloading	0 (programmed, SPI programming enabled)
WDTON <sup>(3)</sup>	4	Watchdog Timer Always On	1 (unprogrammed)
EESAVE	3	EEPROM memory is preserved through the Chip Erase	1 (unprogrammed), EEPROM not reserved
BOOTSZ1	2	Select Boot Size (see <a href="#">Table 27-7 on page 284</a> , <a href="#">Table 27-10 on page 285</a> and <a href="#">Table 27-13 on page 286</a> for details)	0 (programmed) <sup>(4)</sup>
BOOTSZ0	1	Select Boot Size (see <a href="#">Table 27-7 on page 284</a> , <a href="#">Table 27-10 on page 285</a> and <a href="#">Table 27-13 on page 286</a> for details)	0 (programmed) <sup>(4)</sup>
BOOTRST	0	Select Reset Vector	1 (unprogrammed)

**Fuse Low Byte:** Permite controlar el funcionamiento del reloj desde los bits CKSEL3, CKSEL2, CKSEL1 y CKSEL0. Donde se puede seleccionar la fuente de reloj dentro del microcontrolador. El bit CKOUT permite que la señal del reloj se muestre en el PORTB0. Luego existe otro tipo de reloj, el CKDIV8, el cual divide la señal de reloj en 8. Por último se puede seleccionar el tiempo en que comienza el reloj con las funciones SUT1 y SUT0.

Table 28-9. Fuse Low Byte

Low Fuse Byte	Bit No	Description	Default Value
CKDIV8 <sup>(4)</sup>	7	Divide clock by 8	0 (programmed)
CKOUT <sup>(3)</sup>	6	Clock output	1 (unprogrammed)
SUT1	5	Select start-up time	1 (unprogrammed) <sup>(1)</sup>
SUT0	4	Select start-up time	0 (programmed) <sup>(1)</sup>
CKSEL3	3	Select Clock source	0 (programmed) <sup>(2)</sup>
CKSEL2	2	Select Clock source	0 (programmed) <sup>(2)</sup>
CKSEL1	1	Select Clock source	1 (unprogrammed) <sup>(2)</sup>
CKSEL0	0	Select Clock source	0 (programmed) <sup>(2)</sup>

Table 9-1. Device Clocking Options Select<sup>(1)</sup>

Device Clocking Option	CKSEL3...0
Low Power Crystal Oscillator	1111 - 1000
Full Swing Crystal Oscillator	0111 - 0110
Low Frequency Crystal Oscillator	0101 - 0100
Internal 128kHz RC Oscillator	0011
Calibrated Internal RC Oscillator	0010
External Clock	0000
Reserved	0001

Factory Default:  
LFuse = 01100010

Arduino Default:  
LFuse = 11111111

Note: 1. For all fuses "1" means unprogrammed while "0" means programmed.

2. ¿Qué opciones de oscilador tiene el uC? Explique las diferentes opciones con sus palabras
  - Low POver Crystal Oscilator: De esta manera el cristal de oscilación consume la mayor cantidad de energía, aunque puede presentar ruido.
  - Full Swing Crystal Oscilator: Controla el ruido y maneja otras señales de entrada de reloj.
  - Low Frequency Crystal Oscillator: Se optimiza de manera eficiente el uso del reloj con una frecuencia de 32.768KHz.
  - Calibrated internal RC Oscillator: provee un aproximado de frecuencia de 8.0KHz.
  - Internal Clock: Controla el microcontrolador a través de una señal de reloj externa.
  - Reserved: No tiene un uso específico.

3. ¿Cuál es la diferencia entre un SFR y un GPR?

Los SFR se encuentran en la memoria de datos y en el General Purpose Registers. Estos controlan la operación del dispositivo mientras que los GPR se encuentran donde se almacenan los datos y las operaciones.

Los GPR controlan todos los periféricos y funciones de trabajo.