pulseInLong()

[Advanced I/O]

Description

pulseInLong() is an alternative to [pulseIn()](https://www.arduino.cc/reference/en/language/functions/advanced-io/pulsein) which is better at handling long pulse and interrupt affected scenarios.

Reads a pulse (either HIGH or LOW) on a pin. For example, if value is HIGH, pulseInLong() waits for the pin to go from LOW to HIGH, starts timing, then waits for the pin to go LOW and stops timing. Returns the length of the pulse in microseconds or gives up and returns 0 if no complete pulse was received within the timeout.

The timing of this function has been determined empirically and will probably show errors in shorter pulses. Works on pulses from 10 microseconds to 3 minutes in length. This routine can be used only if interrupts are activated. Furthermore the highest resolution is obtained with large intervals.

Syntax

pulseInLong(pin, value)  
pulseInLong(pin, value, timeout)

Parameters

pin: the number of the Arduino pin on which you want to read the pulse. Allowed data types: int.  
value: type of pulse to read: either [HIGH](https://www.arduino.cc/reference/en/language/variables/constants/constants) or [LOW](https://www.arduino.cc/reference/en/language/variables/constants/constants). Allowed data types: int.  
timeout (optional): the number of microseconds to wait for the pulse to start; default is one second. Allowed data types: unsigned long.

Returns

The length of the pulse (in microseconds) or 0 if no pulse started before the timeout. Data type: unsigned long.

Example Code

The example prints the time duration of a pulse on pin 7.

int pin = 7;

unsigned long duration;

void setup() {

Serial.begin(9600);

pinMode(pin, INPUT);

}

void loop() {

duration = pulseInLong(pin, HIGH);

Serial.println(duration);

}

Notes and Warnings

This function relies on micros() so cannot be used in [noInterrupts()](https://www.arduino.cc/reference/en/language/functions/interrupts/nointerrupts) context.