

Trabajo Práctico N° 1 – Manejo de Puertos

Objetivo:

Usar los registros de los puertos, ver utilidad de *R pullup* interna

Práctica:

- 1- Crear una rama **TrabajoPractico1** en el repositorio local
- 2- Cambiarse a la rama **TrabajoPractico1** en el repositorio local y crear el directorio **TP1**
- 3- En el directorio **TP1** Hacer un programa (**ParteA.asm**) que haga parpadear un LED Conectado en el PIN 2 (Usar la rutina de retardo Dada). Realizar un informe con el diagrama esquemático del hardware, un diagrama en bloques del código y demás requisitos del informe (**Informe.pdf**)
- 4- Realizar un **Commit** al repositorio local y un **Push** al repositorio remoto en github con los cambios realizados y crear un **Issue** para que los docentes verifiquen el código
- 5- Continuar trabajando en la rama **TrabajoPractico1** y hacer un programa que prenda un LED cuando se presiona el pulsador 1 y quede parpadeando hasta que se apague cuando se presiona el pulsador 2 (**ParteB.asm**). Agregar al informe el diagrama esquemático del hardware, un diagrama en bloques del código y demás requisitos del informe (**Informe.pdf**)
El LED está conectado a un pin del microcontrolador y los pulsadores a otros dos pines a elección.
- 6- Realizar un **Commit** al repositorio local y un **Push** al repositorio remoto en github con los cambios realizados y crear un **Pull Request** para que los docentes verifiquen el código y se integre la rama **TrabajoPractico1** a la rama **Main**
- 7- Una vez aprobado el Trabajo Practico, el docente cierra el **Pull Request**
- 8- Realizar un **Merge** de la rama **TP1** a la rama **Main** en el repositorio local y remoto en github

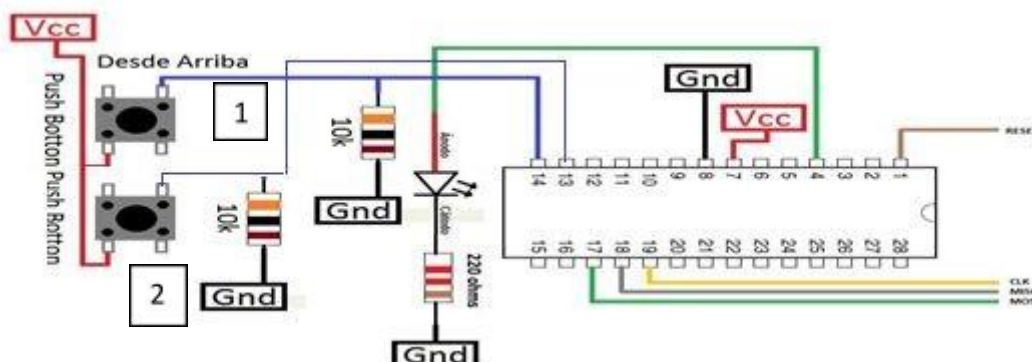
Responder en el Informe:

Cómo modifica el circuito y el programa, para usar la resistencia de *pullup* interna de los ports al conectar el pulsador. ¿Se ahorra algún componente?

Materiales

- 1 LED
- 1 Resistencia de 220 Ohms
- 2 Resistencia de 10 Kohms
- 2 Pulsador
- 1 Microcontrolador ATmega328p
- Programador USBasp V3.0

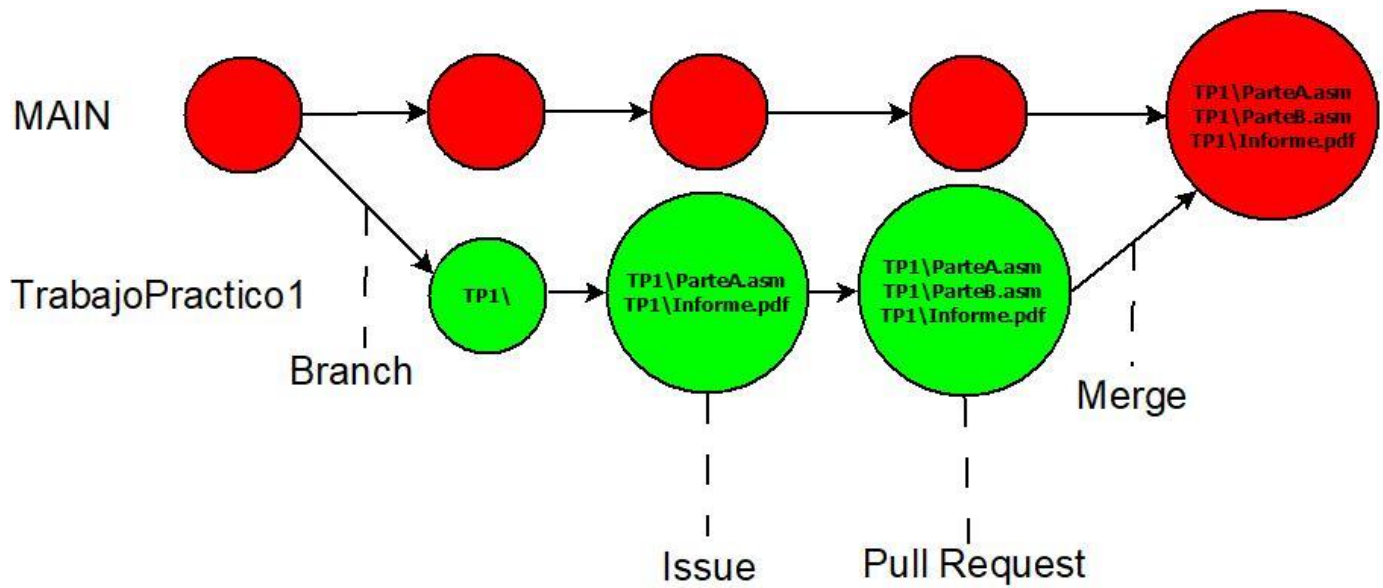
Diagrama Esquemático



Lectura recomendada

“The AVR microcontroller and embedded systems. Embedded system using Assembly and C”. Autores: MUHAMMAD ALI MAZIDI, SARMAD NAIMI, SEPEHR NAIMI

Diagrama del repositorio



Ejemplos de generación de un retardo

<pre> ; Rutina de Retardo de 5ms ;----- delay5ms: ldi Temp1, 66 ; para 8mhz ; 1 ciclo LOOP0: ldi temp2, 200 ; 1 ciclo LOOP1: dec temp2 ; 1 ciclo brne LOOP1 ; 1 si es falso 2 si es ;verdadero dec Temp1 ; 1 brne LOOP0 ; 2 ret </pre>	<pre> ;----- Calculo del tiempo de retardo) Llamada 5ms * 1 Ldi Temp1 * 1 LDI Temp 2 * 66 dec temp2 * 200 brne loop 2 * 200 dec temp1 * 66 brne Loop2 * 66 ret * 1 Tiempo= (5+1+66+(200+400)*66+66+132+4)/8000000= 0,00498seg ~= 5ms </pre>
--	---

; Assembly code

; Delay 8 000 000 cycles

; 1s at 8.0 MHz

ldi r18, 41

ldi r19, 150

ldi r20, 128

L1: dec r20

brne L1

dec r19

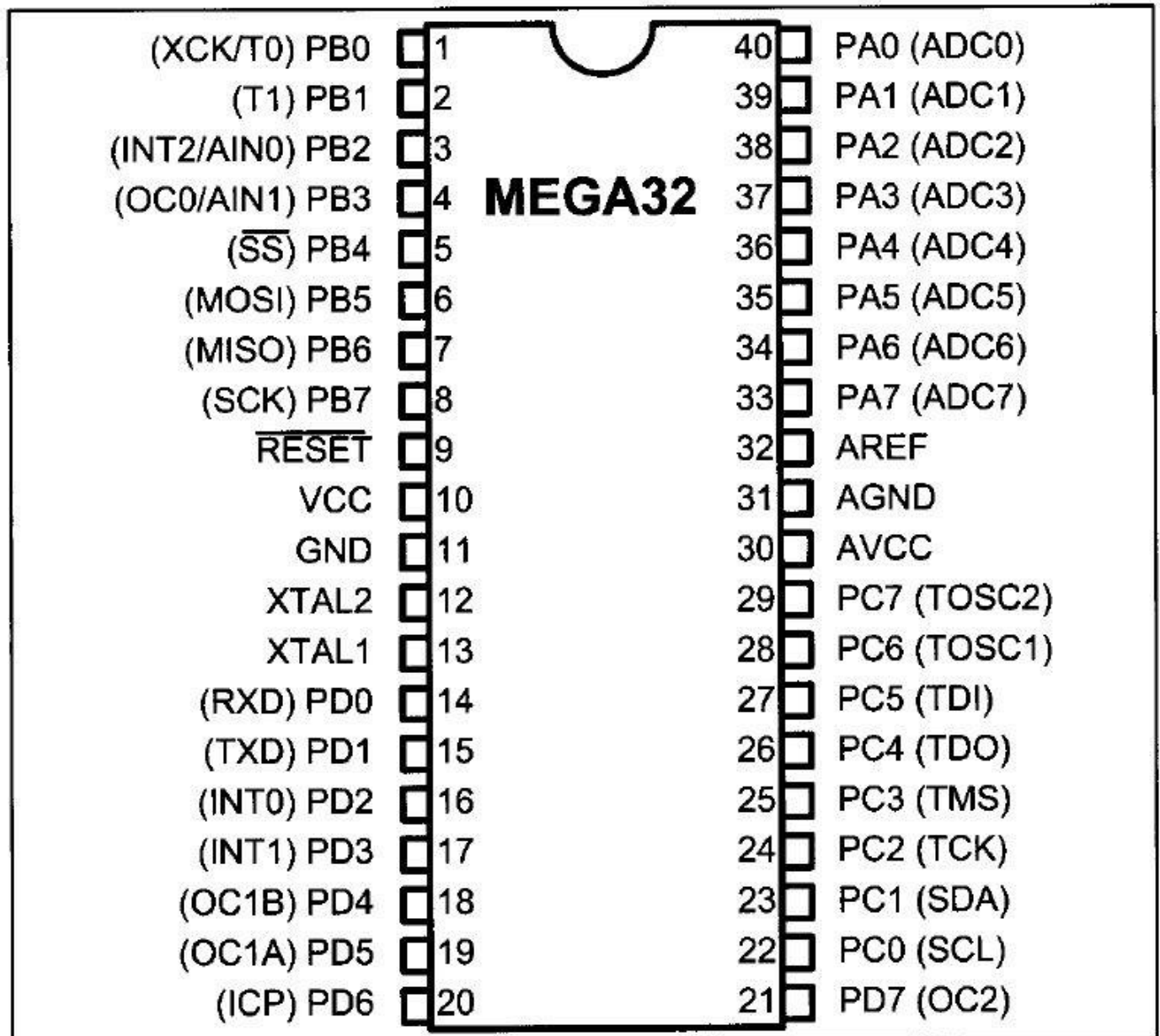
brne L1

dec r18

brne L1

Apéndice

Pinout ATmega32



Pinout ATmega328/p

Pinout

Figure 5-1. 28-pin PDIP

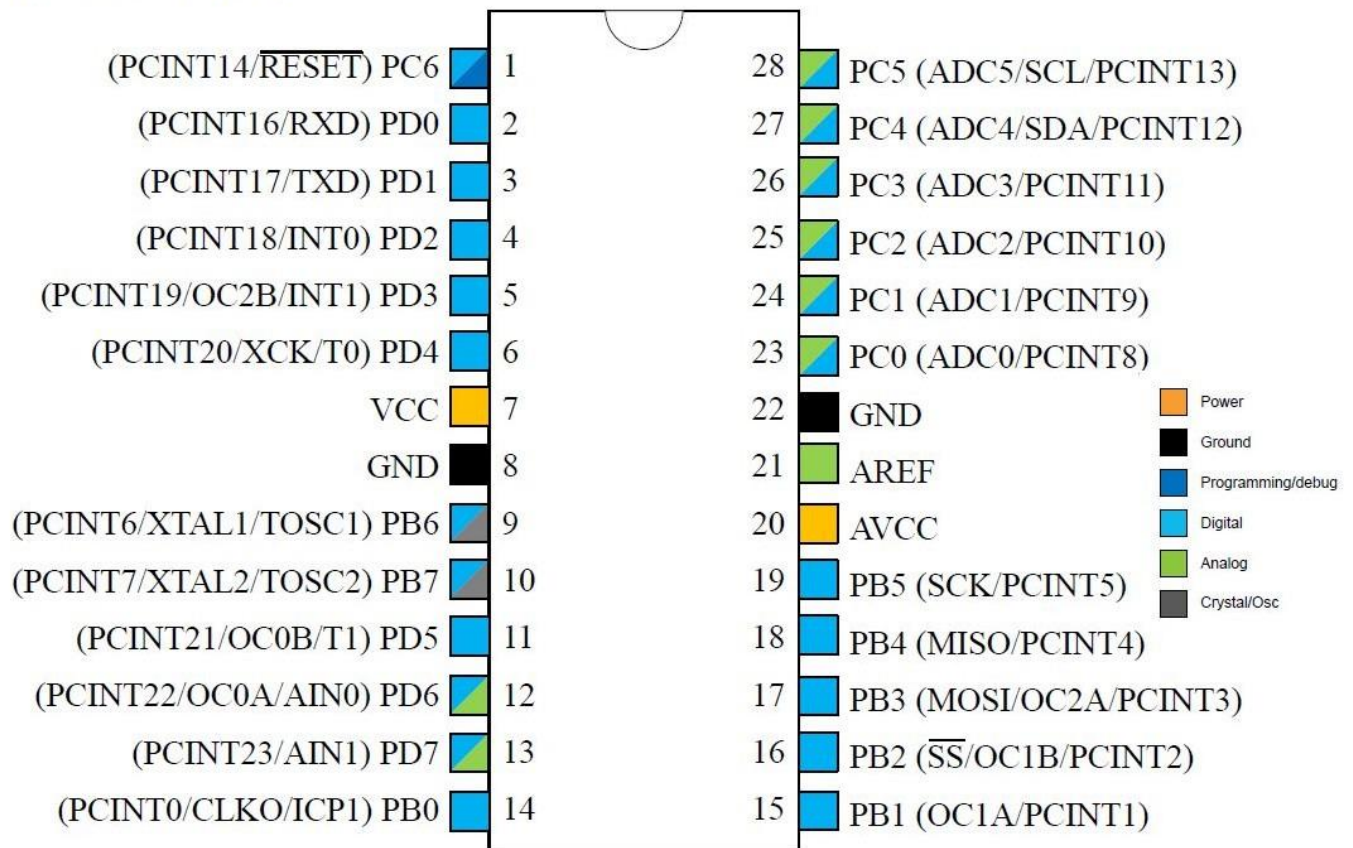


Diagrama de Arduino Uno

