

# CIRCUITOS DIGITALES Y MICROCONTROLADORES 2022

Facultad de Ingeniería  
UNLP

Teclado Matricial

Ing. José Juárez

# Teclados Matriciales

- Arreglo de pulsadores en filas y columnas.
- Utiliza el mínimo posible de pines.
- 4x4 permite leer 16 teclas con 8 pines.
- Varios formatos: 2x2, 3x4, etc.



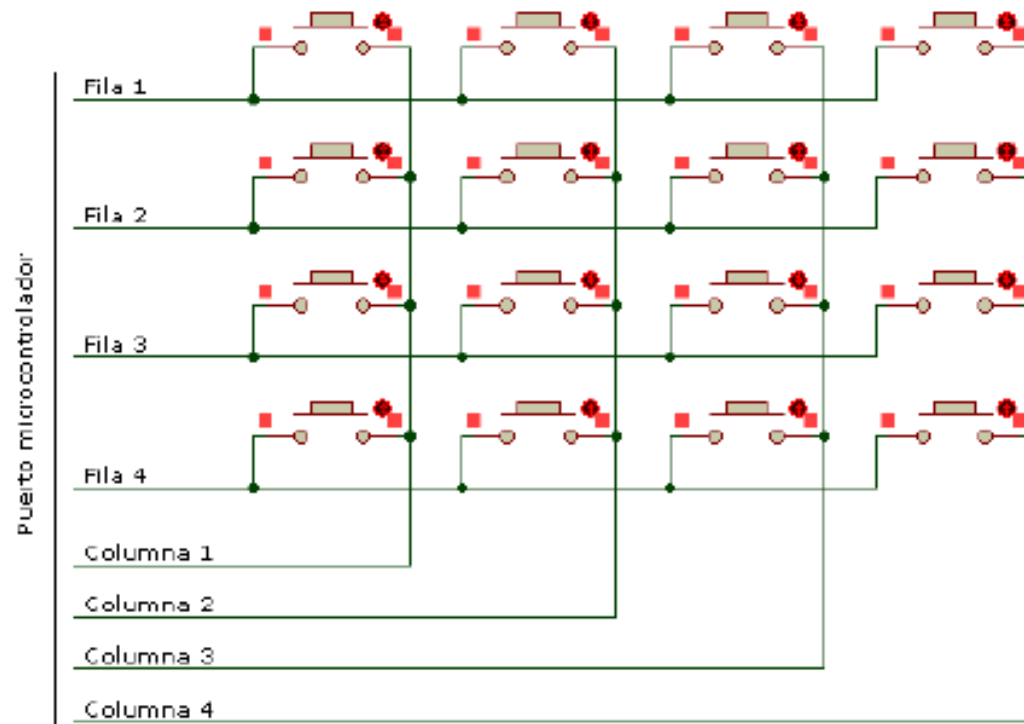
# Conexionado a un uC

- Teclado matricial 4x4

Una SALIDA  
del uC por  
cada fila



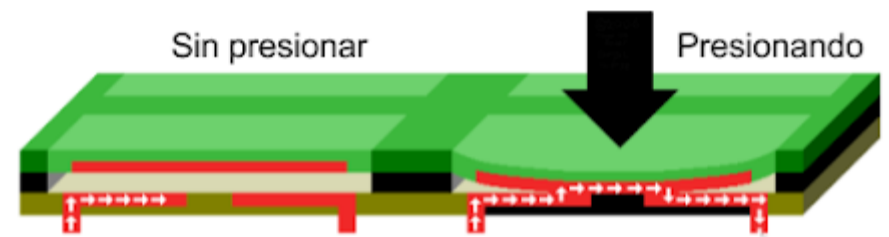
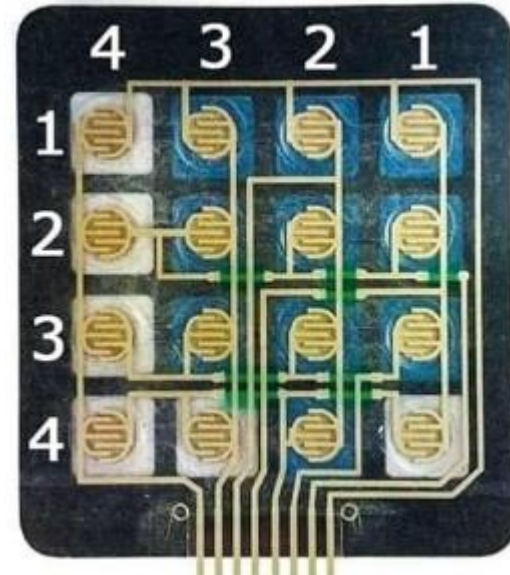
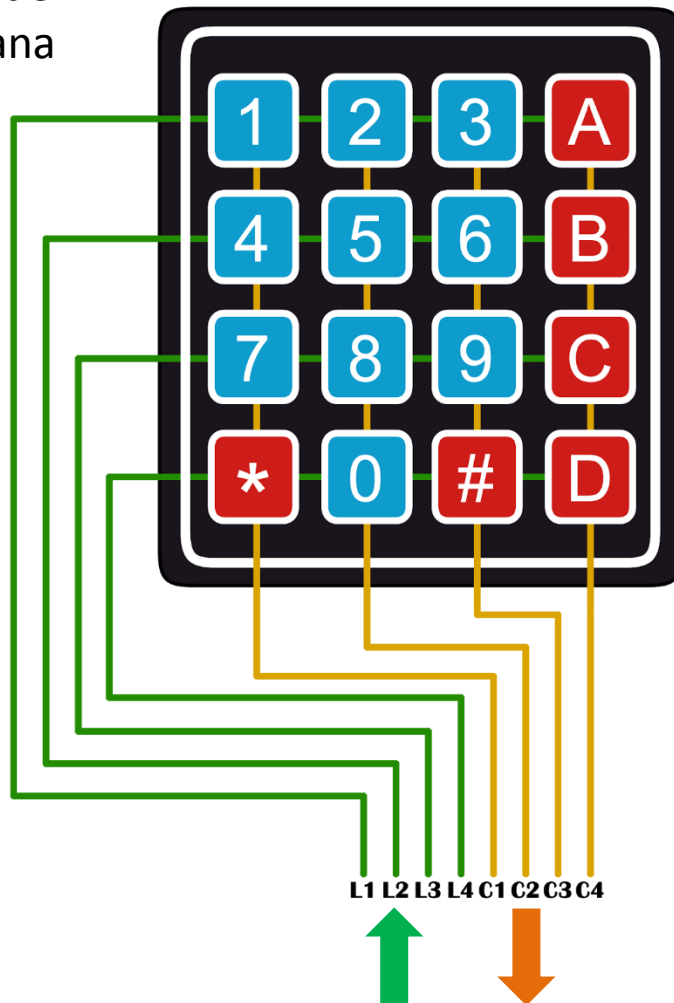
Una ENTRADA  
del uC por  
cada Columna



**ATENCIÓN! Las entradas deben configurarse con resistencias de Pull-UP o Pull-Down**

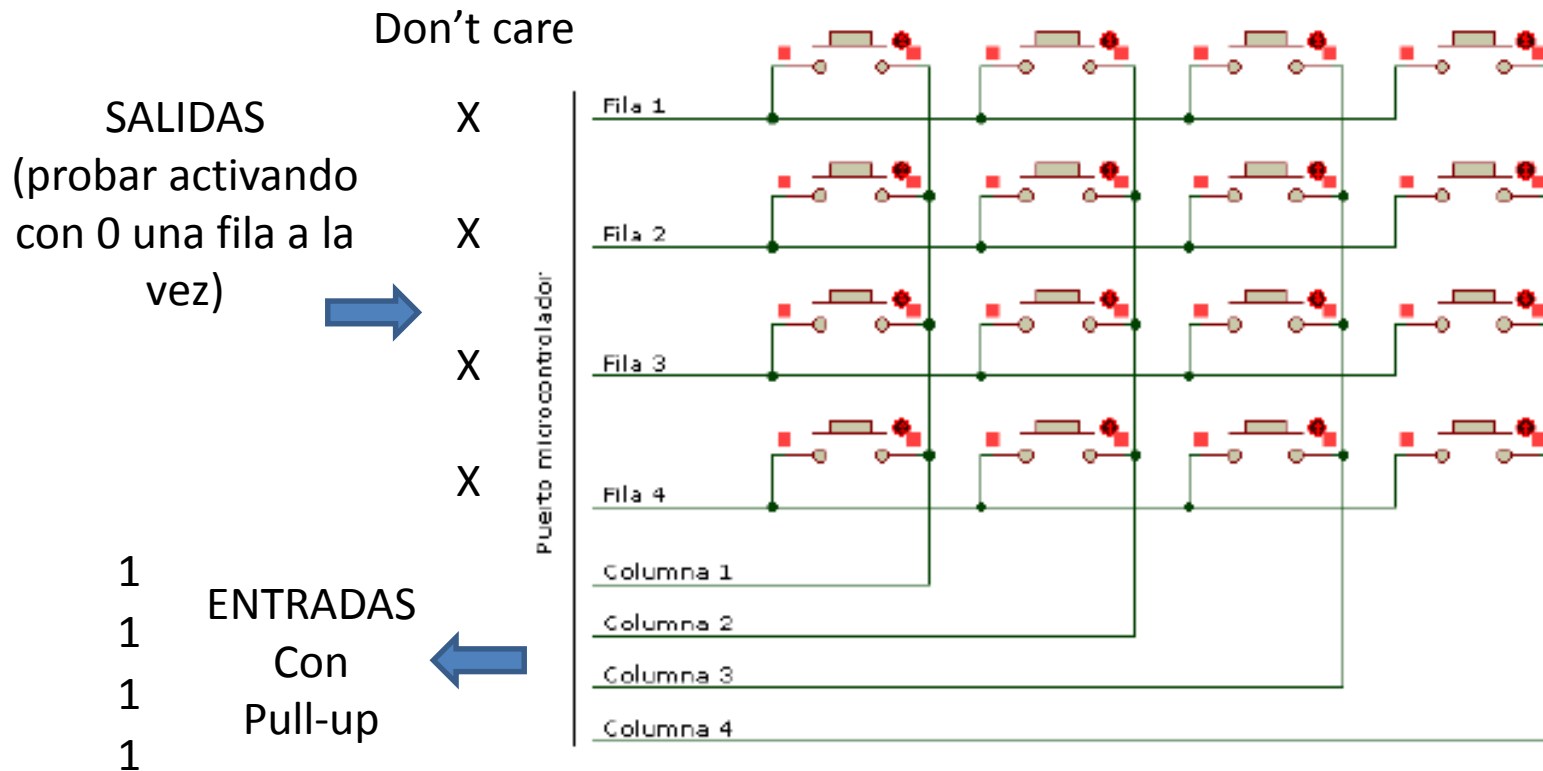
# Conexionado a un uC

Ejemplo con  
teclado de  
membrana



# Funcionamiento

- No hay tecla presionada



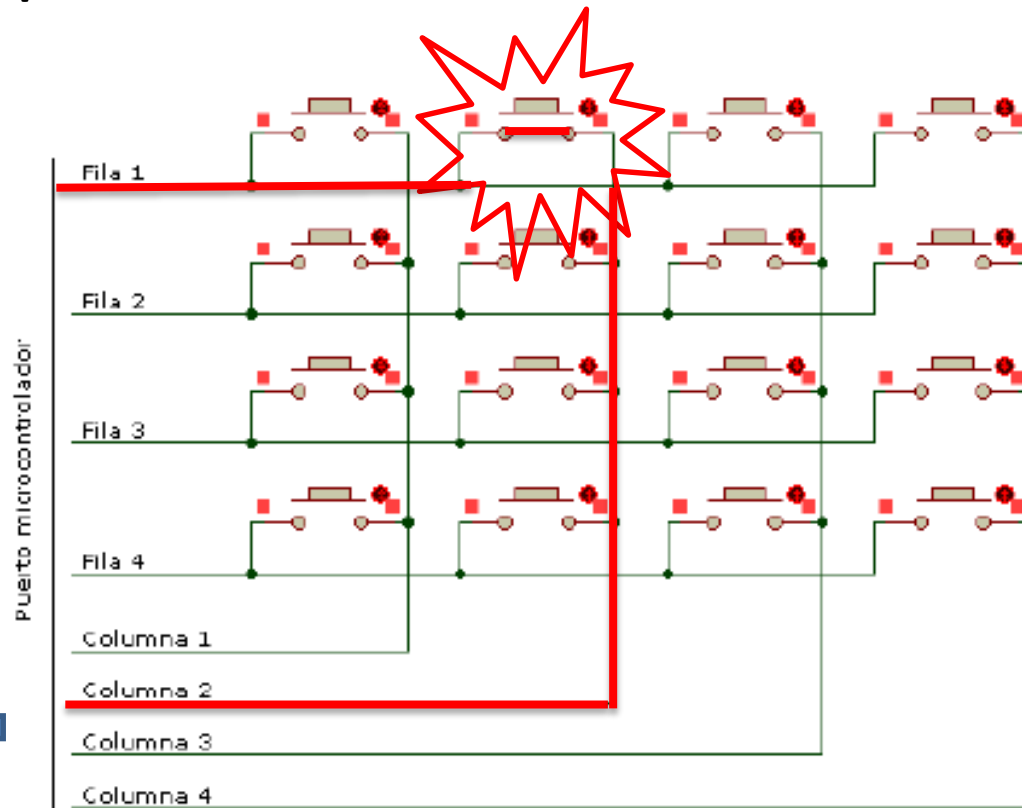
# Funcionamiento

- Si hay tecla presionada

SALIDA → 0  
(activa en 0)

Resto de las  
Filas permanecen  
como entradas Hi z

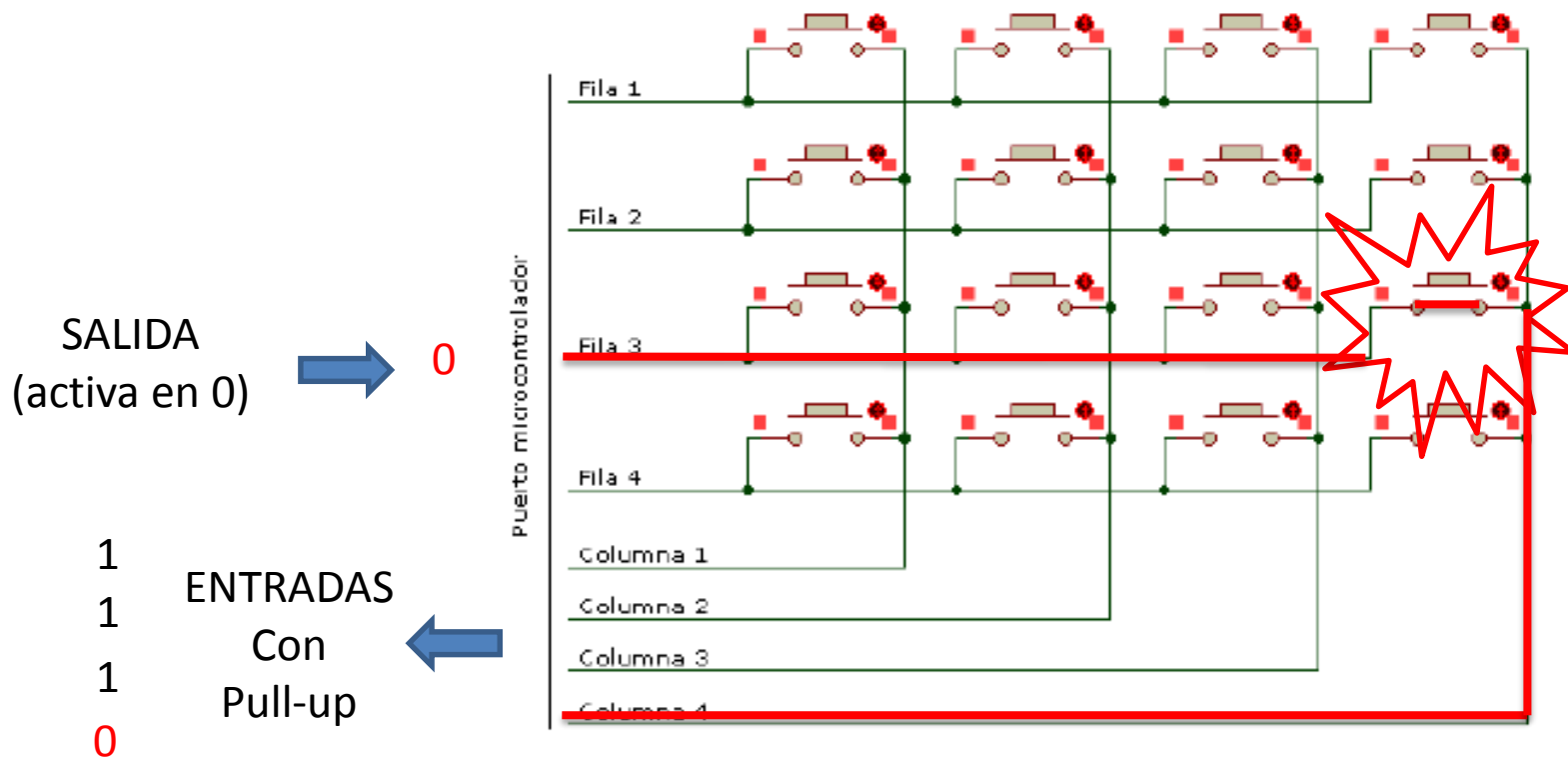
1  
0  
1  
1  
ENTRADAS  
Con  
Pull-up



En las Entradas del uC se lee 1011 cuando la combinación de salida es 0111

# Funcionamiento

- otra tecla presionada



En las Entradas del uC se lee 1110 cuando la combinación de salida es 1101

# Funcionamiento

Ejemplo: función para sondeo de teclado 4x4 (conectado a PORTB)

```
uint8_t KepadUpdate(void)
{
    uint8_t r,c;

    PORTB |= 0x0F;

    for(c=0;c<4;c++)
    {
        DDRB&=~(0xFF);

        DDRB|=(0x80>>c);

        for(r=0;r<4;r++)
        {
            if(!(PINB & (0x08>>r)))
            {
                return (r*4+c);
            }
        }
    }

    return 0xFF; //tecla No presionada
}
```

---



# TP2

- Ejercicio 1:
  - Función no bloqueante:
    - `uint8_t KEYPAD_Scan (uint8_t *key)`
    - Devuelve 0 si al barrer filas del teclado no encuentra tecla presionada
    - Devuelve 1 indicando que encontró una tecla presionada y la retorna por el parámetro `key`

# TP2

- Ejercicio 1:
  - ¿Cuántas veces x segundo debemos encuestar el teclado?
    - Depende de la aplicación pero supongamos que un “usuario” no va a presionar más de 1 tecla x segundo
      - ¿ Encuestamos entonces cada 500ms (2Hz) ?
      - Si queremos menor “tiempo de respuesta” encuestamos cada 100ms
  - ¿Qué sucede si el usuario deja presionado el botón 1 seg en cada pulsación?
    - Si encuestamos cada 100ms detectaremos 10 veces la misma tecla
  - ¿tenemos problemas con el efecto de rebote?
    - Si. Hay que implementar “software debouncing”

# Anti-rebote y detección múltiple

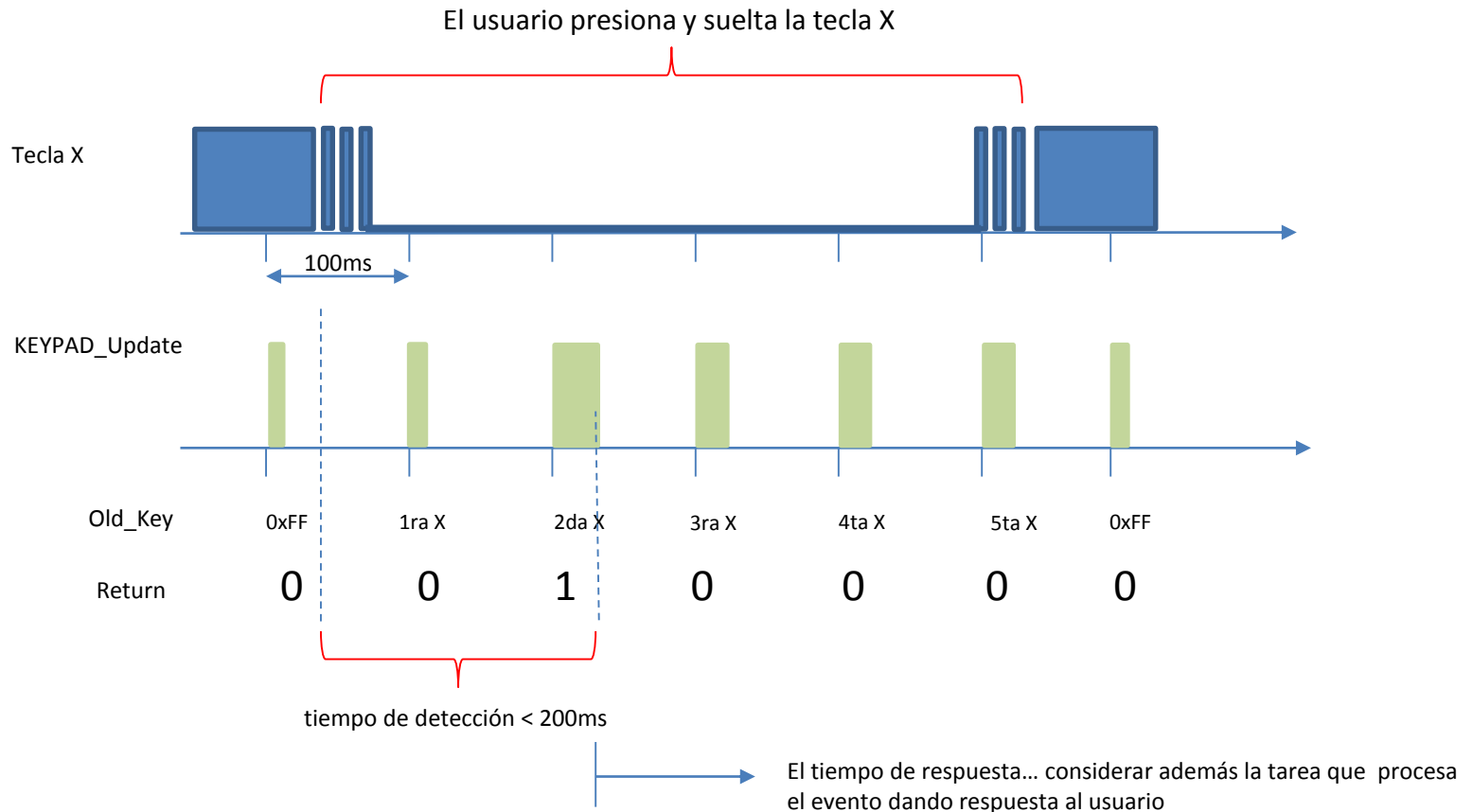
```
/******  
FUNCION PARA ESCANEAR UN TECLADO MATRICIAL Y DEVOLVER LA  
TECLA PRESIONADA UNA SOLA VEZ. TIENE DOBLE VERIFICACION Y  
MEMORIZA LA ULTIMA TECLA PRESIONADA  
DEVUELVE:  
0 -> NO HAYNUEVA TECLA PRESIONADA  
1 -> HAY NUEVA TECLA PRESIONADA Y ES *pkey  
*****/  
uint8_t KEYPAD_Update (uint8_t *pkey)  
{  
    static uint8_t Old_key;  
    uint8_t Key, Last_valid_key=0xFF; // no hay tecla presionada  
  
    if(!KEYPAD_scan(&Key)) {  
        Old_key=0xFF; // no hay tecla presionada  
        Last_valid_key=0xFF;  
        return 0;  
    }  
    if(Key==Old_key) { //2da verificación  
        if(Key!=Last_valid_key){ //evita múltiple detección  
            *pkey=Key;  
            Last_valid_key = Key;  
            return 1;  
        }  
    }  
    Old_key=Key; //1era verificación  
    return 0;  
}
```

Se debe encuestar periódicamente



# Anti-rebote y detección múltiple

- Análisis temporal



# Esquemático del KIT

