



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE COMPUTO

# Key

## Practica 10

Materia:

Arquitectura de computadoras

Profesor:

Castillo Cabrera Gelacio

Alumno:

Cortés Piña Oziel

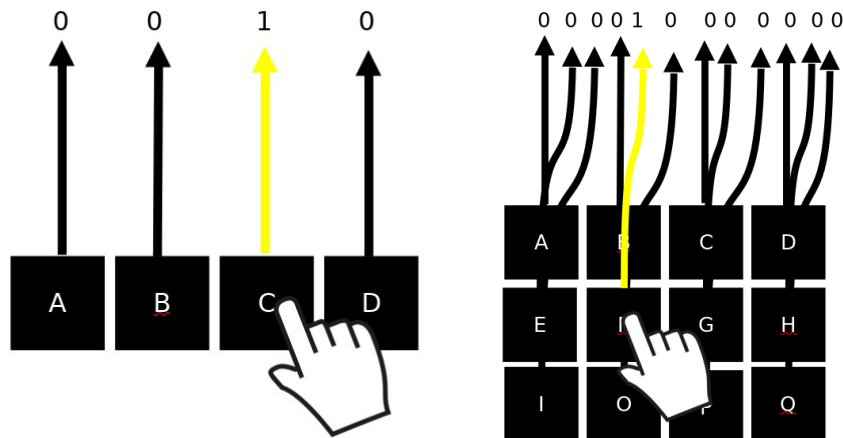
Grupo:

3CM12

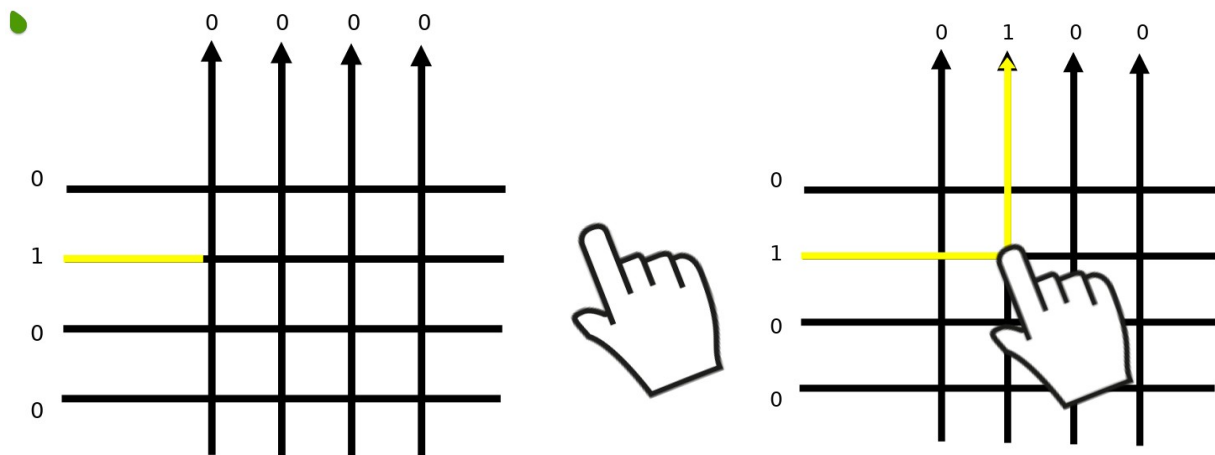


# Introducción

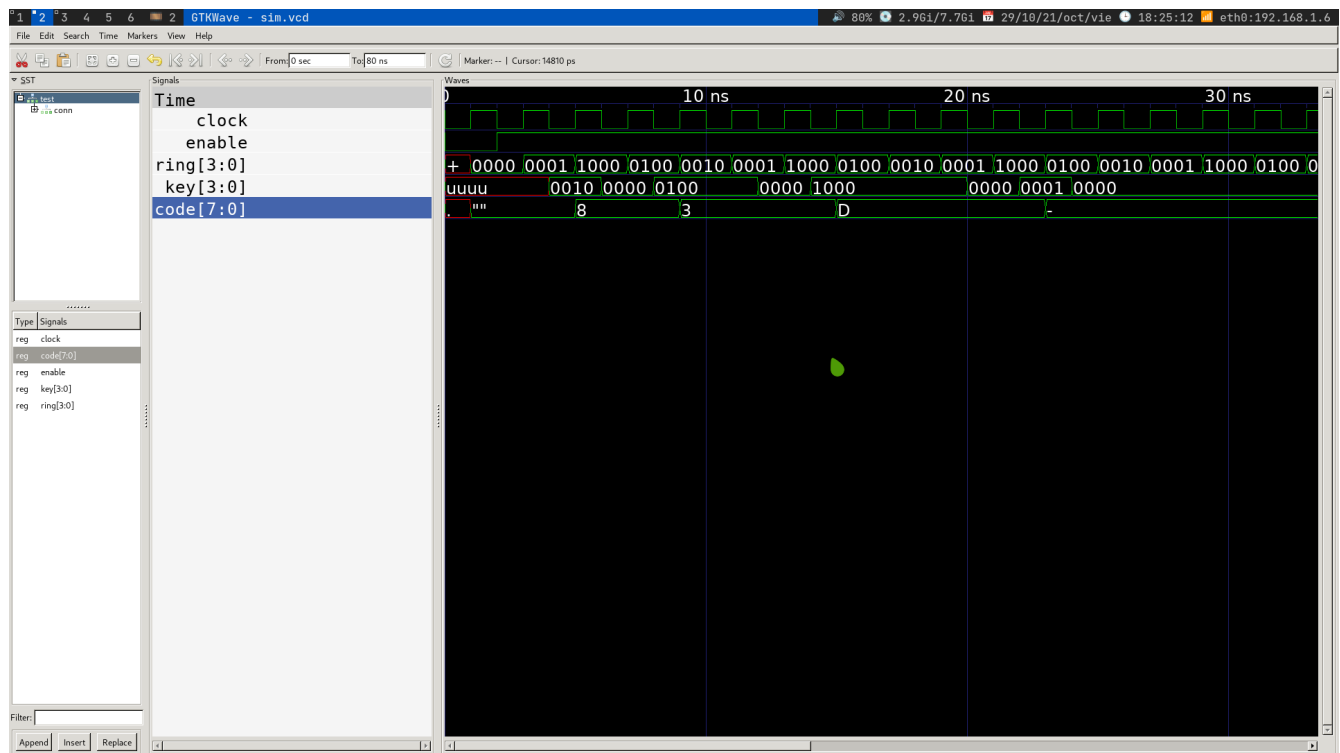
Para implementar un teclado dentro de nuestro arsenal podemos simplificarlo mucho y usar un vector de bits, en el cual cada bit representará un tecla que fue presionada por el usuario. Aunque simple con el paso del tiempo si requerimos de implementar más teclas puede volverse un serio problema este vector al necesitar de volverlo más grande y añadir más cables convirtiéndolo en un teclado fácil de fabricar pero horrible de administrar en circuitos.



La solución optima actual es la de implementar una matriz en que dos vectores de menor tamaño serán usados para la interpretación de la tecla que esta introduciendo el usuario. Para ello se implementan una serie de cables que atravesarán la matriz por uno de sus lados. Y otros cables que estarán conectados por otro de sus lados donde se implementará un contador de anillo. Este contador proveerá de energía a los botones una fila a la vez así pues será cuando el usuario presione uno de los botones del teclado que la señal llegará al segundo vector. Ahora dentro del circuito podremos hacer uso de un codificador que recibirá ambas señales y como resultado nos dará la letra o carácter del teclado.



# Simulación en GHDL y GTKWAVE



## Análisis de vectores

El anillo recorre las teclas, cuando estas son cero asignan un valor 0 a una variable para indicar que no esta pulsando nada. En caso contrario se da el carácter y se asigna un 1.

RING \ KEY	0000	0001	0010	0100	1000
0000					
0001		A	9	8	7
0010		B	6	5	4
0100		C	3	2	1
1000		D	*	+	-

## Conclusión

La manera en la que se esta implementado un teclado puede ser de tediosa codificación o diseño del hardware para la implementación pero es un ahorro de recursos al no tener que implementar un cable por cada tecla, lo que a grandes escalas como los teclados de escritorio son una gran ventaja. Al ser una codificación por hardware incluso puede llegar a ser una ventaja para el programador de alto nivel al leer el teclado del usuario.