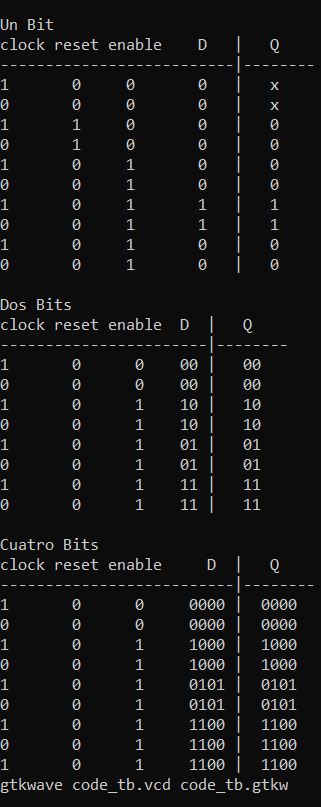
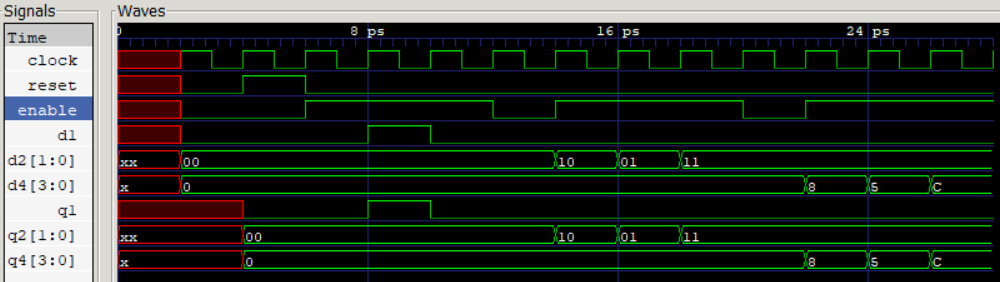
Laboratorio 9

Ejercicio 1:

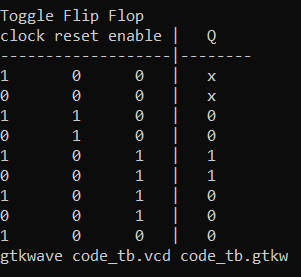
Para este ejercicio se creó un flip flop tipo D con el uso de un bloque always y condicionales. Además, haciendo uso de este primer módulo, se crearon módulos de flip flops tipo D de 2 y 4 bits. Para esto se crearon inputs y outputs de 2 y 4 bits y se instanciaron 2 y 4 flip flops respectivamente para asignarle valores a cada una de las salidas según cada una de las entradas. A estos módulos se les incluyó reset y enable.

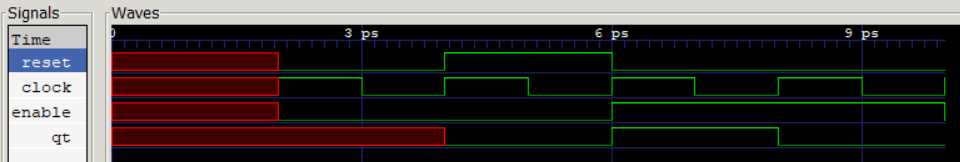




Ejercicio 2:

En este ejercicio se creó un flip flop tipo T o Toggle Flip Flop en base a un flip flop tipo D. Para esto se conectó el negado de la salida del flip flop a su entrada. Este módulo también contó con enable y reset.



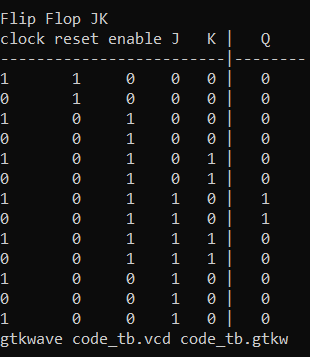


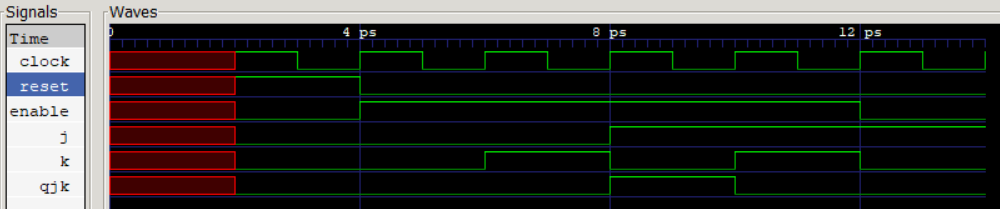
Graphical user interface, diagram, application

Description automatically generated

Ejercicio 3:

En este módulo se creó un Flip Flop JK con enable y reset. Este muestra un valor de Q previo cuando JK son 0, de 0 cuando J es 0 y K es 1, de 1 en el caso contrario y del negado de Q previo cuando ambos son 1.



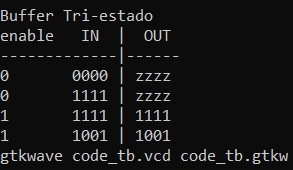


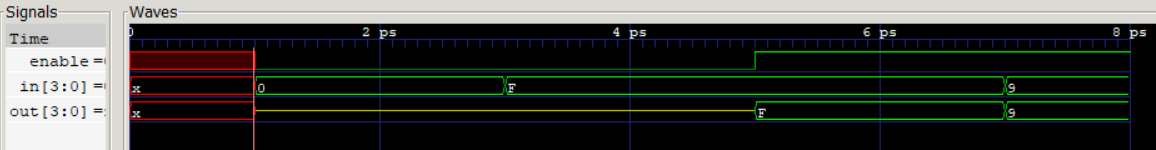
Diagram, schematic

Description automatically generated

Ejercicio 4:

Se realizó un Buffer Tri-estado en un módulo utilizando un operador ternario, el cual indica que siempre que el enable está apagado, el resultado sea alta impedancia.





Ejercicio 5:

Se creó una look up table en un módulo usando los bloques casex, el cual toma en cuenta los don’t cares de las tablas, lo cual fue útil para implementar las 128 posibilidades deseadas para crear todas las instrucciones de control para todos los opcodes posibles cuando se ancle al microcontrolador.

