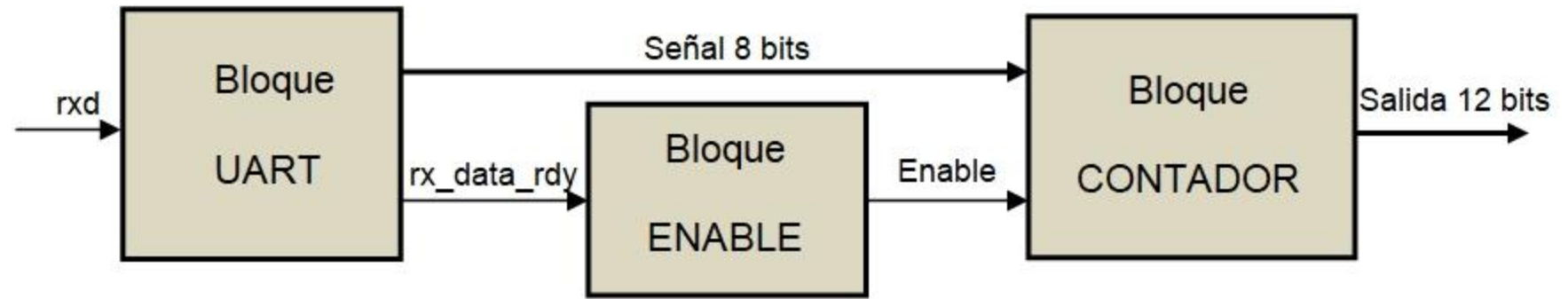


Contador Controlado por UART

Materia: Circuitos Lógicos Programables

Alumno: Roberto Federico Farfán
farfan.roberto.f@gmail.com

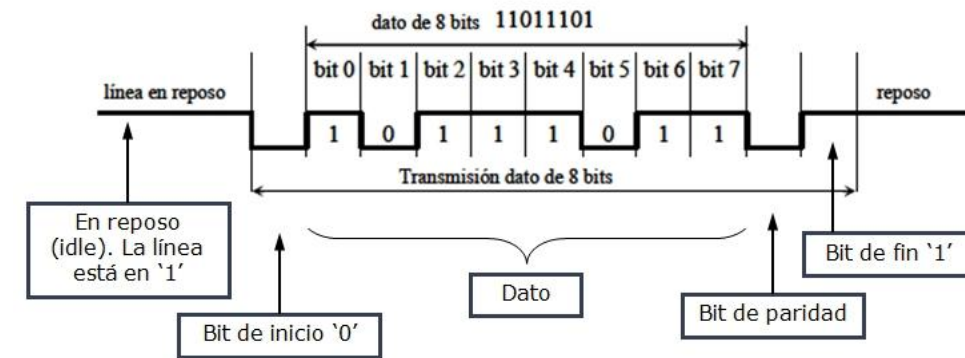
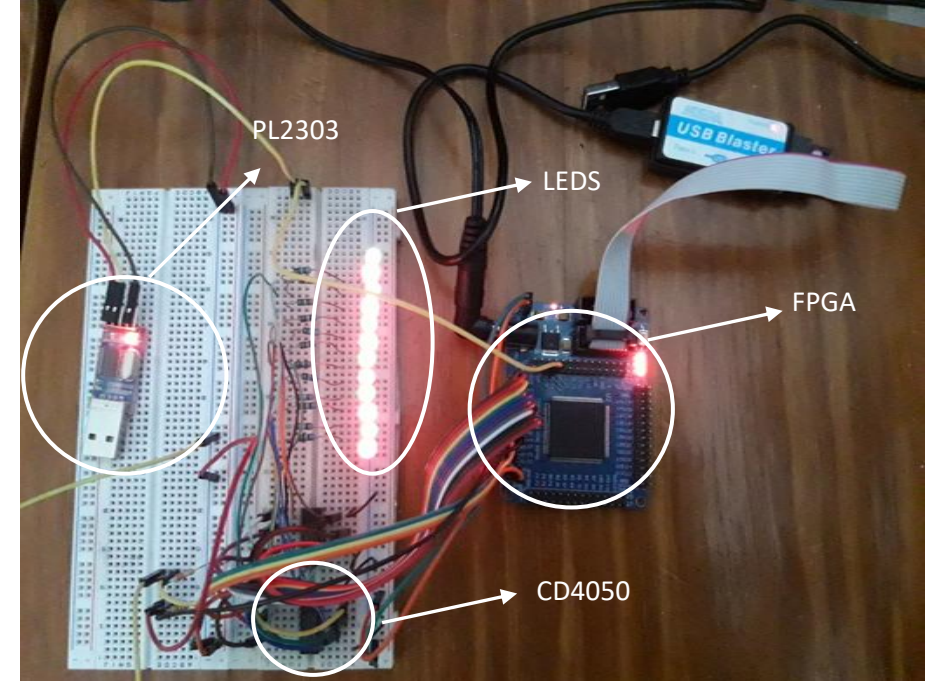
Objetivo.



- Implementar un contador de 12 bits controlado por UART.
- Se implementará en una FPGA Altera Cyclone II EP2C5T144, mediante código VHDL.
- El sistema digital se divide en bloques lógicos, los cuales se desarrollaron en las clases.
- En el trabajo se simulan los bloques en las distintas etapas del desarrollo, utilizando software como GHDL y GTKwave.

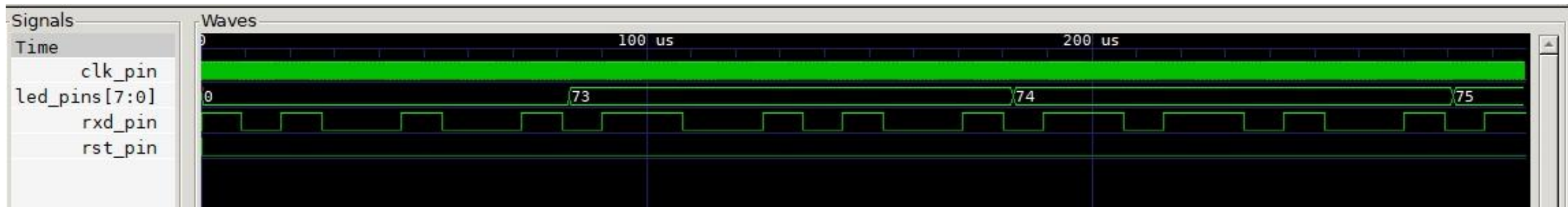
Introducción.

- Elementos: FPGA Altera Cyclone II, dos cd4050, un módulo PL2303, 12 resistencias y 12 diodos leds.
- El contador recibe caracteres:
- Al recibir el carácter I, incrementará una unidad en el contador.
- Al recibir el carácter D, decrementará una unidad en el contador.



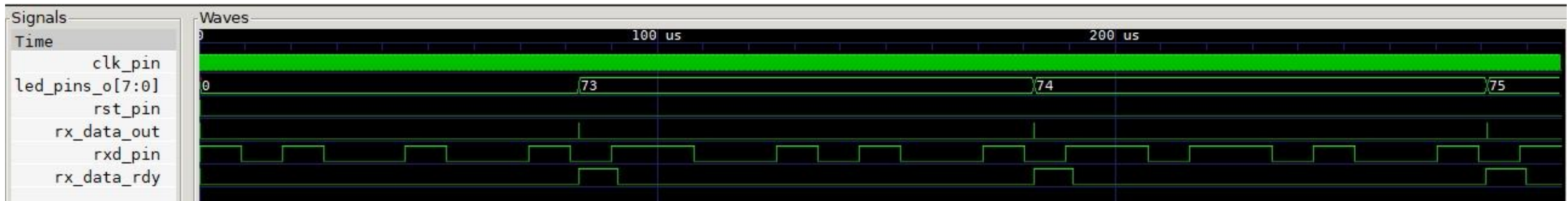
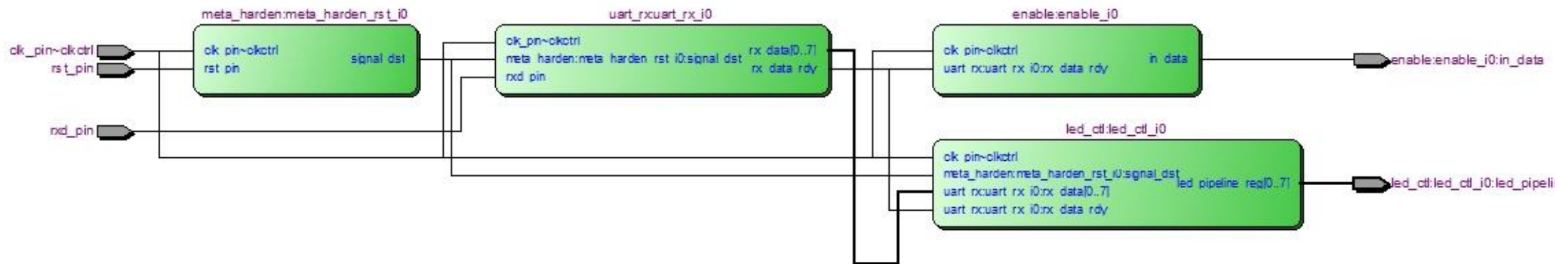
Bloque UART

- Se modificaron los archivos `uart_led.vhd` y `led_ctl.vhd`, que incorporaban como entrada la variable `btn_pin`.



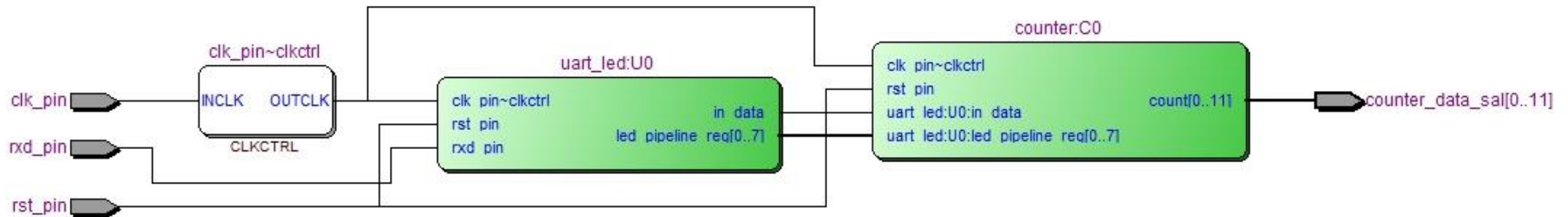
Bloque UART-Habilitación

- Del bloque UART se obtiene como salida la variable rx_data_rdy.
- Inicia en cero y se coloca a uno al tomar la lectura del bit 8 del carácter que ingresa.



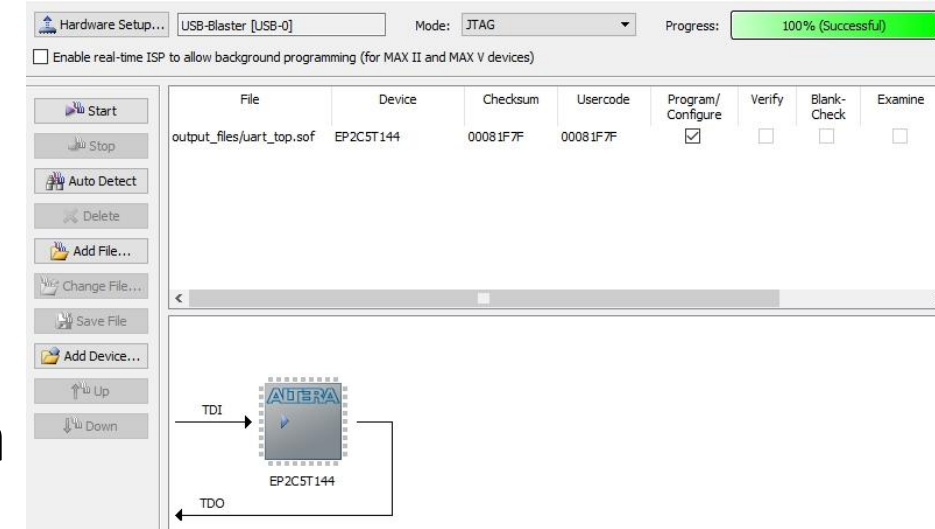
Bloque UART-Habilitación-Contador

- El contador recibe como entrada un pin **ret_pin**, el pin **clk**, los 8 bits de entrada y el *enable*, indicado como in data.



Implementación

- Se utilizó Quartus para depurar el programa y grabar el código.



Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength
in clk_pin	Input	PIN_17	1	B1_N0	PIN_17	3.3-V LV...default)		24mA (default)
out counter_data_sal[11]	Output	PIN_55	4	B4_N1	PIN_142	3.3-V LV...default)		24mA (default)
out counter_data_sal[10]	Output	PIN_53	4	B4_N1	PIN_141	3.3-V LV...default)		24mA (default)
out counter_data_sal[9]	Output	PIN_52	4	B4_N1	PIN_7	3.3-V LV...default)		24mA (default)
out counter_data_sal[8]	Output	PIN_51	4	B4_N1	PIN_144	3.3-V LV...default)		24mA (default)
out counter_data_sal[7]	Output	PIN_48	4	B4_N1	PIN_136	3.3-V LV...default)		24mA (default)
out counter_data_sal[6]	Output	PIN_47	4	B4_N1	PIN_3	3.3-V LV...default)		24mA (default)
out counter_data_sal[5]	Output	PIN_45	4	B4_N1	PIN_137	3.3-V LV...default)		24mA (default)
out counter_data_sal[4]	Output	PIN_44	4	B4_N1	PIN_8	3.3-V LV...default)		24mA (default)
out counter_data_sal[3]	Output	PIN_43	4	B4_N1	PIN_143	3.3-V LV...default)		24mA (default)
out counter_data_sal[2]	Output	PIN_42	4	B4_N1	PIN_135	3.3-V LV...default)		24mA (default)
out counter_data_sal[1]	Output	PIN_41	4	B4_N1	PIN_139	3.3-V LV...default)		24mA (default)
out counter_data_sal[0]	Output	PIN_40	4	B4_N1	PIN_9	3.3-V LV...default)		24mA (default)
in rst_pin	Input	PIN_63	4	B4_N0	PIN_18	3.3-V LV...default)		24mA (default)
in rxd_pin	Input	PIN_28	1	B1_N1	PIN_21	3.3-V LV...default)		24mA (default)

- Una vez desarrollada la interfaz, se utilizó la interfaz CoolTerm para enviar caracteres de la PC a la FPGA.
- Todas las pruebas fueron exitosas.

Gracias!