

INSTITUTO TECNOLÓGICO DE COSTA RICA
ÁREA ACADÉMICA DE INGENIERÍA MECATRÓNICA

MT 4002

Laboratorio de Electrónica Digital

Proyecto 3:

Transmisor de mensajes por código Morse

José Fabio Navarro Naranjo – 2019049626

Adrián Dittel Retana – 2019007945

Gabriel González Rodríguez – 2019057548

Emmanuel Naranjo Blanco – 2019053605

David Rodríguez Camacho – 2019024546

Profesora: Ana María Murillo Morgan

Semestre II – 2021

Contenido

I.	Descripción del proyecto	3
II.	Diagramas del Diseño Propuesto	4
i.	Primer nivel	4
ii.	Segundo nivel.....	5
iii.	Tercer nivel	6
III.	Desarrollo del Diseño	7
i.	Módulo Receptor	7
ii.	Módulo Traductor	7
iii.	Módulo Memoria.....	10
iv.	Módulo Divisor de frecuencia	11

I. Descripción del proyecto

El código Morse es un sistema de representación de letras y números mediante señales emitidas de manera intermitente. Así, cada letra y número se representa mediante una combinación de puntos y guiones, donde los puntos y los guiones, en realidad representan la duración de las señales en alto.

A	• —	U	• • —
B	— • • •	V	• • • —
C	— • — •	W	• — —
D	— • •	X	— • • —
E	•	Y	— • — —
F	• • — •	Z	— — • •
G	— — •		
H	• • • •		
I	• •		
J	• — — —		
K	— • —	1	• — — — —
L	• — • •	2	• • — — —
M	— —	3	• • • — —
N	— •	4	• • • • —
O	— — —	5	• • • • •
P	• — — •	6	— • • • •
Q	— — • • —	7	— — • • •
R	• — — •	8	— — — • •
S	• • •	9	— — — — •
T	—	0	— — — — —

Figura 1. Representación de letras y números mediante el código Morse internacional.

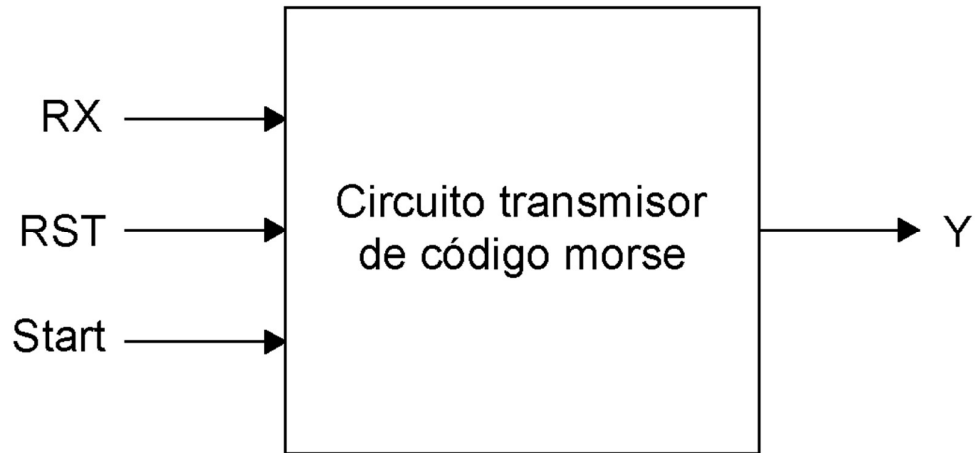
Reglas:

- Un punto tiene una duración de una unidad de tiempo.
- El guion tiene una duración de tres unidades de tiempo.
- El espacio entre señales de una misma letra tiene la misma longitud que un punto.
- El espacio entre letras es de tres unidades de tiempo.
- El espacio entre palabras es de 7 unidades de tiempo.

II. Diagramas del Diseño Propuesto

Se adjuntan los distintos diagramas propuestos para la solución del problema.

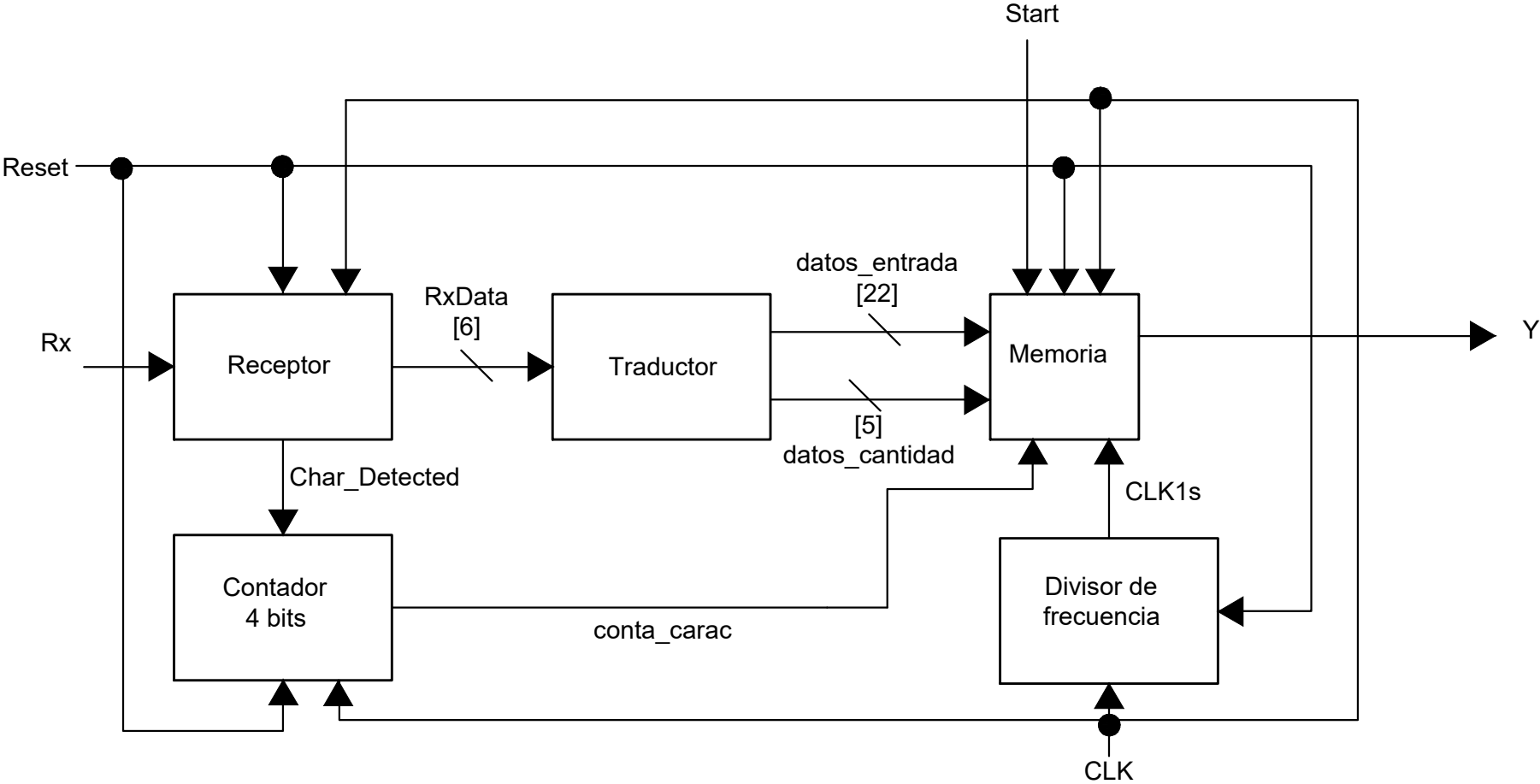
i. Primer nivel



ii. Segundo nivel

Se presenta el diagrama de segundo nivel del circuito propuesto.

PRODUCED BY AN AUTODESK STUDENT VERSION



PRODUCED BY AN AUTODESK STUDENT VERSION

TEC | Tecnológico
de Costa Rica

PROYECTO 3: TRANSMISOR CÓDIGO MORSE

DIVISOR DE FRECUENCIA:
DIAGRAMA DE SEGUNDO NIVEL

ÁREA ACADÉMICA DE INGENIERÍA MECATRÓNICA

CURSO: MT4002 - LAB. ELECTRÓNICA DIGITAL

INTEGRANTES:

ADRIÁN DITTEL RETANA

GABRIEL GONZÁLEZ RODRÍGUEZ

EMMANUEL NARANJO BLANCO

JOSÉ FABIO NAVARRO NARANJO

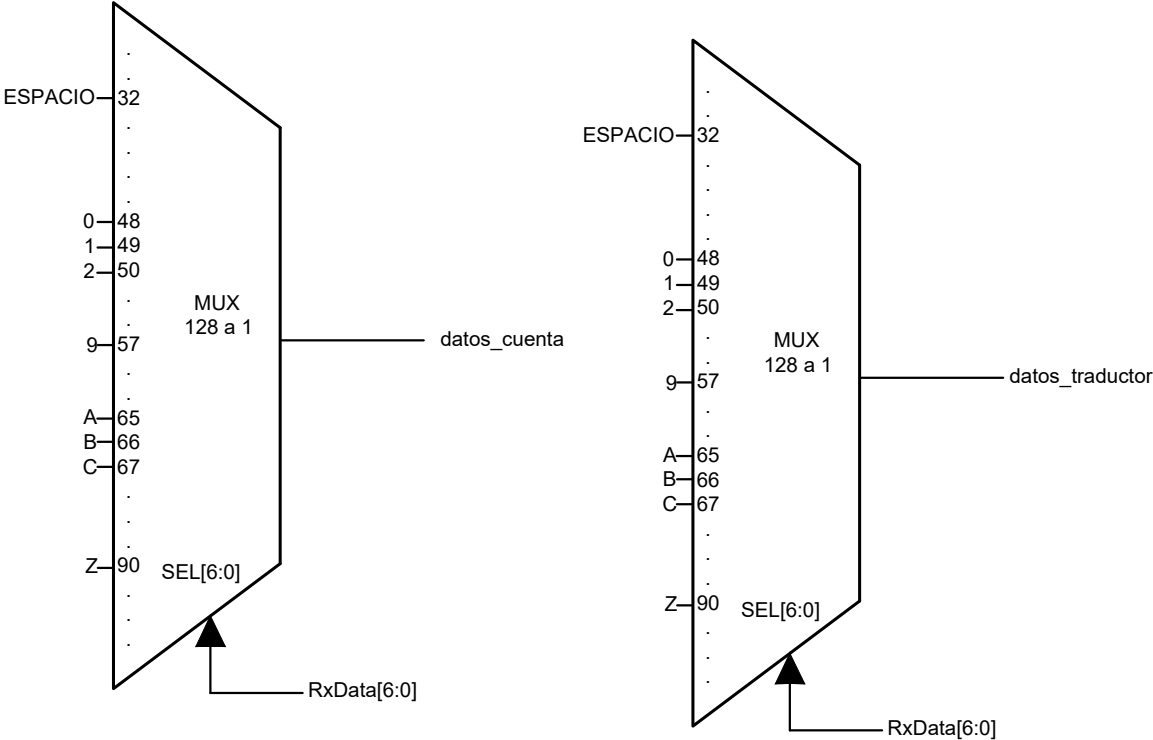
DAVID RODRÍGUEZ CAMACHO

LÁMINA

iii. Tercer nivel

Se presentan los diagramas de tercer nivel del circuito propuesto. Incluyen el diseño del módulo traductor, módulo de memoria y módulo de división de frecuencia.

PRODUCED BY AN AUTODESK STUDENT VERSION



PRODUCED BY AN AUTODESK STUDENT VERSION

TEC | Tecnológico
de Costa Rica

PROYECTO 3: TRANSMISOR CÓDIGO MORSE

TRADUCTOR:
DIAGRAMA DE TERCER NIVEL

ÁREA ACADÉMICA DE INGENIERÍA MECATRÓNICA

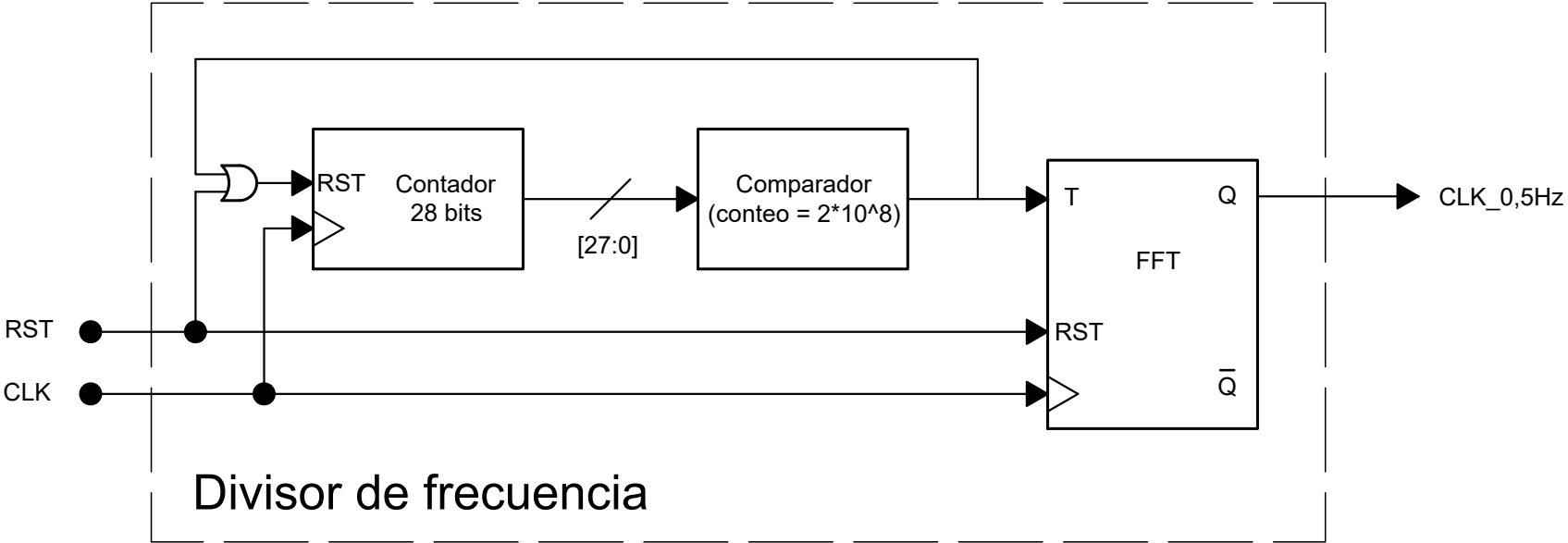
CURSO: MT4002 - LAB. ELECTRÓNICA DIGITAL

INTEGRANTES:

ADRIÁN DITTEL RETANA
GABRIEL GONZÁLEZ RODRÍGUEZ
EMMANUEL NARANJO BLANCO
JOSÉ FABIO NAVARRO NARANJO
DAVID RODRÍGUEZ CHAMACO

LÁMINA





TEC | Tecnológico
de Costa Rica

PROYECTO 3: TRANSMISOR CÓDIGO MORSE

DIVISOR DE FRECUENCIA:
DIAGRAMA DE SEGUNDO NIVEL

ÁREA ACADÉMICA DE INGENIERÍA MECATRÓNICA

CURSO: MT4002 - LAB. ELECTRÓNICA DIGITAL

INTEGRANTES:

ADRIÁN DITTEL RETANA

GABRIEL GONZÁLEZ RODRÍGUEZ

EMMANUEL NARANJO BLANCO

JOSÉ FABIO NAVARRO NARANJO

DAVID RODRÍGUEZ CHAMACO

LÁMINA

III. Desarrollo del Diseño

Se presenta el desarrollo conceptual de cada módulo, con sus respectivas bases teóricas.

i. Módulo Receptor

La transmisión de datos entre la computadora y la FPGA se lleva a cabo por medio del protocolo UART (comunicación serial), para lo cual se utilizó el módulo Receptor, que maneja el conjunto de datos a la entrada (en serie) a fin de transmitirlos a la salida de forma paralela. De esta forma a la salida del módulo Receptor se obtienen 10 bits en paralelo, correspondiendo el primero al bit de “start”, los ocho del medio al código ASCII del carácter presionado en el teclado, y el último al bit de “stop”, de la señal recibida.

Para este módulo se utilizó la teoría planteada en las técnicas de comunicación por medio del puerto serial, y se tomó como referencia el código de Verilog del siguiente proyecto: [UART Communication on Basys 3, FPGA Dev Board Powered by Xilinx Artix 7 Part II : 10 Steps - Instructables](#) (alexwonglik, 2021).

Adicional a esto se agregó otra salida llamada “charDetected” que envía una señal en alto al módulo Contador de Caracteres cada vez que ingresa un carácter nuevo.

ii. Módulo Traductor

El módulo traductor recibe su nombre debido a su finalidad, la cual consiste en el primer paso hacia la traducción del dato recibido en código ASCII a su correspondiente valor en código morse. En este módulo se tiene como entrada los 7 bits en paralelo del dato ingresado, sea número o letra, codificado en ASCII estándar proveniente de la comunicación serial tratada en el módulo anterior, y se presentan dos salidas que partirán hacia el siguiente módulo del proyecto.

Para estas salidas, una se trata de una codificación diseñada a conveniencia para cumplir con las restricciones del problema y otra indica el tamaño de palabra del carácter específico que se requiere convertir a código morse.

Para esto, se escogió el uso de dos multiplexores, uno para cada salida, que trabajan como una especie de memoria que según el carácter escogido, la salida contiene un conjunto de bits propio. De este modo, en el selector de ambos Mux se trata de los 7 bits codificados en ASCII, que seleccionará una de las 128 entradas disponibles, de las cuales se utilizan únicamente 37, los 10 números decimales, las 26 letras del alfabeto y 1 destinada para el espacio entre palabras.

Como se observa en la Figura 1, cada elemento tiene su propia traducción en morse, donde se consideró que cada punto corresponde a un uno (1), cada espacio en un mismo caracter es un cero (0), cada raya son tres unos (111), y se colocan tres ceros (000) para representar los espacios entre caracter. Además, se empleó una entrada destinada para el espacio entre palabras, que son cuatro ceros (0000), que al complementarse con los tres 0, indicaría un espacio entre palabra de 7 ceros. Esto se tomando en cuenta el código morse de la Figura de derecha a izquierda.

Por otro lado, ya que se trabaja con palabras de tamaño constante, se escogió trabajar con 22 bits de longitud ya que el caracter más grande requiere esta cantidad de bits. No obstante, dado que no todas las letras, ni números requieren el uso de todos los bits, estos espacios se completaron con 0 y se decidió implementar un segundo multiplexor que, con el mismo selector, indique el tamaño de palabra del caracter. Esto para que en el siguiente módulo se tenga como entrada la codificación diseñada y su cantidad.

Se presenta en la Tabla 1 en resumen el uso de los Mux, que indica para el seleccionador en código ASCII, su salida de cada Mux, además, se presentó el diagrama de este módulo traductor en la sección de diagramas. Por último, se entiende que las demás entradas que no se utilizan se colocan en 0 o tierra.

Tabla 1. Uso de multiplexores para la etapa del módulo traductor según el selector.

SEL en ASCII	Salida codificada	Cantidad iniciando en 0	SEL en ASCII	Salida codificada	Cantidad iniciando en 0
ESPACIO	22'b0	3	A	22'b0000000000000000011101;	7
			B	22'b0000000000000010101011;	11
0	22'b0001110111011101110111;	21	C	22'b0000000000001011101011;	13
1	22'b0000011101110111011101;	19	D	22'b0000000000000000101011;	9
2	22'b0000000111011101110101;	17	E	22'b000000000000000000000001;	3
3	22'b0000000001110111010101;	15	F	22'b0000000000000000101110101;	11
4	22'b0000000000011101010101;	13	G	22'b0000000000000000101110111;	11
5	22'b00000000000000101010101;	11	H	22'b00000000000000001010101;	9
6	22'b00000000000010101010111;	13	I	22'b000000000000000000000101;	5
7	22'b0000000001010101110111;	15	J	22'b0000000001110111011101;	15
8	22'b0000000101011101110111;	17	K	22'b000000000000000111010111;	11
9	22'b0000010111011101110111;	19	L	22'b000000000000000101011101;	11
			M	22'b00000000000000001110111;	9
			N	22'b0000000000000000010111;	7
			O	22'b0000000000011101110111;	13
			P	22'b00000000000010111011101;	13
			Q	22'b0000000001110101110111;	13
			R	22'b00000000000000001011101;	9
			S	22'b0000000000000000010101;	7
			T	22'b000000000000000000000111;	5
			U	22'b00000000000000001110101;	9
			V	22'b000000000000000111010101;	11
			W	22'b000000000000000111011101;	11
			X	22'b0000000000011101010111;	13
			Y	22'b0000000001110111010111;	15
			Z	22'b0000000000010101110111;	13

iii. Módulo Memoria

Este módulo presenta varias etapas interconectadas por registros en paralelo, multiplexores, contadores y un comparador, un decodificador y un FF-JK, que en conjunto conforman la sección del proyecto denominada memoria. Al tener en cuenta el modelo presentado en la sección de diagramas, se inicia con una etapa de entrada y guardado de datos, y se finaliza con una sección de salida de datos que se dirigen hacia el LED final del circuito.

Primeramente, se reciben las dos entradas provenientes del módulo traductor y la salida charDetected derivada del módulo receptor de caracteres, las cuales son de 22, 5 y 1 bit respectivamente. La lógica de entrada y salida están interconectadas por doce registros en paralelo que pueden almacenar hasta 27 bits (tomando en cuenta ambas salidas del traductor), cuyo funcionamiento está regido por un decodificador de 4 a 16.

Las entradas de este dispositivo corresponden al resultado de un circuito contador cuyo conteo está relacionado con la señal charDetected proveniente a raíz de la comunicación serial cada vez que llega un nuevo caracter. De este modo, el decodificador distribuirá los datos provenientes del traductor a lo largo de los 12 registros, limitando así la cantidad de elementos que pueden ingresar al sistema.

En cuanto a la salida, se dispone de los datos de cada registro paralelo en las entradas de dos multiplexores, uno de 22 bits para la información y otro de 5 bits para la longitud de palabra. Para ambos, se emplea un selector que corresponde al resultado del conteo de ciclos de reloj una vez activada la señal de start. Esta señal se modela mediante un FF-JK y la conexión estratégica de una compuerta OR para el control de activación del circuito.

Posteriormente, la salida del Mux del mensaje se alambran a las entradas de otro Mux, cuyo selector depende de otro contador que se activa de igual forma con el start, este cuenta desde 0 hasta que la salida del comparador indique que se alcanzó la longitud del carácter que se está transmitiendo. Luego, la señal de este comparador activará una serie de resets y permitirá que el contador que controla ambos Mux de información y cantidad, pase a su estado siguiente, repitiendo el proceso las veces que sea necesario.

Por último, cabe destacar que este módulo trabaja con un clock de 1 s.

iv. Módulo Divisor de frecuencia

Este módulo se encarga de dividir la frecuencia del CLK principal al que opera la FPGA, a una frecuencia que el ojo humano pueda detectar con facilidad para así leer el código morse en el LED de la salida del circuito. Esto permite definir cuántos ciclos de reloj equivale la mínima unidad de tiempo del transmisor solicitadas en las reglas del proyecto.

Este se compone de un contador, un comparador y un Flip Flop T, como se presentó en la sección de Diagramas, encargados de dividir la frecuencia de 100 MHz del CLK global a una señal con periodo de 2s o 0.5 Hz. Para esto se utilizó un contador de 28 bits y un comparador de 28 bits, lo cual brindaría una salida con periodos de tiempo en alto de 1 s.