



Instituto Tecnológico de Costa Rica

Escuela de Ingeniería Electrónica

Taller de diseño digital (EL3313)

Prof. Kaleb Alfaro Bonilla, M.Sc.

Estudiantes:

2021155576 - Bolaños Solís Oscar

2022207101 - Bou Espinoza Raquel

2022195304 - Lobo Campos Andrés

2021021770 - Sánchez Herrera Angélica

Lab. 1: Cuestionario Previo

I Semestre 2026

Cuestionario Previo

1. Investigue las características de las familias TTL bajo las series 74*xx en particular las diferencias entre las variantes * ∈ {L, LS y HC}.

Los circuitos TTL (Transistor-Transistor Logic) son una familia lógica implementada con transistores bipolares (BJT), donde el transistor cumple funciones tanto de conmutación lógica como de amplificación de corriente (1). Dentro de la familia TTL existen varias subfamilias, entre ellas: Fast TTL, Schottky TTL, High-Power TTL, Low-Power TTL y Advanced Schottky TTL.

Las principales características de las familias TTL son el fan-out, la disipación de potencia, el retardo de propagación y el margen de ruido, parámetros que determinan el desempeño eléctrico y temporal de las puertas lógicas [1].

Las diferencias entre las variantes L, LS y HC son las siguiente [2] s:

Categoría	Serie 74HC	Serie 74LS	Serie 74L
Tecnología	CMOS de alta velocidad con transistores NMOS y PMOS.	TTL Schottky de bajo consumo con transistores Schottky.	TTL de bajo consumo usando transistores bipolares (sin Schottky).
Consumo de potencia	Generalmente muy bajo, adecuado para dispositivos alimentados por batería.	Consume más que 74HC, pero menos que la TTL estándar.	Menor que la TTL estándar (74xx), pero mayor que 74LS y mucho mayor que 74HC.
Velocidad	CMOS de alta velocidad, a menudo comparable o superior a 74LS.	Conocida por su conmutación rápida dentro de la familia TTL.	Más lenta que 74LS; prioriza bajo consumo a costa de velocidad.
Niveles de voltaje	Compatible con TTL y CMOS, rango de voltaje más amplio (2–6 V).	Diseñada para niveles de voltaje TTL (5 V).	Niveles de voltaje TTL (5 V).

Categoría	Serie 74HC	Serie 74LS	Serie 74L
Inmunidad al ruido	Generalmente mejor inmunidad al ruido.	Inmunidad al ruido moderada.	Menor inmunidad al ruido que 74HC y similar o inferior a 74LS.
Rango de temperatura	Normalmente mayor rango de operación.	Rangos comerciales/industriales típicos de TTL.	Similar a TTL estándar, generalmente más limitado que CMOS.
Información adicional	Velocidades de transición similares. 74HC es un buen reemplazo de 74LS. 74HCT es más compatible con los umbrales de entrada TTL.	Las entradas flotantes tienden a HIGH en TTL. Más rápida que 74L.	Familia TTL temprana de bajo consumo. Hoy casi en desuso, reemplazada por 74LS y 74HC.

2. Investigue las características de la familia CMOS 4000.

R/ La familia CMOS 4000 es una de las primeras series comerciales basadas en tecnología CMOS. Fue diseñada para ofrecer bajo consumo y amplio rango de operación en comparación con las familias TTL. Está basada en transistores MOSFET complementarios como los PMOS en la red pull-up y NMOS en la red pull-down. A diferencia de TTL que está basada en BJT, en el caso de CMOS idealmente no hay corriente estática cuando la salida está en estado estable. La corriente se consume principalmente durante los cambios de estado. Una de sus principales ventajas es su amplio rango de voltaje donde VDD típicamente está entre 3V y 15V, y VSS usualmente conectado a tierra. Los niveles lógicos son proporcionales al voltaje de alimentación. Esto implica que los márgenes de ruido aumentan con el voltaje de alimentación. Su consumo de potencia estática es extremadamente bajo. Tiene tiempos de propagación típicos de 50 ns a 200 ns, lo que la hace adecuada para aplicaciones de baja velocidad. La impedancia de entrada es muy alta y la corriente de entrada es despreciable. Sus aplicaciones típicas son en sistemas de baja frecuencia, osciladores RC, temporizadores y generadores de señales.[3]

3. Investigue que cuidados deben tenerse al manipular las tecnologías CMOS.

Se debe tener cuidado con la manipulación del CMOS ya que al tener una impedancia muy alta se expone daños por descargas electroestáticas, por ejemplo, fácilmente al tocar la terminal de entrada con el dedo. [4]

Por lo que se recomienda:

- Antes de manipular un CMOS tocar el gabinete metálico o chasis de un instrumento conectado u otra fuente de tierra.
- Antes de pasarle un CMOS a un compañero, especialmente en un día seco o de invierno, tocarlo primero antes de entregárselo.

4. Investigue el significado de los parámetros VIL, VIH, VOL, VOH, IIK, IOK

- VIL: Voltaje de entrada bajo, representa el valor de voltaje de entrada correspondiente al valor mínimo de la señal (0).
 - VIH: Voltaje de entrada alto, representa el valor de voltaje de entrada correspondiente al valor máximo de la señal (1).
 - VOL: Voltaje de salida bajo, representa el valor de voltaje de salida correspondiente al valor mínimo de la señal (0).
 - VOH: Voltaje de salida alto, representa el valor de voltaje de salida correspondiente al valor máximo de la señal (1).
 - IIK: Corriente de entrada, se refiere a cuanta corriente se ocupa para la operación del circuito.
 - IOK: Corriente de salida, se refiere a la corriente que puede ser entregada por el circuito.
- [5]

5. Investigue qué son los tiempos de propagación t_{PD} , t_{PLH} y t_{PHL} y los tiempos de transición t_t , t_r y t_f .

El **tiempo de propagación** es el intervalo que transcurre entre una transición en la entrada de una puerta lógica y el inicio de la transición correspondiente en la salida. Se definen dos tiempos de propagación fundamentales:

- t_{PLH} : mide el tiempo transcurrido desde una transición en la entrada hasta que la salida comienza a cambiar del nivel lógico bajo al nivel lógico alto.
- t_{PHL} : mide el tiempo transcurrido desde una transición en la entrada hasta que la salida comienza a cambiar del nivel lógico alto al nivel lógico bajo.

El **tiempo de propagación promedio** t_{PD} se define como el promedio aritmético entre t_{PLH} y t_{PHL} [6].

Los **tiempos de transición** describen la duración del cambio de estado de la señal de salida:

- t_r (rise time): tiempo necesario para que la señal pase del nivel lógico bajo al nivel lógico alto.
- t_f (fall time): tiempo necesario para que la señal pase del nivel lógico alto al nivel lógico bajo.
- t_t : intervalo total durante el cual ocurre la transición de la señal.

6. Investigue qué significa el termino fan-out y cuáles valores típicos se encuentran en las familias TTL y CMOS.

El fan-out es el número máximo de entradas lógicas estándar que puede manejar una salida sin sobrepasar las especificaciones eléctricas de niveles de voltaje y corriente. La mayoría de las puertas lógicas transistor-a-transistor TTL admiten hasta 10 puertas o dispositivos digitales adicionales. Una TTL típica tiene una distribución de 10. Las puertas CMOS admiten más de 50 dispositivos conectados, lo que les otorga una tasa de distribución superior a 50. [7]

7. Para cada una de las variantes TTL y CMOS especifique en una tabla:

- a) rango de tensión eléctrica de alimentación VCC o VDD, VSS
- b) rango de tensiones de entrada y salida
- c) tiempos de propagación y transición

	TTL 74L	TTL 74LS	TTL 74HC	CMOS 4000
rango de tensión eléctrica de alimentación	4.75 V – 5.25 V	4.75 V – 5.25 V	2 V a 6 V	3 V – 15 V
rango de tensiones de entrada	Bajo: 0 V – 0.7 V Indefinido: 0.7 V – 2.0 V Alto: 2.0 V – 5.25 V	Bajo: 0 V – 0.8 V Indefinido: 0.8 V – 2.0 V Alto: 2.0 V – 5.25 V	2 V: Bajo: 0 - 0.5 V Indefinido: 0.5-1.5 V Alto: 1.5-2 V 4.5 V: Bajo: 0 -1.35 V Indefinido 1.35-3.15 V Alto: 3.15-4.5 V 6 V: Bajo: 0 a 1.8 V Indefinido: 1.8-4.2 V Alto: 4.2-6 V	VDD = 5 V: Bajo 0 – 1.5 V Alto 3.5 – 5 V VDD = 10 V: 0 – 3 V 7 – 10 V VDD = 15 V: 0 – 4 V 11 – 15 V
rango de tensiones de salida	Bajo: 0 V – 0.3 V alto: 2.4 V – 5.25	Bajo: 0 V – 0.5 V Alto: 2.7 V – 5.25 V	bajo: 0-0.1 V alto: 2 V) 1.9 V-2V 4.5 V) 4.4 V-4.5 V 6 V) 5.9 V-6 V	VDD = 5 V: Bajo 0 – 0.05 V Alto: 4.95 – 5 V VDD = 10 V: Bajo 0 – 0.05 V Alto 9.95 – 10 V VDD = 15 V: Bajo 0 – 0.05 V Alto 14.95 – 15 V
tiempos de propagación	31 ns – 60 ns t _{phl} 35 ns – 60 ns t _{plh}	15 ns t _{plh} 15 ns t _{phl}	2 V: 90 ns – 115 ns 4.5 V: 18 ns – 23 ns 6 V: 15 ns – 20 ns	5 V: 60 – 200 ns 10 V: 30 – 100 ns 15 V: 20 – 60 ns
tiempos de transición		15 ns rise and fall time	2 V: 75 ns – 95 ns 4.5 V: 15 ns – 19 ns 6 V: 13 ns – 16 ns	5 V: 50 – 150 ns 10 V: 30 – 80 ns 15 V: 20 – 60 ns

De [8], [9], [10], [11].

8. Revise la hoja de datos de los circuitos integrados 74*00,74*02,74*04,74*14, 4001, 4011, 4069 y 40106. Resuma para qué sirve cada uno.

- 74*00

Compuerta NAND (AND negada) positiva de 2 entradas, familia TTL, con 4 compuertas por circuito integrado. [12]

- 74*02

Compuerta NOR (OR negada) de 2 entradas, familia TTL, con 4 compuertas por circuito integrado. [13]

- 74*04

Compuerta NOT (inversor) de alta velocidad, familia TTL, con 6 compuertas por circuito integrado. [14]

- 74*14

Compuerta NOT (inversor) de Schmitt trigger, familia TTL, con 6 compuertas por circuito integrado; gracias a el efecto Schmitt, los inversores tienen distintos puntos de disparo lo que ayuda a prevenir ruido y también pueden detectar pendientes muy lentas. [15]

- 4001

Compuerta NOR (OR negada) de 2 entradas, tecnología CMOS, con 4 compuertas por circuito integrado. [16]

- 4011

Compuerta NAND (AND negada) de 2 entradas, tecnología CMOS, con 4 compuertas por circuito integrado. [17]

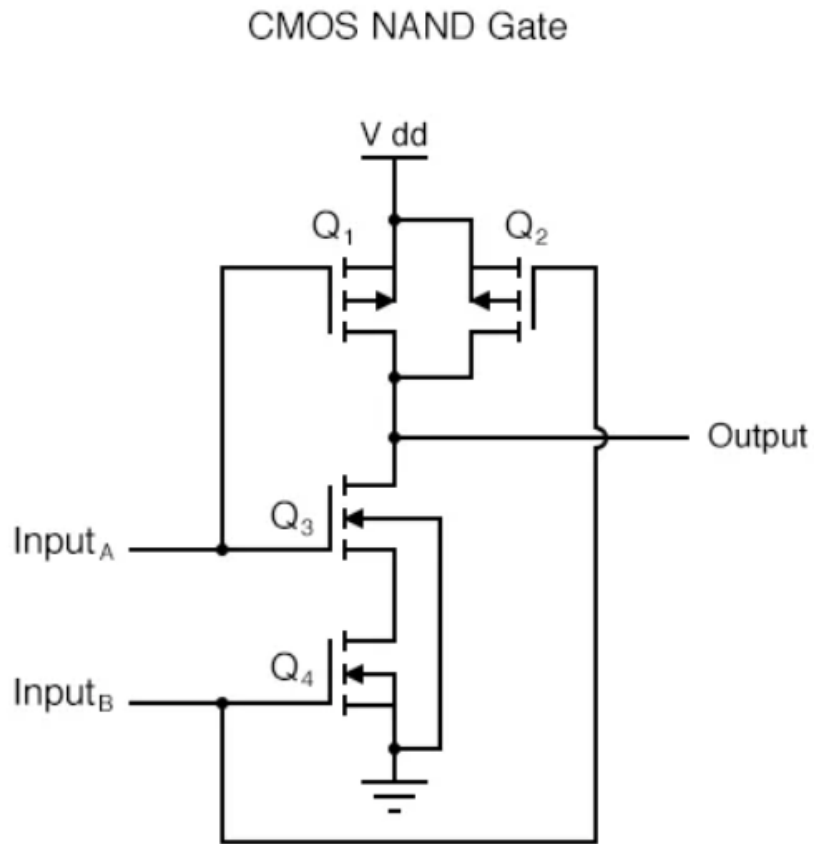
- 4069

Compuerta NOT (inversor) de uso general, tecnología CMOS, con 6 compuertas por circuito integrado. [18]

- 40106

Compuerta NOT (inversor) de Schmitt trigger, tecnología CMOS, con 6 compuertas por circuito integrado; gracias a el efecto Schmitt, los inversores tienen distintos puntos de disparo lo que ayuda a prevenir ruido y también pueden detectar pendientes muy lentas. [19]

9. Revise la estructura básica, a nivel de transistores, de una compuerta NAND en circuitos integrados CMOS.



Una compuerta NAND implementada en tecnología CMOS está compuesta por cuatro transistores MOSFET: dos PMOS conectados en paralelo formando la red de pull-up, y dos NMOS conectados en serie formando la red de pull-down [20].

A	B	PMOS	NMOS	Salida
0	0	ON	OFF	1
0	1	ON	OFF	1
1	0	ON	OFF	1
1	1	OFF	ON	0

10. Investigue sobre el concepto y el uso de los de circuitos pull-up y pull-down en electrónica digital.

R/ Un circuito pull-up conecta un nodo a VCC mediante una resistencia. Un circuito pull-down conecta un nodo a GND mediante una resistencia. Su propósito es definir un estado lógico cuando ningún dispositivo está forzando activamente el nodo. Las entradas CMOS tienen alta impedancia, entonces si se dejan flotantes pueden captar ruido y oscilar aleatoriamente entre estados. Un pull-up o pull-down fija un valor por defecto. Un valor típico de resistencia pull-up es de 4,7 kΩ, pero puede variar según la aplicación. Sus aplicaciones comunes son la interfaz de dispositivos, microcontroladores y convertidores A/D.[21]

11. Investigue que es un circuito disparador Schmitt (Schmitt trigger). Revise las características

técnicas del circuito 74*14.

Un Schmitt Trigger es un circuito comparador con histéresis, es decir, posee dos umbrales distintos de conmutación umbral superior V_{T+} y umbral inferior V_{T-} , en el que la histéresis es la diferencia entre ambos umbrales $V_H = V_{T+} - V_{T-}$, esto permite que no haya conmutaciones indeseadas, lo que implica que elimina el ruido, convierte señales analógicas lentas en digitales limpias y evita oscilaciones cerca del umbral.[22]

El circuito 74*14 puede encontrarse en las configuraciones 74LS14, 74S14 y 74HC14. Analizando las características de 74HC14:

74HC14	Parámetro	Valor / Rango
Rango de tensión de alimentación	VCC (VDD)	2 V – 6 V
Rango de tensiones de entrada	Bajo	0 – 1.35 V (VCC = 4.5V)
	Alto	3.15 V – VCC
Rango de tensiones de salida	Bajo	0 – 0.33 V
	Alto	3.84 V – VCC
Tiempos de propagación	t_{PLH} t_{PHL}	14 – 23 ns (según VCC)
Tiempos de transición	t_r t_f	8 – 19 ns (según VCC)

12. Investigue en que consiste la modulación de ancho de pulso (PWM).

PWM o modulación por ancho de pulso es una técnica que permite codificar una información analógica en la forma del ancho del pulso en una señal periódica; esto típicamente se presenta como unas variaciones en el ciclo de trabajo de una señal cuadrada.

Esta técnica es utilizada en varias aplicaciones como controlar la velocidad de motores, luminosidad de LEDs o para controlar el voltaje en fuentes conmutadas. [23]

13. Investigue que es el efecto de rebote y típicos circuitos anti-rebote (debouncing circuits)

R/El **efecto de rebote** es un fenómeno propio de interruptores mecánicos, en el cual al accionar un botón o contacto se producen múltiples transiciones rápidas antes de que la señal se estabilice. Esto puede provocar que un sistema digital interprete varias activaciones en lugar de una sola [24]

Para eliminar este problema se utilizan **circuitos anti-rebote (debouncing)**, entre los cuales se encuentran los filtros **RC**, los **flip-flops** y los circuitos basados en **disparadores Schmitt**. Estos métodos permiten obtener una señal digital limpia y estable a partir de un interruptor mecánico [24].

14. Explique qué es el modelado de comportamiento y de estructura en diseño digital. Brinde un ejemplo de cada uno.

R/ En diseño digital con HDL existen diferentes niveles de abstracción: modelado de comportamiento y estructural. El modelado comportamental describe qué hace el sistema, sin especificar cómo está construido físicamente. Se enfoca en la función lógica. No se describen compuertas específicas, solo la relación funcional. Ejemplo en Verilog: se define un multiplexor simple de 2 a 1.[25]

```
módulo mux2to1 ( entrada cable a, b, sel, salida cable out ) ;
```

```
    asignar = ( sel)? b : a;
```

```
módulo final
```

El modelado estructural describe explícitamente la interconexión de componentes. Se construye el circuito como si fuera un diagrama de bloques textual. Ejemplo en Verilog: se define un sumador completo.

```
módulo full_adder (entrada a, b, cin , salida suma, cout ) ;
```

```

cable s1, c1, c2;

xor (s1, a, b);

xor (suma, s1, cin );

y (c1, a, b);

y (c2, s1, cin );

o ( cout , c1, c2);

módulo final

```

15. Explique el proceso de síntesis lógica en el diseño de circuitos digitales, tanto para el desarrollo de un ASIC como para una FPGA.

Primeramente, la síntesis lógica es el proceso en el que mediante una descripción en HDL como lo son Verilog, System Verilog y VHDL se transforma en una implementación de hardware como compuertas o bloques físicos.

La síntesis lógica de ASIC y FPGA siguen etapas como: descripción en HDL, análisis y elaboración, optimización lógica, mapeo tecnológico y generación de netlist sin embargo ambas tienen su diferenciador, en FPGA, el proceso se orienta al mapeo hacia bloques programables predefinidos, mientras que en ASIC se realiza una optimización profunda hacia una biblioteca tecnológica específica, culminando en la fabricación del circuito integrado.

Proceso de Síntesis de la FPGA	
Etapas	Descripción
1. Descripción RTL	El diseñador escribe el circuito en HDL (Verilog/SystemVerilog/VHDL) describiendo su comportamiento y estructura a nivel de registros y transferencia de datos. En esta etapa no se trabaja con compuertas físicas, sino con una representación abstracta del hardware.
2. Elaboración	La herramienta de síntesis analiza el código, verifica su sintaxis, resuelve parámetros, expande jerarquías y genera una representación intermedia del diseño lista para ser optimizada.

3. Síntesis lógica	El RTL es transformado en una red lógica optimizada. Se eliminan redundancias, se simplifican expresiones booleanas y se organizan los recursos para cumplir con restricciones de área y tiempo.
4. Mapeo tecnológico	La lógica optimizada se traduce a los recursos físicos disponibles en la FPGA, como LUTs, flip-flops, bloques DSP y memorias internas. Aquí el diseño deja de ser abstracto y pasa a depender de la arquitectura específica del dispositivo.
5. Place & Route	Se determina la ubicación física de cada bloque dentro de la FPGA y se enrutan las interconexiones entre ellos. Esta etapa afecta directamente el cumplimiento de las restricciones de timing.
6. Análisis de tiempo (STA)	Se verifica que todas las rutas críticas cumplan con las restricciones de frecuencia y tiempos establecidos. Si no se cumplen, el diseño debe optimizarse nuevamente.
7. Generación de bitstream	Se crea el archivo final que configura la FPGA. Este archivo programa la matriz lógica y las interconexiones internas del dispositivo.

Proceso de Síntesis de ASIC	
Etapas	Descripción
1. Descripción RTL	El diseño se describe en HDL a nivel de registros y transferencia de datos (RTL). En esta etapa se define el comportamiento funcional del circuito sin considerar aún detalles físicos de implementación.
2. Elaboración	La herramienta de síntesis analiza el código, verifica su consistencia, resuelve jerarquías y parámetros, y genera una representación interna del diseño lista para optimización.
3. Síntesis lógica	El diseño RTL se transforma en una red de compuertas lógicas optimizadas. Se realizan simplificaciones

Proceso de Síntesis de ASIC	
Etapa	Descripción
	booleanas y ajustes para cumplir con restricciones de área, potencia y tiempo.
4. Mapeo tecnológico	La red lógica optimizada se traduce a una biblioteca específica de celdas estándar (AND, OR, flip-flops, buffers, etc.) perteneciente a la tecnología del proceso de fabricación seleccionada.
5. Floorplanning	Se define la organización general del chip, asignando áreas para bloques funcionales, memorias y rutas principales de interconexión.
6. Placement	Las celdas estándar se ubican físicamente dentro del área del chip de acuerdo con criterios de optimización de área y rendimiento.
7. Routing	Se diseñan las interconexiones metálicas que conectan todas las celdas del circuito, respetando reglas físicas del proceso tecnológico.
8. Verificación física (DRC/LVS)	Se comprueba que el diseño cumpla con las reglas de fabricación y que el layout físico sea eléctricamente equivalente al esquema lógico original.
9. Generación del archivo GDSII (Tape-out)	Se produce el archivo final que será enviado a fabricación. A partir de este punto, el diseño se convierte en un circuito integrado físico permanente.

El proceso de síntesis lógica en FPGA y ASIC parte de una descripción común en RTL, pero se diferencian en la etapa de implementación física. En la FPGA, la lógica se mapea hacia recursos programables como LUTs y flip-flops, culminando en la generación de un bitstream reconfigurable, mientras que en la ASIC la síntesis continúa hacia el diseño físico detallado, incluyendo placement, routing y verificación, hasta la fabricación del circuito integrado. Esto implica mayores costos y tiempos de desarrollo, pero permite una optimización superior en área, potencia y rendimiento.[26]

16. Investigue sobre la tecnología de FPGAs. Describa el funcionamiento de la lógica programable en general, así como los componentes básicos de una. Una FPGA es un circuito integrado, que a diferencia de otros circuitos integrados como ASICs, CPUs o ICs que solo tienen una estructura y función predeterminada en su diseño; una FPGA puede ser reprogramada y adaptarse a lo que se necesita en el lugar de aplicación.

Para lograr esto la FPGA está construida como una matriz de bloques lógicos, las FPAs estas construidas con las siguientes partes:

- **Bloques lógicos:** Son las soluciones modulares básicos de una FPGA. Pueden programarse para realizar diversas funciones lógicas.
- **Interconexiones:** Son cables programables que conectan los bloques lógicos.
- **Bloques de E/S:** Se utilizan para interconectar la FPGA con otros periféricos y componentes.

[27]

Finalmente para poder controlar el comportamiento de estas compuestas lógicas se debe de usar un programa de edición de lenguaje de descripción de hardware, la función de este programa es traducir un programa creado en un lenguaje de programación que sea más fácil de entender, y transformar ese programa en las instrucciones necesarias para configurar los bloques lógicos dentro de la FPGA, las interconexiones entre estos y a los periféricos necesarios para correr el programa deseado a nivel de hardware, en vez de usar un sistema operativo.

17. Investigue sobre el protocolo UART y como se implementaría la sección de TX.

El protocolo URAT es un protocolo de comunicación serial asincrónico, consta de 2 pines RX que se encarga de recibir las señales y TX que se encarga de enviar las señales, cabe recalcar que al conectar 2 dispositivos el TX de uno se conecta al RX del otro y viceversa; el protocolo URAT envía información en bloques, estos bloques están formados por:

1 bit de inicio: indica cuando se debe de empezar a leer el mensaje, normalmente se ve como una transición de alto a bajo

5 a 9 bits de mensaje: esta es la información que se desea transmitir, puede ser de 5 o 8 bits, si no se utiliza un bit de parodia se pueden mandar mensajes de 9 bits.

0 o 1 bit de parodia: es un bit utilizado por el receptor para saber si no se cambió ningún bit del mensaje durante la transmisión, la parodia funciona contando el número de bits que son 1 en el mensaje, si es par el bit de parodia será 0, si es impar el bit de parodia será 1.

1 o 2 bits de final: al terminar el mensaje se pasa la línea de bajo a alto y se mantiene en alto por 1 o 2 bits.

[28]

A la hora de enviar datos por URAT se debe de mantener en cuenta que la velocidad de transmisión en baudios sea la misma entre el receptor y el transmisor, seguidamente que

la conexiones entre los dispositivos sea correcta, es decir TX con RX y viceversa, y finalmente hay que configurar la transmisión para que siga el formato de los bloques de datos del protocolo URAT.

Referencias bibliográficas:

- [1] Elprocus, “Transistor transistor logic (TTL),” [Online]. Available: <https://www.elprocus.com/transistor-transistor-logic-ttl/>
- [2] Texas Instruments, *The TTL Logic Family*, Application Report SDLA006, 2008. [Online]. Available: <https://www.ti.com/lit/an/sdla006/sdla006.pdf>
- [3] A. Limited, “The Ultimate Guide to 4000 Series Integrated Circuits,” [Online]. Available: <https://www.allelcoelec.com/blog/The-Ultimate-Guide-to-4000-Series-Integrated-Circuits.html>
- [4] J. F. Wakerly, *Diseño digital*. Pearson Educación, 2001.
- [5] “Familia lógica de circuitos integrados,” Scribd. [Online]. Available: <https://www.scribd.com/presentation/410179852/Familia-Logica-de-Circuitos-Integrados>
- [6] Universidad de Sevilla, “Familias lógicas,” [Online]. Available: https://www.dte.us.es/docencia/etsii/gii-ti/cedti/grupopilarparra/material/notas-tema-3/familias_logicas_libroETC.pdf
- [7] R. Sheldon, “Fan-out,” *WhatIs*, Aug. 25, 2022. [Online]. Available: <https://www.techtarget.com/whatis/definition/fan-out>
- [8] Texas Instruments, *SN74L00 Quad 2-Input NAND Gates, Series 54L/74L Low-Power TTL Integrated Circuits Datasheet*, Dallas, TX, USA, 1971. [Online]. Available: https://ftpmirror.your.org/pub/misc/bitsavers/components/ti/_dataBooks/1971_TI_IntegratedCircuits/08.pdf
- [9] Texas Instruments, *SN74LS00 Quad 2-Input NAND Gates*, Dallas, TX, USA, 2023. [Online]. Available: <https://www.ti.com/lit/ds/symlink/sn74ls00.pdf>
- [10] Texas Instruments, *SN74HC00, SN54HC00: Quadruple 2-Input NAND Gates*, Datasheet Rev. H, Aug. 2021. [Online]. Available: <https://www.ti.com/lit/ds/symlink/sn74hc00.pdf>
- [11] Texas Instruments, *CD4001B CMOS Quad 2-Input NOR Gate*, Dallas, TX, USA, 2021. [Online]. Available: <https://www.ti.com/lit/ds/symlink/cd4001b.pdf>
- [12] Texas Instruments, “7400 Quadruple 2-Input Positive-NAND Gates,” Datasheet. [Online]. Available: <https://www.alldatasheet.com/html-pdf/840854/TI1/7400/51/1/7400.html>

- [13] Fairchild Semiconductor, “7402 Quad 2-Input NOR Gates,” Datasheet. [Online]. Available: <https://www.alldatasheet.com/html-pdf/50887/FAIRCHILD/7402/1219/3/7402.html>
- [14] STMicroelectronics, “7404 Hex Inverter,” Datasheet. [Online]. Available: <https://www.alldatasheet.com/html-pdf/23020/STMICROELECTRONICS/7404/1618/1/7404.html>
- [15] Texas Instruments, “7414 Hex Schmitt-Trigger Inverters,” Datasheet. [Online]. Available: <https://www.alldatasheet.com/html-pdf/27375/TI/7414/18/1/7414.html>
- [16] RCA, *RCA CMOS Databook*, 1983. [Online]. Available: https://archive.org/details/bitsavers_rcadataBoook_74957274
- [17] RCA, *RCA CMOS Databook*, 1983, p. 81. [Online]. Available: https://archive.org/details/bitsavers_rcadataBoook_74957274/page/n81/mode/2up
- [18] RCA, *RCA CMOS Databook*, 1983, p. 239. [Online]. Available: https://archive.org/details/bitsavers_rcadataBoook_74957274/page/n239/mode/2up
- [19] RCA, *RCA CMOS Databook*, 1983, p. 405. [Online]. Available: https://archive.org/details/bitsavers_rcadataBoook_74957274/page/n405/mode/2up
- [20] All About Circuits, “CMOS gate circuitry,” [Online]. Available: <https://www.allaboutcircuits.com/textbook/digital/chpt-3/cmos-gate-circuitry/>
- [21] T. Agarwal, “Working of pull-up and pull-down resistors with examples,” *ElProCus*, Jul. 15, 2022. [Online]. Available: <https://www.elprocus.com/pull-up-and-pull-down-resistors-with-applications/>
- [22] Texas Instruments, *SN74HC14 Hex Schmitt-Trigger Inverters*, Datasheet, 2021. [Online]. Available: <https://www.ti.com/lit/ds/symlink/sn74hc14.pdf>
- [23] “PWM: How duty cycle controls power and signals,” Dec. 31, 2025. [Online]. Available: <https://analogcircuitdesign.com/pulse-width-modulation-pwm/>
- [24] GeeksforGeeks, “Switch debounce in digital circuits,” [Online]. Available: <https://www.geeksforgeeks.org/digital-logic/switch-debounce-in-digital-circuits/>
- [25] Ramanathan, “Parte 18: Tipos de modelado en Verilog,” DigiKey, Jun. 16, 2025. [Online]. Available: <https://www.digikey.com/en/maker/tutorials/2025/part-18-types-of-modeling-in-verilog>
- [26] D. Harris and S. Harris, *Digital Design and Computer Architecture*, 2nd ed. Morgan Kaufmann, 2012.
- [27] Supermicro, “¿Qué es una FPGA?,” [Online]. Available: <https://www.supermicro.com/es/glossary/fpga> (accessed Feb. 26, 2026).

[28] E. Peña and M. Grace, "UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter," *Analog Dialogue*, Analog Devices. [Online]. Available: <https://www.analog.com/en/resources/analog-dialogue/articles/uart-a-hardware-communication-protocol.html> (accessed Feb. 26, 2026).

[1] Elprocus, "Transistor transistor logic (TTL)," [Online]. Available: