**UART RX**

**LIBRARY** **IEEE**;

**USE** **IEEE.STD\_LOGIC\_1164.ALL**;

**USE** **IEEE.NUMERIC\_STD.ALL**;

**ENTITY** **rx\_uart** **IS**

**PORT**(CLK: **IN** **STD\_LOGIC**;

RX: **IN** **STD\_LOGIC**;

DATA: **OUT** **STD\_LOGIC\_VECTOR**(**7** **DOWNTO** **0**);

BUSY: **OUT** **STD\_LOGIC**);

**END** **rx\_uart**;

**ARCHITECTURE** **BEAS** **OF** **rx\_uart** **IS**

**SIGNAL** INDEX: **INTEGER** **RANGE** **0** **TO** **9** := **0**;

**SIGNAL** PRESCL: **INTEGER** **RANGE** **0** **TO** **434** := **0**; -- PARA BAUDAJE PRESCL = 50 MHz/115200

**SIGNAL** RX\_FLAG: **STD\_LOGIC** := '0';

**SIGNAL** DATA\_AUX: **STD\_LOGIC\_VECTOR**(**9** **DOWNTO** **0**);

**BEGIN**

**PROCESS**(CLK)

**BEGIN**

**IF** RISING\_EDGE (CLK) **THEN**

**IF** (RX\_FLAG = '0' **AND** RX = '0') **THEN**

INDEX <= **0**; -- INICIAR EN POSICIÓN CERO

PRESCL <= **0**;

BUSY <= '1'; -- PUERTO SERIE OCUPADO

RX\_FLAG <= '1'; -- ESTA RECIBIENDO

**END** **IF**;

**IF** RX\_FLAG = '1' **THEN**

DATA\_AUX(INDEX) <= RX;

**IF** PRESCL < **434** **THEN**

PRESCL <= PRESCL + **1**;

**ELSE**

PRESCL <= **0**;

**END** **IF**;

**END** **IF**;

**IF** PRESCL = **217** **THEN**

**IF** INDEX < **9** **THEN**

INDEX <= INDEX + **1**;

**ELSE**

**IF** (DATA\_AUX(**0**) = '0' **AND** DATA\_AUX(**9**) = '1') **THEN** -- START = 0 & STOP = 1

DATA <= DATA\_AUX(**8** **DOWNTO** **1**);

**ELSE**

DATA <= (**OTHERS** => '0');

**END** **IF**;

RX\_FLAG <= '0';

BUSY <= '0';

**END** **IF**;

**END** **IF**;

**END** **IF**;

**END** **PROCESS**;

**END** **BEAS**;

**UART TX**

**LIBRARY** **IEEE**;

**USE** **IEEE.STD\_LOGIC\_1164.ALL**;

**USE** **IEEE.NUMERIC\_STD.ALL**;

**ENTITY** **uart\_tx** **IS**

**PORT**(CLK: **IN** **STD\_LOGIC**;

TX\_OUT: **OUT** **STD\_LOGIC**;

BUSY: **IN** **STD\_LOGIC**; -- INICIA COMUNICACION

SW: **IN** **INTEGER** **RANGE** **0** **TO** **3**);

**END** **uart\_tx**;

**ARCHITECTURE** **BEAS** **OF** **uart\_tx** **IS**

**SIGNAL** START: **STD\_LOGIC**;

**SIGNAL** PRESCL: **INTEGER** **RANGE** **0** **TO** **434** := **0**; -- BAUDAJE = 50MHz/PRESCL

**SIGNAL** INDEX: **INTEGER** **RANGE** **0** **TO** **9** := **0**; -- SELECCIONA QUE BIT SE ENVIA

**SIGNAL** DATA\_FRAME: **STD\_LOGIC\_VECTOR**(**9** **DOWNTO** **0**) ; -- VECTOR DE 8 BITS DE DATOS, BIT DE START Y BIT DE STOP

**SIGNAL** TX\_FLAG: **STD\_LOGIC** := '0';

**SIGNAL** DATA: **STD\_LOGIC\_VECTOR**(**7** **DOWNTO** **0**) := (**OTHERS** => '0'); -- DATO DE 8 BITS

**SIGNAL** HEX\_DATA: **STD\_LOGIC\_VECTOR**(**7** **DOWNTO** **0**) := (**OTHERS** => '0');

**SIGNAL** PLACE: **INTEGER** **RANGE** **0** **TO** **1**; -- TAMAÑO DE ARRAY\_TX

**SIGNAL** DELAY: **INTEGER** := **10000**; -- RETARDO ENTRE DATOS

**SIGNAL** CONTA: **INTEGER** := **0**;

-----------------------------------------------------------------------------------------

**TYPE** ARRAY\_TX **IS** **ARRAY** (**0** **TO** **1**) **OF** **STD\_LOGIC\_VECTOR**(**7** **DOWNTO** **0**);

**SIGNAL** ASC\_DATA : ARRAY\_TX := (X"30", X"0A"); -- X0A SALTO DE LINEA

**BEGIN**

**PROCESS**(CLK, BUSY)

**BEGIN**

ASC\_DATA(**0**) <= HEX\_DATA; -- PONGO EL DATO QUE ASIGNO

**IF** BUSY = '1' **THEN**

**IF** FALLING\_EDGE(CLK) **THEN**

**IF** CONTA = DELAY **THEN**

CONTA <= **0**;

START <= '1';

DATA <= ASC\_DATA(PLACE);

**IF** PLACE = **1** **THEN**

PLACE <= **0**;

**ELSE**

PLACE <= PLACE + **1**;

**END** **IF**;

**ELSE**

CONTA <= CONTA + **1**;

START <= '0';

**END** **IF**;

**END** **IF**;

**END** **IF**;

**END** **PROCESS**;

**PROCESS**(CLK)

**BEGIN**

**IF** RISING\_EDGE(CLK) **THEN**

**IF** (TX\_FLAG = '0' **AND** START = '1') **THEN**

TX\_FLAG <= '1';

DATA\_FRAME(**0**) <= '0'; -- BIT DE START

DATA\_FRAME(**9**) <= '1'; -- BIT DE STOP

DATA\_FRAME(**8** **DOWNTO** **1**) <= DATA; -- 8 BITS (DATOS)

**END** **IF**;

**IF** TX\_FLAG = '1' **THEN**

**IF** PRESCL < **433** **THEN**

PRESCL <= PRESCL + **1**;

**ELSE**

PRESCL <= **0**;

**END** **IF**;

**IF** PRESCL = **217** **THEN** -- (434/2)

TX\_OUT <= DATA\_FRAME (INDEX);

**IF** INDEX < **9** **THEN**

INDEX <= INDEX + **1**;

**ELSE**

TX\_FLAG <= '0';

INDEX <= **0**;

**END** **IF**;

**END** **IF**;

**END** **IF**;

**END** **IF**;

**END** **PROCESS**;

**WITH** (SW) **SELECT** -- MUX ASCII (HEX)

HEX\_DATA <= X"30" **WHEN** **0**,

X"31" **WHEN** **1**,

X"32" **WHEN** **2**,

X"33" **WHEN** **3**,

X"34" **WHEN** **OTHERS**;

**END** **BEAS**;

| **dec** | **bin** | **hex** | **símbolo** | **dec** | **bin** | **hex** | **símbolo** | **dec** | **bin** | **hex** | **símbolo** | **dec** | **bin** | **hex** | **símbolo** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 00000000 | 00 | ␀ | **32** | 00100000 | 20 |  | **64** | 01000000 | 40 | @ | **96** | 01100000 | 60 | ` |
| **1** | 00000001 | 01 | ␁ | **33** | 00100001 | 21 | ! | **65** | 01000001 | 41 | A | **97** | 01100001 | 61 | a |
| **2** | 00000010 | 02 | ␂ | **34** | 00100010 | 22 | " | **66** | 01000010 | 42 | B | **98** | 01100010 | 62 | b |
| **3** | 00000011 | 03 | ␃ | **35** | 00100011 | 23 | # | **67** | 01000011 | 43 | C | **99** | 01100011 | 63 | c |
| **4** | 00000100 | 04 | ␄ | **36** | 00100100 | 24 | $ | **68** | 01000100 | 44 | D | **100** | 01100100 | 64 | d |
| **5** | 00000101 | 05 | ␅ | **37** | 00100101 | 25 | % | **69** | 01000101 | 45 | E | **101** | 01100101 | 65 | e |
| **6** | 00000110 | 06 | ␆ | **38** | 00100110 | 26 | & | **70** | 01000110 | 46 | F | **102** | 01100110 | 66 | f |
| **7** | 00000111 | 07 | ␇ | **39** | 00100111 | 27 | ' | **71** | 01000111 | 47 | G | **103** | 01100111 | 67 | g |
| **8** | 00001000 | 08 | ␈ | **40** | 00101000 | 28 | ( | **72** | 01001000 | 48 | H | **104** | 01101000 | 68 | h |
| **9** | 00001001 | 09 | ␉ | **41** | 00101001 | 29 | ) | **73** | 01001001 | 49 | I | **105** | 01101001 | 69 | i |
| **10** | 00001010 | 0A | ␊ | **42** | 00101010 | 2A | \* | **74** | 01001010 | 4A | J | **106** | 01101010 | 6A | j |
| **11** | 00001011 | 0B | ␋ | **43** | 00101011 | 2B | + | **75** | 01001011 | 4B | K | **107** | 01101011 | 6B | k |
| **12** | 00001100 | 0C | ␌ | **44** | 00101100 | 2C | , | **76** | 01001100 | 4C | L | **108** | 01101100 | 6C | l |
| **13** | 00001101 | 0D | ␍ | **45** | 00101101 | 2D | - | **77** | 01001101 | 4D | M | **109** | 01101101 | 6D | m |
| **14** | 00001110 | 0E | ␎ | **46** | 00101110 | 2E | . | **78** | 01001110 | 4E | N | **110** | 01101110 | 6E | n |
| **15** | 00001111 | 0F | ␏ | **47** | 00101111 | 2F | / | **79** | 01001111 | 4F | O | **111** | 01101111 | 6F | o |
| **16** | 00010000 | 10 | ␐ | **48** | 00110000 | 30 | 0 | **80** | 01010000 | 50 | P | **112** | 01110000 | 70 | p |
| **17** | 00010001 | 11 | ␑ | **49** | 00110001 | 31 | 1 | **81** | 01010001 | 51 | Q | **113** | 01110001 | 71 | q |
| **18** | 00010010 | 12 | ␒ | **50** | 00110010 | 32 | 2 | **82** | 01010010 | 52 | R | **114** | 01110010 | 72 | r |
| **19** | 00010011 | 13 | ␓ | **51** | 00110011 | 33 | 3 | **83** | 01010011 | 53 | S | **115** | 01110011 | 73 | s |
| **20** | 00010100 | 14 | ␔ | **52** | 00110100 | 34 | 4 | **84** | 01010100 | 54 | T | **116** | 01110100 | 74 | t |
| **21** | 00010101 | 15 | ␕ | **53** | 00110101 | 35 | 5 | **85** | 01010101 | 55 | U | **117** | 01110101 | 75 | u |
| **22** | 00010110 | 16 | ␖ | **54** | 00110110 | 36 | 6 | **86** | 01010110 | 56 | V | **118** | 01110110 | 76 | v |
| **23** | 00010111 | 17 | ␗ | **55** | 00110111 | 37 | 7 | **87** | 01010111 | 57 | W | **119** | 01110111 | 77 | w |
| **24** | 00011000 | 18 | ␘ | **56** | 00111000 | 38 | 8 | **88** | 01011000 | 58 | X | **120** | 01111000 | 78 | x |
| **25** | 00011001 | 19 | ␙ | **57** | 00111001 | 39 | 9 | **89** | 01011001 | 59 | Y | **121** | 01111001 | 79 | y |
| **26** | 00011010 | 1A | ␚ | **58** | 00111010 | 3A | : | **90** | 01011010 | 5A | Z | **122** | 01111010 | 7A | z |
| **27** | 00011011 | 1B | ␛ | **59** | 00111011 | 3B | ; | **91** | 01011011 | 5B | [ | **123** | 01111011 | 7B | { |
| **28** | 00011100 | 1C | ␜ | **60** | 00111100 | 3C | < | **92** | 01011100 | 5C | \ | **124** | 01111100 | 7C | | |
| **29** | 00011101 | 1D | ␝ | **61** | 00111101 | 3D | = | **93** | 01011101 | 5D | ] | **125** | 01111101 | 7D | } |
| **30** | 00011110 | 1E | ␞ | **62** | 00111110 | 3E | > | **94** | 01011110 | 5E | ^ | **126** | 01111110 | 7E | ~ |
| **31** | 00011111 | 1F | ␟ | **63** | 00111111 | 3F | ? | **95** | 01011111 | 5F | \_ | **127** | 01111111 | 7F | ␡ |