Created By: Antonio Jimenez

Requirements: Configuration Registers

* CONFIG\_REG\_00: Every single output of the configuration register module shall be 0 when ‘i\_reset’ is set to 1, with the exception of ‘o\_afmr1’, ‘o\_afmr2’, ‘o\_afmr3’, ‘o\_afmr4’, ‘o\_afir1’, ‘o\_afir2’, ‘o\_afir3’, and ‘o\_afir4’.
* CONFIG\_REG\_01: ‘o\_reg\_ack’ shall output 1 for one clock cycle when any bit within ‘i\_rs\_vector’ is set to 1 for the previous clock cycle, else ‘o\_reg\_ack’ is set to 0.
* CONFIG\_REG\_02: ‘o\_reg\_error’ shall output 1 for one clock cycle whenever there is an attempt to write the non-default value to the reserved bits of an internal register within the previous clock cycle, else ‘o\_reg\_error’ is set to 0.
* CONFIG\_REG\_03: ‘o\_interrupt’ shall output 1 when any of the bits in the internal interrupt status register is set to 1 while the corresponding bit positions in the internal interrupt enable register bits are set to 1, else ‘o\_interrupt’ is set to 0.
* CONFIG\_REG\_04: ‘o\_soft\_reset’ shall be set to 1 for one clock cycle when ‘i\_sys\_clk’ transitions from 0 to 1 while bit 31 of i\_reg\_w\_bus is set to 1 and i\_rs\_vector bit 0 is set to 1, else ‘o\_soft\_reset’ is set to 0.
* CONFIG\_REG\_05: ‘o\_tx\_fifo\_data[0:127]’ shall output the data that is stored in the four internal registers from least significant bit to most significant bit in this corresponding order: tx ID[0:31], tx DLC[0:31], tx DW1[0:31], and tx DW2[0:31].
* CONFIG\_REG\_06: ‘o\_tx\_w\_en’ shall be set to 1 for one clock cycle when ‘i\_sys\_clk’ transition from 0 to 1 while i\_tx\_full is 0 and i\_rs\_vector[13] was set to 1 in the previous clock cycle, else ‘o\_tx\_w\_en’ is set to 0.
* CONFIG\_REG\_07: ‘o\_hpb\_data[0:127]’ shall output the data that is stored in the four internal registers from least significant bit to most significant bit in this corresponding order: hpb ID[0:31], hpb DLC[0:31], hpb DW1[0:31], and hpb DW2[0:31].
* CONFIG\_REG\_08: ‘o\_hpb\_r\_en’ shall be set to 1 for one clock cycle when ‘i\_sys\_clk’ transitions from 0 to 1 while i\_rs\_vector[17] was set to 1 in the previous clock cycle, else ‘o\_hpb\_r\_en’ shall be set to 0.
* CONFIG\_REG\_09: ‘o\_hpb\_full’ shall be set to 1 for one clock cycle when ‘i\_sys\_clk’ transitions from 0 to 1 while i\_r\_neg\_w is 0 and i\_rs\_vector is 0x20000 in the previous clock cycle, else o\_hpb\_full is 0.
* CONFIG\_REG\_10: ‘o\_lback’ shall output the value of the internal mode select register at bit 30.
* CONFIG\_REG\_11: ‘o\_sleep’ shall output the value of the internal mode select register at bit 31 synched to i\_can\_clk.
* CONFIG\_REG\_12: ‘o\_sjw[1:0]’ shall output the value of the internal bit timing register from the corresponding bit position; [24:23] synched to i\_can\_clk.
* CONFIG\_REG\_13: ‘o\_ts2[2:0]’ shall output the value of the internal bit timing register from the corresponding bit position; [27:25] synched to i\_can\_clk.
* CONFIG\_REG\_14: ‘o\_ts1[3:0]’ shall output the value of the internal bit timing register from the corresponding bit position; [31:28] synched to i\_can\_clk.
* CONFIG\_REG\_15: ‘o\_rx\_r\_en’ shall output 1 for one clock cycle when ‘i\_sys\_clk’ transitions from 0 to 1 while i\_r\_neg\_w is 1, i\_rx\_empty is 0, and i\_rs\_vector is 0x200000 in the previous clock cycle, else ‘o\_rx\_r\_en’ is 0.
* CONFIG\_REG\_16: ‘o\_uaf1’ shall output the value of the internal acceptance filter register at bit 31.
* CONFIG\_REG\_17: ‘o\_uaf2’ shall output the value of the internal acceptance filter register at bit 30.
* CONFIG\_REG\_18: ‘o\_uaf3’ shall output the value of the internal acceptance filter register at bit 29.
* CONFIG\_REG\_19: ‘o\_uaf4’ shall output the value of the internal acceptance filter register at bit 28.
* CONFIG\_REG\_20: ‘o\_afmr1’ shall output the value stored in the internal acceptance filter mask register 1.
* CONFIG\_REG\_21: ‘o\_afmr2’ shall output the value stored in the internal acceptance filter mask register 2.
* CONFIG\_REG\_22: ‘o\_afmr3’ shall output the value stored in the internal acceptance filter mask register 3.
* CONFIG\_REG\_23: ‘o\_afmr4’ shall output the value stored in the internal acceptance filter mask register 4.
* CONFIG\_REG\_24: ‘o\_afir1’ shall output the value stored in the internal acceptance filter ID register 1.
* CONFIG\_REG\_25: ‘o\_afir2’ shall output the value stored in the internal acceptance filter ID register 2.
* CONFIG\_REG\_26: ‘o\_afir3’ shall output the value stored in the internal acceptance filter ID register 3.
* CONFIG\_REG\_27: ‘o\_afir4’ shall output the value stored in the internal acceptance filter ID register 4.
* CONFIG\_REG\_28: The internal register, rx ID, shall store the value, ‘i\_rx\_fifo\_data[31:0]’, when ‘o\_rx\_r\_en’ is 1 and ‘i\_sys\_clk’ transitions from 0 to 1.
* CONFIG\_REG\_29: The internal register, rx DLC, shall store the value, ‘i\_rx\_fifo\_data[63:32]’, when ‘o\_rx\_r\_en’ is 1 and ‘i\_sys\_clk’ transitions from 0 to 1.
* CONFIG\_REG\_30: The internal register, rx DW1, shall store the value, ‘i\_rx\_fifo\_data[95:64]’, when ‘o\_rx\_r\_en’ is 1 and ‘i\_sys\_clk’ transitions from 0 to 1.
* CONFIG\_REG\_31: The internal register, rx DW2, shall store the value, ‘i\_rx\_fifo\_data[127:96]’, when ‘o\_rx\_r\_en’ is 1 and ‘i\_sys\_clk’ transitions from 0 to 1.
* CONFIG\_REG\_32: Bit position 27 in the internal error status register shall store the value input into i\_acker every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_33: Bit position 28 in the internal error status register shall store the value input into i\_berr every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_34: Bit position 29 in the internal error status register shall store the value input into i\_ster every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_35: Bit position 30 in the internal error status register shall store the value input into i\_fmer every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_36: Bit position 31 in the internal error status register shall store the value input into i\_crcer every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_37: Bit position 20 in the internal status register shall store the value input into ‘i\_acfbsy’ every clock cycle.
* CONFIG\_REG\_38: Bit position [24:23] in the internal status register shall store the value input into ‘i\_estat’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_39: Bit position 25 in the internal status register shall store the value input into ‘i\_errwrn’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_40: Bit position 26 in the internal status register shall store the value input into ‘i\_bbsy’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_41: Bit position 27 in the internal status register shall store the value input into ‘i\_bidle’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_42: Bit position 28 in the internal status register shall store the value input into ‘i\_normal’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_43: Bit position 29 in the internal status register shall store the value input into ‘i\_sleep’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_44: Bit position 30 in the internal status register shall store the value input into ‘i\_lback’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_45: Bit position 31 in the internal status register shall store the value input into ‘i\_config’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_46: Bit position 20 in the internal interrupt status register shall store the value input into ‘i\_wakeup’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_47: Bit position 21 in the internal interrupt status register shall store the value input into ‘i\_sleep’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_48: Bit position 22 in the internal interrupt status register shall store the value input into ‘i\_bsoff’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_49: Bit position 23 in the internal interrupt status register shall store the value input into ‘i\_error’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_50: Bit position 24 in the internal interrupt status register shall store the value input into ‘i\_rxnemp’ every clock cycle.
* CONFIG\_REG\_51: Bit position 25 in the internal interrupt status register shall store the value input into ‘i\_rxoflw’ every clock cycle.
* CONFIG\_REG\_52: Bit position 26 in the internal interrupt status register shall store the value input into ‘i\_rxuflw’ every clock cycle.
* CONFIG\_REG\_53: Bit position 27 in the internal interrupt status register shall store the value input into ‘i\_rxok’ every clock cycle.
* CONFIG\_REG\_54: Bit position 28 in the internal interrupt status register shall store the value output of ‘o\_hpb\_full’ every clock cycle.
* CONFIG\_REG\_55: Bit position 29 in the internal interrupt status register shall store the value input into ‘i\_tx\_full’ every clock cycle.
* CONFIG\_REG\_56: Bit position 30 in the internal interrupt status register shall store the value input into ‘i\_txok’ every clock cycle.
* CONFIG\_REG\_57: Bit position 31 in the internal interrupt status register shall store the value input into ‘i\_arblst’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_58: Bit position [23:16] in the internal error count register shall store the value input into ‘i\_rec’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_59: Bit position [31:24] in the internal error count register shall store the value input into ‘i\_tec’ every clock cycle synched to i\_sys\_clk.
* CONFIG\_REG\_60: Bit position 21 in the internal status register shall store the value input into ‘i\_txfull’ every clock cycle.