AIDAPAL PLUG

Manual Analysis

```
1  int __fastcall UART_reset_and_configure_uart(int result)
2  {
3    int v1; // r3
4
5    UART_UART_SRR_REG = 7;                        // reset UART
6    v1 = UART_UART_LCR_REG;
7    UART_UART_LCR_REG = v1 | 0x80;                // set Divisor Latch Access Bit, required to set
8                                                  // buad rate via DLL/DLH reg
9    UART_UART_DLF_REG = result;                   // set the rate divisor fractional part - 0x1106
10   UART_UART_RBR_THR_DLL_REG = BYTE1(result);    // set the low byte of divisor
11   UART_UART_IER_DLH_REG = BYTE2(result);        // set the hibyte of the divisor
12   UART_UART_LCR_REG = 3;                        // datalen select b11 is 8bits
13   UART_UART_IIR_FCR_REG = 7;                    // setup read interrupt register
14   UART_UART_IER_DLH_REG = BYTE2(result) & 0xFE;
15   return result;
16 }
```

Aidapal Analysis

```
1  // This function configures various UART registers with the provided input value.
2  // It sets specific bits in the Software Reset Register, Line Control Register,
3  // Divisor Latch Fraction Register, Receive Buffer Register Threshold, Interrupt
4  // Enable Register, and Interrupt Identification Register/FIFO Control Register
5  // based on the input value.
6  int __fastcall configureUartRegisters_264c(int inputValue)
7  {
8    int previousLcrValue; // r3
9
10   UART_UART_SRR_REG = 7;
11   previousLcrValue = UART_UART_LCR_REG;
12   UART_UART_LCR_REG = previousLcrValue | 0x80;
13   UART_UART_DLF_REG = inputValue;
14   UART_UART_RBR_THR_DLL_REG = BYTE1(inputValue);
15   UART_UART_IER_DLH_REG = BYTE2(inputValue);
16   UART_UART_LCR_REG = 3;
17   UART_UART_IIR_FCR_REG = 7;
18   UART_UART_IER_DLH_REG = BYTE2(inputValue) & 0xFE;
19   return inputValue;
20 }
```

Pseudocode-A

```
1  int __fastcall sub_264C(int result)
2  {
3    int v1; // r3
4
5    UART_UART_SRR_REG = 7;
6    v1 = UART_UART_LCR_REG;
7    UART_UART_LCR_REG = v1 | 0x80;
8    UART_UART_DLF_REG = (unsigned __int8)result;
9    UART_UART_RBR_THR_DLL_REG = BYTE1(result);
10   UART_UART_IER_DLH_REG = BYTE2(result);
11   UART_UART_LCR_REG = 3;
12   UART_UART_IIR_FCR_REG = 7;
13   UART_UART_IER_DLH_REG = BYTE2(result) & 0xFE;
14   return result;
15 }
```

```
000026A6 sub_264C:14 (26A6) (Synchronized with IDA View-A, IDA View-B)
```

aiDAPal Results

aiDAPal Function Name

☑ configureUartRegisters

aiDAPal Comment

☑ This function configures various UART registers with the provided input value. It sets specific bits in the Software Reset Register, Line Control Register, Divisor Latch Fraction Register, Receive Buffer Register Threshold, Interrupt Enable Register, and Interrupt Identification Register/FIFO Control Register based on the input value.

aiDAPal Variables

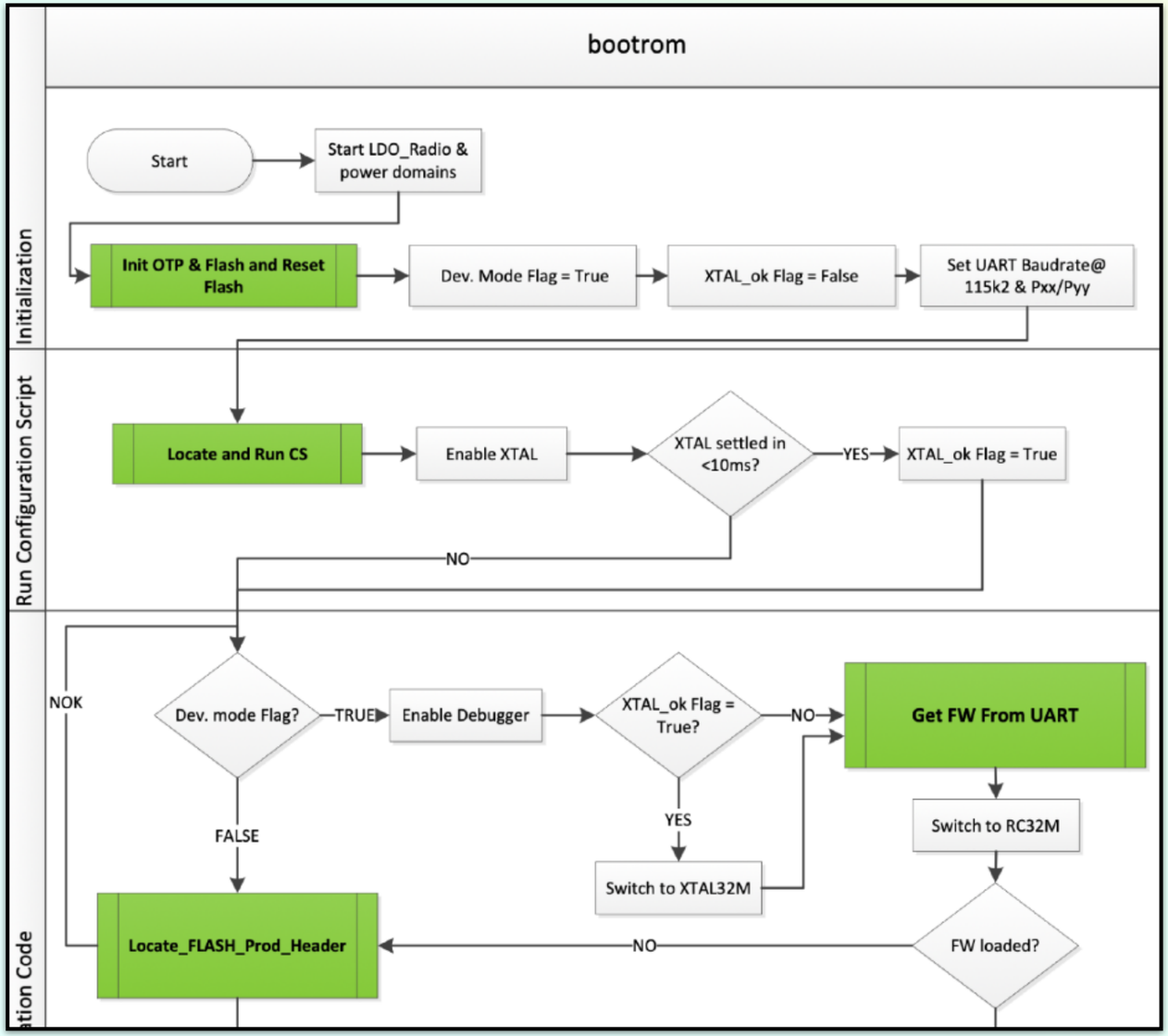
☑ result inputValue ☑ v1 previousLcrValue

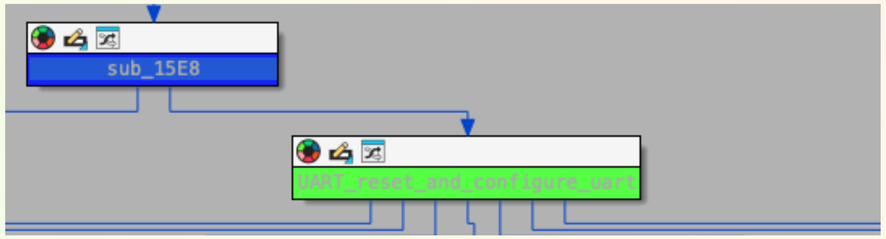
Accept

Cancel

https://github.com/atredispartners/aidapal

NAVIGATION SYSTEM





REN_da1469x_3v3_DST_20220421.pdf