MAINLINE

1. Set up stacks for supervisor mode and IRQ mode
2. Initialize GPIO1 clock (Clock Module base: =0x44E0_0000)
    a. Write #0x02 to CM_PER_GPIO1_CLKCTRL (offset: 0xAC)
3. Initialize UART2 clock
    a. Write #0x02 to CM_PER_UART2_CLKCTRL (offset: #0x70)
4. Set up GPIO1_31 for falling edge interrupt (GPIO1 base: =0x4804_C000)
    a. RMW GPIO1_FALLINGDETECT register
        i. Read GPIO1_FALLINGDETECT(offset: #0x14C)
        ii. Modify (set bit 31) #0x8000_0000
        iii. Write back
    b. Enable button interrupt for bit 31 of GPIO1
        i. Write #0x8000_0000 to GPIO1_IRQSTATUS_SET_0 (offset: #0x34)
5. Initialize INTC for GPIO1_31, UART2, and Timer7 (INTC base: =0x4820_0000)
    a. Reset INTC
        i. Write #0x2 to INTC config reg (offset: #0x10)
    b. Unmask INTC INT 98 (98 - 32 - 32 - 32 = bit 2) (GPIOINT1A)
        i. Write #0x04 to INTC_MIR_CLEAR3 (offset: #0xE8)
    c. Unmask INTC INT 74 (74 - 32 - 32 = bit 10) (UART2)
        i. Write #0x400 to INTC_MIR_CLEAR2 (offset: #0xC8)
6. Map UART2
    a. Read CONF_SPI0_D0 (Address: =0x44E1_0954)
    b. Modify (set bit 4 and 0) #0x11 for MODE1 (pin21 of P9)
    c. Write back
7. Initialize UART2 Baud rate, etc. (UART2 base: =0x4802_4000)
    a. Switch to mode A and set 8-bit data format
        i. Write #0x83 to line control register UART_LCR (offset: 0x0C)
    b. Enable 9.6kbps Baud rate
        i. Write #0x01 to divisor latch high DLH (offset: 0x04)
        ii. Write #0x38 to divisor latch low DLL (offset: 0x00)
    c. Enable 16x UART mode
        i. Write #0x00 to MDR1 (offset: 0x20)
    d. Switch back to operational mode
        i. Write #0x03 to UART_LCR (offset: #0x70)
    e. Turn off FIFO
        i. Write #0x00 to FIFO control register (offset: #0x8)
8. Enable IRQ interrupt
    a. Clear bit 7 in CPSR
9. Wait loop

INT_DIRECTOR

1. Save registers
2. Check if interrupt from UART2
    a. Check bit 10 in INTC_PENDING_IRQ2 (Address: =0x4820_00D8)

```
        b. If bit 10 = 0, go check button
        c. If bit 10 = 1, check bit 0 of IIR_UART (Address: =0x4802_4008)
            i. If bit 0 = 0, go to TALKER_SVC
            ii. If bit 0 = 1, go to RETURN_SVC
    3. Check if interrupt from button (INTC base: =0x4820_0000)
        a. Check bit 2 in INTC_PENDING_IRQ3 (offset: #0xF8)
        b. If bit 2 = 0, go to RETURN_SVC
        c. If bit 2 = 1, check bit 31 of GPIO1_IRQSTATUS_0 (Address: =0x4804_C02C)
            i. If bit 31 = 1, go to BUTTON_SVC
            ii. If bit 31 = 0, go to RETURN_SVC

BUTTON_SVC
    1. Turn off GPIO1_31 interrupt
        a. Write 0x8000_0000 to GPIO1_IRQSTATUS_0 (Address: =0x4804_C02C)
    2. Enable INTC for new interrupt aka turn off NEWIRQA bit
        a. Write #0x1 to INTC_CONTROL (Address: =0x4820_0048)
    3. Enable UART2 interrupt signals
        a. Write #0x02 to IER_UART (Address: =0x4802_4004)
    4. Restore registers and return to wait loop

TALKER_SVC
    1. Write character to THR (Address: =0x4802_4000)
        a. Load byte, increment counter
        b. If counter = 0, disable UART2 interrupt signal
            i. Write #0x0 to IER_UART2 (Address: =0x4802_4004)
            ii. Restore registers and return to wait loop
    2. Go to RETURN_SVC

RETURN_SVC
    1. Enable new IRQ interrupt
        a. Write #0x1 to INTC_CONTROL (Address: =0x4820_0048)
        b. Restore registers and return to wait loop
```