UART Programming

Clock and Baud Rate Generation

1. UART Clock Configuration (UARTCC) Register
    1. Out of 32 bits, only lowest 4 bits are used to choose Clock Source (CS): 0x0 means System Clock, 0x5 PIOSC
2. UART Control (UARTCTL) Register
    1. Only 5th bit (HSE) is used to select different clock rate.
    2. HSE=1 – System Clock is divided by 8, HSE=0 – System Clock is divided by 16
3. UART Integer Baud Rate Divisor (UARTIBRD)
4. UART Fractional Baud Rate Divisor (UARTFBRD)

$$\text{Buad Rate} = \text{UARTSysClk} / (\text{BRDI} + \text{BRDF}) * \text{ClkDiv}$$

$$\text{UARTFBRD} = \text{Fraction} * 64 = 0.5$$

UART Module Control & Status Register

1. UART Control (UARTCTL)
2. UART Line Control Register (UARTLCHR)
3. UART Data Register (UARTDR)
4. UART Receive Status/Error Clear Register (UARTRSR/UARTECR)
5. UART Flag Register (UARTFR)
6. UART IrDA Low Power Register (UARTILPR)
7. UART 9-bit Self Address Register
8. UART 9-bit Self Address mask
9. UART Peripheral Properties Register
10. Transmit FIFO (TxFIFO) and Receive FIFO (RxFIFO)

Important points

- All bits are cleared on the system reset
- UARTCTL should not be modified when UART is enabled. Sequence to follow
  - Disable UART
  - Wait for the end of Tx or Rx
  - Flush FIFO by clearing FEN (4^{th} bit) in UARTLCRH (UART Line Control)
  - Modify UARTCTL
  - Enable UART

UART Line Control (UARTLCRH)

Used to configure serial data communication parameters such as data length, parity, and stop bit

- Write to UARTLCRH whenever Baud Rate parameters are modified

UART Receive Status/Error Clear Register

UARTRSR/UARTECR

- Out of 32 bit only last 8 bits are used
- Bit 1 – parity enabled
- Bit 2 – select 1 for odd parity
- Bit 3 – select 1 for Two Stop bits
- Bit 4 – Enable FIFO, becomes 1 byte holding register, if 0 is selected
- Bit 6:5 – WLEN 0x0: 5 bits, 0x2: 7bits
- Bit 7 – Stick Parity Select

UART DATA Register (UARTDR)

- Actually two data registers, one for transmit and another for receive
- Bridge between MCU and FIFOs to store data temporarily
  - Temporarily store data that MCU wants to transmit before writing into TxFIFO
  - Store data received from RxFIFO

UART Flag Register (UARTFR)

- 32 bit Flag register used to monitor status of UART
- TxFF, RxFF, and BUSY are cleared (set to ‘0’)
- TxFE and RxFE are set to ‘1’
- Bit 3 – indicates if UART is busy in transmitting data

UART Initialization and Configuration

1. Configure and initialize UART related ports
    1. Enable and clock the appropriate GPIO module
    2. Set AFSEL for the corresponding pin
    3. Configure Peripheral Mux Control (PMCx)
2. Initialize and configure clock source and baud rate
    1. Enable the clock for UART using RCGCUART register
    2. For 115,200 baud rate, 8 bit data length, one stop bit, no parity, no interrupt, FIFOs disabled
3. Initialize and configure UART module
    1. Disable the UART by clearing the UARTEN bit (bit 0) in UARTCTL
    2. Write UARTIBRD & UARTFBRD
    3. Write desired serial communication in UARTLCRH
    4. Setup 0x00 to the UARTCC to select System Clock
    5. Enable UART module 0 by setting the UARTEN bit in UARTCTL

```
BRD = 16,000,000/ (16*115,200)
    = 8.6806 ;
UARTFBDR = (0.6806*64 + 0.5) = 44
```