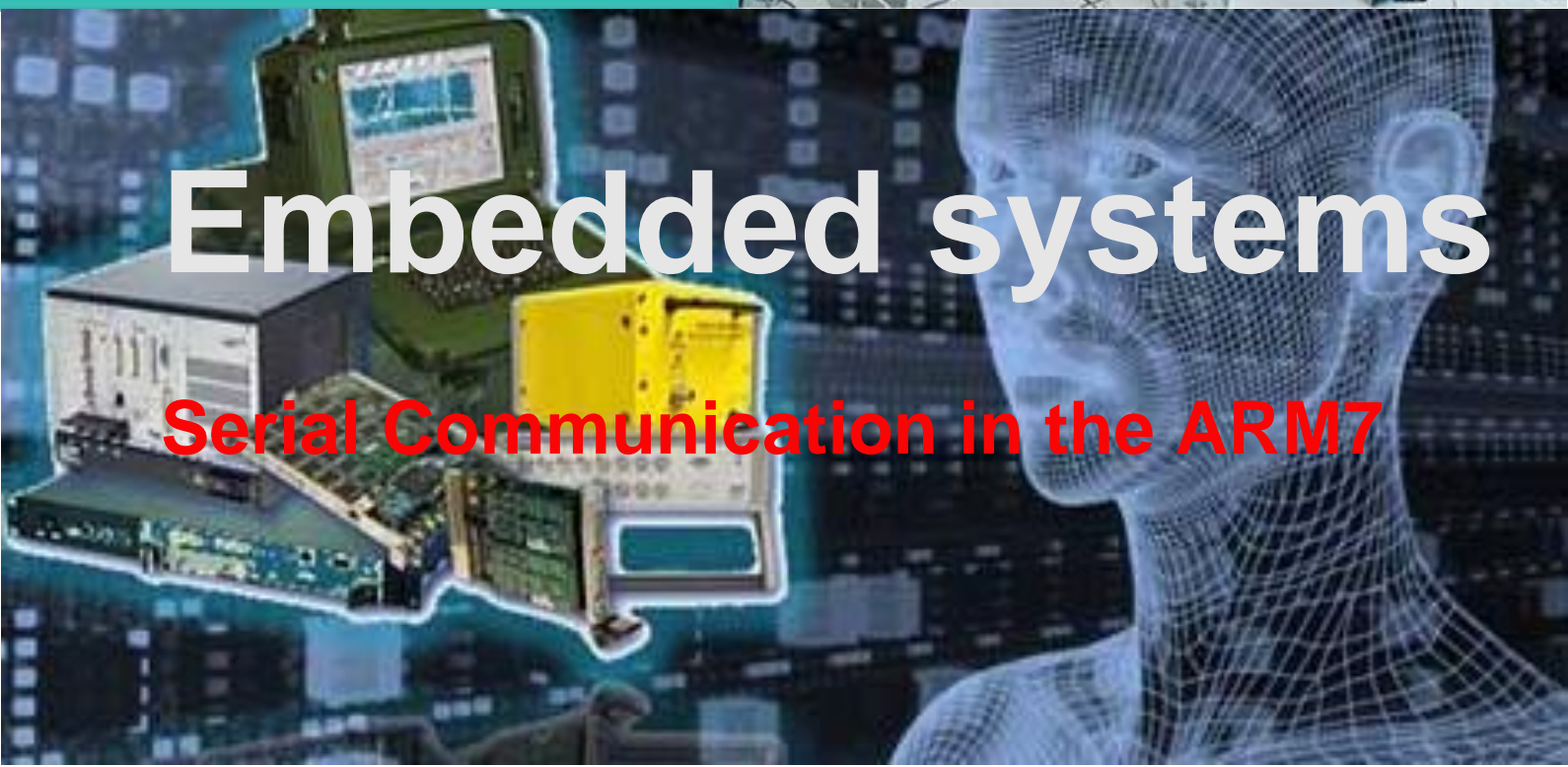




Andhra Pradesh State Skill Development Corporation



Embedded systems

Serial Communication in the ARM7



Serial Communication.

AIM: UART communication.

Software Required: Keil IDE and putty

Components Required:

1. System -1
2. ARM7 development board -1
3. USB to RS-232 cable -1
4. 9V Adapter -1
5. Connecting Wires -Required

Theory:

UART in LPC2148 ARM7 Microcontroller- Serial Communication

As we all know UART is widely used serial communication protocol in embedded system-based applications. Almost all microcontrollers have built-in on-chip UART support. LPC2148 ARM7 core supports two UART in it, UART0 and UART1. UART0 can be used as general purpose UART and also can support ISP Programming through it, whereas UART1 has additional modem support. Both have built in baud rate generator and 16-byte transmit and receive FIFOs. For UART0 the TxD Pin is at P0.0 and RxD Pin is at P0.1 and similarly for UART1 the TxD Pin is at P0.8 and RxD Pin is at P0.9.

Calculate Baud Rate for UART in LPC2148 ARM7

Baud Rate for UART in LPC2148 ARM7 Microcontroller can be calculated by given equation:

$$\text{BaudRate} = \frac{PCLK}{16 \times (256 \times U0DLM + U0DLL) \times \left(1 + \frac{DIVADDVAL}{MULVAL}\right)}$$

PCLK: Peripheral Clock Frequency (In MHz)

U0DLM, U0DLL: These are standard UART0 baud rate generator divider registers

MULVAL, DIVADDVAL: These registers are fraction generator values. They must meet following condition $0 < MULVAL, DIVADDVAL \leq 15$ with $MULVAL=0$ treated as $MULVAL=1$

Example Project:

Let's look at real world example where we will send string from LPC2148. This message will be received and displayed on terminal emulator or serial console which configured to appropriate COM Port of PC. I believe now you're familiar with how UART works. In our program we will use following configuration to establish proper communication:

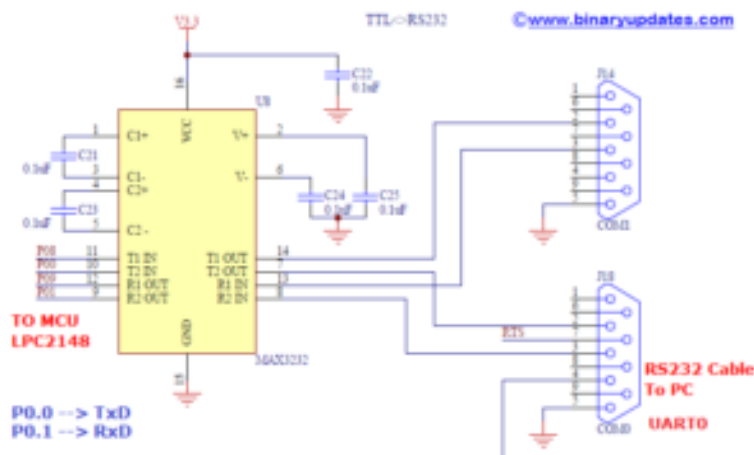
- **Baud Rate = 9600 baud** (with PCLK=60Mhz)
- **Data Length = 8 bits**
- **No Parity Bit**
- **and 1 Stop Bit**

To work out on this project we need following things to be setup:

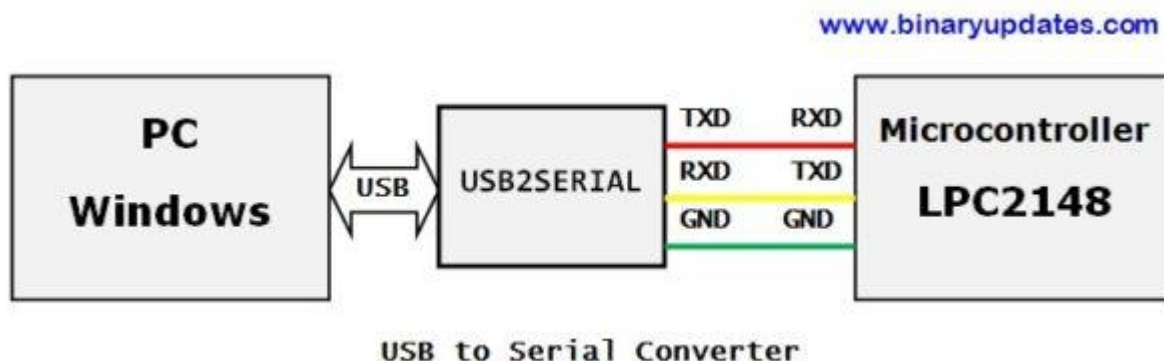
Software Requirements: Install Keil uVision4, Flash Magic, PuTTY (terminal emulator or serial console)

Hardware Requirements: LPC2148 Development Board, RS232 Cable (USB Serial Converter), Power Adapter (9V-500mA).

Circuit Diagram: UART in LPC2148 ARM7 Microcontroller



Circuit Diagram-UART in LPC2148 ARM Microcontroller





Procedure:-

1. Open Keil μ Vision from the icon created on your desktop.
2. Create a new project on Kiel with the appropriate name and destination.
3. Take a new text file and write the code in a text editor.
4. Save the text file with ".c" extension.
5. add ".c" file to source group and check errors and warnings.
6. change the target options and create a hex file.
7. Now open flash magic to burn hex file into the development board.
8. Connect the hardware circuit and Connect your development Board to the USB port of your computer.
9. In the flash-magic window select the target device, serial port, board rate, and hex file.
10. Click on the start button to burn the hex file to the development board.
11. after uploading press the reset button and check the output.

Source Code:

```
#include<lpc214x.h>

// user function declaration
void uart_init(void);
void uart_data(unsigned char);
char uart_rx(void);
void uart_enter(void);
void uart_string(unsigned char []);
void uart_rx_string(void);
void uart_num(unsigned int ) ;
// main function
int main(){
    char b;
    uart_init();
    while(1){
        b=uart_rx();
        uart_data(b);
        //uart_enter();
        //uart_string(" APSSDC");
        uart_num(9876) ;
        uart_enter();
    }
}

// user function definition
void uart_init(void){
    PINSEL0|=0x00000005;
    U0LCR=0x83;
    U0DLL=97;
    U0LCR=0x03;
```



```
}

void uart_data(unsigned char ch){
    U0THR=ch;
    while(!((U0LSR)&(0x20)));
}

char uart_rx(void){
    unsigned char a;
    while(!((U0LSR)&(0x01)) );
    a=U0RBR;
    return(a);
}

void uart_enter(void){
    uart_data(13);
    uart_data(10);
}

void uart_string(unsigned char ch[]){
    int x;
    for(x=0;ch[x]!='\0';x++){
        uart_data(ch[x]);
    }
}

void uart_num(unsigned int m){
    unsigned int d=0,e[10];
    while(m>0){
        e[d]=m%10;
        m=m/10;
        d++;
    }
    while(d>0){
        d--;
        uart_data(e[d]+48);
    }
}
```

Result: -When a character is given to the Virtual terminal then the led will turn ON and for another character led will turn OFF.