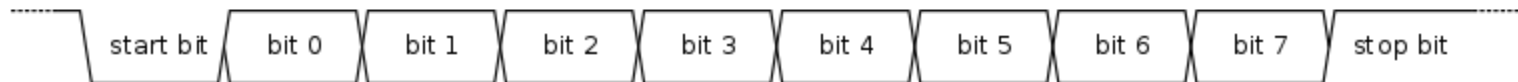


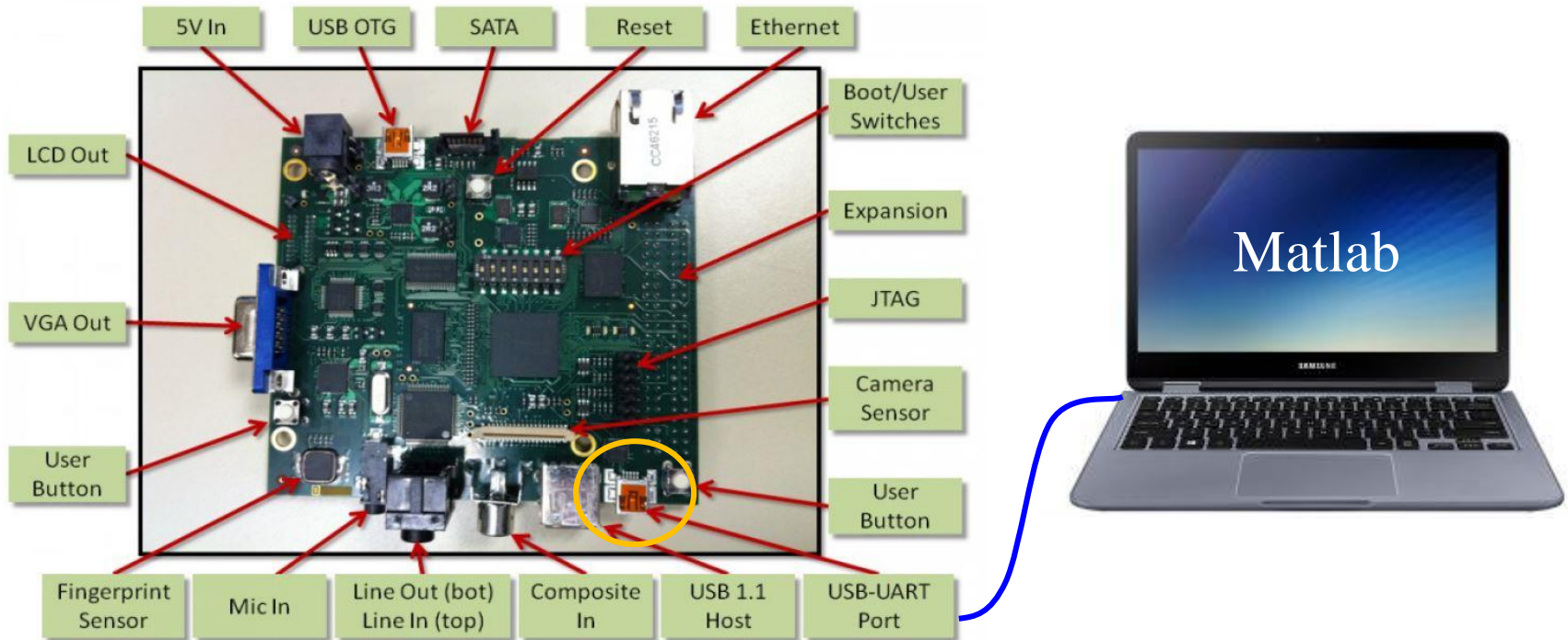
UART

- A universal asynchronous receiver/transmitter (UART), is a computer hardware device for **asynchronous serial communication** in which the data format and transmission speeds are configurable.



UART - PC

- USB-UART Port



Baudrate

- The primary factor affecting UART throughput is the **baud rate**. Table 1 shows the throughput achieved at various baud rates, set by the divisor and over-sampling mode. Figure 1 shows that the throughput scales linearly with baud rate.

$$\text{Divisor} = \frac{\text{UART Input Clock Frequency}}{\text{Desired Baud Rate} \times 16}$$

$$\text{Divisor} = \frac{\text{UART Input Clock Frequency}}{\text{Desired Baud Rate} \times 13}$$

Table 2-1 Baud Rate Examples for 150-MHz UART Input Clock and 16× Oversampling Mode

Baud Rate	Divisor Value	Actual Baud Rate	Error (%)
2400	3906	2400.154	0.01
4800	1953	4800.372	0.01
9600	977	9595.701	−0.04
19200	488	19211.066	0.06
38400	244	38422.131	0.06
56000	167	56137.725	0.25
128000	73	129807.7	0.33
3000000	3	3125000	4.00

Table 2-2 Baud Rate Examples for 150-MHz UART Input Clock and 13× Oversampling Mode

Baud Rate	Divisor Value	Actual Baud Rate	Error (%)
2400	4808	2399	−0.01
4800	2404	4799.646	−0.01
9600	1202	9599.386	−0.01
19200	601	19198.771	−0.01
38400	300	38461.538	0.16
56000	206	56011.949	0.02
128000	90	128205.128	0.16
3000000	4	2884615.385	−4.00

Figure 1. Baudrate supported on LCDK

UART Communication

- Communication between PC and LCDK
- Why? – Control/monitor LCDK using Matlab
- UART functions in LCDK side
 - **void Init_UART2(Uint32);** // Initialize UART Baudrate
 - **void Write_UART2(Uint8);** // Send int data to PC
 - **void Puts_UART2(char *);** // Send char data to PC
 - **Uint8 Read_UART2();** // Receive data from PC
 - **Uint8 IsDataReady_UART2();** // Check receive buffer
 - **Uint8 IsTxReady_UART2();** // Check transmit buffer



- **UART Receive and Send example**

```
#include "DSP_Config.h"
```

```
int main()
```

```
{  
    // initialize DSP board  
    DSP_Init();  
    // call StartUp for application specific code defined in each application directory  
    StartUp();  
    Init_UART2(115200);           // Set a baudrate  
    while(1) {  
        if(IsDataReady_UART2()){ // Check if UART receive data  
            Write_UART2(Read_UART2()); // Send the received data to PC  
        }  
    }  
}
```

MATLAB functions

- Create an Object

```
delete(instrfindall);          % delete previous connection
serialPort = 'COM3';          % set a serial port (UART)
serialObject = serial(serialPort, 'BaudRate', 115200);
% define an object
fopen(serialObject);          % open the UART connection
```

- Send data

```
fwrite(serialObject, senddata);
```

- Receive data (datatype: int16, uint16, uint8,...)

```
receiveddata = fread(serialObject, buffersize, 'datatype')
```

- Receive data (char*)

```
fscanf(serialObject)
```

- Close the Object

```
fclose(serialObject);
```

Send char to PC

Matlab

```
clear all; close all; clc;
delete(instrfindall); % delete all connection
% initialize COM port and baudrate
s = serial('COM6', 'BaudRate', 115200);
% open a connection
fopen(s);
% read data from buffer
fscanf(s)
% close the connection
fclose(s);
```

CCS main.c

```
char *hello = {"Hello World."}; // data to send

int main()
{
    DSP_Init();

    Init_UART2(115200); // initialize UART baudrate as 115200

    Puts_UART2(hello); // send the data
}
```

Send uint8 to PC

Matlab

CCS main.c

```
uint8_t UARTbuffer[256];

int main()
{
    DSP_Init();
    Init_UART2(115200);

    for(mm=0; mm<256; mm++){
        while(IsTxReady_UART2()==0){
            ; // Wait until Tx buffer is ready
        }
        // Send the received data to PC
        Write_UART2(Uartbuffer[mm]);
        wait(10000); // delay the next transmission
    }
}
```

```
clear all; close all; clc;
%% Create the serial object
delete(instrfindall);
serialPort = 'COM6';
serialObject = serial(serialPort, 'BaudRate',115200);
fopen(serialObject);

sample = fread(serialObject,256,'uint8');
sam = typecast(uint8(sample),'int8');

fclose(serialObject);
delete(serialObject);
clear serialObject;
```


Send uint16 to PC

Matlab

CCS main.c

```
volatile union {
    Uint16 ui16;
    Uint8 ui8[2];
} UARTout;

int main()
{
    DSP_Init();
    Init_UART2(115200);
    for(mm=0; mm<256; mm++){
        UARTout.ui16 = (Uint16)UARTbuffer[mm];
        for(uu=0; uu<2; uu++){
            while(IsTxReady_UART2()==0){
                ;
            }
            Write_UART2(UARTout.ui8[uu]); // Send the received data to PC
            wait(10000);
        }
    }
}
```

```
clear all; close all; clc;
%% Create the serial object
delete(instrfindall);
serialPort = 'COM6';
serialObject = serial(serialPort, 'BaudRate',115200);
fopen(serialObject);

sample = fread(serialObject,256,'uint16');
sam = typecast(uint16(sample),'int16');

fclose(serialObject);
delete(serialObject);
clear serialObject;
```