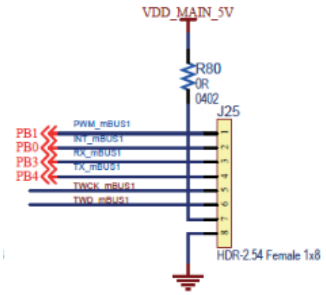
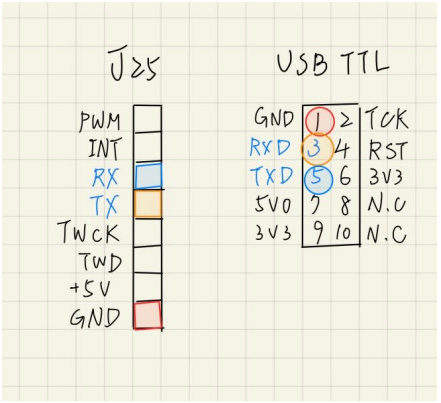
- Summary:
  [UART hardware communication protocol]

  - mikroBUS 1:



  - Env:



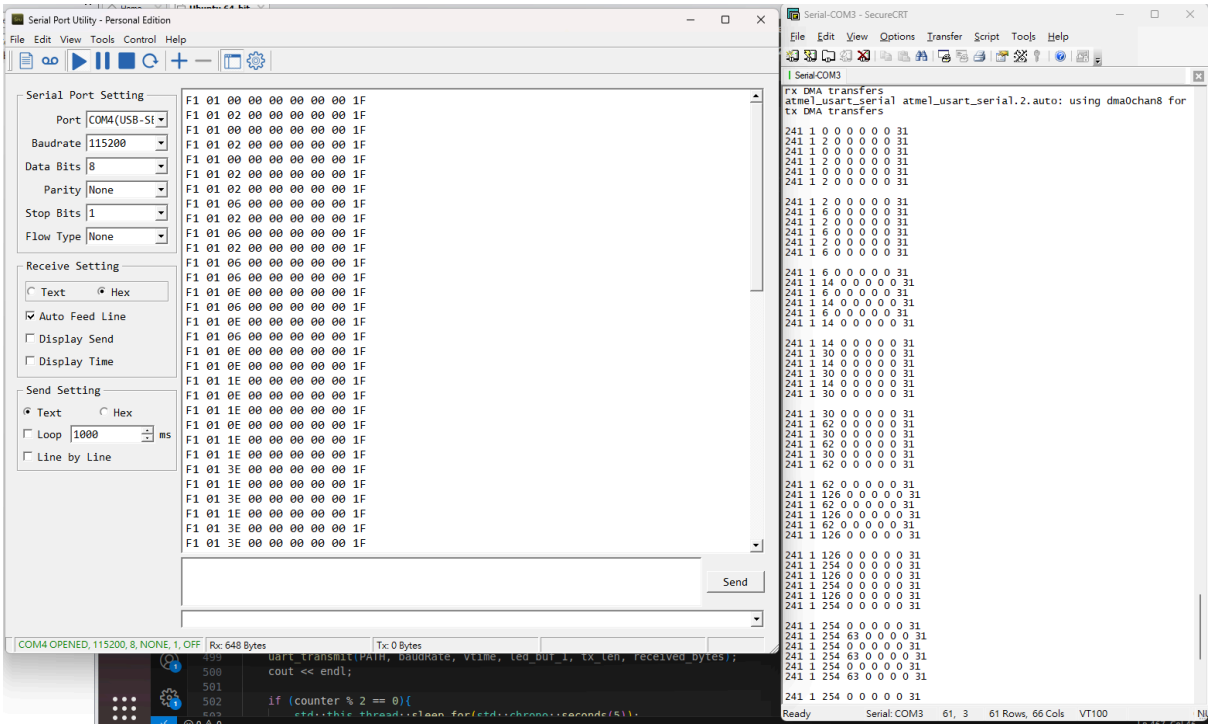
- Code:

```
#include <iostream>
#include <stdio.h>
#include <fcntl.h>
#include <unistd.h>
#include <termios.h>
#define PATH "/dev/ttyS1"
#define BUF_SIZE 255
using namespace std;

uint8_t* uart_transmit(const char* DEV_PATH, speed_t baudRate, int vtime, uint8_t tx_data[],
int tx_len, int& received_bytes){

    int fd;
    int ret;
    received_bytes = 0;
    uint8_t* rx_data = new uint8_t[BUF_SIZE];
    struct termios options;
    /* open uart */
```

```
  fd = open(DEV_PATH, O_RDWR | O_NOCTTY);
  if (fd < 0)
  {
        printf("ERROR open %s ret=%d\n\r", DEV_PATH, fd);
        return 0;
  }
  /* configure uart */
  tcgetattr(fd, &options);
  options.c_cflag &= ~PARENB;
  options.c_cflag &= ~CSTOPB;
  options.c_cflag &= ~CSIZE;
  options.c_cflag |= CS8;
  options.c_cc[VTIME] = vtime; // read timeout 10*100ms
  options.c_cc[VMIN] = 0;
  options.c_lflag &= ~(ICANON | ECHO | ECHOE | ISIG);
  options.c_oflag &= ~OPOST;
  options.c_iflag &= ~(ICRNL | IXON);
  cfsetispeed(&options, baudRate);
  cfsetospeed(&options, baudRate);
  options.c_cflag |= (CLOCAL | CREAD);
  tcflush(fd, TCIFLUSH);
  tcsetattr(fd, TCSANOW, &options);

  /* write uart */
  ret = write(fd, tx_data, tx_len);
  if (ret != tx_len) {
        printf("ERROR write ret=%d\n", ret);
  }
  while ((ret = read(fd, rx_data, BUF_SIZE - 1)) > 0) {
        received_bytes += ret;
  }

  return rx_data;


  close(fd);


}

Int main(){
      speed_t baudRate = B115200;
      int vtime = 10;
      int tx_len = 9;
      int received_bytes =0;
```

```
        uart_transmit(PATH, baudRate, vtime, buf, tx_len, received_bytes);

        return 0;
    }
```

- Result:



- Spec

通訊參數：

UART Asynchronous mode、 Band：9600、DATA1：8、Parity：no、Stop：1

設定數值 LED

	Start	Command	DATA1	DATA2	DATA3	END
數值	0xF1	0x00-0xFF	0x00-0xFF	0x00-0xFF	0x00-0xFF	0x1F