

Nextion Display Interface with TI microcontroller (TMS320F280025C)

Problem Statement:

As per your suggestion, I have checked program and this correct. But in standalone mode after turn OFF and then ON both Display and MCU communication is not working. The data is transmitted by the display but MCU not received. After disconnect the TX pin (MCU) and connect again communication start again without any problem.

Here I am detailing the program and waveform of the TX and RX pin.

As per your suggestion, I have checked the program and it is correct. However, in standalone mode, after turning OFF both the Display and MCU and then ON again, the communication is not working. The data is being transmitted by the display as shown below in waveform, but the MCU is not responding it. After disconnecting the TX pin (MCU) and reconnecting it, the communication starts again without any problem.

Here, I am detailing the program and the waveform of the TX and RX pin."

//The configuration of SCI is below:

```
void mySCI0_init(){
    SCI_clearInterruptStatus(mySCI0_BASE, SCI_INT_RXFF | SCI_INT_TXFF | SCI_INT_FE |
SCI_INT_OE | SCI_INT_PE | SCI_INT_RXERR | SCI_INT_RXRDY_BRKDT | SCI_INT_TXRDY);
    SCI_clearOverflowStatus(mySCI0_BASE);
    SCI_resetTxFIFO(mySCI0_BASE);
    SCI_resetRxFIFO(mySCI0_BASE);
    SCI_resetChannels(mySCI0_BASE);
    SCI_setConfig(mySCI0_BASE, DEVICE_LSPCLK_FREQ, mySCI0_BAUDRATE,
(SCI_CONFIG_WLEN_8|SCI_CONFIG_STOP_ONE|SCI_CONFIG_PAR_NONE));
    SCI_disableLoopback(mySCI0_BASE);
    SCI_performSoftwareReset(mySCI0_BASE);
    SCI_enableInterrupt(mySCI0_BASE, SCI_INT_RXFF | SCI_INT_TXFF);
    SCI_setFIFOInterruptLevel(mySCI0_BASE, SCI_FIFO_TX2, SCI_FIFO_RX2);
    SCI_enableFIFO(mySCI0_BASE);
    SCI_enableModule(mySCI0_BASE);
}
```

//Interrupt details:

```
interrupt void INT_mySCI0_RX_ISR(void)
{
    start_flag=3;
    //Parameters for transmit the VDC voltage to display
    tx_buffer[0] = 0x76; //(V)
    tx_buffer[1] = 0x6f; //(o)
    tx_buffer[2] = 0x6c; //(l)
    tx_buffer[3] = 0x74; //(t)
    tx_buffer[4] = 0x2E; //(.)
    tx_buffer[5] = 0x74; //(t)
    tx_buffer[6] = 0x78; //(x)
    tx_buffer[7] = 0x74; //(t)
    tx_buffer[8] = 0x3D; //(=)
    tx_buffer[9] = 0x22; //(")
    tx_buffer[10] = ((measured_voltage/100)+0x30); //(Voltage)
    tx_buffer[11] = (((measured_voltage%100)/10)+0x30); //(Voltage)
    tx_buffer[12] = ((measured_voltage%10)+0x30); //(Voltage)
```

```

        tx_buffer[13] = 0x22; //"
        tx_buffer[14] = 0xFF;
        tx_buffer[15] = 0xFF;
        tx_buffer[16] = 0xFF;
    //end transmit the VDC voltage to display

    Data_Available = SCI_getRxFIFOStatus(mySCI0_BASE);
    if (Data_Available > 0)
    {

        //receivedChar = SCI_readCharNonBlocking(mySCI0_BASE); //For 8-bit UART data transfer

        overFlowStatus= SCI_getOverflowStatus(SCIA_BASE);
        // Receive 17 Byte in Uart ISR
        SCI_readCharArray(SCIA_BASE, rx_buffer, 17);
        //If the First Byte is 0x65, then move data in an actual variable from the Receive Buffer
        if (rx_buffer[0] == 0x65)
        {
            voltage = (((rx_buffer[7] - 0x30) * 100) + ((rx_buffer[8] -
0x30) * 10) + (rx_buffer[9] - 0x30));
            frequency = (((rx_buffer[12] - 0x30) * 100) + ((rx_buffer[13] -
0x30) * 10) + (rx_buffer[14] - 0x30));
            duty=(((rx_buffer[15] - 0x30)*10) + (rx_buffer[16] - 0x30));
        }

    }

    SCI_writeCharArray(SCIA_BASE, tx_buffer, 17);
    measured_voltage = voltage;

    SCI_clearOverflowStatus(SCIA_BASE);
    SCI_clearInterruptStatus(SCIA_BASE, SCI_INT_RXFF);
    Interrupt_clearACKGroup(INTERRUPT_ACK_GROUP9);

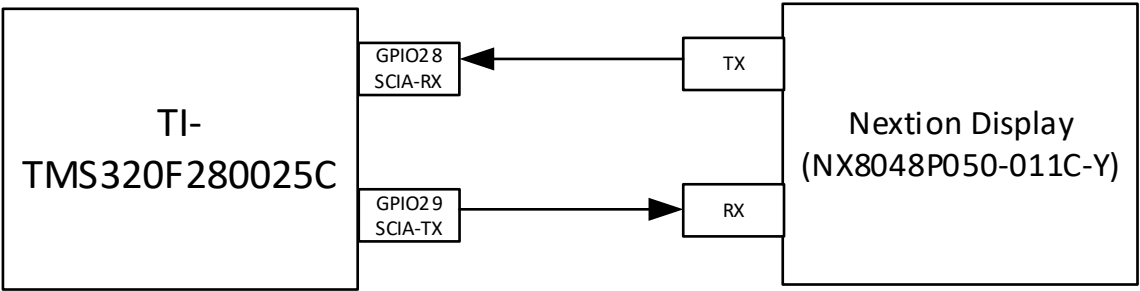
    rxStatus = SCI_getRxStatus(SCIA_BASE);
    if((rxStatus & SCI_RXSTATUS_ERROR) != 0)
    {
        //
        //If Execution stops here there is some error
        //Analyze SCI_getRxStatus() API return value
        //
        ESTOP0;
    }

    counter++;

interrupt void INT_mySCI0_TX_ISR(void)
{
    SCI_clearOverflowStatus(SCIA_BASE);
    SCI_clearInterruptStatus(SCIA_BASE, SCI_INT_TXFF);
    Interrupt_clearACKGroup(INTERRUPT_ACK_GROUP9);
    Interrupt_clearACKGroup(INTERRUPT_ACK_GROUP9);
}

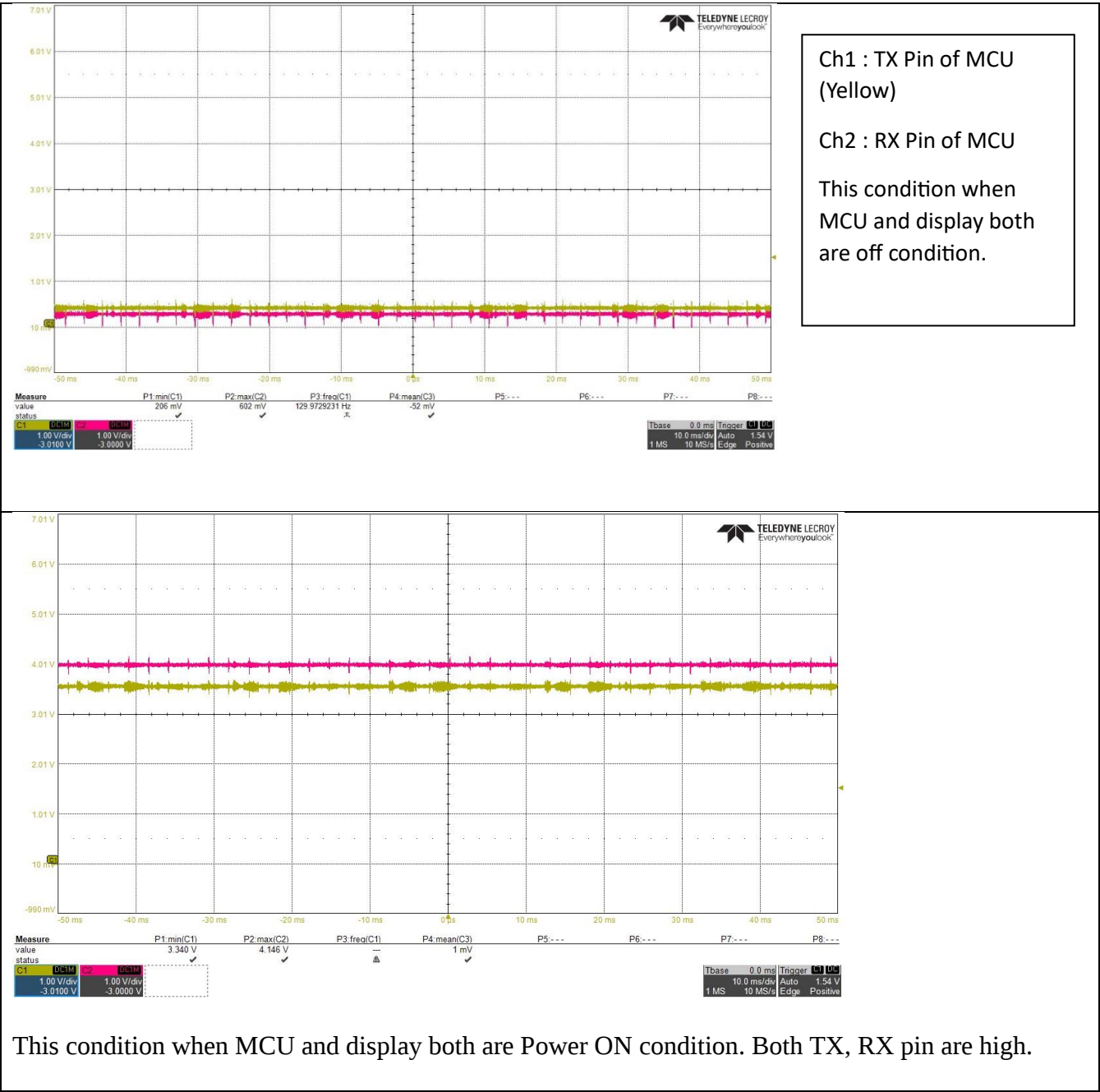
```

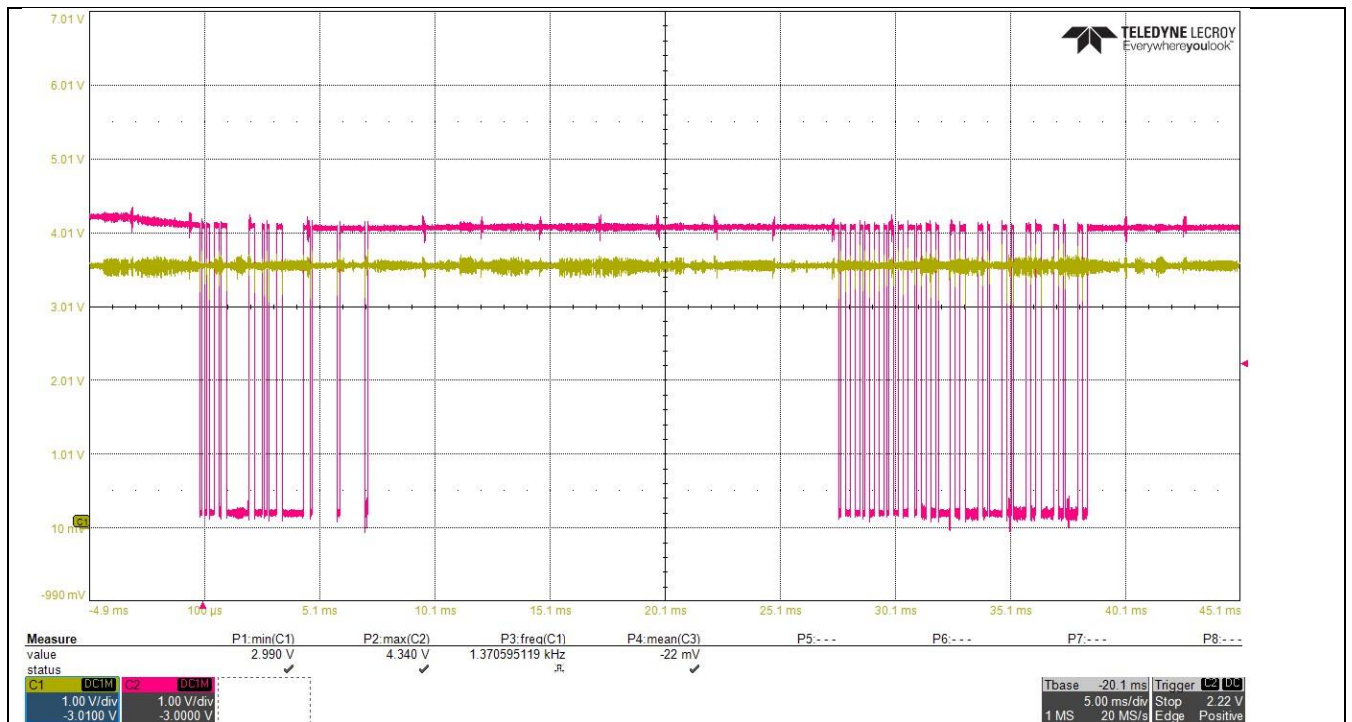
The connection diagram is below:



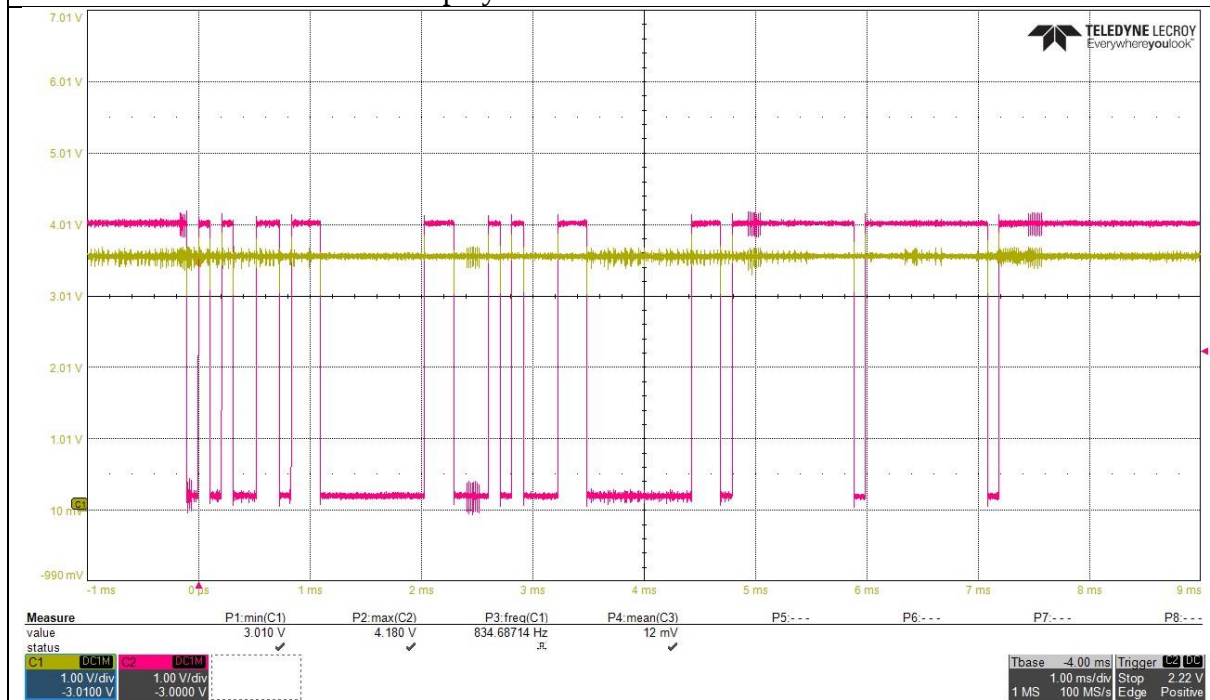
The waveform at different conditions:

MCU: TI-TMS320F280025C

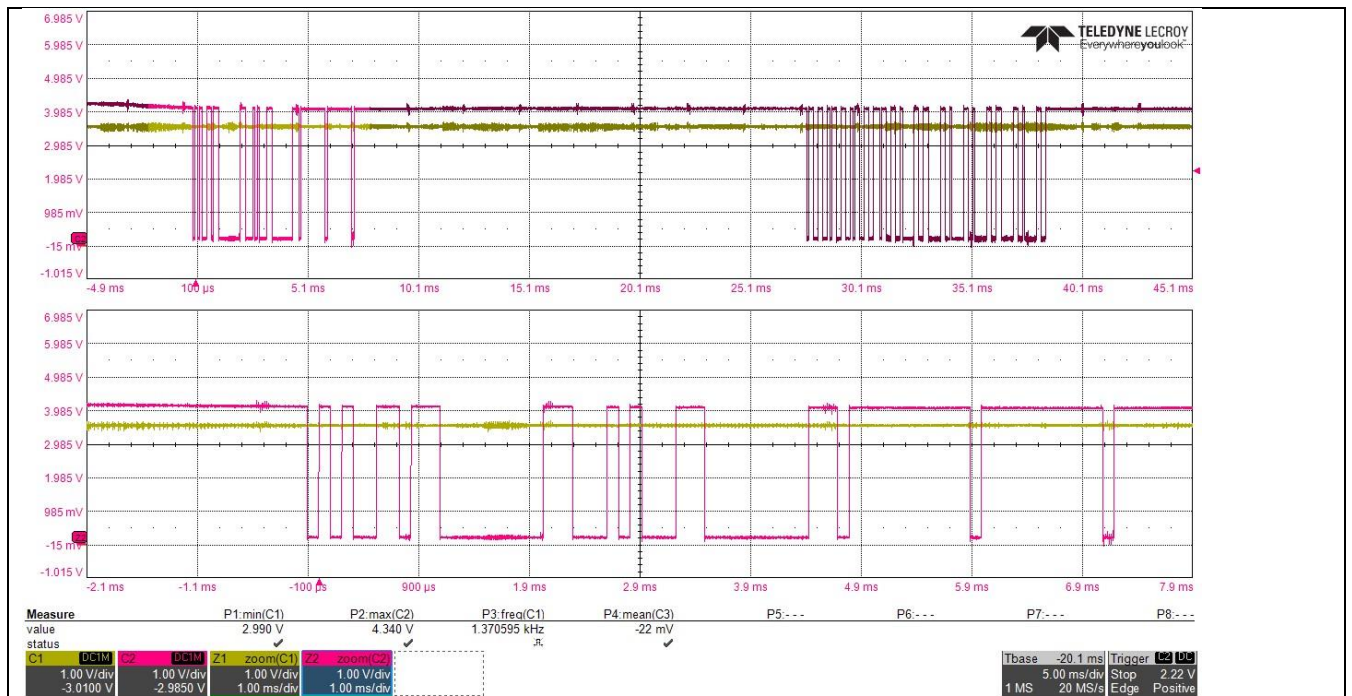




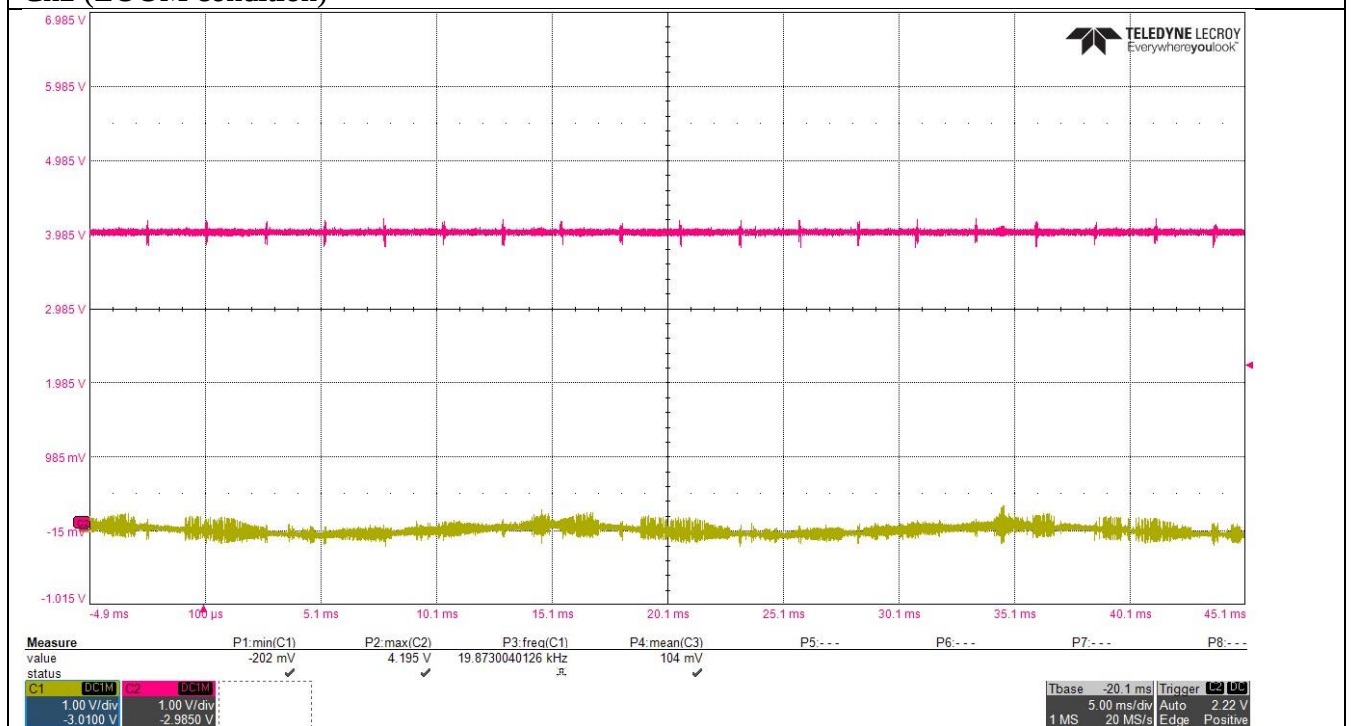
Now data is transmitted from display to MCU Ch2



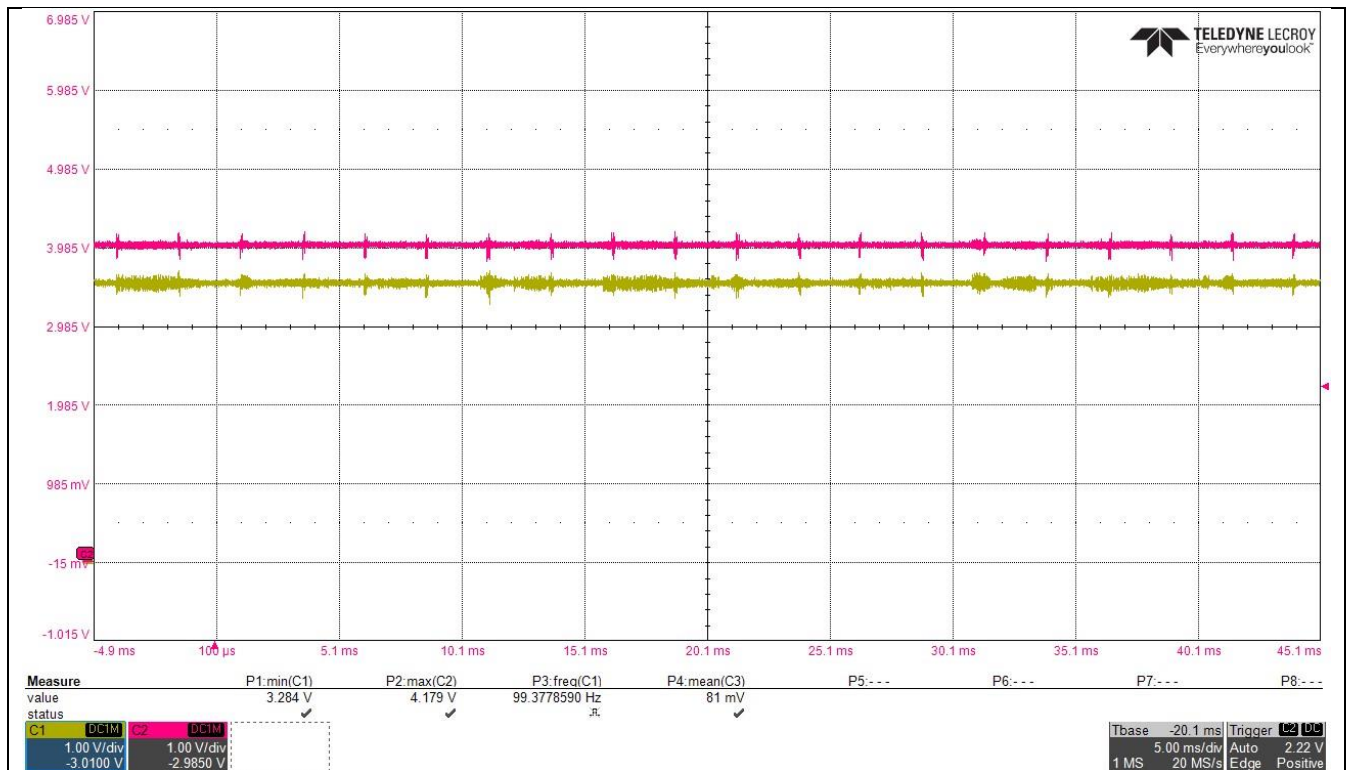
Now data is transmitted from display to MCU Ch2 (ZOOM condition)



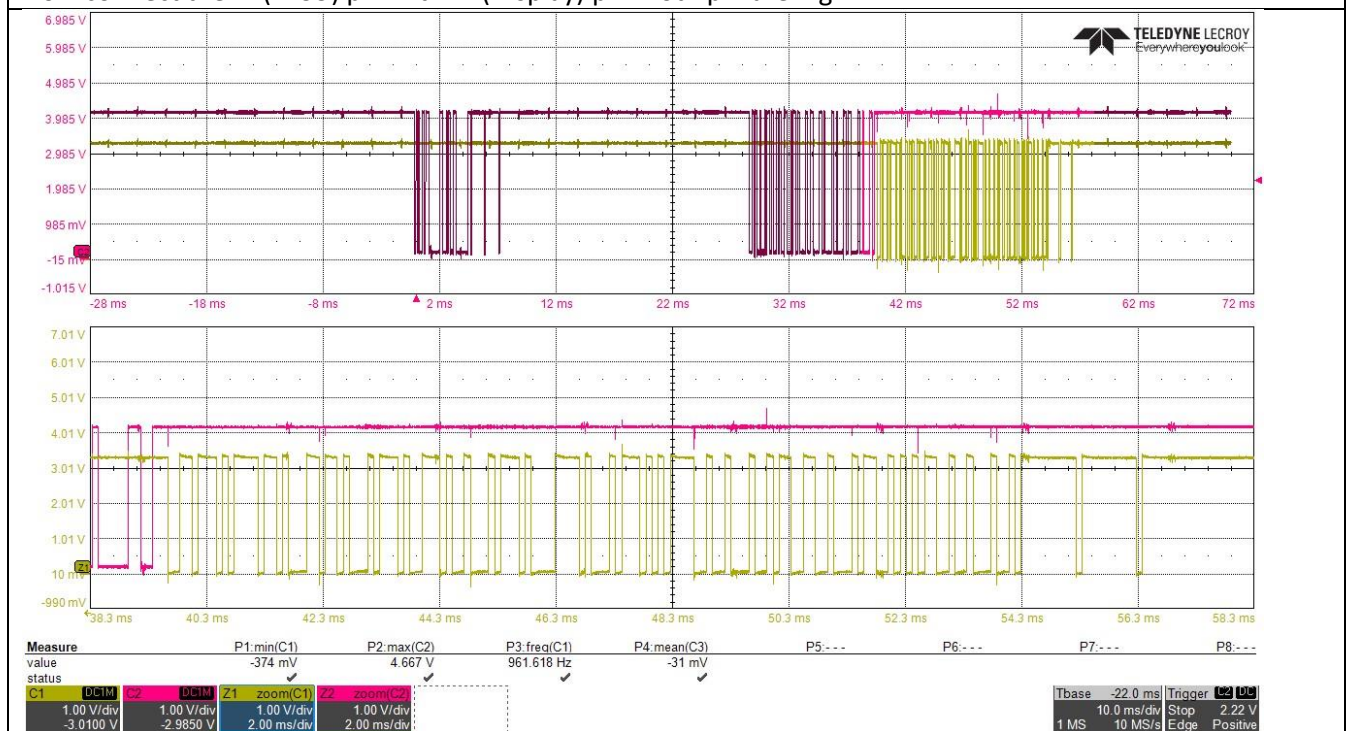
Now data is transmitted from display to MCU Ch2. Now data is transmitted from display to MCU Ch2 (ZOOM condition)



For start the communication I have reset/disconnect the TX(MCU) pin. The above condition when disconnect the TX(MCU) pin.



Now connect the TX(MCU) pin with RX(Display) pin. Both pin are high.



Now communication is start