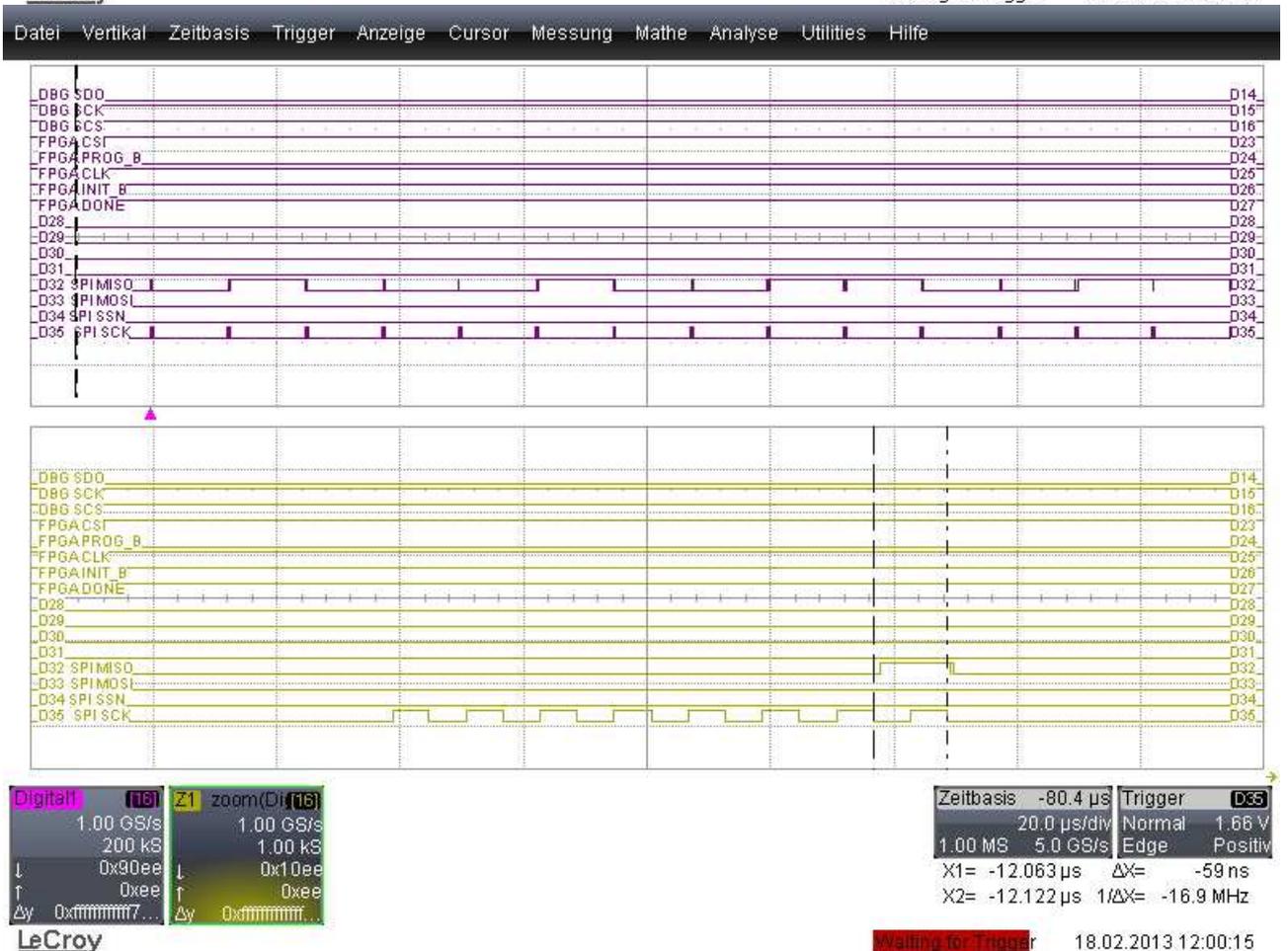
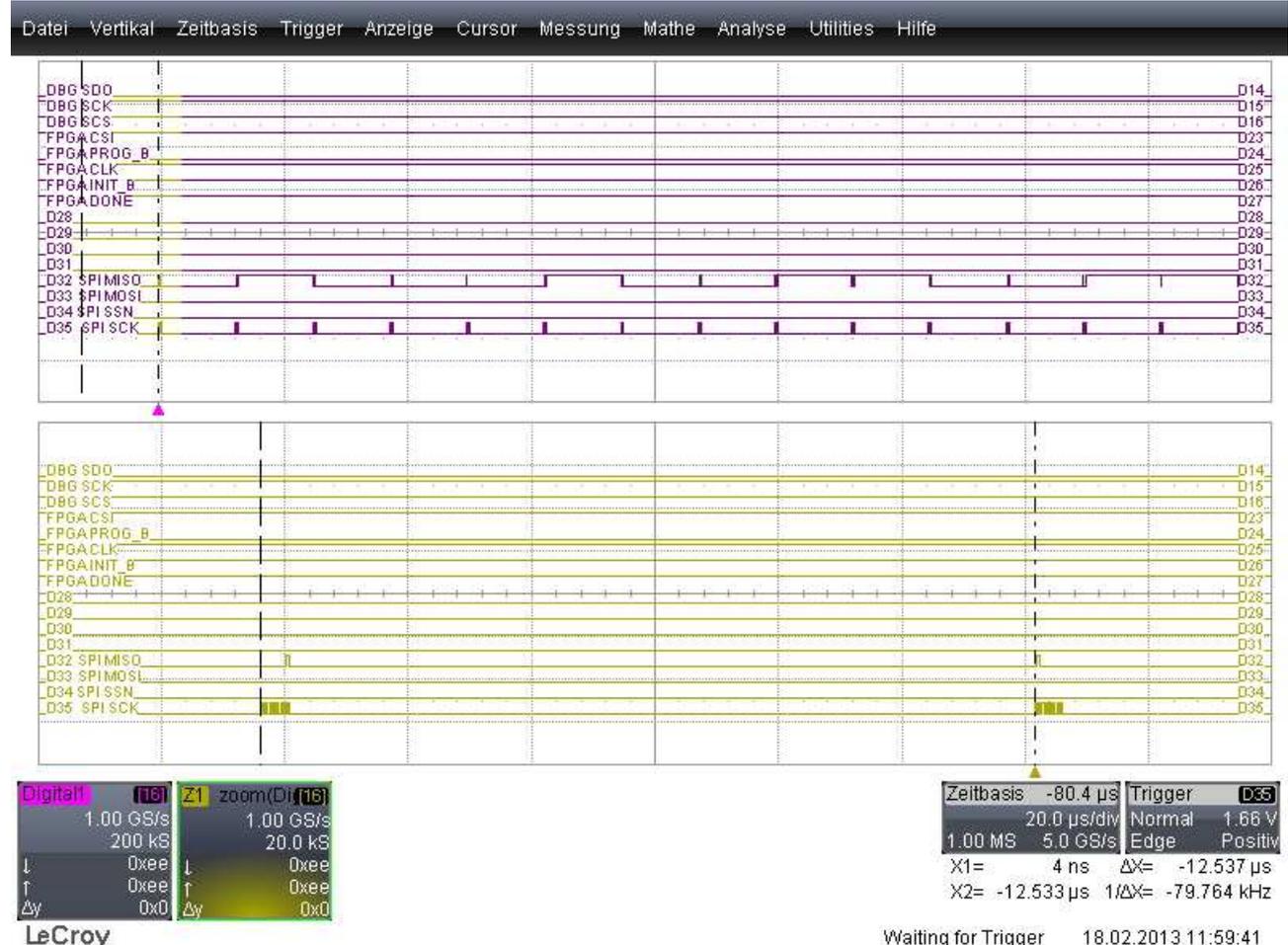


CYFX3 Bootloader API used SPI 8Bit, 33000000 Hz Clock
 CyFx3BootSpiReceiveWords (pData, 4096)



```

CyFx3BootErrorCode_t IS_SpiInit (void)
{
    CyFx3BootErrorCode_t nRet = CY_FX3_BOOT_SUCCESS;
    CyFx3BootSpiConfig_t cfg;
    CyFx3BootSpiInit ();

    cfg.isLsbFirst = CyFalse;      /* Data shift mode - CyFalse: MSB first,
                                   CyTrue: LSB first */
    cfg.cpol       = CyFalse;      /* Clock polarity - CyFalse(0): SCK idles low,
                                   CyTrue(1): SCK idles high */
    cfg.cpha       = CyFalse;      /* Clock phase - CyFalse(0): Slave samples at
                                   idle-active edge, CyTrue(1): Slave samples
                                   at active-idle edge */
    cfg.ssnPol     = CyFalse;      /* Polarity of SSN line. CyFalse (0): SSN is
                                   active low, CyTrue (1): SSN is active high. */
    cfg.ssnCtrl    = CY_FX3_BOOT_SPI_SSN_CTRL_FW; /* SSN control */
    cfg.leadTime   = CY_FX3_BOOT_SPI_SSN_LAG_LEAD_ONE_CLK; /* Time between SSN's assertion and first SCLK's
                                   edge. This is at the beginning of a
                                   transfer and is valid only for hardware
                                   controlled SSN. Zero lead time is not
                                   supported. */
    cfg.lagTime    = CY_FX3_BOOT_SPI_SSN_LAG_LEAD_ONE_CLK; /* Time between the last SCK edge to SSN's
                                   de-assertion. This is at the end of a
                                   transfer and is valid only for hardware
                                   controlled SSN. For CPHA = 1, lag time
                                   cannot be zero. */

    cfg.clock      = 33000000;     /* SPI clock frequency in Hz. */
    cfg.wordLen    = 8;           /* Word length in bits. Valid values are 4 - 32. */
    nRet = CyFx3BootSpiSetConfig (&cfg);

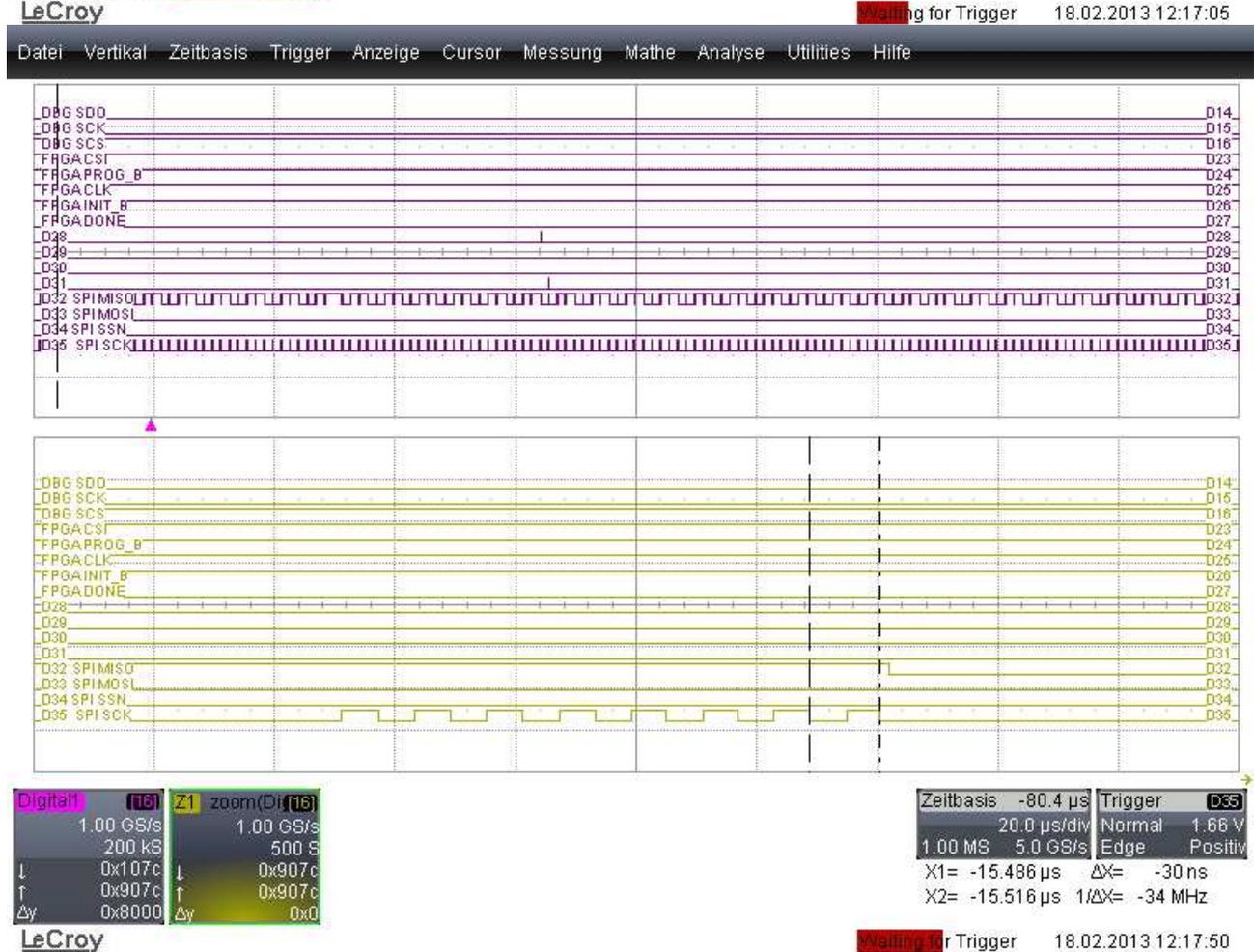
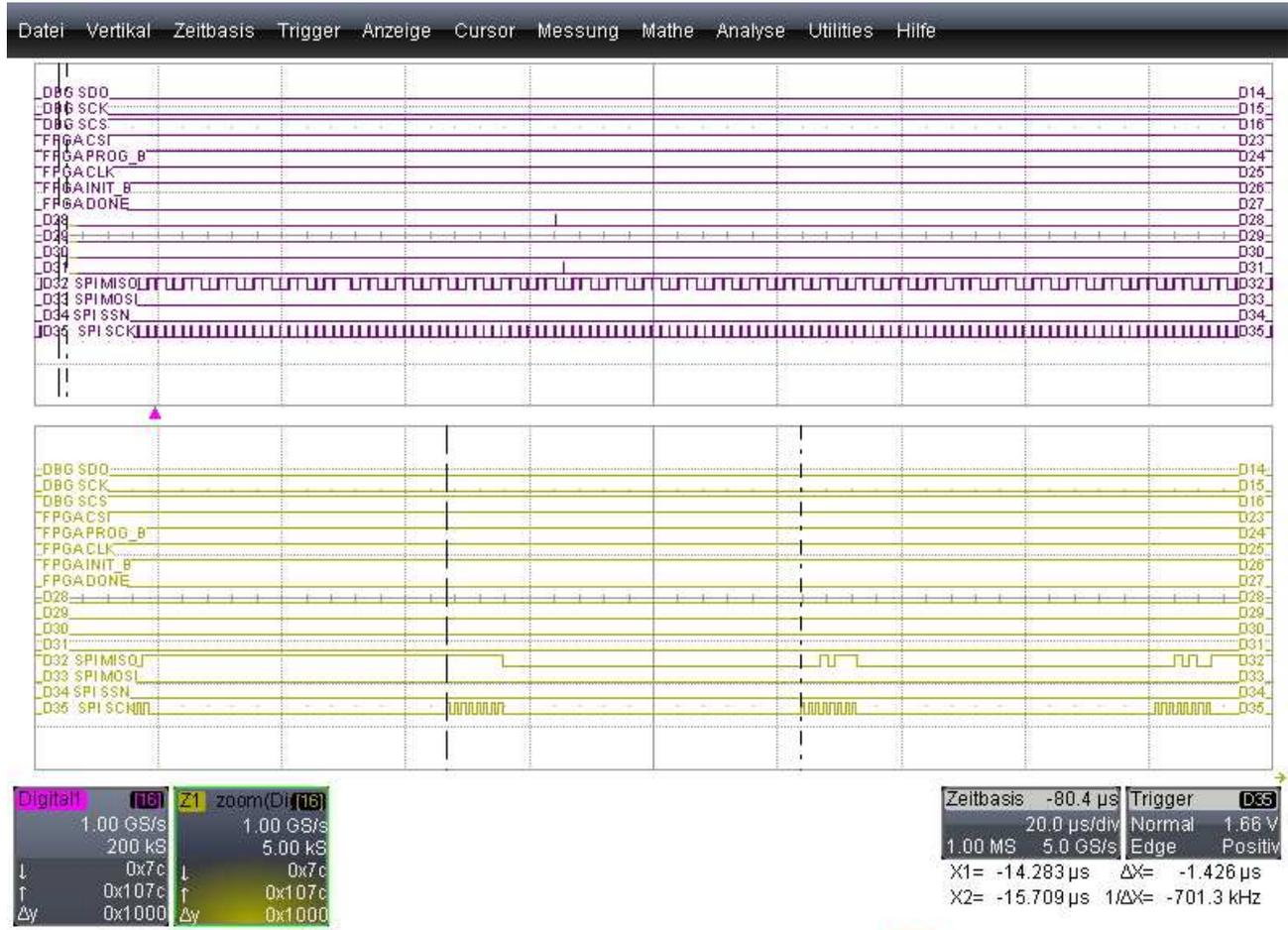
    IS_SpiSetSsnLine (IS_TRUE);

    return nRet;
}

```

CY FX3 SDK API used SPI 8Bit, 33000000 Hz Clock

CyU3PspiReceiveWords (pData, 4096)



```

IS_ERROR_CODE IS_SpiInit (IS_U32 u32SpiClock)
{
    IS_ERROR_CODE nRet = CY_U3P_SUCCESS;
    CyU3PSpiConfig_t cfg;
    nRet = CyU3PSpiInit ();

    cfg.isLsbFirst = CyFalse;      /* Data shift mode - CyFalse: MSB first,
                                   CyTrue: LSB first */
    cfg.cpol       = CyFalse;      /* Clock polarity - CyFalse(0): SCK idles low,
                                   CyTrue(1): SCK idles high */
    cfg.cpha       = CyFalse;      /* Clock phase - CyFalse(0): Slave samples at
                                   idle-active edge, CyTrue(1): Slave samples
                                   at active-idle edge */
    cfg.ssnPol     = CyFalse;      /* Polarity of SSN line. CyFalse (0): SSN is
                                   active low, CyTrue (1): SSN is active high. */
    cfg.ssnCtrl    = CY_U3P_SPI_SSN_CTRL_FW; /* SSN control */
    cfg.leadTime   = CY_U3P_SPI_SSN_LAG_LEAD_ONE_CLK; /* Time between SSN's assertion and first SCLK's
                                   edge. This is at the beginning of a
                                   transfer and is valid only for hardware
                                   controlled SSN. Zero lead time is not
                                   supported. */
    cfg.lagTime    = CY_U3P_SPI_SSN_LAG_LEAD_ONE_CLK; /* Time between the last SCK edge to SSN's
                                   de-assertion. This is at the end of a
                                   transfer and is valid only for hardware
                                   controlled SSN. For CPHA = 1, lag time
                                   cannot be zero. */
    cfg.clock      = 33000000;     /* SPI clock frequency in Hz. */
    cfg.wordLen    = 8;           /* Word length in bits. Valid values are 4 - 32. */
    nRet = CyU3PSpiSetConfig (&cfg, IS_SpiCb);

    nRet = CyU3PSpiSetSsnLine (CyTrue);
    return nRet;
}

```