Software Requirements Specification (SRS)

For projects

1533 (FAST LINK)

Version: 1.0

RESTRICTION OF USE, DUPLICATION, OR DISCLOSURE OF PROPRIETARY INFORMATION

This document contains proprietary information that is the sole property of DSIT. The document is submitted to the recipient for his use only. By receiving this document, the recipient undertakes not to duplicate the document or to disclose in part, or in whole any of the information contained herein to any third party without receiving beforehand written permission from DSIT.

B.A. Microwaves LTD.

10 Marconi St. P.O.B. 25482, Haifa 3125401

Tel. +972-4-8202715 Fax. +972-4-8202716

bamw@bamicrowaves.co.il www.bamicrowaves.co.il

Table of Contents

1 Changes

Name	Change	Version	Date
Roee Zinoue	First Edition.	1.0	25/01/18
Roee Zinoue	Second Edition.	1.1	14/02/18

2 *Project 1533*:

2.1 Introduction:

This document describes the SW operation of the FAST LINK unit.

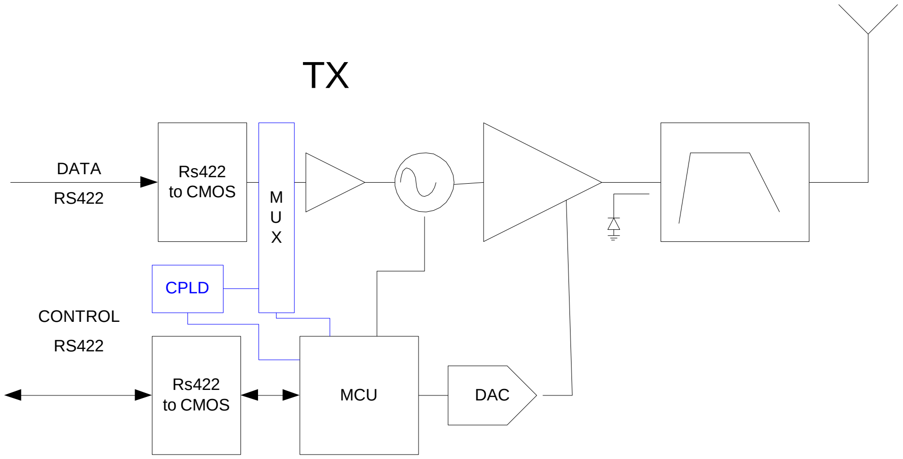
The system will consist two RF units:

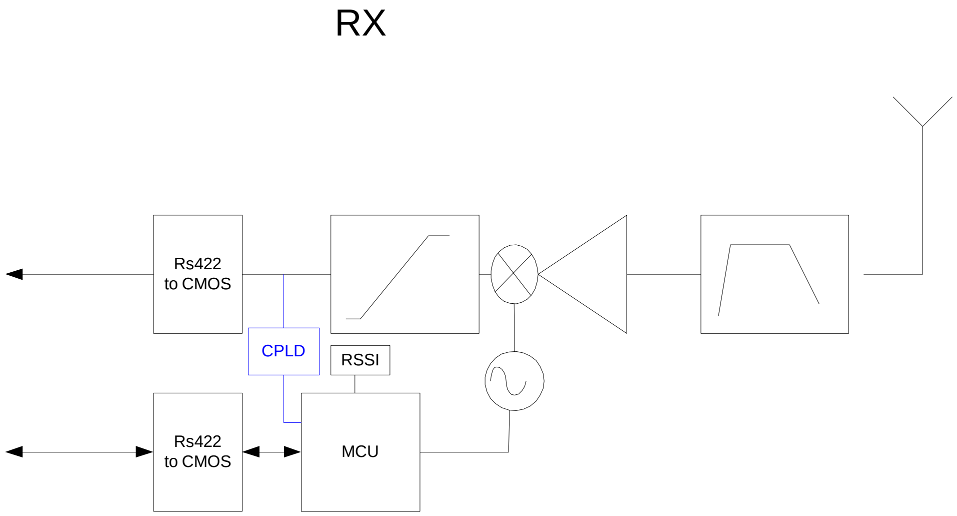
Transmission unit: will output RF signal from selected source.

Receiving unit: Will receive RF signal and export measure data to PC host application.

Via serial connection we will able to control state of the transmitting / receiving units. The system also have digital clock to indicate about the state of the system operation or system failures.

2.2 System block diagram





2.3 Objective

Main object of the MCU is to transmit RF signals and then measure data about the income RF signals states and export it to Host PC application.

Transmission unit functionalities:

a. The transmission unit will transmit RF signal from two sources:
   - External source via serial communication interface.
   - From on board CPLD unit during BIT mode.

   This desire source will be selected by command sanded from on board MCU unit.

b. The transmitted RF signal data specification describe at PLL configuration section.

c. The transmission unit will transmit one tone at same pattern at each transmission, this signal data rate will be set on 9Mbps.

d. The transmission unit under BIT mode will transmit RF signal for 2 seconds period time.

e. When the transmission unit will be under BIT mode, ACK message will be send via serial communication to PC host application.

f. The transmission unit after system restart will return to transmit RF signals from external source.

g. The transmission unit will measure using A2D the following data:
   - Forward power.
   - If the signal latched by the on board synthesizer.
   - Return power.

   Then this data will be send to PC host application at each user request.

h. The A2D sensors can be turn ON or OFF by user request via serial command.

Receiving unit functionalities:

a. The receiving unit will measure the following RF data using A2D unit:
   - RSSI.
   - Income signal power.

   Then this data will be send to PC host application at each user request.

b. The A2D sensors can be turn off by user request via serial command.

c. The income signal power will be used to insert the receiving unit to work under low power mode or high power mode if it reach or under certain threshold. The system will output the power mode via connected GPIO pin.

d. The connected synthesizer will calibrate according to current power mode.

e. The receiving unit will be able to work under stand-by mode (at this mode the system will not listen to any income RF signals).

MCU functionalities:

a. The MCU will always check the serial communication state (even at BIT mode) and will indicate the system user about any failures related via on board digital clock.
b. The serial communication interface between PC host application and on board MCU unit will based on RS-422.
c. The on board MCU unit also will start to collect samples data from on board ADC unit at the Transmission / Receiving units immediate after system initialize.
d. The A2D samples will be stored on the internal MCU Flash unit, then can be read at PC host application at user request (the data from flash will be read on packets, each packet size contain 32 samples).
e. To indicate the system user via on board digital clock about several system operation state.
f. To turn ON or OFF on board CPLD unit at the Transmission / Receiving units.
g. The communication interface between on board MCU and CPLD will be based on serial communication RS-422.

2.4 PLL configurations

The on board MCU need to configure on board PLL-Synthesizers to the following configurations:

Tx signals:
PLL parameters:
PLL type: ADF4113
RF reference to PLL = 10 MHz
PRESCALER: 64
Frequencies steps: 10 kHz
Registers values:
prog = 0b110010010000000010010011 = 0xC90093
prog = 0b000000000000111110100000 = 0x000FA0

Tx signals		
Description	Range	Step size
RF_{out} – Low	5.75 – 5.85 GHz	1 MHz
RF_{out} – High	5.15 – 5.25 GHz	1 MHz
$\mathrm{PLL}_{\text{receive signal}} = \frac{RF_{out}}{2}$		
PLL – receive signal (Low)	2.575-2.625 GHz	0.5 MHz
PLL – receive signal (High)	2.8625-2.9125 GHz	0.5 MHz

Rx signals:
PLL parameters:
PLL type: ADF4350
RF reference to PLL = 10 MHz
PRESCALER: 8
Frequencies steps: 10 kHz
IF FREQ=480MHz
Registers values:
R1 = 0x08008321 R2 = 0x18005EC2 R3 = 0x000004B3 R4 = 0x0085003C
R5= 0x00580005

Rx signals		
Description	Range	Step size
RF_{in} – Low	5.15-5.25 GHz	1 MHz
RF_{in} – High	5.75-5.85 GHz	1 MHz
$\mathrm{PLL}_{\text{receive signal}} = \frac{(RF_{in} - 480\ MHz)}{2}$		
PLL – receive signal (Low)	2.335-2.385 GHz	0.5 MHz
PLL – receive signal (High)	2.6225-2.6725 GHz	0.5 MHz

2.5 Errors indications and system failures treatment

Indications about system operation state and might come system failures will be describe at future document steps.

2.6 Synthesizer registers values

Note: Please check Analog devices datasheet to learn about the way of frequency in the sensitizer unit.

In order to configure the registers of both ADF4113 and ADF4350 we use Analog devices simulator (Int N-PLL software).

3 Host PC serial control protocol

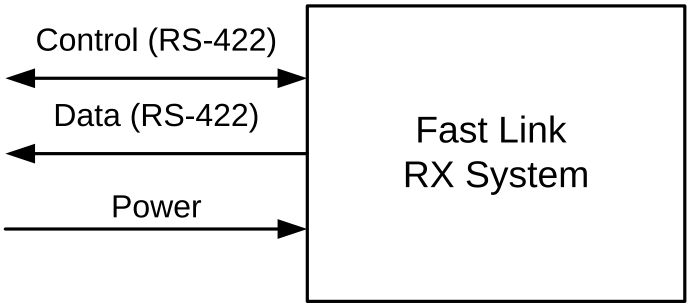
The on board MCU unit will be controller by serial communication channel (based on RS-422) and via hyper terminal application. The following table will configure that commands that the host can send to the MCU.

3.1 Serial communication configuration

a. Baud – rate: 115200 bps.
b. 8 bit data size.
c. 1 start bit, 1 stop bit.
d. No CTS, RTS.

3.2 Data / control directions

For both Rx and Tx the flow data / control direction can be describe as:



Frame description:
Each sending request command will begin with $ char delimiter and end with <cr> (carriage return).
On each request the MCU will send ACK response command that start with "OK <cr>$" and end with carriage return.

3.3 Host commands

TX commands				
#	Description	data size	Command	Example
1	Init TX system: init TX unit system, this command include carrier frequency that the system start to work with (Frequency will set in steps of 1 MHz and in allowed range).	4 bits	$TI XXXX <cr> Frequency in MHz XXXX	$TI 1230 <cr> Init Tx unit and set frequancy of 1230 MHz
2	Frequency change: set frequency of the TX unit (in steps of 1 MHz and in allowed range).	2 bytes	$TF XXXX <cr> XXXX = Frequency in MHz	$TF 1234 <cr> Set TX unit frequancy at 1234 MHz.
4	Get status: Get status from TX unit: ○ Synth (PLL). Lock indication. ○ Input power status. ○ FREQ=XXXX (value in MHz). ○ REV POWER =±XX (value in dB). ○ FWD POWER =+XX (value in dB). ○ TEMP=XX.X (value in Celsius degree). ○ TRANSMIT= ON/OFF. ○ LIGHT=ON/OFF. ○ UNIT ID=XXX. ○ UNIT DC=YYXX. ○ SW VERSION=<string>. ○ Input power status. ○ UNIT TYPE=X (type T or R). ○ BIT = ON/OFF. Each data value will be separate from each other with comma (",").	3 bytes	$TQ <cr>	When we receive: 1, 1, 4230, N32, 22, 38.2, 1, 1, 123, 1234, S4321, 0, 1, 1. It will indicate that the system set on: ○ Synth (PLL)= lock ○ input power status on ○ FREQ= 4230 MHZ ○ REV POWER=-32 dB. ○ FWD POWER=22 dBm ○ TEMP=38.2 c ○ TRANMSIT=ON ○ LIGHT=ON ○ UNIT ID=123 ○ UNIT DC=1234 ○ SW VERSION=S4321 ○ INPUT POWER=OFF ○ UNIT TYPE=T ○ BIT=ON
5	BIT mode: start / stop	4 bits	$TB 1 / 0 < cr> 1: ON 0: OFF Default: 0	$TB 1 <cr>
	Start / stop transmit: insert / stop TX unit to work at full operation mode.	4 bits	$TX 1 / 0 < cr> 1: ON 0: OFF Default: 0	$TX 1 <cr> Set TX unit on.

RX commands				
#	Description	Byte size	Command	Example
1	Init RX system: init RX unit system, this command include carrier frequency that the system start to work with. (Frequency will set in steps of 1 MHz and in allowed range).	4 bits	$RI XXXX <cr> XXXX = 4 digits Frequency in MHz	$RI 1234 <cr>
2	Start / stop receive: insert / stop RX unit to work at full operation mode.	4 bits	$RX 1 / 0 < cr> 1: ON 0: OFF Default: 0	$RX 1 <cr>
3	Get status: Get status from TX unit: ○ Synt (PLL). Lock indication. ○ Device temperature= XX.X (value in Celsius degree) ○ Input power status (ON, OFF) ○ FREQ=XXXX (value in MHz). ○ RSSI=± XX (value in dBm). ○ TEMP=XX.X (value in Celsius degree). ○ LIGHT=ON/OFF. ○ UNIT ID=XXX. ○ UNIT DC=YYXX. ○ SW VERSION=<string>. ○ UNIT TYPE=X (type T or R). ○ Compression point= XX (Value in dbm) ○ PASS / FAIL bit (PASS = number of income good signals reach predefine threshold).	3 bytes	$RQ<cr>	When we receive: 1, 38.2, 1, 1230, N32, 45.2, 1, 123, 1234, 000A, 0, 23, 1 It will indicate that the system set on: ○ Synth (PLL): lock. ○ Device temp: 38.2 c. ○ Input power: on. ○ FREQ: 1230 (MHz). ○ RSSI: -32 dBm. ○ TEMP: 45.2 c. ○ LIGHT: IN ○ UNIT ID: 123 ○ UNIT DC: 1234 ○ SW VERSION: 10. ○ UNIT TYPE: R. ○ Compression point: 23 dBm. ○ BIT mode PASS.
4	Compression point range: The command will set low or high compression point range.	4 bits	$RCR 1 / 0 <cr> 1: High. 0: Low Default: 0	$RCR 1 Set RX Compression point to work at high range.
	Compression point operation: The command will compression point in automatic or manual work mode.	4 bits	$RCP 1 / 0 <cr> 1: Automatic. 0: Manual. Default: 0	$RCP 1 Set RX Compression point at automatic work mode.
5	BIT mode: start / stop	4 bits	$RB 1 / 0 < cr> 1: ON 0: OFF Default: 0	$RB 1 <cr>

4 Software Update Session

The MFE software update component is a piece of software that is responsible for checking the integrity of the operational software, loading it and running it.
The software update component is also responsible for getting a new software image from the host, verify its integrity and saving it to internal flash.

To start the software update process, a power up should be performed with the PROG_EN discrete line set to ‘0’. When PROG_EN equals ‘0’ the software update process does not load the operational software from flash but instead waits for configuration data from the host.
The software update process will sample the PROG_EN discrete line for approx. TBD milliseconds. If while sampling the PROG_EN discrete its value does not equal ‘0’ the software update process will load the operational software (if it exists) and jump to it.

As in the operational application, the communication channel is an RS422 Half Duplex, 1Mb/s communication channel as described in section 5.1.
Also, the software update component also uses the RFFC1662 framing as described in section 5.3 and the message / response packet structure is the same.

To verify that the software update component has indeed loaded correctly, the host should request the version string from the MFE. The Version string form the MFE software update component is in the form of B.x where B denotes that this is a software update component version.

When starting a new flash programming operation, the software update process will start by erasing the flash sections affected by the data received, this step is destructive and data saved on that sector will be lost.

The software update component cannot update itself, in case that the software update component needs to be updated a physical access to the MFE will be needed (a PICit3 or similar debugger will be needed to program the MFE controller via ICSP).

Calibration parameters are not programmed via the software update process, and will not be effected by a software upgrade.

4.1 Software Update Messages

As stated above, the software update component uses the same communication channel, framing and message format as the operational software, this allows the host to use the same codebase for communicating with the MFE software update component .

The MFE software update component can receive the following messages:

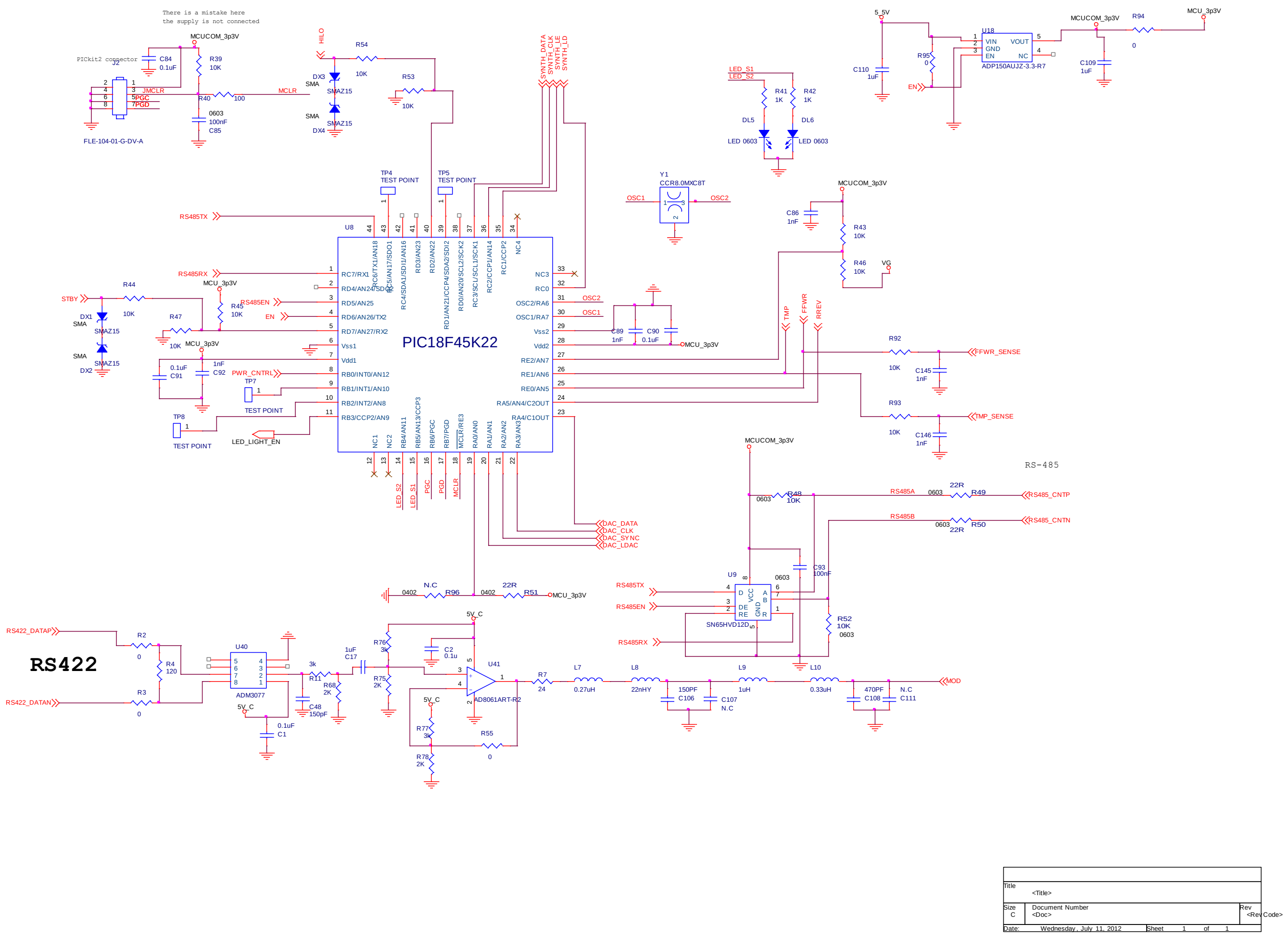
- Version Request.
- Set Data Line.
- Get Data Line.
- Finished Update Process.

The MFE will respond with the following responses:

- Version Response.
- Data Line Status.
- Data Line Data.
- Ack.

5 Hardware

5.1 MCU Schematic



5.2 Microcontroller

Recommended PIC	**PIC18F45K22 – 8 bit core**
Operating voltage	**3.3V**
Inputs (TTL / converter)	**TBD**
Outputs (TTL / converter)	**TBD**
POR	**Available**
Internal clock	**8MHz and up to 64Mhz**
Pin count	**40**

- Flash Memory
  - The PIC microcontroller has 32Kbytes of internal flash memory. This memory will be used for both the software update component software and for the operational software.
- RAM Memory
  - The PIC microcontroller has 1536 bytes of internal RAM, the software will use this memory for its stack & heap.
- Peripherals support:
  - Connectivity: 2-UART, 2-SPI, 2-I2C2-MSSP(SPI/I2C).
  - ADC: 28 ch, 10-bit.

5.3 GPIO and Analog Pin Assignments

MCU NET	I/O	SCHEMATIC NET	TX FUNCTION	RX FUNCTION	EXPLAIN
RC7/RX1	I	RS422RX	RS422_RX	RS422_RX	RS422 Control RX
RD4/AN24/SDO2			NC	NC	
RD5/AN25		RS422EN	NC	NC	
RD6/AN26/TX2	O	EN	POWER_EN	POWER_EN	
RD7/AN27/RX2	I	N16785247	STBY_IN	STBY_IN	When HI unit goes into STBy (EN=0) . If Low the according to Software word)
Vss1		GND_POWER			
Vdd1		MCU_3p3V			
RB0/INT0/AN12	O	PWR_CNTRL	PA_ON		operating when negative power supply is ok and when software control used
RB1/INT1/AN10		N16949169	NC	NC	
RB2/INT2/AN8		N16920411	NC	NC	
RB3/CCP2/AN9	O	LED_LIGHT_EN	MET_EN	MET_EN	when Hi enables EXTERNAL METER BACKLIGHT
NC1					
NC2					
RB4/AN11	O	LED_S2	LED_S2	LED_S2	Blinks when not locked , Lights up when PLL locked

RB5/AN13/CCP3	O	LED_S1	LED_S1	LED_S1	IN TX : on in full power . Blinks in low power mode , IN RX : Blinks when below RSSI THRESHHOLD, Lights above
RB6/PGC		PGC	PGC	PGC	
RB7/PGD		PGD	PGD	PGD	
M\C\L\R\/RE3		MCLR	MCLR	MCLR	
RA0/AN0		N16785239	NC	NC	
RA1/AN1	O	DAC_LDAC	DAC_LDAC	DAC_LDAC	Controls DAC AD5312ARMZ for :
RA2/AN2	O	DAC_SYNC	DAC_SYNC	DAC_SYNC	TX: PORT A : analog voltage according to measured output power
RA3/AN3	O	DAC_CLK	DAC_CLK	DAC_CLK	TX: PORT B: Set negative voltage according to software control . HI /LO only
RA4/C1OUT	O	DAC_DATA	DAC_DATA	DAC_DATA	RX: PORT A: analog voltage according to measured RSSI , outputs 1,11,111,1111 , PORT B: Not used
RA5/AN4/C2OUT	IAN	RREV	RREV	RSSI	TX: Analog input detecting REV power for reading through

					software .RX : RSSI
RE0/AN5	IAN	FFWR	FFWR	FFWR	TX : Analog input for FWD power to show on meter and in software.RX: not used
RE1/AN6	IAN	TMP	TMP	TMP	Reads Temperature parameters , reads in status .
RE2/AN7	IAN	N16785439	VG_MONITOR	NC	Monitors voltage , if Above ?Thereshhold , PA_ON is off
Vdd2		MCU_3p3V			
Vss2		GND_POWER			
OSC1/RA7		OSC1	OSC1	OSC1	8MHZ EXTERNAL CRYSTAL
OSC2/RA6		OSC2	OSC2	OSC2	
RC0	I	SYNTH_LD	SYNTH_LD	SYNTH_LD	IN TX: ADF 4113 , IN RX : ADF4350 (Need to decrease 480MHz IF)
NC3					
NC4					
RC1/CCP2	O	SYNTH_LE	SYNTH_LE	SYNTH_LE	
RC2/CCP1/AN14	O	SYNTH_CLK	SYNTH_CLK	SYNTH_CLK	
RC3/SCL/SCL1/SCK1	O	SYNTH_DATA	SYNTH_DATA	SYNTH_DATA	
RD0/AN20/SCL2/SCK2					
RD1/AN21/CCP4/SDA2/SDI2		N1678585842			
RD2/AN22	I	N16785806	HILO		TX ONLY, When "1" the

					tx power is set to low . When "0" Controlled by software
RD3/AN23					
RC4/SDA1/SDI1/AN16					
RC5/AN17/SDO1		N1678585824			
RC6/TX1/AN18	I	RS422TX	RS422_TX	RS422_TX	RS422 Control TX