ARA RF Advanced Front End (ARAFE): Slave Communication Document

This document details the communication protocol to the ARA RF Advanced Front End (ARAFE) slave (quad) modules. This communication protocol is typically performed by the ARAFE Master, in response to commands from software.

Overall Physical Layer

The ARAFE slave communication is performed over the +15V DC power supply line, using an on-off keying (OOK) signaling mechanism at approximately 8 MHz. "On" is interpreted as a digital 0, and "off" is interpreted as a digital 1. Characters are then sent as a typical UART, at 9600 bps, 8 bits, no parity, and 1 stop bit.

Packets from Master (bytes)

'!'	'M'	'!'	command	argument	0xFF

Responses from Slave (bytes)

'!'	'S'	'!'	ack	0xFF

Normal Commands (command 0-15)

Command structure is [type][flag][sub-command]=[7:6][5:4][3:0]

Available types are:

0x00: attenuator

0x40: sensor

0x80: info

0xC0: misc

Normal Commands (command 0-15)

Command	Description
Type 0x00	
00	Signal channel 0
01	Signal channel 1
02	Signal channel 2
03	Signal channel 3
04	Trigger channel 0
05	Trigger channel 1
06	Trigger channel 2
07	Trigger channel 3
08-3F	Repeat as above: 0x08 = sig chan 0, 0x09 = sig chan 1, 0x0E=0x3F=trig chan 3, etc.
Type 0x40	
40	Sensor 0
41	Sensor 1
42	Sensor 2
43	Sensor 3

44-47	Last two bits of last ADC
48-7F	Repeat as above
Type 0x80	
80-9F	Read/Write Device Info Eight bit argument: 1,0,0,W/R,A3,A2,A1,A0 A[3:0] = address of byte in the device_info W/R = 1 for write, 0 for read That is, 85 = read device info 5, 95 = write device info 5 (8->9 for switch from read to write)
A0-BF	Shadow of 80-9F (ie, A1 = 81)
Type 0xC0	
C0	Turn on/off 12V for channel 0
C1	Turn on/off 12V for channel 1
C2	Turn on/off 12V for channel 2
C3	Turn on/off 12V for channel 3
C4	Turn on/off 5V regulator
C5	Turn on/off 12V regulator
C6	Turn on/off 5V regulator (Shadow C4)
C7	Turn on/off 12V regulator (Shadow C5)
C8-FE	Repeat as above
0XFF	**Flash**

Device Info

Each ARAFE quad has a 16-byte 'device info' structure, stored in the microcontroller information memory B space (".infoB"), which contains the default settings for the attenuators as well as the default power on/off behavior.

These can be written to (for index < 12) and read from, and then stored permanently in flash so that initial power on behavior can be controlled.

Device Info Table

Index	Description
0	Default signal attenuator value for channel 0
1	Default signal attenuator value for channel 1
2	Default signal attenuator value for channel 2
3	Default signal attenuator value for channel 3
4	Default trigger attenuator value for channel 0
5	Default trigger attenuator value for channel 1
6	Default trigger attenuator value for channel 2
7	Default trigger attenuator value for channel 3
8	Default P2OUT value (see text)
9	Default P3OUT value (see text)
10	Serial number MSB
11	Serial number LSB
12	Major firmware version
13	Minor firmware version

14	0x12
15	0x34

Each of the 16 parts of the device info structure is 8-bit chunk of memory. Index 0 through 7 hold the default attenuator values. Eight bits is enough to accommodate all of the 0 to 127 attenuation settings (0 = 00000000 = 0x00 and 127 = 11111111 = 0x7F). Index 8 and 9 contain the default values for 16 of the microcontroller pins (1 = pin high on power on, 0 = pin low on power on). Index 10 holds the most significant bit of the serial number, and 11 holds the least significant bit of the serial number, meaning the serial number can be anything from 0 = 0x0000 to 65535 = 0xFFFF. Index 12 and 13 are reserved for the major and minor revisions of the firmware.

The P2OUT and P3OUT values control which pins are enabled / have voltage automatically at power on. Note that improperly programming these values may cause some problems with the default attenuator programming, but most likely not. (In detail, P2OUT/P3OUT can also set the LE pin for each attenuator, which is supposed to be pulsed high after the data is clocked in. The LE pin is set low at the beginning, so this should not cause problems, but there is some possibility).

The P2OUT and P3OUT registers control the following settings; these are taken from the schematic.

Pin	**Setting**		**Pin**	**Setting**
P2.0	SATT_CS3		P3.0	TATT_CS3
P2.1	12V_EN3		P3.1	EN_5V
P2.2	TATT_CS2		P3.2	SATT_CS2
P2.3	SATT_CS1		P3.3	12V_EN2
P2.4	TATT_CS1		P3.4	12V_EN1
P2.5	TATT_CS0		P3.5	SATT_CS0
P2.6	SPI_CLK_MSP430		P3.6	12V_EN0
P2.7	SPI_DATA_MSP430		P3.7	EN_12V

The default values during debugging are to set all attenuator values to zero, to disable all 12V lines, but to enable 5V rails. This means that P2OUT = [0,0,0,0,0,0,0,0] = 0x00, but P3OUT = [1,0,0,0,0,0,1,0] = 0x82. By default, the +5V and +12V regulators turn on automatically. **None** of the 12V 0/1/2/3 turn on to prevent DC voltage from being present on the RF input (via the bias tee) by default.

One useful command for the future will be to turn on the 5V and 12V regulators, and turn on all of the 12V outputs. This would be P2OUT = [0,0,0,0,0,0,1,0] = 0x02 and P3OUT = [1,1,0,1,1,0,1,0] = 0xDA.

Default Power Enable Locations

Voltage	**Bit to Set to Turn On By Default**
5V	P3OUT | 0x02
12V	P3OUT | 0x80
12V 0	P3OUT | 0x40
12V 1	P3OUT | 0x10
12V 2	P3OUT | 0x08
12V 3	P2OUT | 0x02

Writing/Reading/Flashing Device Info [Need to redo this]

To **write** to the device info, the command should be a bitwise OR of 0x80 with the device info address, and the argument should be the value. That is, to write 0x12 to device info 10, you would send command = 0x80 | (10), and argument = 0x12.

To **read** from the device info, the command should be a bitwise OR of 0x40 with the device info address. The argument is unused. The value read from the device info is contained in the *ack* byte.

To **flash** the device info, send command 0xFF.

Housekeeping [Need to redo this]

There are 3 sensors in the ARAFE slave, and 4 possible housekeeping commands. The sensors are *nominally* 10 bits, but only 1 byte is returned with each command. The most significant 8 bits are returned with each sensor read, and the low 2 bits can be obtained with a 0x44 through 0x47 command.

Command	Return 'ack' value
0x10	MSP430 temperature, top 8 bits
0x11	+5V current, top 8 bits
0x12	+12V current, top 8 bits
0x13	Fault detection (Fault if very low if 0x12 is high), top 8 bits
0x44->0x47	Low 2 bits of last conversion

The fault detection circuit works by comparing the value read with the +12V current.

- +12V current roughly 0, fault detection > 0: **no fault** and **no current**
- +12V current roughly 0, fault detection roughly 0: **fault** on one of the outputs
- +12V current above 0, fault detection above 0: **normal operation**

The remaining case (+12V current above 0, fault detection 0) should never occur.