

Scalable Signal Processor: Operating Manual

Table of Contents

1. Introduction
2. TCP/IP Configuration
3. Command Format
4. Data Format

Introduction

The Scalable Signal Processor is an Ethernet-based 3-channel 100 MHz analog to digital converter with onboard configurable signal processing. The A/D conversion is performed by 3 AD9446 16-bit converters. The sample frequency is defined by programming an AD9510 clock generator. The signal processing and Ethernet communication is handled by the onboard Virtex-4 SX35 FPGA using the embedded microblaze microprocessor. The configuration for the FPGA is stored in a CompactFlash memory, so it is relatively easy to reconfigure the device. TCP/IP configuration parameters are stored in EEPROM. Most boards can share the same program with only the unique configuration parameters stored in EEPROM.

Each channel consists of an A/D converter and a pre-adder block. Then the data from the three channels is passed to the scan processing block, which separates the input into scans, coadds multiple scans and transmits the combined scan data to the embedded microprocessor for broadcast to the host system.

Although designed to support conversions up to 100 MHz, this board is not actually intended to provide high data throughput to the host system. Instead, onboard processing is used to combine fast, repetitive signals to produce an average waveform at a slower net sample rate. There are two configuration parameters, NA and NC, which define how the averaging is performed. NA specifies how many sequential samples are averaged in the pre-adder stage before the data is passed on to the scan processor. NC specifies how many consecutive scans are coadded to produce the final output scan. Coadding of scans is the process of combining consecutive scans where the nth sample in the output is the sum of the nth sample from each of the input scans. I use the term 'averaging' loosely here. The SSP board always adds samples. Division by the product of NA and NC is performed by the host-based drivers to report true floating point average values.

TCP/IP Configuration

The SSP is configured using a web interface. The web configuration program is selected by setting the System ACE configuration address switches (SW1 switches 5, 6 & 7) to the OFF position. SW1 switches 1-4 determine the IP address used for the configuration web interface according to the following table. '0' here corresponds to the 'OFF' position of the corresponding switch.


SW1 Configuration Settings							
1	2	3	4	5	6	7	Address
0	0	0	0	0	0	0	10.0.0.200
0	0	0	1	0	0	0	10.0.0.201
0	0	1	0	0	0	0	10.0.0.202
0	0	1	1	0	0	0	192.168.0.200
0	1	0	0	0	0	0	192.168.0.201
1	1	1	1	0	0	0	Stored Address

SSP Configuration Utility - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://10.0.0.200/

Harvard Anderson Group
Data Acquisition Software



Scalable Signal Processor Configuration Utility

SSP Board Serial Number: ARP_SSP_001

Fabrication Date: 07 / 01 / 2008

Configuration Date: 11 / 26 / 2008

MAC Address: 00 : 0A : 35 : 55 : 55 : 40

IP Address: 10 . 0 . 0 . 220

IP Netmask: 255 . 255 . 255 . 0

IP Gateway: 10 . 0 . 0 . 1

Notes: Configuration from flash boot

Length: 64 Checksum: 17158 Version: 0

© 2008 President and Fellows of Harvard College

After setting the switches, provide either a power cycle or SysACE RESET to initialize the web server. You can then access the configuration utility at the specified address, e.g. **http://10.0.0.200/** for the configuration selected with all switches OFF.

The screen shot above shows how the configuration utility allows you to configure all the usual TCP/IP parameters as well as some basic bookkeeping information.

After the configuration has been successfully submitted and written to EEPROM, you can switch any of the switches 5, 6 or 7 to the ON position and perform a SysACE RESET or power cycle to reinitialize the board using the new parameters.

Note: There is no web server in the operational configuration. You can verify the network configuration by using the standard ping network utility. Beyond that, you will need to run one of the data acquisition programs to verify full operation.

Command Format

The SSP listens for commands on TCP Port 1500.

The port number is defined via the `SSP_SERVER_PORT` macro defined in `ssp_ad.h`, which is referenced by both the client and server programs. It is not currently a configurable parameter, so changing the port would require creating a new image and rebuilding the clients.

Commands are sent and acknowledged one at a time. There is currently no provision for sending multiple commands in a single TCP packet.

Command codes consist of two uppercase letters. All commands with numerical arguments start with the letter N and are followed with a colon and the number, optionally preceded by a minus sign where applicable. Each command should be terminated with a network-standard CRLF sequence.

EN	Enable: Start conversions
EX	Quit: Used only for debugging
DA	Disable: End conversions
NS:nnnn	Set Number of Samples: 1– <code>SSP_MAX_SAMPLES</code> (4096)
NA:nnnn	Set Number of Averages: 1– <code>SSP_MAX_PREADD</code> (256)
NC:nnnn	Set Number of Coadds: 1– <code>SSP_MAX_COADD</code> (16383)
NF:nn	Set Frequency Divisor: 1–32
NP:nnnnn	Set UDP Port for scan data
NE:n	Set channel enable register: 1–7, bit-mapped
NU:[-]nnnnn	Select rising trigger and specify trigger level
ND:[-]nnnnn	Select falling trigger and specify trigger level
NT:n	Select trigger channel: 0-3. 3 indicates external trigger
NI:nnnnn	Set UDP Port for Idle temperature reporting (proposed)
AE	Enable Autotrigger
AD	Disable Autotrigger
RI	Return Information (proposed)

With the exception of RI , the return codes are strings similar to HTTP response codes:

200 OK	Normal Successful Response
410 Gone	Successful Response to EX
500 Bad	Invalid command
503 Busy	Service Unavailable

RI will return a string identifying the board by serial number and description as configured using the web configuration tool.

To program the board for data acquisition, an initial DA command should be sent, then values should be specified for each of the ‘N’ commands as well as AE or AD before issuing the EN command.

[Proposed:] When the NI parameter is non-zero and the converter is not enabled, the board will transmit status information periodically to the specified UDP port.

Data Format

Once acquisition begins, incoming scans are transmitted to the specified UDP port on the client machine. The basic scan format

Word	msb	Scan Format 0		lsb
0	0	Format Version: U15 = 0		NwordsHdr: U16 = 6
1		NSamples: U16		NChannels: U16
2		NAvg: U16		NCoadd: U16
3		NSkP: U16		NSkL: U16
4		ScanNum: U32		
5		Spare: U32		
6		Sample(1,1)		
		...		
		Sample(1,NChannels)		
		Sample(2,1)		
		...		
		Sample(2,NChannels)		
		...		
		Sample(NSamples,NChannels)		
		Status Word (see below)		

Scan Format 1 is driven by a software update to add support for temperature monitors and record the raw sample rate. The scan reported by the hardware still comes in as version 0. It is possible that a future update to the IP will bring the hardware and software version numbers back into synch.

Word	Scan Format 1											lsb
0	0	Format Version: U15 = 1					NwordsHdr: U16 = 6					
1	NSamples: U16					NF: U8			NChannels: U8			
2	NAvg: U16					NCoadd: U16						
3	NSkP: U16					NSkL: U16						
4	ScanNum: U32											
5	T_FPGA: S16					T_HtSink: S16						
6	Sample(1,1)											
	...											
	Sample(1,NChannels)											
	Sample(2,1)											
	...											
	Sample(2,NChannels)											
	...											
	Sample(NSamples,NChannels)											
	Status Word (see below)											

	Status Word											lsb		
	Reserved	AE	TR	NT	NE	ADOOR			PAOVF			CAOVF		
Bit	31-16	15	14	13-12	11-9	8	7	6	5	4	3	2	1	0
Channel						2	1	0	2	1	0	2	1	0

Mnemonic	Description
Format Version	Serial number of the format. Currently zero for the format shown.
NWordsHdr	Number of 32-bit words in the header. 6 for this format
NSample	Corresponds to NS command
NChannels	Determined from NE command
NF	Frequency divisor from NF command
NAvg	Corresponds to NA command minus 1
NCoadd	Corresponds to NC command
AE	1 indicates Auto Trigger Enable
TR	1 indicates trigger on rising edge
NT	Trigger Source as defined by NT command
NE	Active channels as defined by NE command
ADOOR	Bit-mapped status indicating A/D Out of Range as reported by the hardware
PAOVF	Bit-mapped status indicating Pre-Adder Overflow
CAOVF	Bit-mapped status indicating Coadd Overflow
NSkP	Number of scans skipped before first coadded scan
NSkL	Number of scans skipped before the last coadded scan
T_FPGA	Temperature of the FPGA as reported by MAX6628
T_HtSink	Temperature of the heat sink on the converters as reported by MAX6661

The ADOOR status is reported by the A/D converter. It is a sticky status value so a non-zero value is reported if any individual sample in a coadded scan sequence was flagged as out of range. Ordinarily this type of overflow will be clearly visible as clipping of a large-amplitude signal, but in the presence of noise where averaging is enabled (NA or NC > 1) the result can be a more subtle distortion as the average signal approaches full scale.

PAOVF is similarly sticky. It indicates that the internal FIFO between the pre-adder and the scan processor overflowed. I don't believe this can actually happen in the current design, but if it did it would produce a time distortion in the reported signal.

CAOVF indicates that there was an arithmetic overflow in the coadder. This sort of overflow will cause a wrap, so it may be possible to recover the full resolution.

NSkP and NSkL are part of the auto-throttling circuitry in the scan processor. If the embedded processor is unable to offload a scan before the following scan is ready to start transmitting, the scan processor must drop scans. The decision to drop scans is made just before the last scan of a coadd sequence begins. If the transmit FIFO is not empty at that time, the incoming scan will be skipped, the NSkL (Number Skipped before Last) value is incremented, and the decision is revisited before the next scan. When the final scan is finally accepted, the NSkL value is recorded in the header, and is then used as input to the pre-skip circuit. This circuit recognizes that NSkL scans had to be skipped on the last

sequence, so it plans to skip that many scans before starting coadding on the next sequence. The point of this is to avoid having a large time gap in the middle of a coadd sequence. If you have to skip scans, it is better to skip them before the coadd sequence rather than in the middle. NSkP (Number Skipped Pre-coadd) records the number of scans skipped before the coadd sequence.

In ordinary operation, it will be desirable to configure the system with enough coadds so that no scans need to be skipped. This maximizes use of the available signal given the bandwidth of the system. On occasion it may be desirable to acquire scans with less averaging in order to investigate noise issues. This circuit makes it possible to accomplish this without causing the system to completely back up on itself.

T_FPGA is a signed, 16-bit value reported by the MAX6628 digital temperature sensor. The low-order 3 bits are not significant. If T_{raw} is the signed integer represented in bits D15–D3, then the temperature reported $T_{FPGA} = T_{raw}/16$ °C. Hence $T_{FPGA} = (T_{FPGA} \gg 3)/16$.

T_HtSink is essentially a signed, 16-bit value as reported by the MAX6661. The low-order 4 bits (B3:0) are not defined. Bit 4 is non-zero if the two bytes come from different conversions. Bits B15:5 are a two's complement integer representing temperature in Celcius at a resolution of .125°. Hence $T_{heatsink} = (T_{HtSink} \& 0xFFE0)/256$.

Packet Fragmentation

The largest unfragmented UDP packet on a standard Ethernet network contains a payload of 1472 octets. While the standards define how larger UDP packets may be fragmented and reassembled, hardware implementations vary in their ability to handle UDP fragments. The SSP therefore avoids the issue by providing its own fragmentation scheme to avoid sending any UDP packets above the maximum size. Each fragment of a scan includes the following header word.

SSP Fragment Header				
bits	31	30	29-16	15-0
value	Frag = 1	Last = 1	SerNum	Offset

Each fragment of a given scan will have the same SerNum value. The Offset value indicates how many 32-bit words precede the current packet. Bit 31 of the header is set for all fragments. Bit 30 is set for the last fragment of a scan.