Product Specification

AHA4540-EVB

Turbo Product Code Evaluation Board

This product is covered under multiple patents held or licensed by Comtech AHA Corporation.

This product is covered by a Turbo Code Patent License from France Telecom - TDF - Groupe des ecoles des telecommunications.

PS4540evb_0405

AHA

A subsidiary of Comtech Telecommunications Corporation

1126 Alturas Drive ■ Moscow ID 83843 ■ tel: 208.892.5600 ■ fax: 208.892.5601 ■ www.aha.com

Table of Contents

List of Figures

List of Tables

1.0 INTRODUCTION

The AHA4540-EVB is a hardware evaluation board that allows incorporation of the AHA4540 Turbo Product Code (TPC) device in a prototype communications system or laboratory test environment. It can be used to demonstrate the forward error correction (FEC) performance gain that TPCs offer over other available solutions. The EVB supports simultaneous encoding and decoding of a serial or parallel data stream. The board can be programmed and configured via an RS-232 interface connected to a PC running the included configuration software to support a wide variety of code configurations.

The AHA4540-EVB can be interfaced with a standard Bit Error Rate Tester (BERT) for data source and data sink or other communications hardware connected to the uncoded side of the EVB. On the channel side of the EVB, coded data is output from the encoder and connected to a modem or other hardware channel model and then input back into the EVB after demodulation.

The AHA4540 device is a single-chip Turbo Product Code (TPC) Forward Error Correction (FEC) Encoder/Decoder. This device integrates independent TPC encoder and decoder functions, and configurable for full or half duplex operation. In addition to TPC coding, the device includes helical interleaving, CRC computation and detection, digital modlation/ demodulation (8PSK, QPSK, 16/ 64/256 QAM), LLR mapping and pseudo random bit stream scrambling. Each of these functional blocks can be independently bypassed. When connected with a PC for configuration via the RS-232 interface the AHA4540-EVB is fully programmable for a wide range of coding configurations and allows all registers in the AHA4540 device to be accessed for configuration.

Figure 1: AHA4540-EVB Photo

1.1 CONVENTIONS, NOTATIONS AND DEFINITIONS

– Active low signals have an “N” appended to the end of the signal name. For example, MCSN and RESETN.
– Channel Rate - The rate at which encoded data is output from the device when encoding or input to the device when decoding. Note that system channel rate may be different due to external synchronization marks or other overhead.
– Code block - A data stream to be encoded or decoded is segmented into blocks for processing by the AHA4540. Data in a code block is configured as either a 2D or a 3D array.
– Code rate - Ratio of the number of data bits to the number of data and ECC bits.
– Data rate - The rate at which unencoded data is input to the device when encoding or output from the device when decoding.
– Full iteration - Decoding all axes of an array (all rows and columns).
– Hard Decision Array (HDA) - The hard decision output. Data is stored as one bit per location.
– Hex values are represented with a prefix of “0x”, such as register “0x00”. Binary values do not contain a prefix.
– Latency - The time from the first bit of an incoming block till the first bit of the same block out.
– (n1,k1)x(n2,k2) - A general representation of a 2D block code for use in the descriptions to follow in this specification. For example, in a (64,57)x(64,57) code; n1,n2=64 represents the length of the data + ECC bits, and k1,k2=57 represents the length of only the data bits. 3D codes are represented as (n1,k1)x(n2,k2)x(n3,k3)
– Soft value - Input to the decoder from either an Analog/Digital Converter (ADC) or digital demodulator.

1.2 FEATURES

PERFORMANCE:

- 148 MHz maximum clock input (S_UCLK, S_CCLK)
- 148 Mbits/sec maximum serial data rate
- Simultaneous independent TPC encoding and decoding
- Access to registers in the AHA4540 device
- Up to 4 bit soft decision input to TPC decoder
- On-board synchronization in the AHA4540
- On-board AHA4540 includes CRC insertion and detection

INTERFACE:

- SMA connectors for all clock and data signals on/off board
- 2 frequency synthesizers on board generate payload side clocks at frequencies up to 60 MHz
- User can supply payload and channel side bit clocks up to 148 MHz max frequency
- RS-232 control interface for board configuration and monitoring

GENERAL:

- PCB form factor (5.30” x 8.80”)
- On board LED status indicators
- Requires 5V power supply @ 2.0 A
- Requires Windows PC for configuration control via RS-232 port
- SMA connectors are used for inputs and outputs for all high speed serial interfaces

1.3 SYSTEM REQUIREMENTS

- 5.0V Power is supplied to the AHA4540-EVB by the user’s power supply. (2.0 Amps maximum)
- Programming of the EVB requires a Pentium class PC running Windows® 95, 98 or NT with a RS-232 port.

1.4 MODES

Mode 1: SERIAL

Uses RS-232 w/ AHAESB windows software for configuration. This mode allows the user to easily switch between codes using an AHAESB script. Data is transfered to and from the device through serial SMA connectors. This mode is selected when internal control register (0x08) bit 0 and bit 1 are cleared, and the CDATA MAP register (0x20) is set to 0x00.

Mode 2: PARALLEL

Uses RS-232 w/ AHAESB windows software for configuration. This mode allows the user to easily switch between codes using an AHAESB script. Data is transfered to and from the device through parallel flat cable connectors. This mode is selected when internal control register (0x08) bit 1 is asserted and register 0x20, CDATA MAP register is programmed to 0x01. USER_EDATA is always enabled and active. It must always be configured as 8 bits wide. USER_DDATA is always enabled. Bits 2 and 5 of Internal Control register (0x08) should be set to enable the USER_DACPT signal.

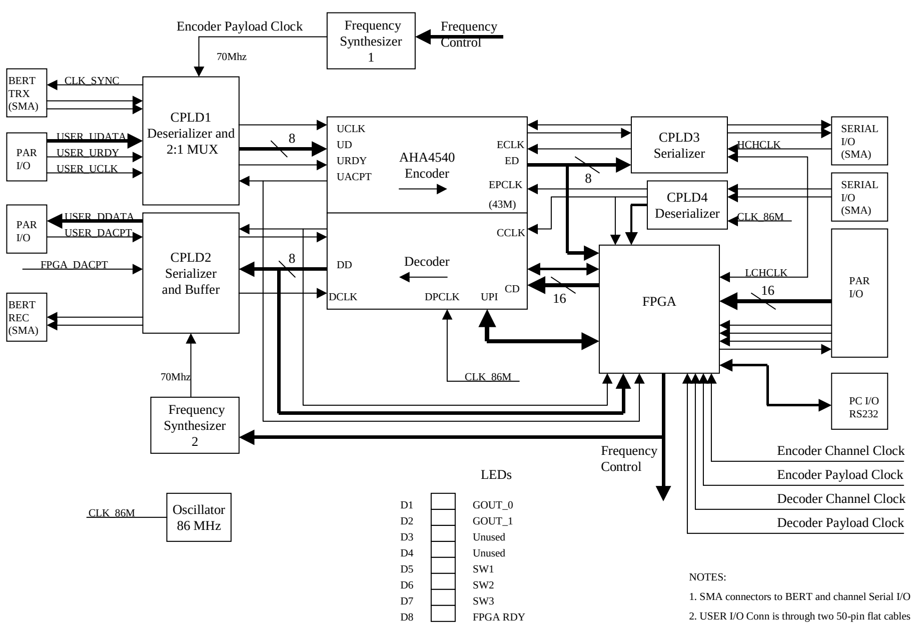
2.0 FUNCTIONAL OVERVIEW

The AHA4540 simultaneously encodes and decodes user provided data using Turbo Product Codes (TPCs). User provided data is clocked into the EVB serially using signals S-UCLK and S_UDATA from coaxial SMA connectors, or with parallel 8-bit transfers using USER_UDATA from the parallel connector JP3 and ready/accept handshake signals. In Figure 2, CPLD_1 contains the multiplexor that selects either serial data with clock, or parallel data from the USER_UDATA bus. Control of this multiplexor is via the AHAESB Windows software provided. The TPC encoder in the AHA4540 device encodes the data, adds FEC bits, then outputs the data to CPLD_3 where the data gets serialized and transmitted to the channel along with the clock, S_EDATA and S_ECLK, and also driven out in 8-bit wide format to the USER_EDATA bus on JP4.

Once the data is output from the evalutation board on either the serial or parallel encoded data interfaces, it is transmitted through an external channel where the data is corrupted by the addition of noise resulting in data bit errors. This corrupted data is clocked into the evaluation board serially using S_CCLK and S_CDATA or using the USER_CDATA 16-bit wide parallel input bus on JP5.

The S_CDATA signal is useful for wrapping around hard decision channel data back into the AHA4540 TPC decoder. For soft decision data the 16-bit parallel port must be used. The CDATA port of the AHA4540 accepts up to 16-bits per clock transfers of received channel data. This data may be I,Q data or soft metrics. The chip can accept up to four I,Q pairs or four soft metrics per transfer. In serial streaming mode the received serial channel data is deserialized into four soft metrics with the lower 3 bits of each metric forced to zero in the FPGA. The TPC decoder in the AHA4540 corrects the data errors and outputs the corrected blocks through CPLD_2 in both serialized clock and data format, S_DDATA, S_DCLK, and parallel format to the USER_DDATA bus on JP2.

Figure 2: AHA4540-EVB Block Diagram



2.1 ENCODING OPERATION

The AHA4540 (U1) performs TPC encoding one block at a time. For example, a 2D block with a (n1,k1)x(n2,k2) code, would correspond to an input block size of k1 x k2 bits. Once an entire block of data has been input to the AHA4540, it is encoded. The encoder would compute the extra ECC bits and output n1 x n2 bits. On board Serializers and Deserializers allow the user to accept a continuous data stream.

2.1.1 ENCODE DATA INPUT

The AHA4540-EVB takes serial data on the S_UDATA SMA connector. The input data is clocked into the EVB using the S_UCLK signal. Input data is latched on the rising edge of the

S_UCLK signal. An output signal CLK_SYNC may be used as a clock request signal for S_UCLK.

Parallel input data mode is supported via the USER_UDATA bus on JP3. The data gets passed through CPLD_1 to the AHA4540. To select parallel mode operation the Internal Control register 0x08 bit-6 must be set by writing the register in a user software script. The scripts are either interactively loaded or named USERTESTx.AHA which allows them to connect to one of the buttons on the AHAESB control software window.

2.1.2 ENCODE DATA OUTPUT

Encoded data is output from the AHA4540-EVB encoder serially on the S_EDATA signal. Output data is strobed out of the EVB on the falling edge of the S_ECLK signal. This signal is the inverted user supplied clock (CSCLK, J9. Encoded data can be strobed out from the EVB at data rates up to 148 Mbits/sec.

Parallel encoded data is available on the USER_EDATA bus on connector JP4. The Internal Control register 0x08 bits 0,1 must be set to 0x02 to enable the USER_EACPT signal. USER_ERDY and USER_ECLK are always driven on JP4.

2.2 DECODING OPERATION

The AHA4540 device on the EVB also operates as a TPC decoder, operating on a single block at a time. For a 2D block with a (n1,k1)x(n2,k2) code, n1 x n2 symbols are input for each block. Once an entire block of data is received by the AHA4540 decoder, it is iteratively decoded, corrupted bits are corrected and then the block of k1 x k2 data bits is output from the decoder.

2.2.1 DECODE DATA INPUT

The input data to the AHA4540 decoder device on the EVB can be either hard decision data (1 bit) or soft decision data (up to 4 bit soft decision via USER_CDATA on the parallel connector JP5). Hard decision serialized data can use the S_CDATA SMA connector for data input. The rising edge of S_CCLK is used to clock in the data when the serialized channel input is used.

Parallel received data from the channel may be used by enabling the USER_CDATA port on JP5. This is a 16-bit bus that gets passed straight through to the AHA4540 TPC decoder. Enabling this requires writing the Internal Control register 0x08 bits 0,1 to 0x02. USER_CCLK is passed through to the AHA4540 CCLK signal in this mode of operation.

2.2.2 DECODE DATA OUTPUT

The AHA4540 decoder outputs the decoded data stream serially from the EVB on the S_DDATA SMA connector J4. Decode data is clocked out using the rising edge of S_DCLK output signal on SMA connector J6. Decoded data is output from the EVB at the programmed data rate at rates up to 148 Mbit/sec. Maximum data rate if using an on-board synthesizer to generate the payload bit clock is 60 Mbit/sec.

Parallel decoded data output is available on connector JP3. The data bus and USER-DRDY are always driven. Enabling USER_DACPT requires setting USER_SEL and PSEL2 (bits, 2,5) in the Internal Control register (0x08).

2.3 CLOCK SOURCES

2.3.1 CRYSTAL OSCILLATORS

There are a total of three crystal oscillators on the EVB. Crystal (Y1) at 86.0 MHz is used for the AHA4540 processing clocks DPCLK and EPCLK (86 Mhz / 2). Crystal (Y2) at 3.6864 MHz is a reference for the RS232 interface in the FPGA. Crystal (Y3) at 30.0 MHz is the reference clock for the frequency synthesizers (U20,U21). The frequency synthesizers multiply this clock by six and use an internal reference clock of 180 Mhz.

2.3.2 FREQUENCY SYNTHESIZERS

The frequency synthesizers defines the uncoded data rates through SMA connector (J2) CLK_SYNC and (J6) S_DCLK. Interactive feedback control inside the FPGA automatically adjusts the uncoded data rate frequency to prevent data underflows or overflows, thus frequency locking the payload side data rate to the channel side data rate.

Maximum frequency for the two synthesizers is 60 MHz. They are programmable by writing the 4-Byte control value in a user script, or selecting the Clock Synth Tab in the Configure window, typing in the frequencies in MHz and selecting the set button. To determine the 4-Byte value use the equations below. The first equation determines the code rate including frame sync bits, and the next one calculates the 4-Byte value that gets programmed via the user script.

SYNTH_CLK_FREQ = Payload Data Rage = [Message_data_bits / (Message_data_bits + ECC_bits + Synchronization_bits)]*CHANNEL_CLCK_FREQ **[1]**

4-Byte value = [Synth Byte 1, Synth Byte 2, Synth Byte 3, Synth Byte 4] = B1,B2,B3,B4 **[2]**

$$B1,B2,B3,B4 = [SYNTH\_CLK\_FREQ * 2^{32}] / 180$$

Programming Synthesizer 1, Encoder Payload Clock, requires writing this value to registers 0x0C, 0x0D, 0x0E, and 0x0F.

2.4 SYNCHRONIZATION SUPPORT

All synchronization support is performed by the AHA4540. See AHA4540 product specification for more infomation.

2.4.1 LED STATUS DEFINITIONS (D1-D8)

Table 1: LED Definition

LEDS	*LED DEFINITION*
D1	GOUT_0
D2	GOUT_1
D3	Unused
D4	Unused
D5	SW1
D6	SW2
D7	SW3
D8	FPGA Programmed

2.4.2 SWITCH DEFINITIONS

Table 2: Switch Definition

PUSH BUTTON	*SWITCH DEFITION*
SW0	Reprogram FPGA
SW1	Undefined
SW2	Undefined
SW3	Reset AHA4540, AD9851 and FPGA

2.4.3 JUMPER CLOCK CONFIGURATION (JP7)

Table 3: Jumper Clock Configuration

JP7 PINS 1,2,3	*ENCODER CLKSYNC SOURCE*
1-2	VCO1
2-3	Frequency Synthesizer 1

2.4.4 JUMPER CLOCK CONFIGURATION (JP8)

Table 4: Jumper Clock Configuration

JP8 PINS 1,2,3	*DECODER CLKSYNC SOURCE*
1-2	VCO2
3-4	User Supplied S_DCLK_IN SMA (J13)
5-6	Frequency Synthesizer 2

2.5 LOOPBACK - TEST 1

Test 1 routes the uncoded input data to the output data through the entire data path of the EVB including the output connections for the user channel. The encoder and decoder are configured for a $(128,120)^2$ code and the channel rate is 50 MHz. Connect the cables to the BERT as specified. Connect an additional SMA to SMA jumper between S_EDATA to S_CDATA and S_ECLK to S_CCLK.

Connect appropriate cables as specified below for proper operation.

BERT transmitter connections:

– S_UDATA (SMA J1): Data input from BERT transmitter
– S_UCLK (SMA J3): Clock input from BERT transmitter
– CLK_SYNC (SMA J2): Optional clock request to BERT transmitter

BERT receiver connections:

– S_DDATA (SMA J4): Data output to BERT receiver
– S_DCLK (SMA J6): Clock output to BERT receiver

Note: Cable lengths for S_UDATA and S_UCLK should be approximately the same length shielded coax (recommended RG58c/u or equivalent, Zo=50ohms). Cables for S_DDATA and S_DCLK should be approximately the same length. (Max length 4ft.)

Channel Clock source:

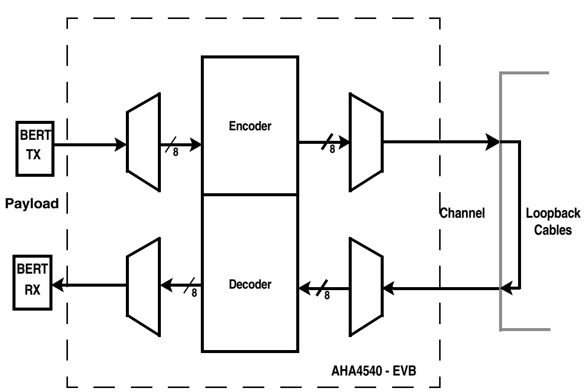
CSCLK (SMA J9): Connect to external clock source with frequency = 50.06 MHz.

- Frequency other than this will require changing the 4-Byte value for the encoder payload clock synthesizer in the software script named TEST1.AHA.

Channel Loopback cables

– Connect S_ECLK (SMA J10) to S_CCLK (SMA J7)
– Connect S_EDATA (SMA J8) to S_CDATA (SMA J5)
– Connect CLK_SYNC (J2) to S_DCLK_IN (J13). This allows the the Encoder synthesizer to be used for both payload rate clock sources.

Figure 3: Encode/Decode Loopback - Test 1 Datapath



JP7 Jumper Settings: (2-3)
JP8 Jumper Settings: (2-3)

2.6 LOOPBACK - TEST 2

Same system configuration as TEST 1. This loopback test uses the $(16,11)^3$ TPC code.

2.7 LOOPBACK - TEST 3

This test implements the $(128, 120)^2$ code, at 50.06 Mb/s channel rate, and allows two separate channel clocks and payload clocks. Both on board synthesizers are used and locked to their appropriate channel rate bit clock.

Same system configuration as TEST 1 except the following:

1) Disconnect cable to S_DCLK_IN(J13)
2) Move jumper JP8 to pins 5,6

2.8 LOOPBACK TEST 4

This is a pass through test requiring one clock source for all channel clocks and payload side clocks and runs at any frequency up to 148 MHz.

Same system configuration as TEST 1 except the following:

1) CLK_SYNC (SMA J2) is not used. Connect the clock request on the BERT Transmitter to the channel bit clock source.

2.9 LOOPBACK - TEST 5

This test uses the same system configuration as TEST 1. It is a high speed, 148.3 Mb/s channel rate test with the $(16,11)^3$ TPC code and a payload data rate of 48 Mb/s. Decoder iterations are set to 3.

2.10 EVB CONFIGURATION FOR MODE 1

1) Connect 5V power supply to JP9 pins 1 and 2.
2) Connect RS-232 cable to EVB and Host PC.
3) Start AHAESB software on Host PC.
4) Connect to board by pressing Connect button (set serial port parameters correctly).
5) Connect SMA-BNC cables to EVB, Data Source/Sink, and Channel Source/Sink as described in Section 2.5.
6) Run AHAESB script to setup the board and program the AHA4540 device to desired configuration.
7) EVB should start encoding, decoding, and passing data.

2.11 MODE 1 SCRIPT WRITING

1) Board reset.
2) Program payload frequency.
3) Program block sizes.
4) Program Sync Word Length.
5) Program the encoder/decoder configuration into the AHA4540 device:
   - 5.1) Assign the AHA4540 address byte into register 0x04.
   - 5.2) Assign the AHA4540 data byte into register 0x05.
   - 5.3) Send a write strobe to the AHA4540 device by using the following line, “inc pluto_write_strobe.aha”.
   - 5.4) (optional) You may read back your written value with the following line, “inc pluto_read_strobe.aha”, this will verify that the correct value was written to the AHA4540 device.
   - 5.5) Repeat these three or four steps for all registers in the TPC encoder/decoder that need to be written after a reset.
6) (Optional) The script may be tied directly to a user test button and executed by selecting the button on the AHAESB software window. This requries naming the script usertest1.AHA, usertest2.AHA, ... usertest5.AHA and locating it in the same directory as the AHAESB.EXE executable software.

2.12 SOFTWARE INSTALLATION

The AHAESB software (AHAESB.exe) is used to configure and monitor the EVB. This software is shipped with the EVB.

2.13 QUICK START

This section briefly describes the steps necessary to power-up the EVB and get Loopback Test1 running, using a standard Bit Error Rate Transmitter/ Receiver (BERT).

1) Install the AHAESB per instructions for Mode 1 above.
2) Provide correct voltage to the EVB 5.0V/GND Power Connector.
3) Connect SMA cables as shown below:

BERT transmitter connections:

– S_UDATA to BERT data output
– S_UCLK to BERT clock output
– CLK_SYNC to BERT external clock input

BERT receiver connections:

– S_DDATA to BERT data input
– S_DCLK to BERT clock input

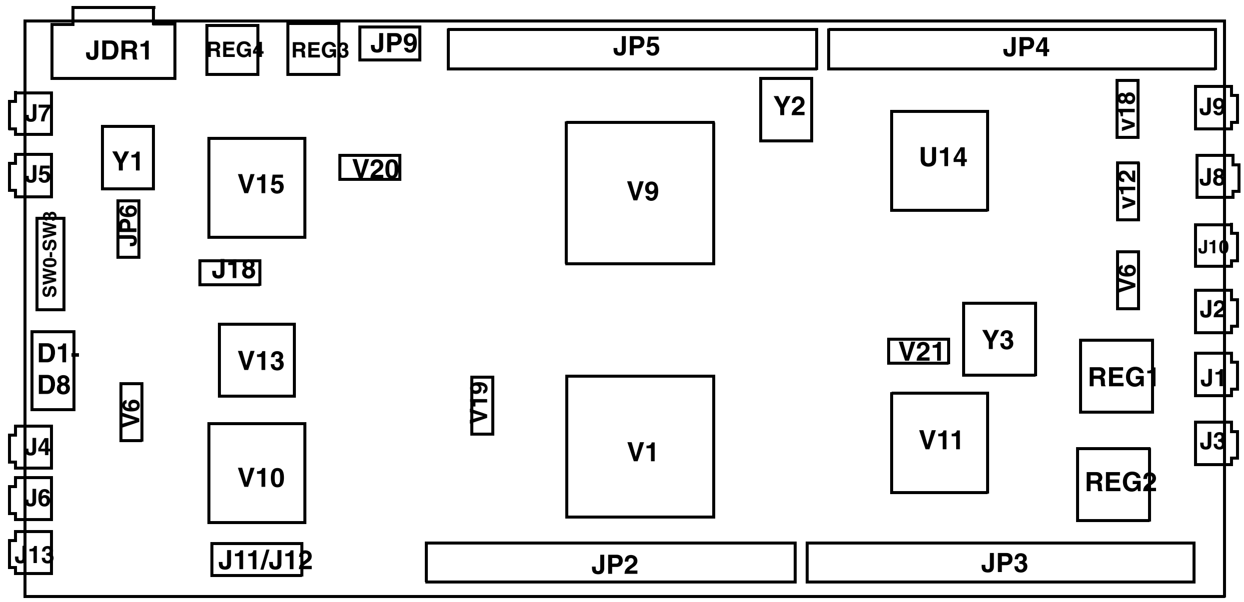
Encode out to Decode in bypass cable:

– S_EDATA to S_CDATA on EVB
– S_ECLK to S_CCLK on EVB

4) Confirm jumpers are connected (2-3), (2-3) for JP7 and JP8.
5) Connect external clock source to CSCLK (J9) for the channel bit-rate clock.
6) Connect loopback cable
   S_EDATA→S_CDATA.
   S_ECLK→S_CCLK.
7) Run Loopback Test 1 by clicking on the appropriate button in the AHAESB SW.
8) Set up BERT transmitter for TTL signal levels. Verify these levels with an oscilloscope.
9) Set up BERT transmitter for external clock sync.
10) BERT receiver should show zero bit errors, and frequency is 43 MHz.
11) The signal monitors in the right side of the software window should show alternating zeros and ones indicating data is flowing from data source to data sink through the EVB.

3.0 HARDWARE DESCRIPTION

Figure 4: AHA4540-EVB Major Component Diagram



Discrete Components

U1: AHA4540 TPC Encoder/Decoder
U9: Xilinx Virtex E FPGA, XCV200E-PQ240
U10: SPROM to program FPGA Controller
U11: Altera 7032 CPLD
U13: Altera 7032 CPLD
U14: Altera 7032 CPLD
U15: Altera 7032 CPLD
U5: Line Driver (Tristatable)
U6: Line Driver (Tristatable)
U12: Line Driver (Tristatable)
U18: Line Driver (Tristatable)
U19: Line Driver (Tristatable)
U20: Frequency Synthesizer for Payload Data Side Clock
U21: Frequency Synthesizer for Payload Data Side Clock

J1: Unencoded Data (Input)
J2: Clock Sync for BERT (Output)
J3: Unencoded Clock (Input)
J4: Decoded Data (Output)
J5: Channel Data (Input)
J6: Decoded Clock (Output)
J7: Channel Clock (Input)
J8: Encoded Data (Output)
J9: Encoder Side Channel Clock (Input)
J10: Encoded Clock (Output)
J13: Decoder Side Channel Clock (Input)

JP6: JTAG Port for Altera CPLD's
JP7: Encoder Side Payload Clock Jumper
JP8: Decoder Side Payload Clock Jumper
JP9: Power Connector
J11: Serial Program Port for FPGA
J12: JTAG Program Port for FPGA

JP2: Decoder Parallel Input:

JP3: Unencoded Parallel Input

JP4: Encoded Parallel Input

JP5: Channel Parallel Input

SW0 - SW1: Resets

SW2 - SW3: Push Buttons

REG1: 5.0V to 2.5V Regulator

REG2: 5.0V to 12V Regulator

REG3: 5.0V to 1.8V Regulator

REG4: 5.0V to 3.3V Regulator

JDR1: RS-232 Connector

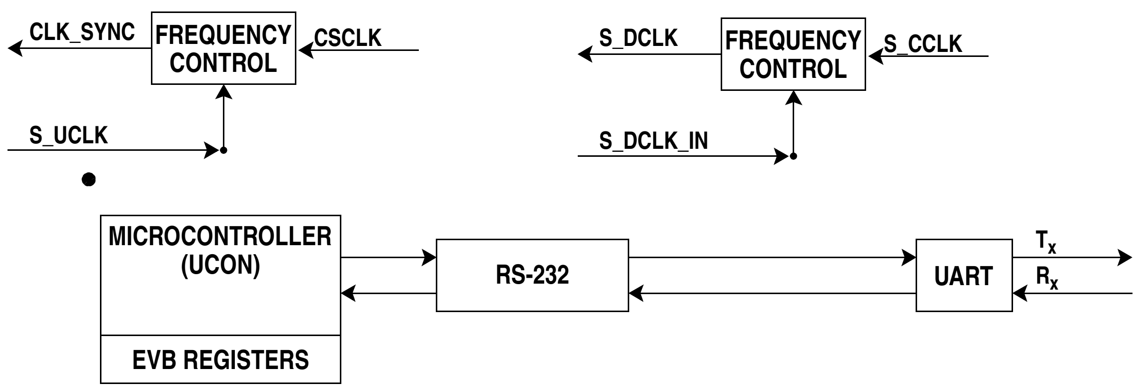
Y1: 86.0 Mhz Clock for AHA4540 EPCLK and DPCLK (EPCLK = DPCLK/2)
Y3: 30 MHz Crystal for Frequency Synthesizer Reference Clock
Y2: 3.6864 MHz Crystal for Serial Port Communication base frequency

4.0 FPGA CONTROLLER TOPLEVEL DESCRIPTION

Hierarchical FPGA VHDL Flow

Ra_pipe - Ready / Accept handshaking pipe stage
UART - Communication with the PC via RS-232 port. The UART block can read and write registers.
RS232con - Takes in Serial RS-232 data and converts it to byte wide data
CZX - Clock Zone crossing boundary
UCON - Microcontroller interface. Manages reads and writes to the register stack. Programs the Frequency Synthesizer.
Freq_ctrl_r5 - This block determines if the payload date frequency should be increased or decreased to avoid an underflow or overflow.
Micro_reg - 8-bit microcontroller register

Figure 5: FPGA Block Diagram



5.0 AHA4540-EVB COMMUNICATION SOFTWARE

Communication and control of the AHA4540-EVB is via the RS-232 interface to a host PC. The AHAESB.exe software is provided for this purpose. The AHAESB software operates under Windows® 95, 98, 2000, or NT. The AHAESB software allows the user to configure the AHA4540 device on the EVB as well as the clock synthesizers and FPGA registers.

5.1 OPERATION OF AHAESB SOFTWARE

To operate the AHAESB software, click on the AHA4540-EVB icon on the Windows® desktop or alternatively execute the program AHA4540 EVB.exe from either a DOS window or the Windows® RUN menu. An application window for AHA4540-EVB should open.

5.1.1 CONNECT BUTTON

This button enables connection from the PC to the AHA4540-EVB via the RS-232 cable. A pop-up menu will appear allowing the user to configure the communication link between host PC and the AHA4540-EVB. Configuration parameters are: Comm Port (COM1,2,3 or 4), Number of Stop Bits (defaults to 1), and Data Rate (defaults to 115,200 Baud).

5.1.2 CONFIGURE BUTTON

This menu allows configuration of the clock synthesizers. Registers in the AHA4540 device can be accessed. See the AHA4540 Product Specification for a complete list of AHA4540 registers. The clock synthesizers are also programmable in the configure menu. Data rate can be controlled from this menu.

5.1.3 INTERACTIVE MODE WINDOW

The interactive mode window allows the user to control signals with the FPGA to allow customized control of the EVB. Script files containing instructions for the EVB can be loaded and saved from this window.

5.1.4 LOG WINDOW

The Log Window records all communication transactions between the PC and the EVB.

If an error were encountered while communicating with the EVB it would be recorded here.

5.1.5 RESET STATUS

Indicates whether or not the board is in reset.

5.1.6 USER TEST MODES

These buttons can be linked to user scripts created by the customer. These buttons provide an easy and quick way to access/run user defined scripts. They must be named usertestX.aha, where X is the button number and reside in the same directory as the executable file for the software. The self test bit must be off and the number of soft bits set.

5.1.7 TEST MODES

These buttons are linked to test scripts created by AHA. These buttons provide an easy and quick way to access/run AHA defined board functionality tests. They must be named testX.aha, where X is the appropriate button number.

5.1.8 AHAESB SCRIPT SYNTAX

The syntax is similar to standard assembly with only a few operations.

WRA – Command used to write to a register in the FPGA.
 i.e. WRA 1E 0F
 This would write a 0F (15-bit sync word length) to register 1E

RDA – Command used to read a register in the FPGA
 i.e. RDA 1E
 If the above write was executed first, the response from the interactive window would yield "0F".

inc – Command used to include another AHAESB script in the current script. Useful for hierarchical script programming.
 i.e. inc reset.aha

SLP – Command used to make the AHAESB SW wait a period of time before the next command is issued.
 i.e. SLP 3E8
 Sleep for 1ms

RST – Generates a board reset

5.2 ESB STATUS

The EVB status window displays the frequencies of the two on-board frequency synthesizers, Encoder Synthesizer and Decoder Synthesizer. It also includes data monitors that monitor some of the signals on the board indicating whether the signals are changing or not.

6.0 REFERENCE GUIDE TO PROGRAMMING EVB REGISTERS

This guide describes all user accessible registers on the evaluation board.

Table 5: List of EVB Registers with Address

ADDRESS (HEX)	NAME
00	LOCAL_UART_ADDRESS
01	RESETS
02	CHIIP_SELECTS
03	CONTROL_SIGNALS
04	ADDRESS
05	OUTPUT_DATA
06	INPUT_DATA
07	INTERRUPT_STATUS
08	INTERNAL_CONTROL
09	OUTPUT_PULSE
0B	BITFILE_VERSION_REGISTER
0C	ENCODE_FREQUENCY_SYNTH_BYTE_1
0D	ENCODE_FREQUENCY_SYNTH_BYTE_2
0E	ENCODE_FREQUENCY_SYNTH_BYTE_3
0F	ENCODE_FREQUENCY_SYNTH_BYTE_4
10	DECODE_FREQUENCY_SYNTH_BYTE_1
11	DECODE_FREQUENCY_SYNTH_BYTE_2
12	DECODE_FREQUENCY_SYNTH_BYTE_3
13	DECODE_FREQUENCY_SYNTH_BYTE_4
20	CDATA_MAP
28	SYNTHESIZERS_CONTROL
29	ENCODE_SYNTH_DELTA_BYTE_2
2A	ENCODE_SYNTH_DELTA_BYTE_3
2B	ENCODE_SYNTH_DELTA_BYTE_4
2C	DECODE_SYNTH_DELTA_BYTE_2
2D	DECODE_SYNTH_DELTA_BYTE_3
2E	DECODE_SYNTH_DELTA_BYTE_4
30	ENC_PAYLOAD_BITS_LSB
31	ENC_PAYLOAD_BITS_MSB
32	ENC_CHANNEL_BITS_LSB
33	ENC_CHANNEL_BITS_MSB
34	ENCODE_SYNC_LENGTH
35	DEC_PAYLOAD_BITS_LSB
36	DEC_PAYLOAD_BITS_MSB
37	DEC_CHANNEL_BITS_LSB
38	DEC_CHANNEL_BITS_MSB
39	DECODE_SYNC_LENGTH
43	SIGNAL_ACTIVITY

6.1 REGISTER DESCRIPTION

0x00 LOCAL UART ADDRESS **reset value [0x00]**

Local storage of register address for UART “RDA” and “WRA” commands

0x01 RESETS **reset value [0x00]**

Reset signals to devices on board.

0x02 CHIP SELECTS **reset value [0x00]**

bit 0 : AHA4540 chip select, MCSN

0x03 CONTROL SIGNALS **reset value [0x00]**

Control signals to on board devices.

bit 7 : Unused, always asserted low
bit 6 : Unused, always asserted low
bit 5 : Unused, always asserted low
bit 4 : Unused, always asserted low
bit 3 : Unused, always asserted low
bit 2 : Unused, always asserted low
bit 1 : AHA4540 read strobe, MRDN_DSN
bit 0 : AHA4540 write strobe, MWRN_RWN1

0.04 ADDRESS **reset value [0x00]**

Address bus to AHA4540 device, MA[5:0}

0x05 OUTPUT DATA **reset value [0x00]**

Data bus to AHA4540 device, MDATA[7:0]

0x06 INPUT DATA **reset value [0x00]**

Data bus from AHA4540 device, MDATA[7:0]

0x07 INTERRUPT STATUS **reset value [undefined]**

Interupts from AHA4540 and GOUTx signals

bit 7 : Unused, always asserted low
bit 6 : Unused, always asserted low
bit 5 : Unused, always asserted low
bit 4 : Unused, always asserted low
bit 3 : Unused, always asserted low
bit 2 : AHA4540 GOUT[1] signal
bit 1 : AHA4540 GOUT[0] signal
bit 0 : AHA4540 Interrupt, MINTN

0x08 INTERNAL CONTROL **reset value [0x00]**

Controls mode of board.

bit 7 : unused, always asserted low
bit 6 : Psel_1, enables USER_UDATA and USER_UCLK
bit 5 : Psel_2, when set along with User_sel enables USER_DACPT
bit 4 : Psel_4, not used
bit 3 : Chsel, not used
bit 2 : User_sel, when set along with Psel_2 enables USER_DACPT
bit 1 : Loopback select_1
bit 0 : Loopback select_0

Loopback select bits 1,0

0x00 : Serial Streaming
0x01 : Edata to Cdata internal loopback
0x02 : USER_EDATA, USER_CDATA parallel ports, and USER_CCLK enabled
0x03 : CRDY and EACPT always asserted high

0x09 OUTPUT PULSE reset value [0x00]

Output Control Pulses.

0x0B BIT FILE VERSION reset value [0x0B]

Returns the bit file version. The upper 4 bits represent the whole number part and the lower 4 bits represent the fractional part.

0X0C ENCODE FREQUENCY SYNTH BYTE 1 reset value [0x00]
0x0D ENCODE FREQUENCY SYNTH BYTE 2 reset value [0x00]
0x0E ENCODE FREQUENCY SYNTH BYTE 3 reset value [0x00]
0x0F ENCODE FREQUENCY SYNTH BYTE 4 reset value [0x00]

These registers set the output frequency of U20, the eoncoder path synthesizer according to the following equation:

$$\text{FREQ} = \frac{[\text{BYTE\_1, BYTE\_2, BYTE\_3, BYTE\_4}] \times 180}{2^{32}} \text{ MHz}$$

0x10 DECODE FREQUENCY SYNTH BYTE 1 reset value [0X00]
0X11 DECODE FREQUENCY SYNTH BYTE 2 reset value [0x00]
0X12 DECODE FREQUENCY SYNTH BYTE 3 reset value [0x00]
0X13 DECODE FREQUENCY SYNTH BYTE 4 reset value [0x00]

These register set the output frequency of U21 the encoder path synthesizer according to the following equation:

$$\text{FREQ} = \frac{[\text{BYTE\_1, BYTE\_2, BYTE\_3, BYTE\_4}] \times 100}{2^{32}} \text{ MHz}$$

0x20 CDATA MAP reset value [0x00]

Maps CDATA port into the AHA4540 as follows

Register Value	Mapping
0x00	Deserialized Hard decision data from SMA (S_CDATA) 4 soft metrics per transfer
0x01	16-bit USER_CDATA mapped straight through
0x02	EDAT to CDATA internal loopback
0x03	USER-EDATA to USER_CDATA external loopback (requires external wire jumpers).
Others	16-bit USER_CDATA mapped straight through

0x28 SYNTHESIZERS CONTROL

bits 7-2: Unused

bit 1 : When set the decoder synthesizer uses the fixed register programmed value for the frequency delta set in registers 0x2C, 0x2D, and 0x2E as follows:

NEW_SYNTH_VALUE = [SYNTH_BYTE 1, 2, 3, 4] +/- [0,00, DELTA_BYTE 2, 3, 4]

Byte 4 is the least significant Byte of the 4-Byte value

bit 0 : When set the encoder synthesizer uses the fixed register programmed value for the frequency delta set in registers 0x29, 0x2A, and 0x2B as follows:

NEW_SYNTH_VALUE = [SYNTH_BTE 1, 2, 3, 4] +/- [0,00, DELTA_BYTE 2, 3, 4]

Byte 4 is the least significant Byte of the 4-Byte value

0x29 ENCODE SYNTH DELTA BYTE 2 reset value [0x00]

0x2A ENCODE SYNTH DELTA BYTE 3 reset value [0x00]
0X2B ENCODE SYNTH DELTA BYTE 4 reset value [0x00]

These register values are added or subtracted to the Encoder frequency synthesizer base value in registers 0x0C, 0x0D, 0x0E, 0x0F when enabled in register 0x28

NEW_SYNTH_VALUE = [SYNTH_BYTE 1, 2, 3, 4] +/- [0x00, DELTA_BYTE 2, 3, 4]

Byte 4 is the least significant Byte of the 4-Byte value.

0x2C DECODE SYNTH DELTA BYTE 2 reset value [0x00]
0x2D DECODE SYNTH DELTA BYTE 3 reset value [0x00]
0x2E DECODE SYNTH DELTA BYTE 4 reset value [0x00]

These register values are added or subtracted to the decoder frequency synthesizer base value in registers 0x10, 0x11, 0x12, 0x13 when enabled in register 0x28.

NEW_SYNTH_VALUE = [SYNTH_BYTE 1, 2, 3, 4] +/- [0x00, DELTA_BYTE 2, 3, 4]

Byte 4 is the least significant Byte of teh 4-Byte value.

0x30 ENCODE PAYLOAD BITS LSB reset value [0x00]
0x31 ENCODE PAYLOAD BITS MSB reset value [0x00]

Set equal to the block size of the encoder TPC unencoded block.

0x32 ENCODE CHANNEL BITS LSB reset value [0x00]
0x33 ENCODE CHANNEL BITS MSB reset value [0x00]

Set equal to the encoder encoded block including the FEC Bits, but not including frame sync bits.

0x34 ENCODE SYNC BITS PER FEC BLK reset value [0x00]

Set equal to (EFSyncLength x EFSyncFreq) as programmed into the AHA4540

0x35 DECODE PAYLOAD BITS LSB reset value [0x00]
0x36 DECODE PAYLOAD BITS MSB reset value [0x00]

Set equal to the block size of the decoder TPC unencoded block

0x37 DECODE CHANNEL BITS LSB reset value [0x00]
0x38 DECODE CHANNEL BITS MSB reset value [0x00]

Set equal to the TPC decoder encoded block size including the FEC bits, but not including frame sync bits.

0x39 DECODE SYNC BITS PER FEC BLK reset value [0x00]

Set equal to (DFSyncLength x DFSync Freq) as programmed into the AHA4540

0x43 SIGNAL ACTIVITY

Indicates activity of selected signal ports by showing current value (multiple reads need to see changes)

bit 7 : decoder payload clk
bit 6 : encoder payload clk
bit 5 : E_CLK
bit 4 : E_DATA (0)
bit 3 : C_CLK
bit 2 : C_DATA
bit 1 : D_CLK
bit 0 : D_DATA (0)

7.0 REGISTER PROGRAMMING

7.1 AHA4540 REGISTERS

The AHA4540 device on the board is accessible for programming and status information through the AH4540-EVB software. From the program select the interactive button. Here you can read and write to registers in the FPGA or AHA4540. Refer to the AHA4540 product specification for a complete description of all registers in the AHA4540 device.

7.2 FREQUENCY SYNTHESIZERS

Two on board synthesizers are programmed via the AHAESB software. From the program select the Clock Synth tab in the configure window. The value entered for Encoder Synth controls the clock synthesizer U21. Clock synthesizer U20 is controled by the entry Decoder Synth. Synthesizer U21 can also be programmed via user scripts by writing the desired values to registers 0x0C, 0x0D, 0x0E, 0x0F as outlined in Section 5.1.8 *AHAESB Script Syntax*. Synthesizer U20 can be written to using register 0x10, 0x11, 0x12, 0x13.

The ratio of the values entered for clock synthesizers should exactly match the code rate of the code (including synchronization bits) as shown in the following equation:

$$\frac{\text{Data Rate}}{\text{Channel Rate}} = \frac{\text{Data Bits}}{\text{Data Bits + ECC Bits + Sync\_word\_bits}}$$

7.3 AHA4540-EVB FPGA REGISTERS

See Section 6.0 *REFERENCE GUIDE TO PROGRAMMING EVB REGISTERS* and Section 6.1 *Register Description.*

8.0 SIGNALS IN/OUT OF EVB

8.1 EVB MAIN BOARD

PIN	CONNECTOR	FUNCTION	TYPE
J1	SMA	S_UDATA	Input
J3	SMA	S_UCLK	Input
J2	SMA	CLK_SYNC	Output
J8	SMA	S_EDATA	Output
J10	SMA	S_ECLK	Output
J9	SMA	CSCLK	Input
J4	SMA	S_DDATA	Output
J6	SMA	S_DCLK	Input
J13	SMA	S_DCLK_IN	Input
J7	SMA	S_CCLK	Input
J5	SMA	S_CDATA	Input
JDR1	9 pin DSUB	RS232 Serial Data Connector	I/O
J11	.1" Header	FPGA Serial Program Header	Input
J12	.1" Header	FPGA JTAG Connector	I/O
JP6	.1" Header	CPLD JTAG Connector	I/O
JP7	1x3	Encoder Payload Clock Selector	Control
JP8	2x3	Deocder Payload Clock Selector	Control
JP9	2x3	DC Power Connector	I/O, Pwr

9.0 ELECTRICAL SPECIFICATIONS

9.1 BOARD POWER

Power is provided from the users power supply.
5 Volts DC at 2.0 Amps JP9 pins 1 and 2.
Power consumption varies depending on customer configuration, but a minimum of 2A is required for factory configuration.

9.2 SIGNAL REQUIREMENTS

BERT signals at SMA connectors should be TTL levels. Program BERT for threshold levels of approximately 1.5V. Validate TTL levels using an oscilloscope.

10.0 TIMING SPECIFICATIONS

Figure 6: Data Input Timings

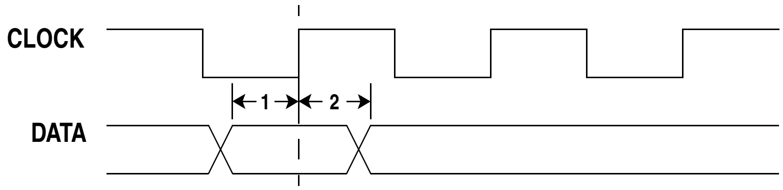


Figure 7: Data Output Timings

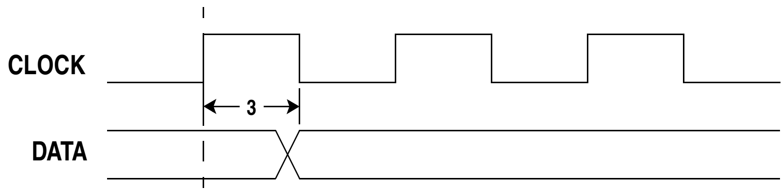


Table 6: Data Input/Output Timings

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	DATA to CLK rising edge	4		ns	
2	DATA from CLK rising edge	4		ns	
3	DATA delay from CLK rising edge		8	ns	

11.0 BOARD DIMENSIONS

PCB board
Overall dimensions are 5.3" x 8.8".

12.0 ORDERING INFORMATION

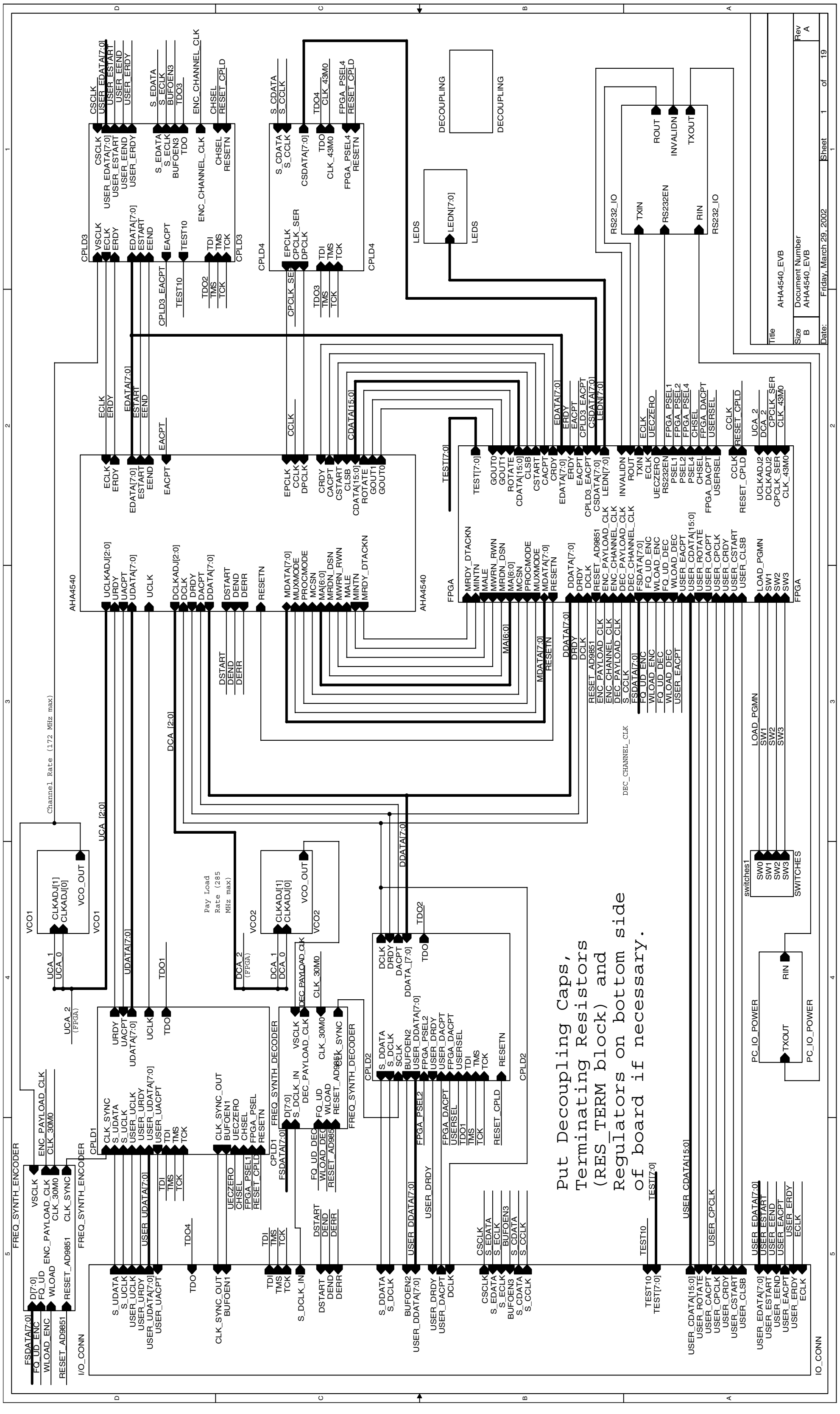
AHA4540-EVB evaluation system kit consists of:

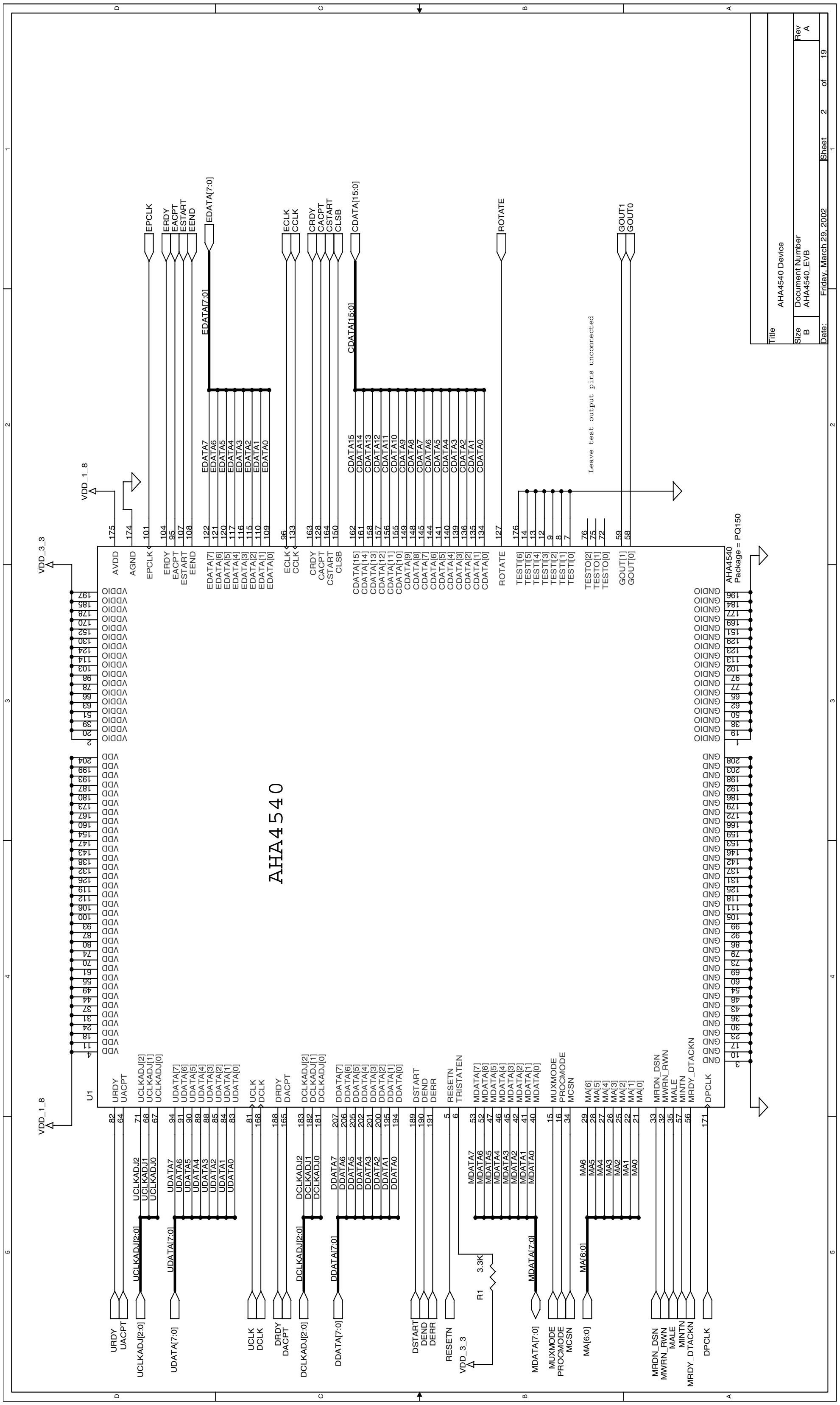
QUANTITY	DESCRIPTION
1	Evaluation board w/AHA4540 encoder/decoder
1	RS-232 interface cable for host PC
1	CD ROM containing software and pdf files of documentss
6	SMA to SMA interconnecting cables (122 cm), with optional BNC adapters
1	Product Specification AHA4540
1	Product Specification AHA4540-EVB

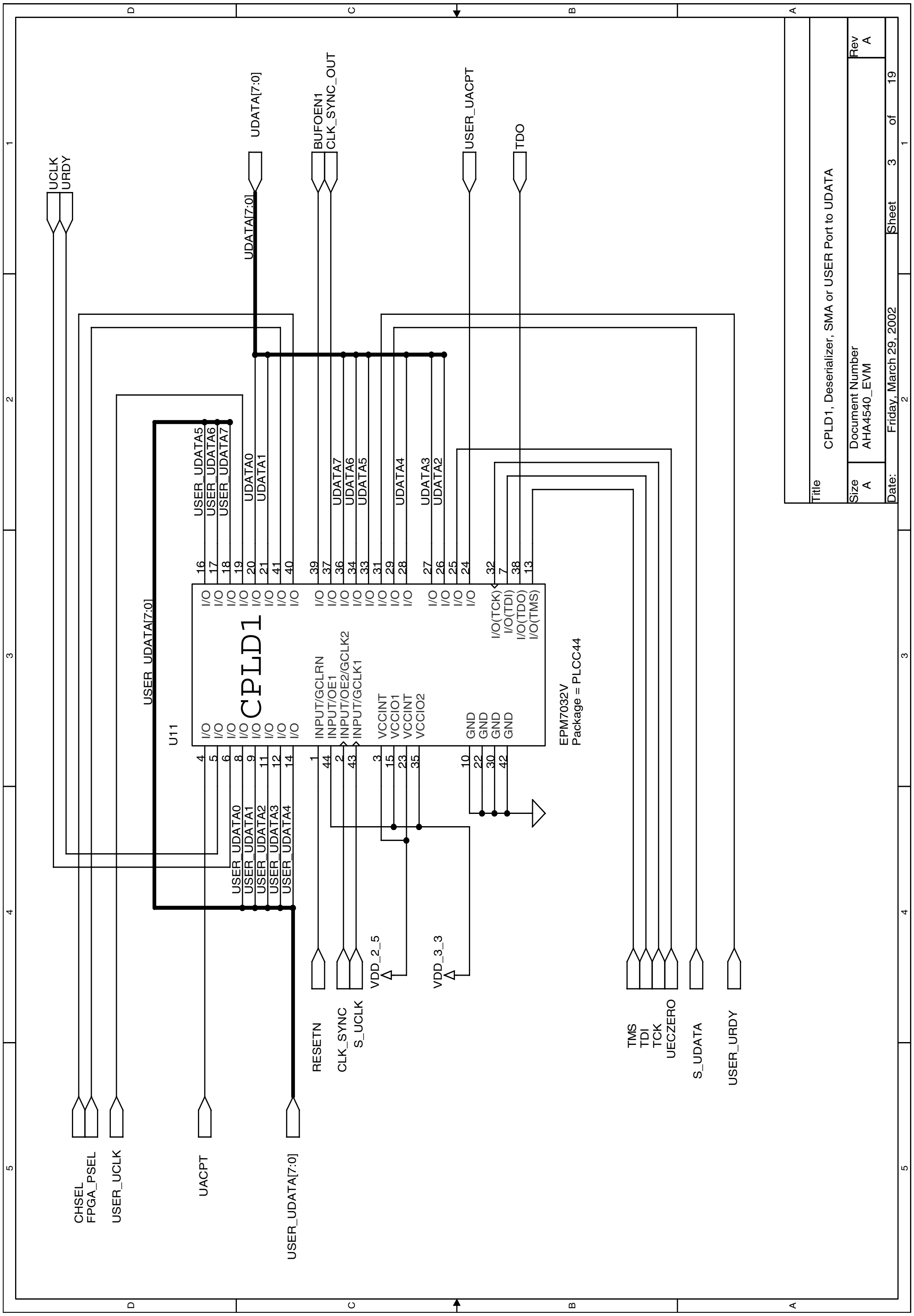
13.0 ABOUT AHA

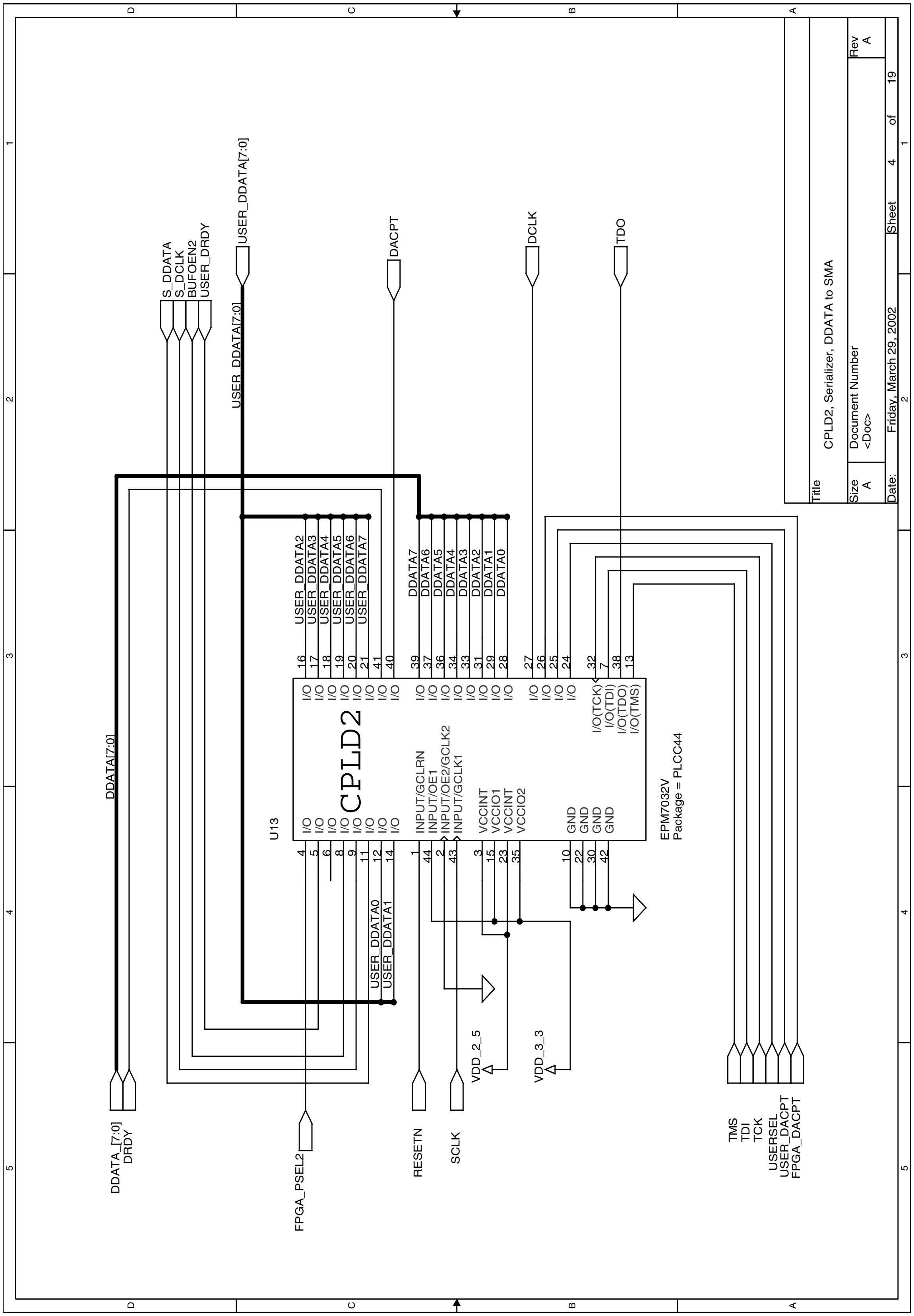
Comtech AHA Corporation (AHA) develops and markets superior integrated circuits, boards, and intellectual property core technology for communications systems architects worldwide. AHA has been setting the standard in Forward Error Correction and Lossless Data Compression technology for many years and provides flexible, cost-effective solutions for today's growing bandwidth and reliability challenges. Comtech AHA Corporation is a wholly owned subsidiary of Comtech Telecommuncations Corp. (NASDAQ: CMTL). For more information, visit www.aha.com.

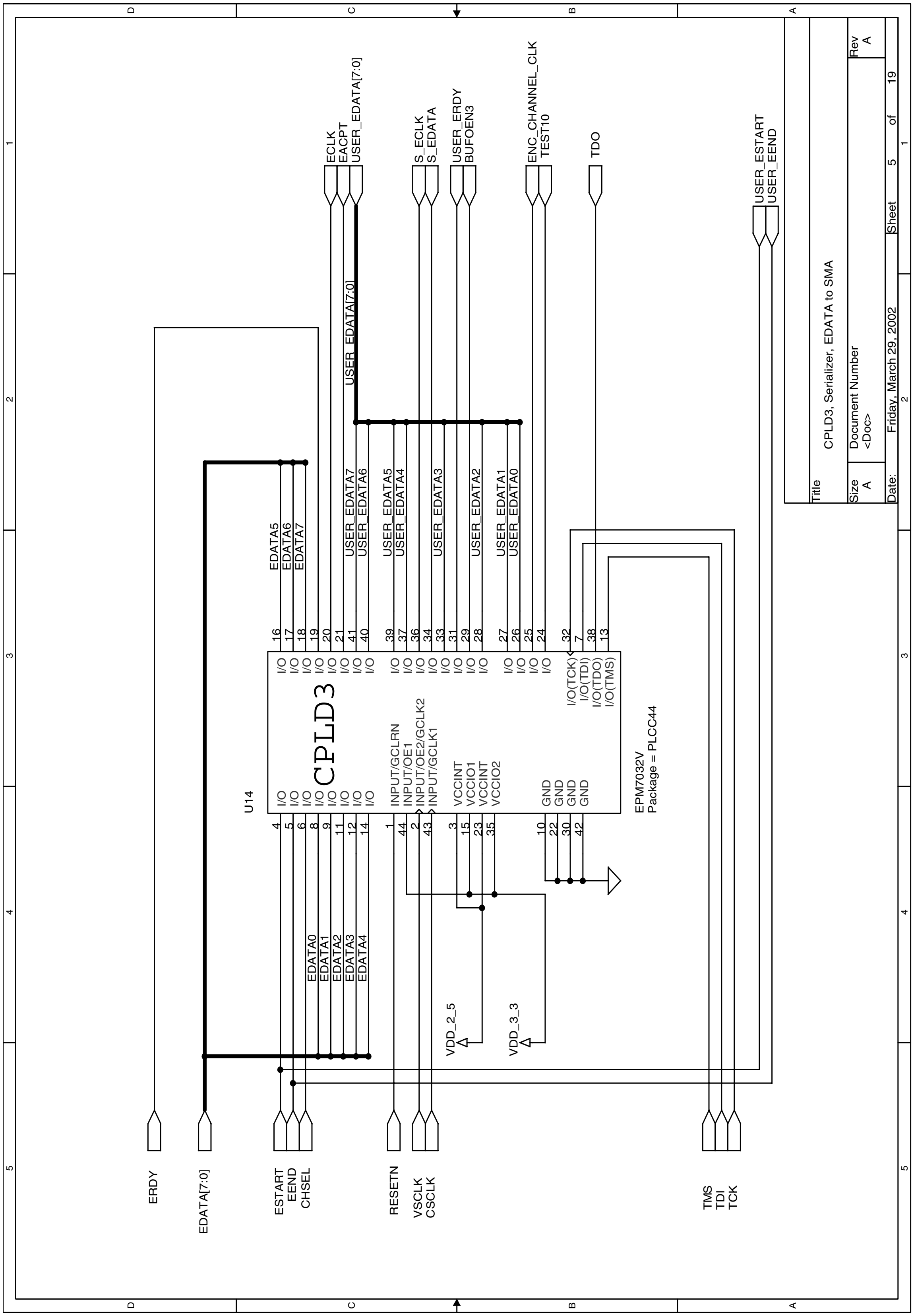
APPENDIX A: BOARD SCHEMATICS

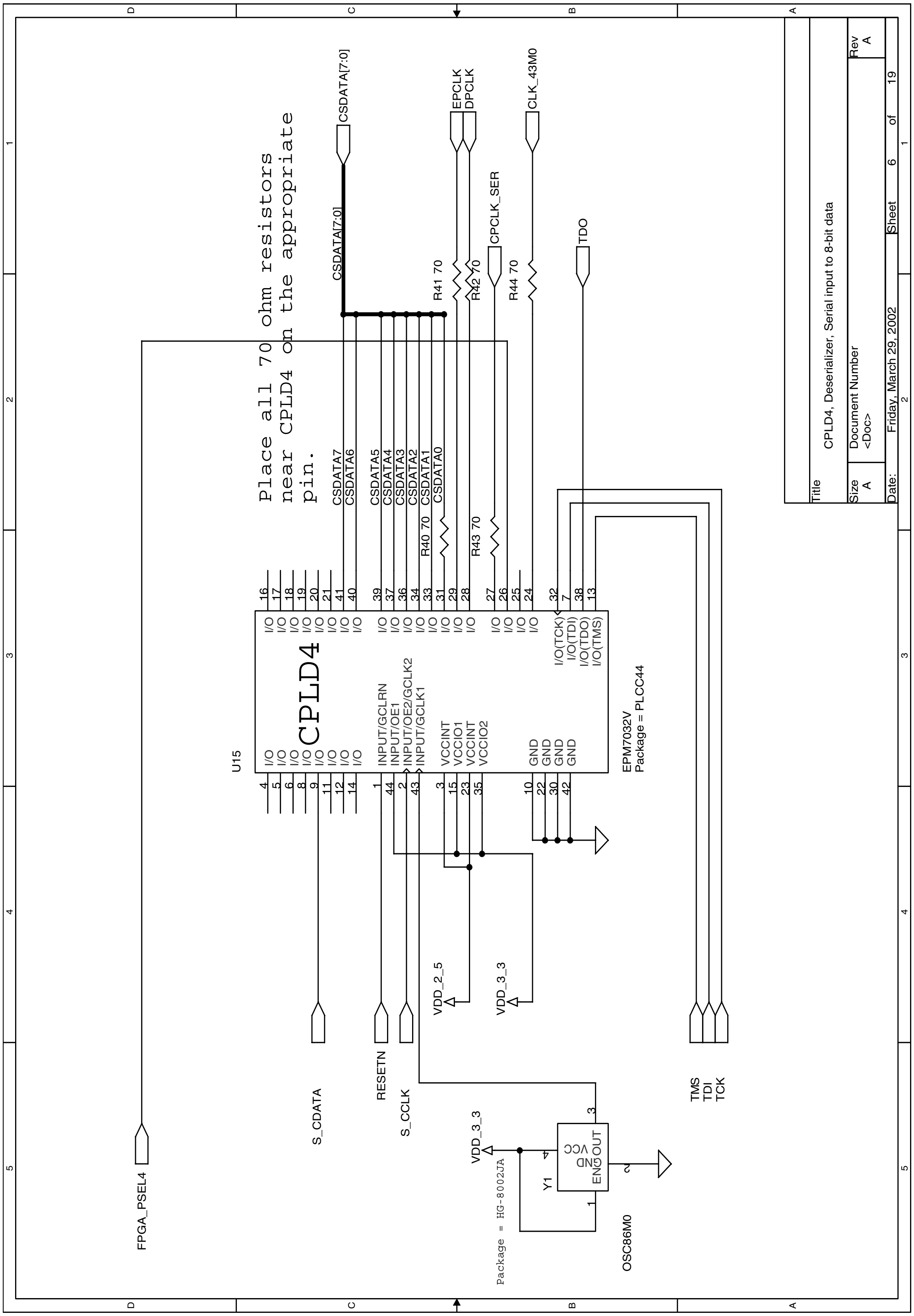


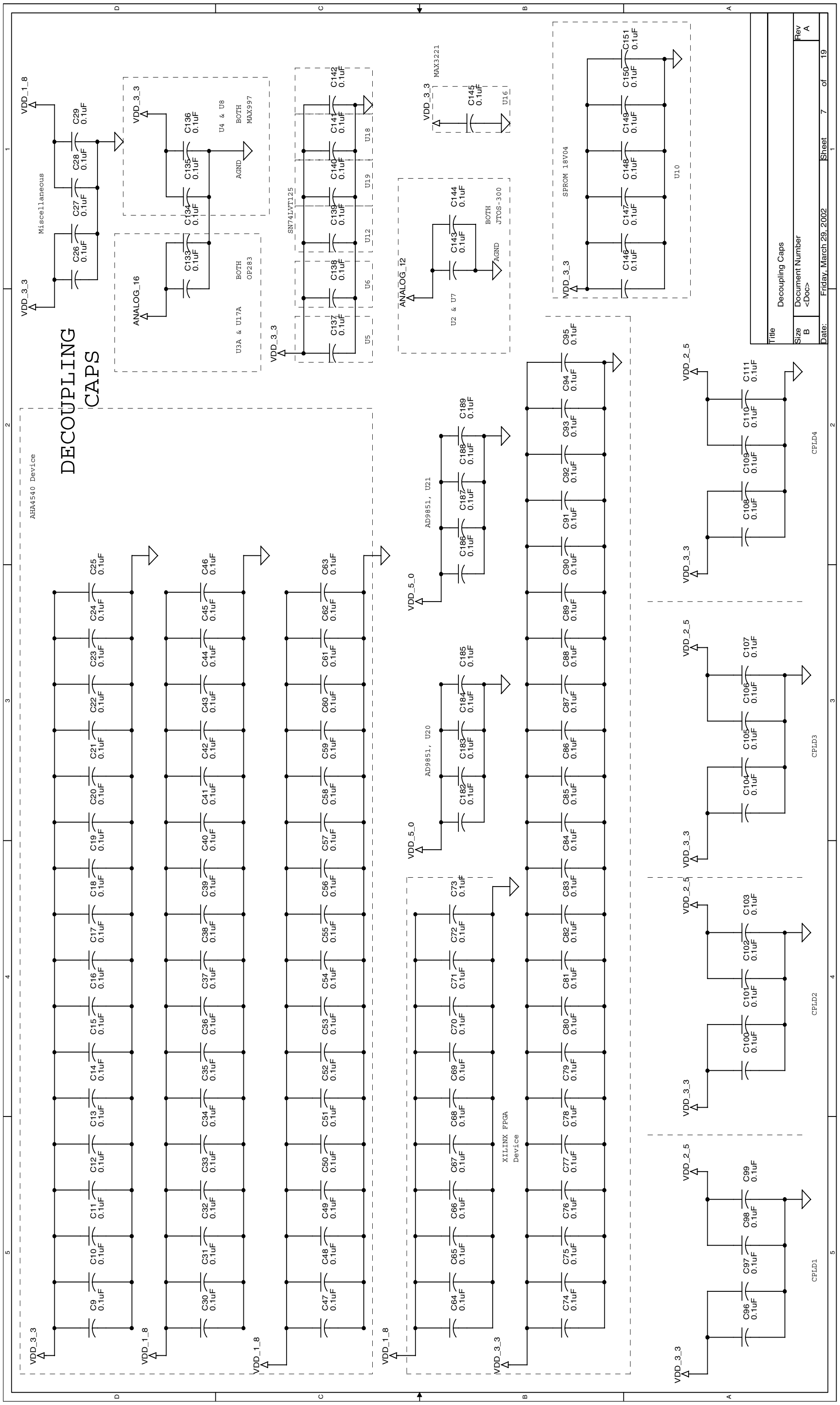

AHA4540
Leave test output pins unconnected
AHA4540
Package = PQ150
Title
AHA4540 Device
Size
B
Document Number
AHA4540_EVB
Rev
A
Date: Friday, March 29, 2002
Sheet 2 of 19

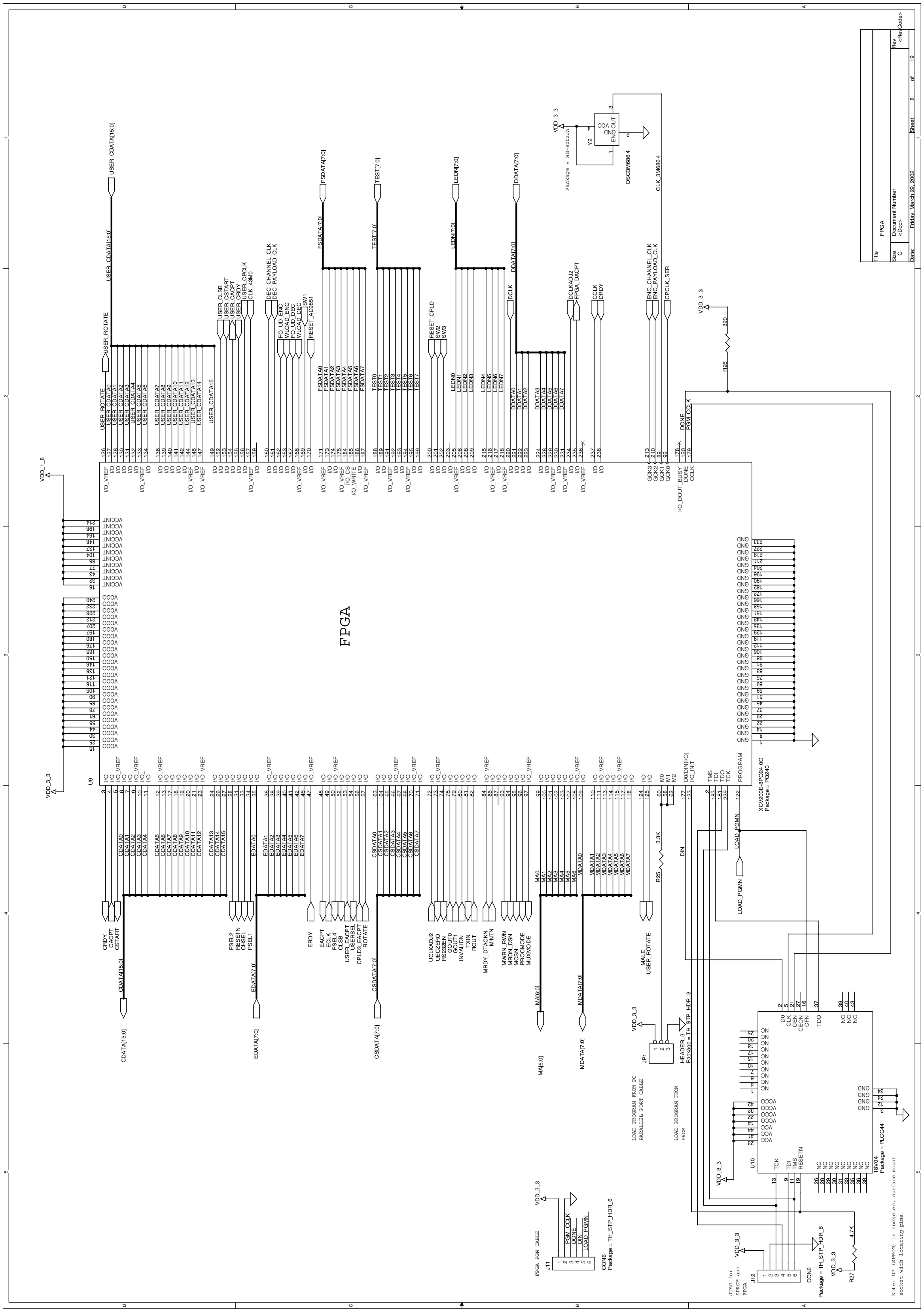

CPLD1, Deserializer, SMA or USER Port to UDATA
Document Number
AHA4540_EVM
Size
A
Rev
A
Date: Friday, March 29, 2002
Sheet 3 of 19
CPLD1
U11
EPM7032V
Package = PLCC44
UCLK
URDY
UDATA[7:0]
BUFOEN1
CLK_SYNC_OUT
USER_UACPT
TDO
USER_UDATA[7:0]
USER_UDATA5
USER_UDATA6
USER_UDATA7
UDATA0
UDATA1
UDATA7
UDATA6
UDATA5
UDATA4
UDATA3
UDATA2
USER_UDATA0
USER_UDATA1
USER_UDATA2
USER_UDATA3
USER_UDATA4
INPUT/GCLRN
INPUT/OE1
INPUT/OE2/GCLK2
INPUT/GCLK1
VCCINT
VCCIO1
VCCINT
VCCIO2
GND
I/O(TCK)
I/O(TDI)
I/O(TDO)
I/O(TMS)
VDD_2_5
VDD_3_3
RESETN
CLK_SYNC
S_UCLK
TMS
TDI
TCK
UECZERO
S_UDATA
USER_URDY
CHSEL
FPGA_PSEL
USER_UCLK
UACPT
USER_UDATA[7:0]

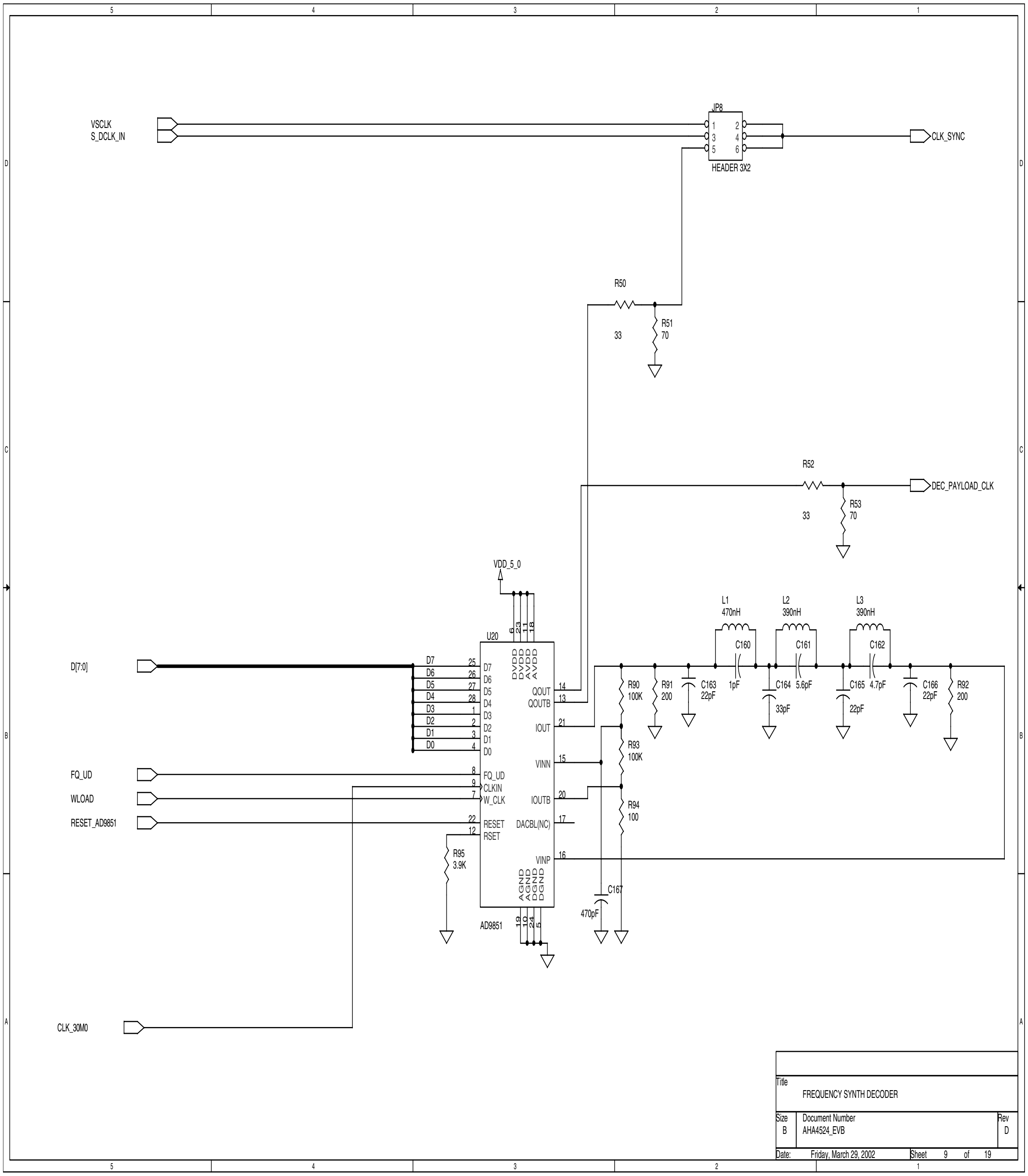

Title
CPLD2, Serializer, DDATA to SMA
Size
A
Document Number
<Doc>
Rev
A
Date: Friday, March 29, 2002
Sheet 4 of 19
U13
CPLD2
EPM7032V
Package = PLCC44
S_DDATA
S_DCLK
BUFOEN2
USER_DRDY
USER_DDATA[7:0]
DACPT
DCLK
TDO
DDATA[7:0]
DDATA_[7:0]
DRDY
FPGA_PSEL2
RESETN
SCLK
VDD_2_5
VDD_3_3
TMS
TDI
TCK
USERSEL
USER_DACPT
FPGA_DACPT
INPUT/GCLRN
INPUT/OE1
INPUT/OE2/GCLK2
INPUT/GCLK1
VCCINT
VCCIO1
VCCINT
VCCIO2
GND
I/O(TCK)
I/O(TDI)
I/O(TDO)
I/O(TMS)

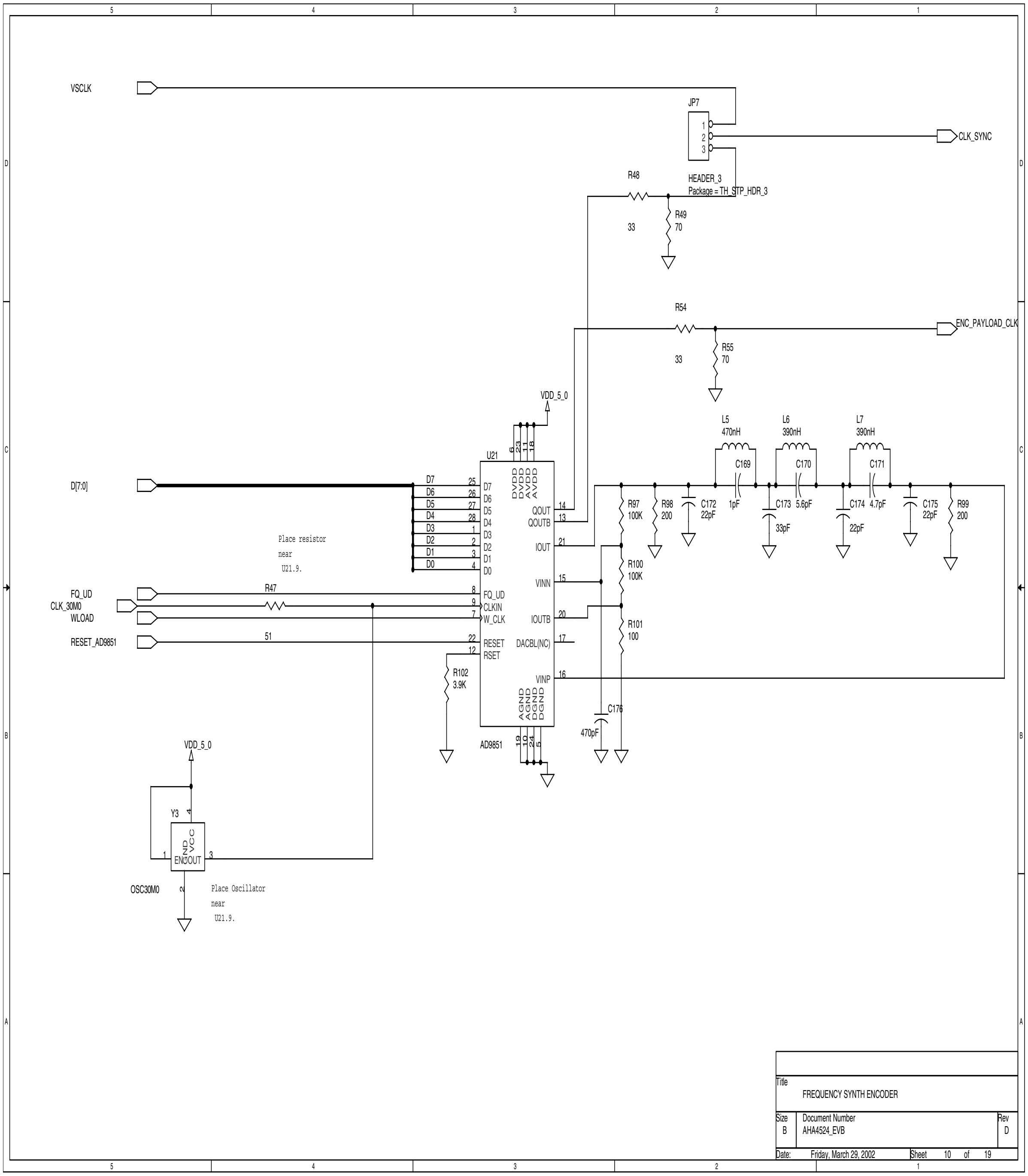

CPLD3
U14
EPM7032V
Package = PLCC44
ERDY
EDATA[7:0]
ESTART
EEND
CHSEL
RESETN
VSCLK
CSCLK
TMS
TDI
TCK
ECLK
EACPT
USER_EDATA[7:0]
S_ECLK
S_EDATA
USER_ERDY
BUFOEN3
ENC_CHANNEL_CLK
TEST10
TDO
USER_ESTART
USER_EEND
VDD_2_5
VDD_3_3
Title
CPLD3, Serializer, EDATA to SMA
Size A
Document Number
<Doc>
Rev A
Date: Friday, March 29, 2002
Sheet 5 of 19

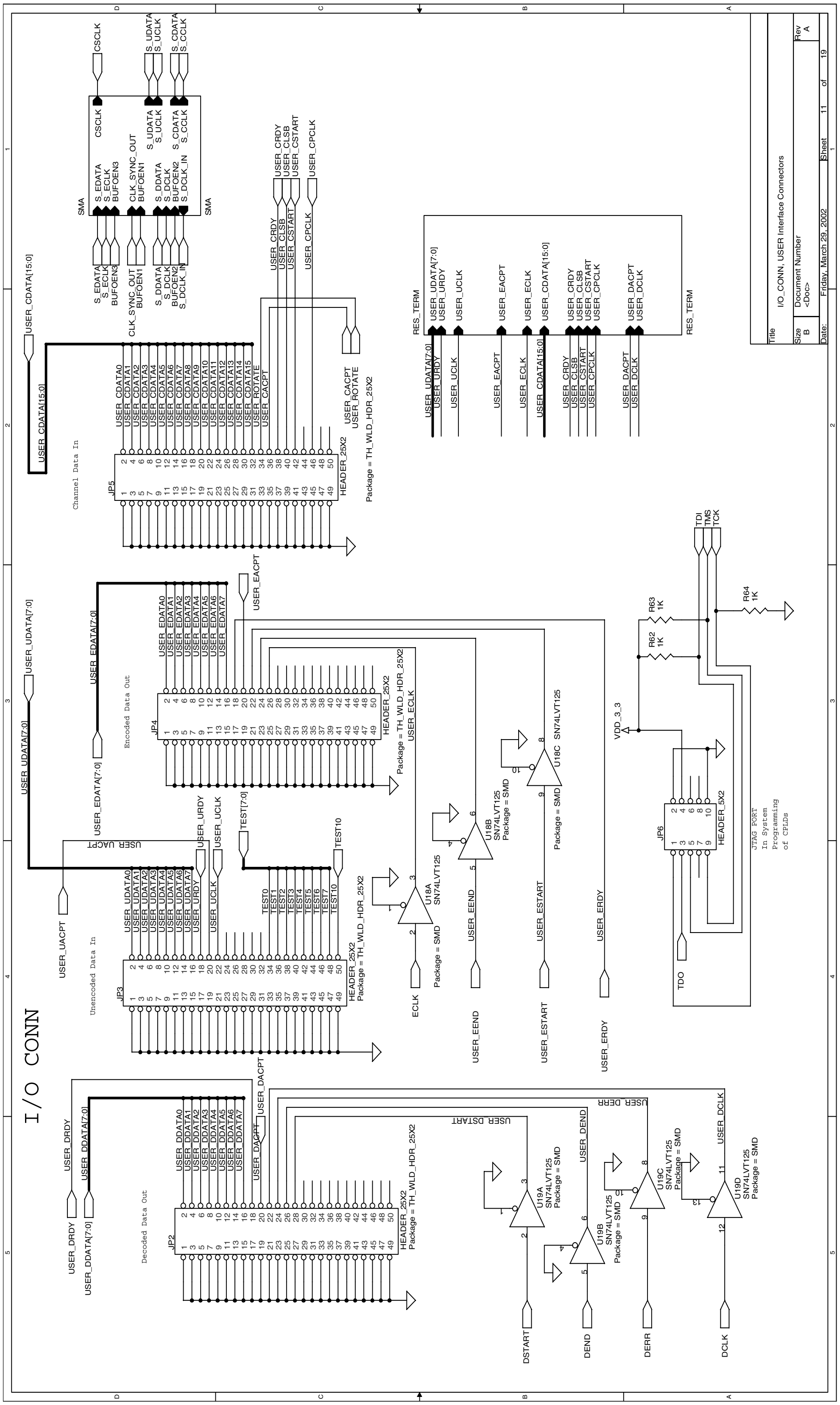

Place all 70 ohm resistors near CPLD4 on the appropriate pin.
CPLD4
U15
EPM7032V
Package = PLCC44
CSDATA[7:0]
EPCLK
DPCLK
CLK_43M0
CPCLK_SER
TDO
R40 70
R41 70
R42 70
R43 70
R44 70
INPUT/GCLRN
INPUT/OE1
INPUT/OE2/GCLK2
INPUT/GCLK1
VCCINT
VCCIO1
VCCINT
VCCIO2
GND
I/O(TCK)
I/O(TDI)
I/O(TDO)
I/O(TMS)
VDD_2_5
VDD_3_3
S_CDATA
RESETN
S_CCLK
TMS
TDI
TCK
FPGA_PSEL4
Y1
OSC86M0
Package = HG-8002JA
Title
CPLD4, Deserializer, Serial input to 8-bit data
Size
A
Document Number
<Doc>
Rev
A
Date: Friday, March 29, 2002
Sheet 6 of 19

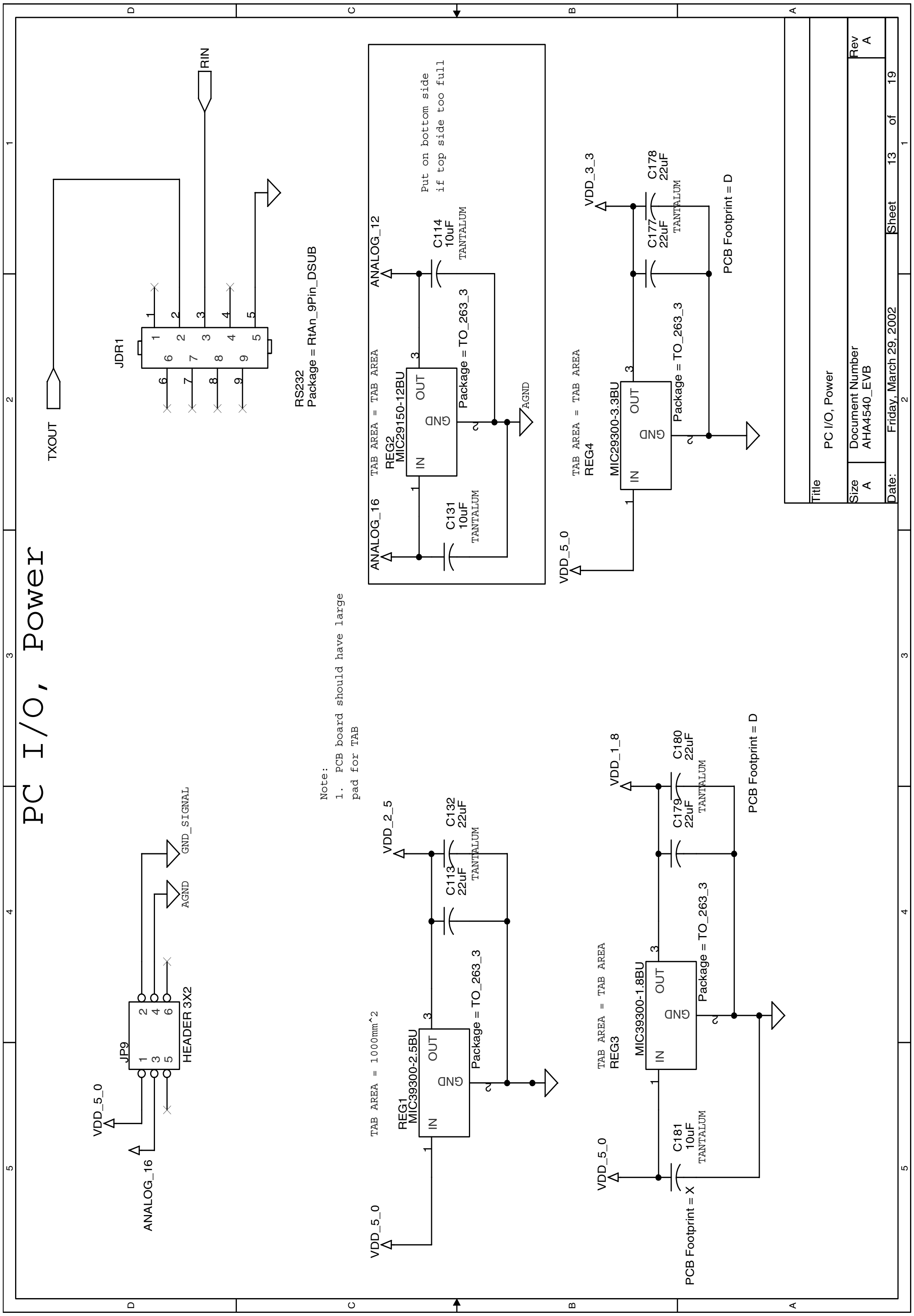

DECOUPLING CAPS
Miscellaneous
AHA4540 Device
XILINX FPGA Device
SN74LVT125
MAX3221
SPROM 18V04
AD9851, U21
AD9851, U20
CPLD1
CPLD2
CPLD3
CPLD4
VDD_3_3
VDD_1_8
VDD_2_5
VDD_5_0
ANALOG_16
ANALOG_12
AGND
Title
Decoupling Caps
Size
B
Document Number
<Doc>
Rev
A
Date: Friday, March 29, 2002
Sheet 7 of 19

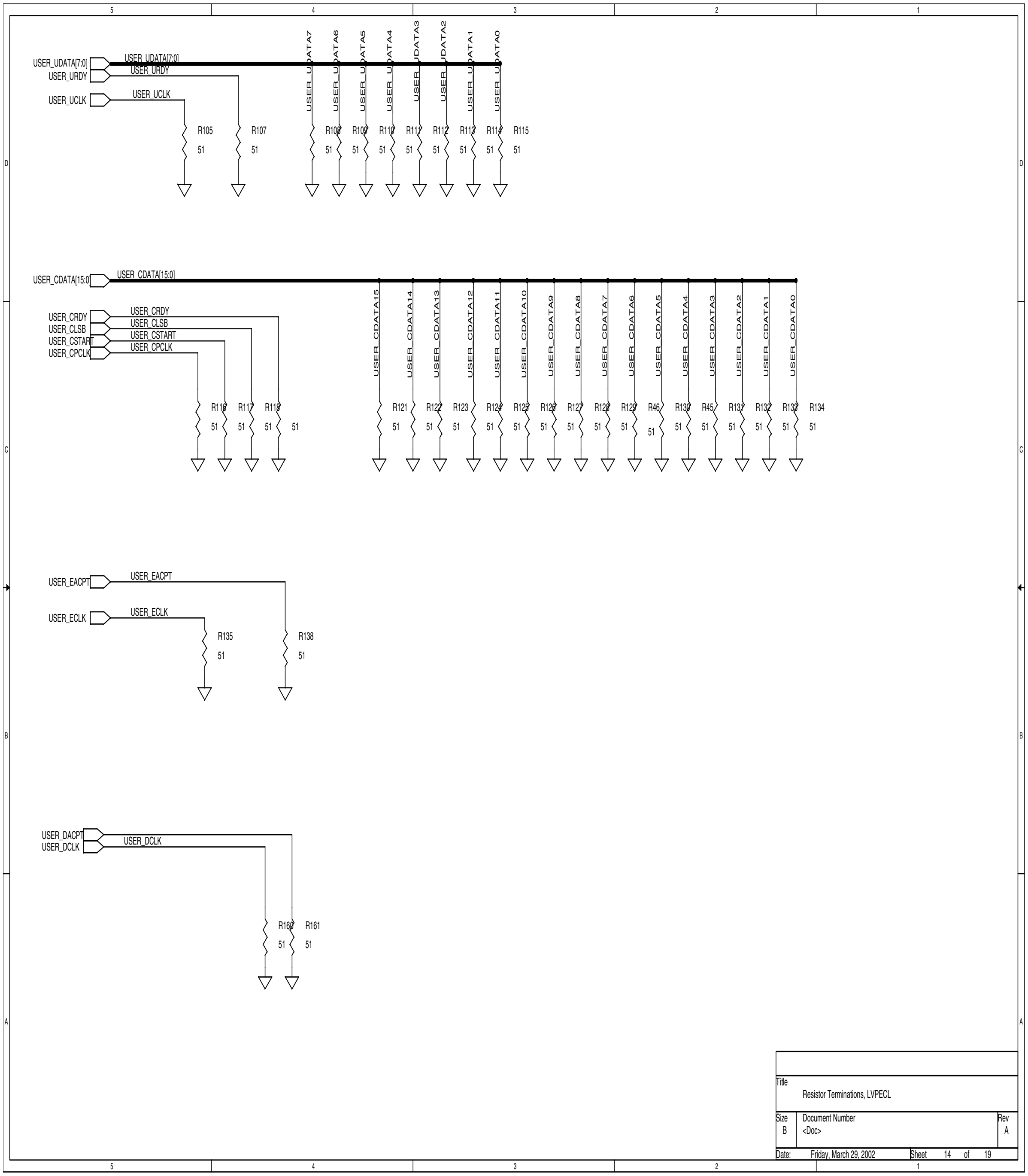

FPGA
Title
FPGA
Document Number
<Doc>
Friday, March 29, 2002

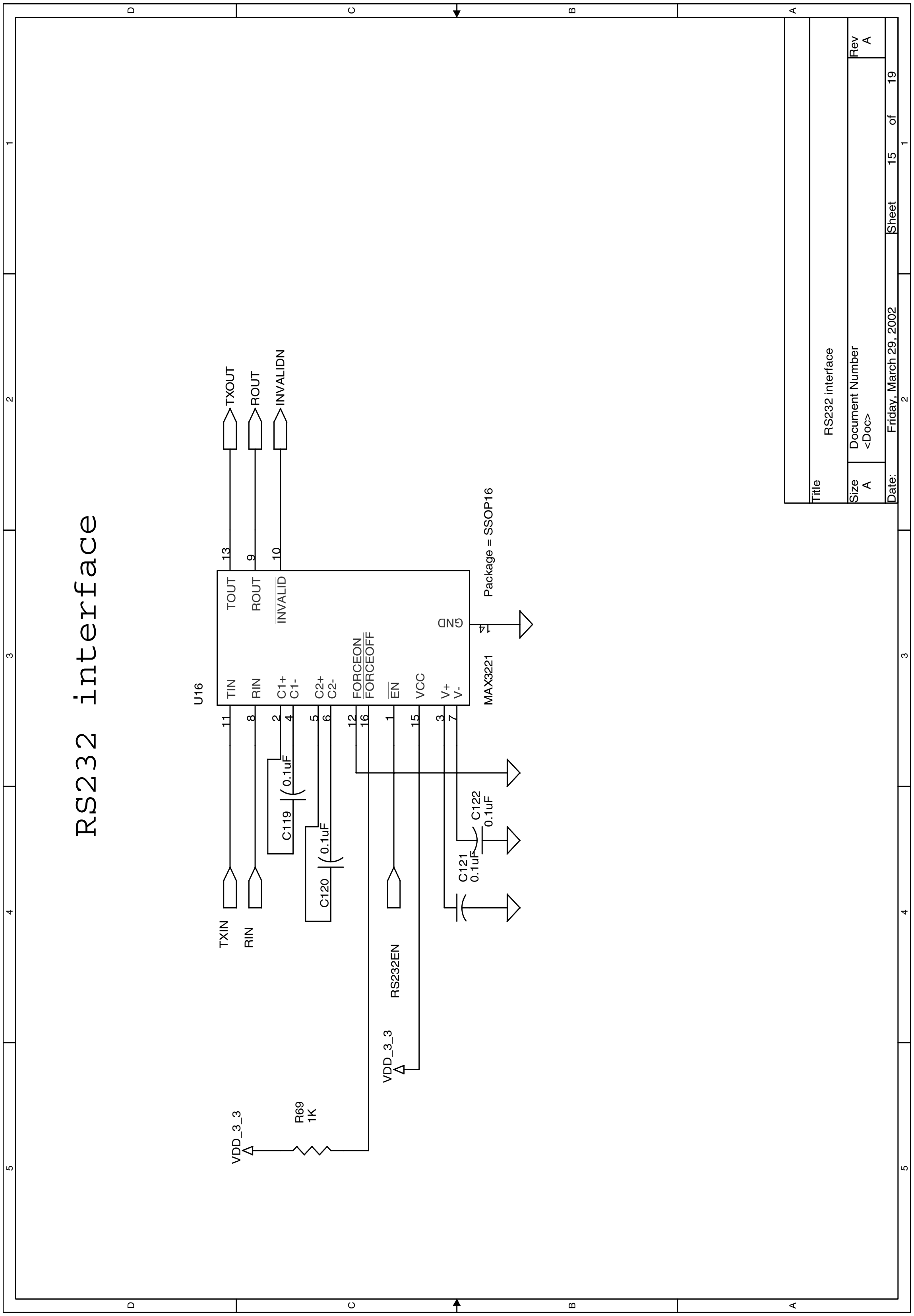

VSCLK
S_DCLK_IN
JP8
HEADER 3X2
CLK_SYNC
R50
33
R51
70
R52
33
R53
70
DEC_PAYLOAD_CLK
VDD_5_0
U20
AD9851
D[7:0]
FQ_UD
WLOAD
RESET_AD9851
CLK_30M0
R95
3.9K
QOUT
QOUTB
IOUT
VINN
IOUTB
DACBL(NC)
VINP
RESET
RSET
FQ_UD
CLKIN
W_CLK
L1
470nH
L2
390nH
L3
390nH
C160
1pF
C161
5.6pF
C162
4.7pF
R90
100K
R91
200
C163
22pF
C164
33pF
C165
22pF
C166
22pF
R92
200
R93
100K
R94
100
C167
470pF
Title
FREQUENCY SYNTH DECODER
Size
B
Document Number
AHA4524_EVB
Rev
D
Date: Friday, March 29, 2002
Sheet 9 of 19

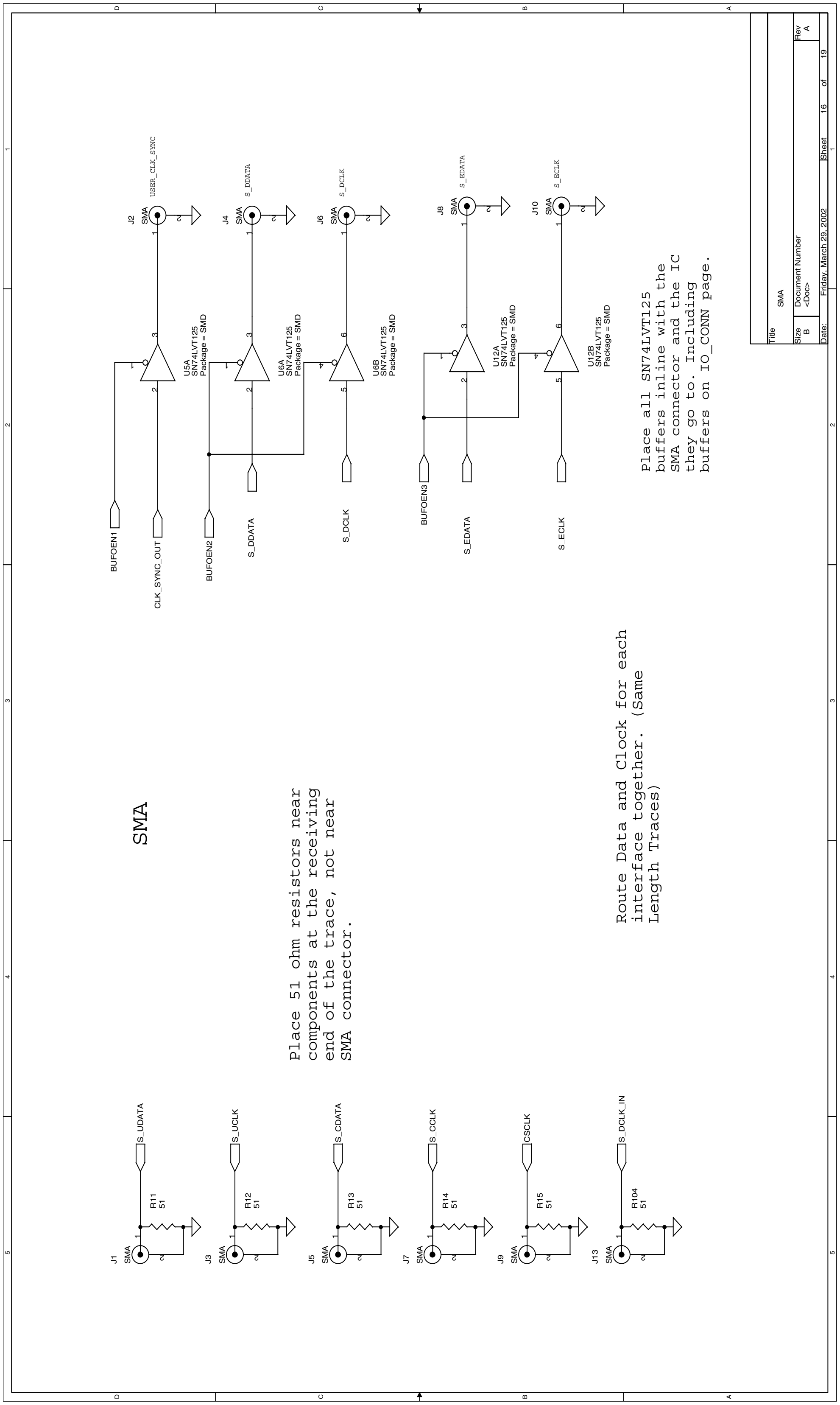

VSCLK
JP7
HEADER_3
Package = TH_STP_HDR_3
CLK_SYNC
R48
33
R49
70
R54
33
R55
70
ENC_PAYLOAD_CLK
VDD_5_0
L5
470nH
L6
390nH
L7
390nH
C169
C170
C171
U21
D[7:0]
D7 D6 D5 D4 D3 D2 D1 D0
DVDD DVDD AVDD AVDD
QOUT
QOUTB
IOUT
VINN
IOUTB
DACBL(NC)
VINP
FQ_UD
CLKIN
W_CLK
RESET
RSET
R97
100K
R98
200
C172
22pF
1pF
C173
5.6pF
33pF
C174
4.7pF
22pF
C175
22pF
R99
200
R100
100K
R101
100
Place resistor
near
U21.9.
FQ_UD
CLK_30M0
WLOAD
RESET_AD9851
R47
51
R102
3.9K
AGND AGND DGND DGND
AD9851
C176
470pF
VDD_5_0
Y3
GND
VCC
EN
OUT
OSC30M0
Place Oscillator
near
U21.9.
Title
FREQUENCY SYNTH ENCODER
Size
B
Document Number
AHA4524_EVB
Rev
D
Date:
Friday, March 29, 2002
Sheet 10 of 19

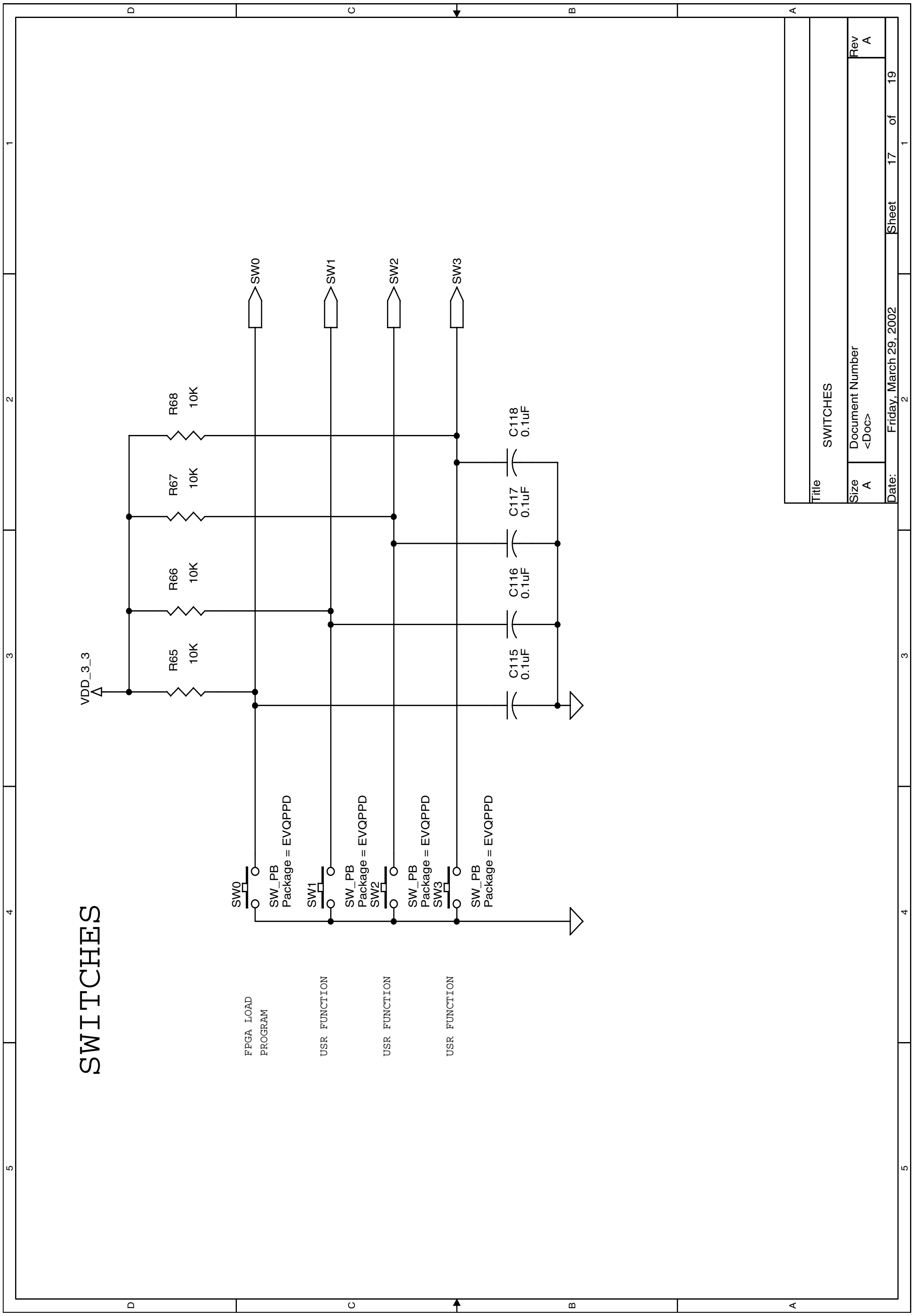

I/O CONN
I/O_CONN, USER Interface Connectors
Friday, March 29, 2002
Sheet 11 of 19

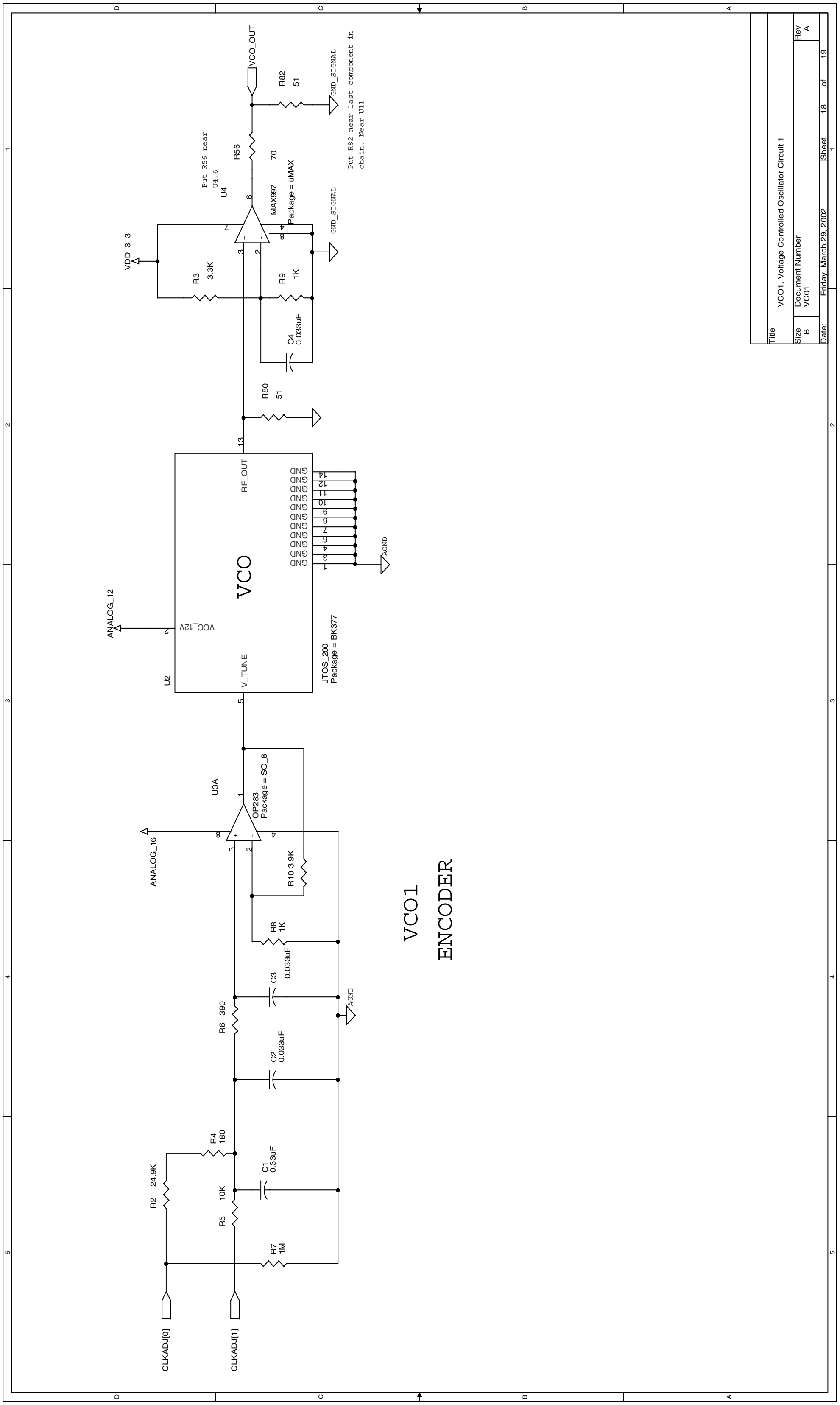

PC I/O, Power
TXOUT
RIN
JDR1
RS232
Package = RtAn_9Pin_DSUB
JP9
HEADER 3X2
VDD_5_0
ANALOG_16
GND_SIGNAL
AGND
Note:
1. PCB board should have large pad for TAB
ANALOG_16
ANALOG_12
TAB AREA = TAB AREA
REG2
MIC29150-12BU
Package = TO_263_3
C131
10uF
TANTALUM
C114
10uF
TANTALUM
Put on bottom side if top side too full
VDD_5_0
VDD_3_3
TAB AREA = TAB AREA
REG4
MIC29300-3.3BU
Package = TO_263_3
C177
22uF
TANTALUM
C178
22uF
PCB Footprint = D
VDD_5_0
VDD_2_5
TAB AREA = 1000mm^2
REG1
MIC39300-2.5BU
Package = TO_263_3
C113
22uF
C132
22uF
TANTALUM
VDD_5_0
VDD_1_8
TAB AREA = TAB AREA
REG3
MIC39300-1.8BU
Package = TO_263_3
C181
10uF
TANTALUM
C179
22uF
C180
22uF
TANTALUM
PCB Footprint = X
PCB Footprint = D
Title
PC I/O, Power
Size
A
Document Number
AHA4540_EVB
Rev
A
Date: Friday, March 29, 2002
Sheet 13 of 19


USER_UDATA[7:0]
USER_URDY
USER_UCLK
USER_CDATA[15:0]
USER_CRDY
USER_CLSB
USER_CSTART
USER_CPCLK
USER_EACPT
USER_ECLK
USER_DACPT
USER_DCLK
R105
R107
R115
R121
R123
R134
R135
R138
R161
51
Title
Resistor Terminations, LVPECL
Size
B
Document Number
<Doc>
Rev
A
Date: Friday, March 29, 2002
Sheet 14 of 19

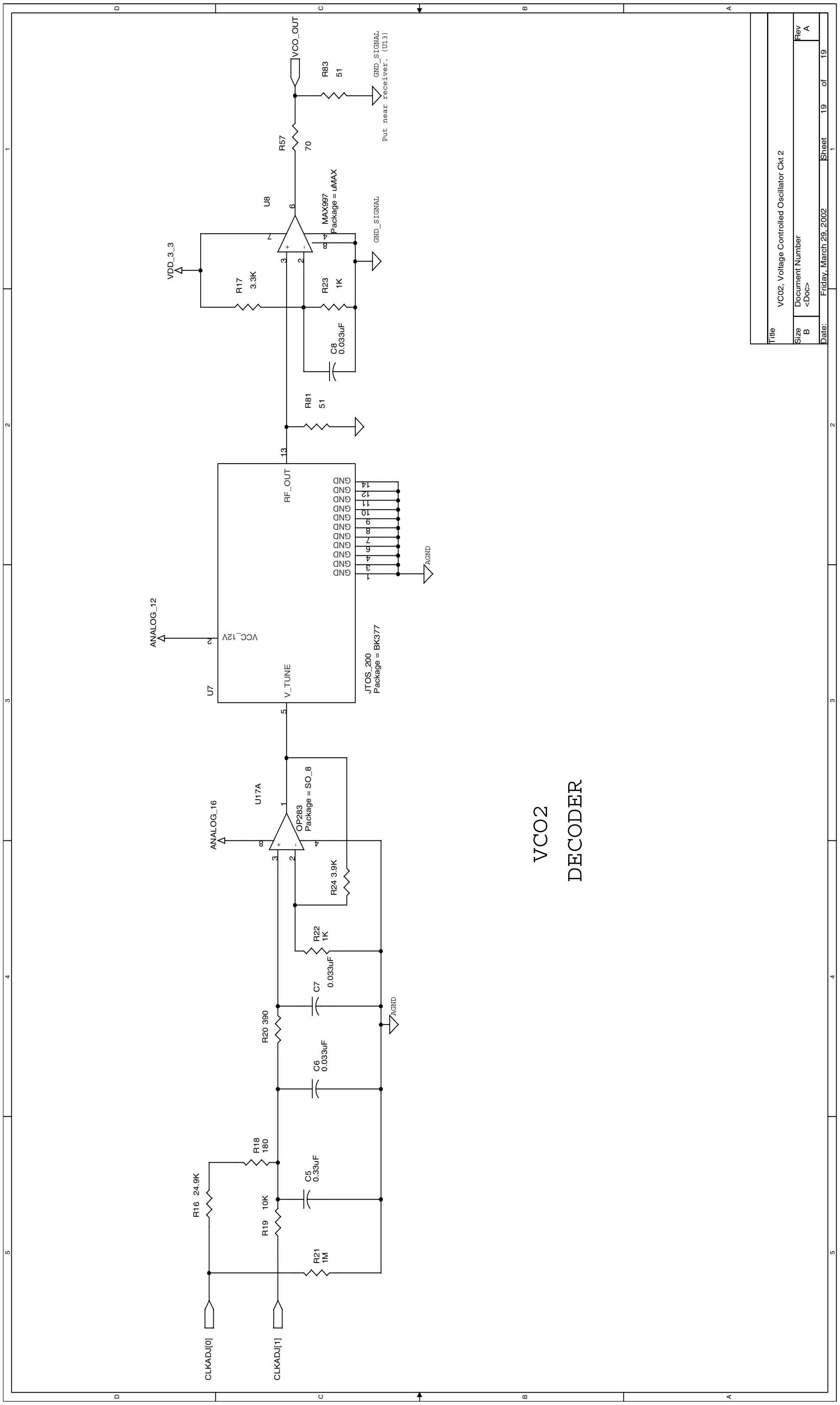

RS232 interface
U16
MAX3221
Package = SSOP16
TIN
RIN
C1+
C1-
C2+
C2-
FORCEON
FORCEOFF
EN
VCC
V+
V-
TOUT
ROUT
INVALID
GND
TXOUT
ROUT
INVALIDN
TXIN
RIN
RS232EN
VDD_3_3
VDD_3_3
R69
1K
C119
0.1uF
C120
0.1uF
C121
0.1uF
C122
0.1uF
Title
RS232 interface
Size
A
Document Number
<Doc>
Rev
A
Date:
Friday, March 29, 2002
Sheet
15
of
19

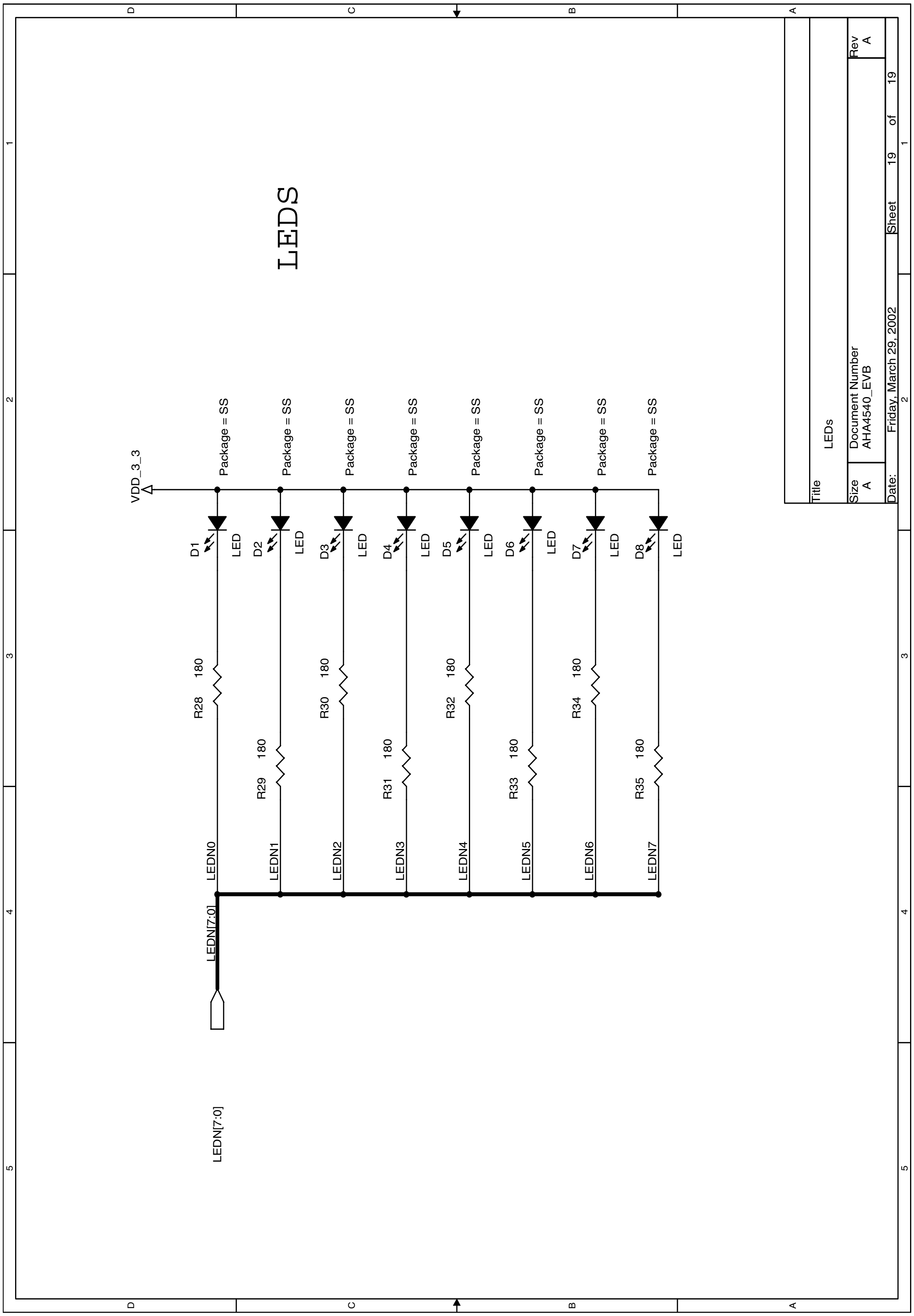

SMA
Place 51 ohm resistors near components at the receiving end of the trace, not near SMA connector.
Route Data and Clock for each interface together. (Same Length Traces)
Place all SN74LVT125 buffers inline with the SMA connector and the IC they go to. Including buffers on IO_CONN page.
J1 SMA R11 51 S_UDATA
J3 SMA R12 51 S_UCLK
J5 SMA R13 51 S_CDATA
J7 SMA R14 51 S_CCLK
J9 SMA R15 51 CSCLK
J13 SMA R104 51 S_DCLK_IN
J2 SMA USER_CLK_SYNC
J4 SMA S_DDATA
J6 SMA S_DCLK
J8 SMA S_EDATA
J10 SMA S_ECLK
U5A SN74LVT125 Package = SMD
U6A SN74LVT125 Package = SMD
U6B SN74LVT125 Package = SMD
U12A SN74LVT125 Package = SMD
U12B SN74LVT125 Package = SMD
BUFOEN1
CLK_SYNC_OUT
BUFOEN2
S_DDATA
S_DCLK
BUFOEN3
S_EDATA
S_ECLK
Title SMA
Size B
Document Number <Doc>
Rev A
Date: Friday, March 29, 2002
Sheet 16 of 19


SWITCHES
VDD_3_3
R65 10K
R66 10K
R67 10K
R68 10K
C115 0.1uF
C116 0.1uF
C117 0.1uF
C118 0.1uF
SW0
SW1
SW2
SW3
SW0 SW_PB Package = EVQPPD
SW1 SW_PB Package = EVQPPD
SW2 SW_PB Package = EVQPPD
SW3 SW_PB Package = EVQPPD
FPGA LOAD PROGRAM
USR FUNCTION
USR FUNCTION
USR FUNCTION
Title
SWITCHES
Size A
Document Number <Doc>
Rev A
Date: Friday, March 29, 2002
Sheet 17 of 19



VCO1
ENCODER
VCO
U2
VCC_12V
V_TUNE
RF_OUT
JTOS_200
Package = BK377
ANALOG_12
ANALOG_16
U3A
OP283
Package = SO_8
U4
MAX997
Package = uMAX
Put R56 near U4.6
Put R82 near last component in chain. Near U11
VCO_OUT
VDD_3_3
GND_SIGNAL
AGND
CLKADJ[0]
CLKADJ[1]
R2 24.9K
R4 180
R5 10K
R6 390
R7 1M
R8 1K
R10 3.9K
R3 3.3K
R9 1K
R56 70
R80 51
R82 51
C1 0.33uF
C2 0.033uF
C3 0.033uF
C4 0.033uF
Title
VCO1, Voltage Controlled Oscillator Circuit 1
Size
B
Document Number
VC01
Rev
A
Date: Friday, March 29, 2002
Sheet 18 of 19


VCO2
DECODER
VCO_OUT
R83
51
GND_SIGNAL
Put near receiver. (U13)
R57
70
U8
MAX997
Package = uMAX
GND_SIGNAL
VDD_3_3
R17
3.3K
R23
1K
C8
0.033uF
R81
51
RF_OUT
GND
AGND
ANALOG_12
VCC_12V
V_TUNE
U7
JTOS_200
Package = BK377
ANALOG_16
U17A
OP283
Package = SO_8
R24 3.9K
R22
1K
C7
0.033uF
R20 390
AGND
C6
0.033uF
R18
180
R16 24.9K
C5
0.33uF
R19 10K
R21
1M
CLKADJ[0]
CLKADJ[1]
Title
VC02, Voltage Controlled Oscillator Ckt 2
Size
B
Document Number
<Doc>
Rev
A
Date: Friday, March 29, 2002
Sheet 19 of 19


LEDS
VDD_3_3
D1 LED Package = SS
D2 LED Package = SS
D3 LED Package = SS
D4 LED Package = SS
D5 LED Package = SS
D6 LED Package = SS
D7 LED Package = SS
D8 LED Package = SS
R28 180
R29 180
R30 180
R31 180
R32 180
R33 180
R34 180
R35 180
LEDN0
LEDN1
LEDN2
LEDN3
LEDN4
LEDN5
LEDN6
LEDN7
LEDN[7:0]
LEDN[7:0]
Title
LEDs
Size
A
Document Number
AHA4540_EVB
Rev
A
Date: Friday, March 29, 2002
Sheet 19 of 19