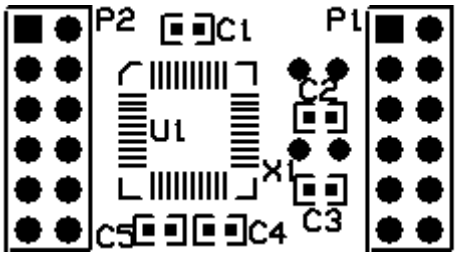
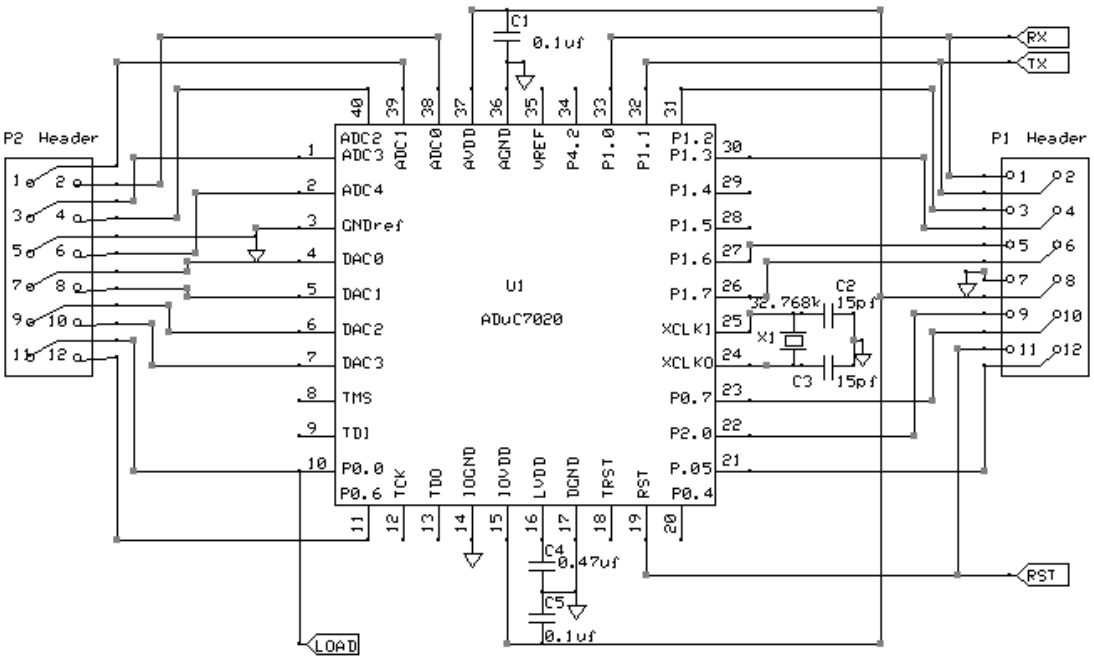
FORTHstamp Board

Chen-Hanson Ting, Offete Enterprises

A Single Chip Computer

FORTHstamp is a true single chip computer, based upon an Analog Devices' chip ADuC7020. It includes a 32-bit ARM7 microprocessor and a host of important peripheral devices, including a serial communication port (UART), many general purpose digital inputs and output ports (GPIO), an analog to digital converter (ADC), 4 digital to analog converters (DAC), and 92 Kbytes of programmable (flash) memory. eForth is installed in flash memory and is used to drive all the devices. It is an excellent platform to use Forth to develop real time applications. It can runs on 3V battery power. Power supply, ground, and all IO signals are brought out on two 12-pin headers. The layout and schematic of FORTHstamp board are shown here:





Signals brought to the two headers P1 and P2 are:

P1 Header:

Pin1	P1.0	RX of UART
Pin2	P1.1	TX of UART
Pin3	P1.2	Digital IO
Pin4	P1.3	Digital IO
Pin5	P1.6	Digital IO
Pin6	P1.7	Digital IO
Pin7		Ground
Pin8		VDD Power 3.3V
Pin9	P2.0	Digital IO
Pin10	P0.7	Digital IO
Pin11	P0.5	Digital IO
Pin12		RESET

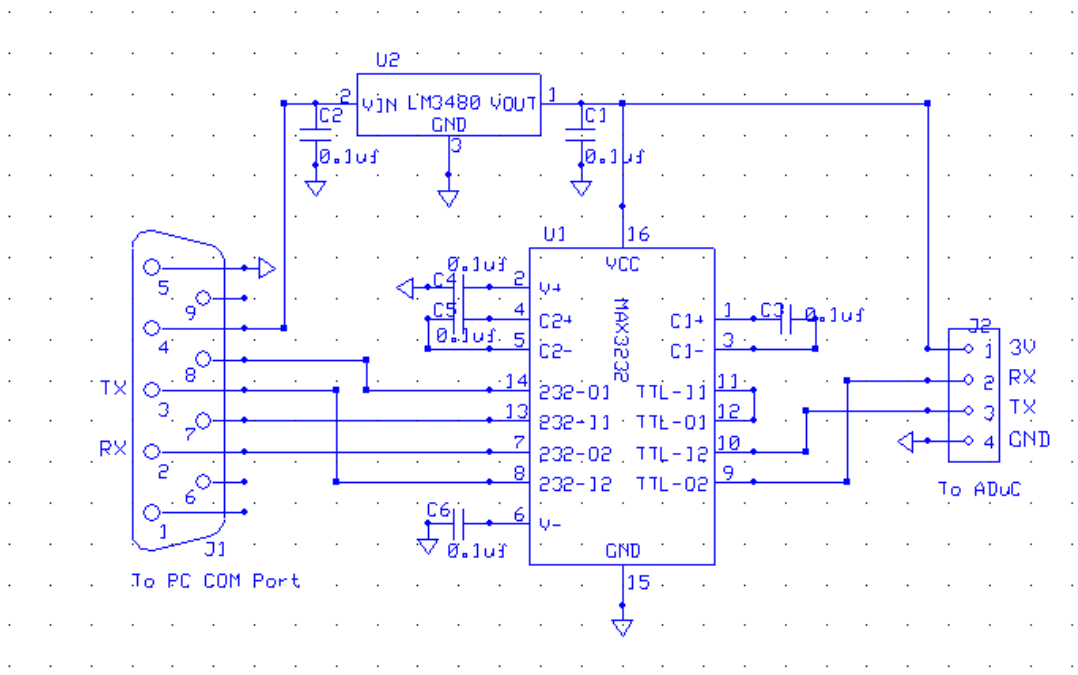
P2 Header:

Pin1		ADC0
Pin2		ADC1
Pin3		Ground
Pin4		ADC2
Pin5		ADC3
Pin6		ADC4
Pin7		DAC0
Pin8		DAC1
Pin9		DAC2
Pin10		DAC3
Pin11	P0.0	LOAD
Pin12	P0.6	Digital IO

A 3 Volt power supply must be connected to Pin8 of Header P1, ground to Pin7 of Header 1. To reset ADuC7020, Pin12 of Header P1 must be pulled to 3 V through a 1k resister and connected to ground through a RESET switch. If you have to reprogram its flash memory, Pin11 of Header P2 must be pulled to ground through a 1k resister and a LOAD switch.

eForth is preloaded on the ADuC7020 chip and can be exercised through the HyperTerminal communication application on most PCs. It must be connected to a COM port of a PC through a UART33 board which has a converter chip like MAX3232 as shown below. The J2 header on UART33 provides connections to

Aduc7020 chip. Pin2 of J2 transmits signals to RX, Pin1 of Header P1, and Pin3 of J2 receives signals from TX, Pin2 of Header P1. J1 is a DB9 connector which connects to a COM port on PC.



J2 Header:

Pin1	3.3V	
Pin2	RX	Connect to Pin1 of P1
Pin3	TX	Connect to Pin2 of P1
Pin4	GND	Connect to Pin7 of P1

FORTHstamp works with Windows through the HyperTerminal program. HyperTerminal must be configured properly at 57600 baud, 8 data bits, 1 stop bit, no parity, and no flow control, to communicate with FORTHstamp. Power up FORTHstamp and HyperTerminal should display the following message:

ADuC eForth v5.10

Here are few eForth commands you can type into the HyperTerminal to test this eForth system:

```
WORDS
HEX 0 200 DUMP
SEE WORDS
HERE .
1 2 + .
```

```
: TEST1 1 2 3 4 5 ;
TEST1
.S
: TEST2 10 FOR R@ . NEXT ;
TEST2
: TEST3 IF 1 ELSE 2 THEN . ;
0 TEST3
1 TEST3
: TEST4 CR ." HELLO, WORLD!" ;
TEST4
```

Using FORTHstamp

Download the data sheet of ADuC702X Series Microconverters from www.analog.com to learn the peripheral devices included in this chip.

Let us try to use P0.0 and P0.6 as input ports, and P0.5 and P0.7 ports as output ports. These ports are configured for GPIO operation by writing a 0 into GP0CON at FFFF0400. GP0DAT at FFFF0420 controls data through this port. The lowest byte always shows the input state of P0. The highest byte controls I/O direction, 1 for outputting and 0 for inputting. The next high byte controls the output data. Here are some examples of the operations:

```
HEX
0 GP0CON !              Set up port P0
20000000 GP0DAT !       Output 0V on P0.5
20200000 GP0DAT !       Output 3.3V on P0.5
GP0DAT ?                Read P0 port
```

DAC0CON at FFFF0600 and DAC0DAT at FFFF0604 control Digital-to-Analog Converter DAC0. If you connect a voltmeter to the output of the digital-to-analog converter DAC0 on Pin7 of Header P2, you can type the commands to change voltage reading on the voltmeter:

```
13 DAC0CON !            Set up DAC0 to output in 0-3.3V range
0 DAC0DAT !             Output 0V
4000000 DAC0DAT !       Output 0.8V
8000000 DAC0DAT !       Output 1.6V
FF00000 DAC0DAT !       Output 3.3V
```

ADCCON at FFFF0500, ADCCP at FFFF0504 and ADCDAT at FFFF0510 control the operations of the Analog-to-Digital Converter ADC. To test ADC, connect the output of DAC0 on Pin7 of Header P2 to ADC0 on Pin1 of Header P2. Then type the following sequence of commands to get the ADC results:

20 ADCCON !	Turn on ADC
0 ADCCP !	Select ADC0 as input channel
E3 ADCCON !	Start ADC0
0 DAC0DAT !	Output 0V
ADCDAT ?	Read conversion results
8000000 DAC0DAT !	Output 1.6V
ADCDAT ?	Read conversion results
FF00000 DAC0DAT !	Output 3.3V
ADCDAT ?	Read conversion results

Programming FORTHstamp

A complete programming and application development system is provided for FORTHstamp in the form of ADuC Metacompiler. Metacompiler is a term used by Forth programmer to describe the process of building a new Forth system on an existing Forth system. ADuC Metacompiler runs under F#, which is an eForth system developed for Windows XP. It produces an Intel HEX file containing eForth system with application programs, which can be downloaded into the flash memory in ADuC 7020, using the flashing utility ARMWSD from Analog Devises.

The ADuC Metacompiler is contained in a zip file ADuCcompiler510.zip. The exact procedure to rebuild ADuC eForth v5.10 system is outlined as follows. Follow this procedure to build your application.

1. Unzip ADuCcompiler510.zip into a folder named ADuCcompiler510.
2. Double click F#.exe to bring a FEX selection window.
3. Double click ADuCcompiler510.fex in this window, and an F# console window opens and shows all the names and addresses of words compiled into the new target system. A new file ADuCef.hex is produced in the same folder.
4. Double click ARMWSD.exe to bring up the ADuC programming window.
5. Browse and select ADuCcompiler510\ADuCef.hex in the file selection window.
6. Connect FORTHstamp board to PC COM port through UART33 board. Apply 3V power to FORTHstamp.
7. Click START button.in ARMWSD window.
8. Hold down LOAD switch to ground Pin P0.0 of ADuC7020, and tap RESET

switch to ground RST pin momentarily. ARMWSD starts erasing, downloading, and verifying ADuCef.hex to ADuC7020.

9. Click RUN button.
10. Click EXIT button.
11. Bring up HyperTerminal on Windows XP. Make sure that the proper COM port is selected and the port is configured at 57600 baud, 1 start bit, 1 stop bit, no parity, and no flow control.
12. You can now type in eForth commands to interact with FORTHstamp.

All the source code required by ADuC Metacompiler are included for you to inspect and to help you write your applications. The following files require your special attention:

ADuCcompiler510.fex	Loader of metacompiler
ARM7asm.F	ARM7 assembler
ADuCkernel.F	Kernel words in ADuC eForth
ADuCeforth.F	High level words in ADuC eForth
ADuCscope.f	Sample application words
ADuCcompiler.f	Compiler commanded used in FORTHstamp

A user's programming manual will be release shortly to discuss all aspects of programming this FORTHstamp Board.

If you have any questions, call me at (650) 571-7639 or write to dr_tingchen@msn.com.