INSTALLING A FLOATING-POINT COPROCESSOR INTO ATARI MEGA ST COMPUTER

(revised)

This text describes how to install a 68881 floating point processor into an
Atari MEGA ST. The only other document on the subject that I could find was
Keith Sabine's description (ref.1) of a circuit using two 74HC133 ICs and a
74HC32 to decode the address, the output of the 74HC32 being connected to
the -AS input of the MC68881. That circuit is incomplete and contains an
error (no "chip select" signal) such that IT CAN'T WORK, though it won't
do any damage either. So here is a description of a correct way to connect
the FPU. The Atari ST is an almost extinct computer breed, but as the 68881
is cheap these days, maybe somebody will find it worth the trouble to make
this modification; total cost should not exceed 40 DEM.

DISCLAIMER
Although this modification worked fine for me, it may not do so for you.
I shall NOT be responsible for any damage, resulting from trying this
modification, to your computer and other equipment, or any damage to any-
body's life and health, or any galactic imbalance whatsoever. Whatever
you do, you do at your own risk. Be advised that in an open computer there
are exposed parts that are at mains line voltage (i.e. 220V AC) and you
can easily get KILLED if you are not careful!

Connecting the MC68881 to the MC68000:

NOTE: prefix "-" will be used for signals that are active in logical low
state, e.g. -AS, -CS, etc.

According to ref.2 the MC68881 FPU connects to the MC68000 CPU not as a
coprocessor, but rather as a "memory-mapped peripheral processor". Address
decoding is system dependent; on the Atari the FPU addresses start at
$FFFA40. This address is decoded from A5 through A23, FC0 and FC1 in the
user data space (FC2 is not used). Address signals A1 through A4 are not
decoded but connected directly to the FPU. Output of the decoder circuit is
connected to the active-low -CS (chip select) input of the FPU.

A0 input of the FPU is connected to logical 0 and -SIZE to +Vcc to indicate
to the FPU that a 16-bit data bus is used.

The 16 least-significant data pins (D0-D15) are connected to the equivalent
pins of the CPU. The 16 most significant data pins (D16-D31) are each
connected to the corresponding less-significant pin (e.g. D0 is connected
to D16, D1 to D17 ... D15 to D31).

-AS, -RESET, R/-W outputs of the CPU connect directly to the corresponding
inputs of the FPU.

-DSACK1 output of the FPU connects to the -DTACK of the CPU; -DSACK0 is not
used.

-LDS and -UDS from the CPU are gated together so that either can assert the
active-low -DS input of the FPU.

-SENSE input of the FPU may optionally be connected to ground, but this is
not required for correct operation. This signal is internally connected to
ground and can be used to indicate to the external hardware that a FPU is
present.

NC pin is reserved for Motorola use and not connected.

Implementation:

I used the 12MHz FPU in a 68-pin PGA package, as I accidentaly bought it
very cheap. Two 74HC11s, one 74HC27 and one 74HC30 IC are used for address
decoding, as shown in CIRCUIT.IMG. I so no point in designing a PCB for a
project that I was not sure will finction properly first time, and so the
circuit was assembled on a piece of universal printed circuit board with
0.1inch raster (one-sided). All components were arranged approximately as
shown in figure in LAYOUT.IMG, the interesting bit being that it was
convenient to install this daughterboard upside-down, i.e. with chips below
the board and all the connections (done with wire-wrap wires) on the upper

side. The reason for this was that the whole assembly occupied very little
height in the computer housing, as the FPU fitted nicely between the MMU
and the 32MHz clock. A connector with long pins was used, the pins
protruding for about 12mm through the board, so that effectively the
expansion port remained accesible for any other add-on that could fit above
the FPU board. As the expansion port did not contain the +5V pins, a couple
of pin sockets was soldered on the main board to the conveniently
positioned strip supplying the Vcc to the MMU. Two long pins were soldered
on the FPU board to mate with these sockets. Therefore the whole FPU board
could be simply plugged-in or removed.   This arrangement, however, proved
unreliable and a piece of cable was added with a connector to mate to the
spare power supply connector on MEGA's board.

A square opening was made in the FPU board, in the space below the FPU, to
improve somewhat air circulation (the upside-down position is not exactly
the optimum for cooling) and to facilitate the extraction of the FPU from
its socket, if needed.

At first, I used the 8MHz clock signal taken from the expansion port, but
later I added an integrated 12MHz clock to the board. As the FPU and CPU
operate asynchronously, the FPU can work at any convenient clock frequency
within its design limits, e.g. 8MHz to 12.5MHz for the 12MHz FPU. For a
16MHz FPU it might be convenient to take the clock signal from the nearby
SHIFTER chip.

A switch Sw, mounted on the back side of the computer, was used to disable
or enable the FPU (disable in "off" state), but after the initial testing
this switch proved completely unnecessary as the FPU made no problems at
all. So both the switch and the resistor can be omitted and the input to
the NOR gate connected to logical 0. It is safe to turn the switch on/off
during computer operation, but some applications may hang if the FPU is
disabled during their execution.

Capacitors C1 (220 microF) and C2 (4.7microF, tantal) and C3-C5 (200nF),
located near the Vcc pins of the chips, were used for decoupling.

The MC68881 dissipates about 0.75W in normal operation, so the whole board
needs about 200mA at 5V supply; make sure that the power supply of your
computer is adeqate for the additional load, having in mind other installed
devices such as a hard disk, etc. Maximum ambient temperature for the
operation of the FPU is 70 degrees Centigrade, so ensure that the cooling
is adequate.

Pin layouts:

See also PINS.IMG

MC68000 CPU (68pin DIP):

```
 1=D4          17=-HALT       33= A5         49= Vcc (+5V)
 2=D3          18=-RESET      34= A6         50= A21
 3=D2          19=-VMA        35= A7         51= A22
 4=D1          20= E          36= A8         52= A23
 5=D0          21=-VPA        37= A9         53= GND
 6=-AS         22=-BERR       38= A10        54= D15
 7=-UDS        23=-IPL2       39= A11        55= D14
 8=-LDS        24=-IPL1       40= A12        56= D13
 9=R/-W        25=-IPL0       41= A13        57= D12
10=-DTACK      26= FC2        42= A14        58= D11
11=-BG         27= FC1        43= A15        59= D10
12=-BGACK      28= FC0        44= A16        60= D9
13=-BR         29= A1         45= A17        61= D8
14=Vcc (+5V)   30= A2         46= A18        62= D7
15=CLK         31= A3         47= A19        63= D6
16=GND         32= A4         48= A20        64= D5
```

MEGA-ST internal expansion port (2 x 32 pins):

```
 b1=D4         b17=-HALT      a1= D5         a17= A20
 b2=D3         b18=-RESET     a2= D6         a18= A19
 b3=D2         b19=-VMA       a3= D7         a19= A18
 b4=D1         b20= E         a4= D8         a20= A17
 b5=D0         b21=-VPA       a5= D9         a21= A16
 b6=-AS        b22=-BERR      a6= D10        a22= A15
 b7=-UDS       b23=-IPL2      a7= D11        a23= A14
```

```
b8=-LDS         b24=-IPL1        a8= D12          a24= A13
b9=R/-W         b25=-IPL0        a9= D13          a25= A12
b10=-DTACK      b26= FC2         a10= D14         a26= A11
b11=-BG         b27= FC1         a11= D15         a27= A10
b12=-BGACK      b28= FC0         a12= GND         a28= A9
b13=-BR         b29= A1          a13= A23         a29= A8
b14=Vcc         b30= A2          a14= A22         a30= A7
b15=CLK         b31= A3          a15= A21         a31= A6
b16=GND         b32= A4          a16= Vcc         a32= A5
```

NOTE: pin layouts for the MC68000 and the expansion port are almost
identical, except that that the pins connected to the Vcc on the MC68000
are connected to GND on the expansion port.

12MHz Clock:

```
 1= NC
 7= GND
 8= CLK
14= Vcc (+5V)
```

MC68881 FPU (68pin PGA):

```
K| A1      R/-W    GND*    -DSACK1 D30     D29     D27     D26     D24     D22
J| A3      Vcc**   -CS     -DSACK0 D31     D28     D25     GND     D23     D21
H|-AS      A2      A0                                      Vcc     GND**   D19
G|-DS      A4                                              D20     D18
F|-SIZE    GND**                                           D17     D16
E| NC      Vcc                                             Vcc     GND
D|-RESET   GND**                                           D12     D15
C| GND     CLK     GND                             D9      D13     D14
B| Vcc*    GND*    GND*    -SENSE  D2      D5      GND     Vcc     D10     D11
A| Vcc**   GND*    D0      D1      D3      D4      D6      D7      D8      GND**
   ---------------------------------------------------------------------------
   1       2       3       4       5       6       7       8       9       10
```

The following is copied from ref.2, whatever it means:

* = new assignment for the A93N mask
**= reserved for future Motorola use

```
Pin group                        Vcc                             GND
-----------------------------------------------------------------------
D31-D16                          H8                              J8
D15-D00                          B8                              B7
-DSACK1,-DSACK0, int.logic       E2,E9   A2,B2,B3,B4,C3,E10,K3
Separate                         --                              C1
Extra                            A1,B1,J2        A10,D2,F2,H9
```

I just connected all the * and ** pins and it worked fine.

Performance:

The increase in floating-point performance relative to the unmodified
computer was impressive. GEMBENCH reported an improvement of about 1200%
with the 8MHz FPU clock and about 1400% with the 12MHz clock. Rotation of a
simple 3D view in DynaCadd 2.0 was speeded up by more than 500%, taking
less than 5 seconds instead of original 25 seconds. Whetstone benchmark in
Prospero FORTRAN reported values of about 175 (at 8MHz) and 200 (at 12MHz)
relative to 50 "kwhets" without the FPU. The interesting thing to note was
that there was not a significant gain in performance when the FPU clock
frequency was changed from 8MHz to 12MHz. Obviously, in most applications
the non-floating-point operations take a lot of CPU time. A conclusion
might be made that, unless the user runs mostly applications that are VERY
"floating-point intensive" there is not much point in installing a very
fast FPU, except if one is accidentaly available. The lower-performance
8MHz chip would do almost the same job.

NOTE 1: Documentation for GEMBENCH stated that some FPU boards "hang when
bombarded with data...". That problem did not appear with this board when
the 8MHz clock was used. It DID appear, however, and only in GEMBENCH (all
other programs worked fine), with 12MHz clock, but ONLY when I used a gate
from a 74HC08 to buffer the clock. When the clock signal was fed directly
to the FPU everything was back to normal (the 74HC08 was then removed as

unnecessary). I did not have an oscilloscope to confirm it, but perhaps the problem could be attributed to inadequate capacitive decoupling that could not cope with continuous switching transients in the additional chip. This behaviour was experienced while the 200nF decoupling capacitors were not installed, but the test was not repeated after they were added.

NOTE 2: Some applications may not recognize the presence of a FPU unless the _FPU cookie is installed (TOS 2.06 + SYSINFO reported a "SFP004 or compatible" FPU, with cookie value set to $00010000).

Revision
────────

This document has been revised to correct several minor discrepances between the original text and the actually built circuit, relating to the power supply cable and the decoupling capacitors. Also, some typing errors in the text have been corrected and my e-mail address updated.

References:
───────────
1. Keith Sabine: 68881 for the 1040, June 19th 1991
2. Motorola semiconductor technical data: M68000 Family Reference;

                              Beograd, October 1996

                              Djordje Vukovic
                     e-mail: vdjole@EUnet.yu
                         or: vdjole@sezam.co.yu (unreliable)

                ────────*────────