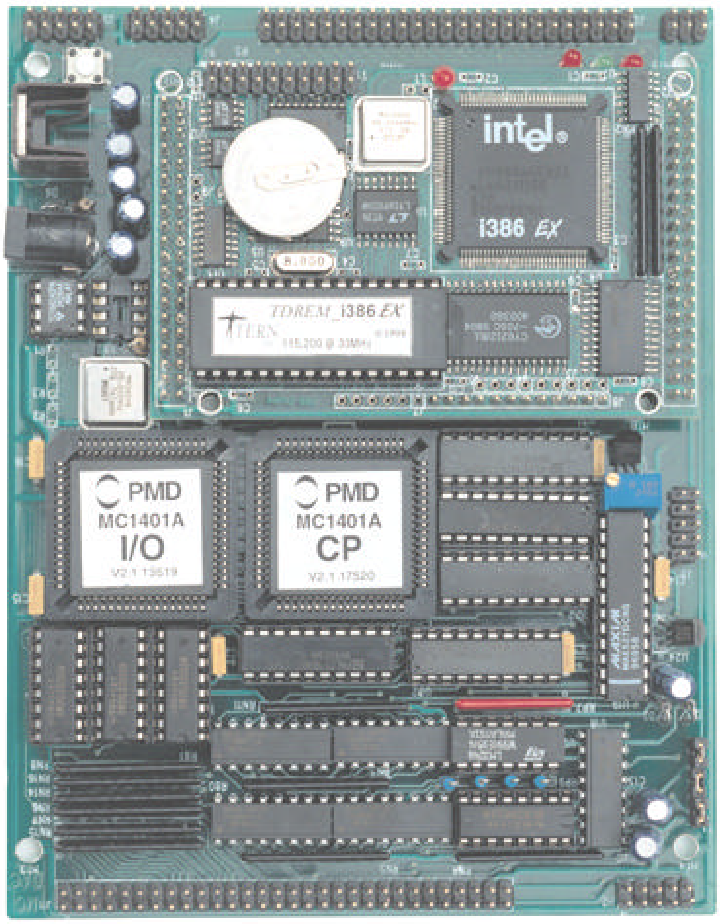
MotionC™

MC1401-DAC and MC1451 versions

C/C++ Programmable DSP 4-axis Motion Controller



Technical Manual

TERN INC.

1724 Picasso Avenue, Suite A, Davis, CA 95616-0547, USA

Tel: 530-758-0180 Fax: 530-758-0181

Internet Email: sales@tern.com *http://www.tern.com*

Version 2.01

August 19, 1999

1724 Picasso Avenue, Suite A, Davis, CA 95616-0547, USA
Tel: 530-758-0180   Fax: 530-758-0181
Internet Email: sales@tern.com ***http://www.tern.com***

Important Notice

TERN is developing complex, high technology integration systems. These systems are integrated with software and hardware that are not 100% defect free. ***TERN products are not designed, intended, authorized, or warranted to be suitable for use in life-support applications, devices, or systems, or in other critical applications. TERN*** and the Buyer agree that ***TERN*** will not be liable for incidental or consequential damages arising from the use of ***TERN*** products. It is the Buyer's responsibility to protect life and property against incidental failure.

TERN reserves the right to make changes and improvements to its products without providing notice.

Table of Contents

Chapter 1: Introduction

1.1 Functional Description

The MotionC is a low-cost, compact, reliable, high performance, C/C++ programmable industrial motion controller for up to 4-axis closed-loop digital servo control or stepper motion control. To allow operating in harsh industrial environments, the MotionC uses 32 opto-couplers. All DAC analog output signals, home switches, limit switches, and fault switches are totally optically isolated from the microcontroller digital circuit, including the DSP motion control chip set (MC1401/1451) and the *Engine* controller. The MotionC can be driven by an A-Engine or an i386-Engine as a host microprocessor module interfacing to the motion control chipset (MC1401/1451, PMD).

The MC1401 is a DSP based advanced multi-axis closed-loop digital servo controller for a large variety of servo motors. The MotionC uses incremental quadrature encoders for position feedback and outputs DC voltages for external power amplifiers. Each axis contains sophisticated trajectory profile and digital servo capabilities, allowing very low position and velocity tracking errors. 4-axis can be programmed either independently or in synchrony to allow advanced multi-axis motion such as circular and continuous-path profiles. It provides electronic gearing, PID or PI control, choice of S-curve, trapezoidal, or contoured velocity profile modes, 1/T counter for stable low velocity motion, automatic motor error shutdown, monitoring travel limit switches, home switches, capture switches, and fault switches.

The MC1451 is functionally similar to the MC1401. However, it is dedicated to the open-loop control of stepping motors and outputs step pulse and direction signals.

The host A-Engine/i386-Engine also provides 11 channels of 12-bit ADC, 24 I/O lines for expansion, 2 RS232 serial ports, and one UART(SCC2691) for multi-drop RS485 networking.

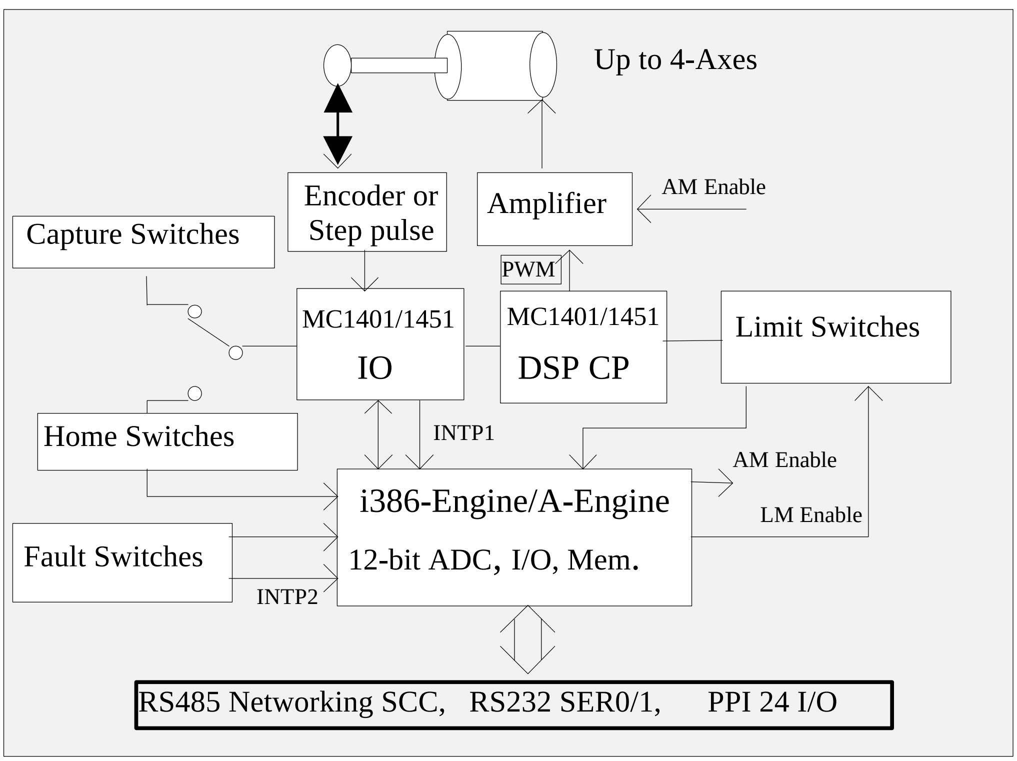


Figure 1.1 Functional block diagram of the MotionC and the i386-Engine/A-Engine

1.2 Features

Standard Features:

- Dimensions: 6.1 x 4.7 inches
- 4-axis closed-loop digital servo control with PMD MC1401 or stepper control with MC1451
- Power consumption: 400 mA at 12V. (On-board +5V regulator)
- Temperature range: -40°C to +80°C
- 2 RS-232 serial channels
- 4 channels high voltage outputs for power amplifier control for MC1401
- 4 axis pulse and direction outputs upto 3 Mpulses/sec for MC1451
- 24 PPI I/O lines for expansion
- Opto-couplers for all home-switches, limit-switches, capture switches, and fault-switches
- 32-bit position, velocity, acceleration and jerk registers
- S-curve, trapezoidal, or contoured velocity profile modes
- Quadrature encoder with index position capture
- Electronic gearing for multi-axis
- PID or PI control, Programmable loop rate to 100 μs
- 1/T counter for stable low velocity motion
- Automatic motor shutdown on motion error
- Driven by an i386-Engine/A-Engine, C/C++ programmable
- EPROM/Flash (up to 512K)
- Watchdog and power-fail protection

Optional Features (* surface-mounted components)**:**

- 32KB, 128KB, or 512KB SRAM* (on Engine controller)
- 11 channels of 12-bit ADC, sample rate up to 10 KHz*
- Up to 4 channels of 12-bit DAC outputs
- SCC2691 UART (on-board) supports 8-bit or 9-bit networking, with RS-232* or RS-485 drivers
- Real-time clock RTC72423*, lithium coin battery* (on Engine controller)

1.3 Physical Description

The physical layout of the MotionC is shown in Figure 1.2.

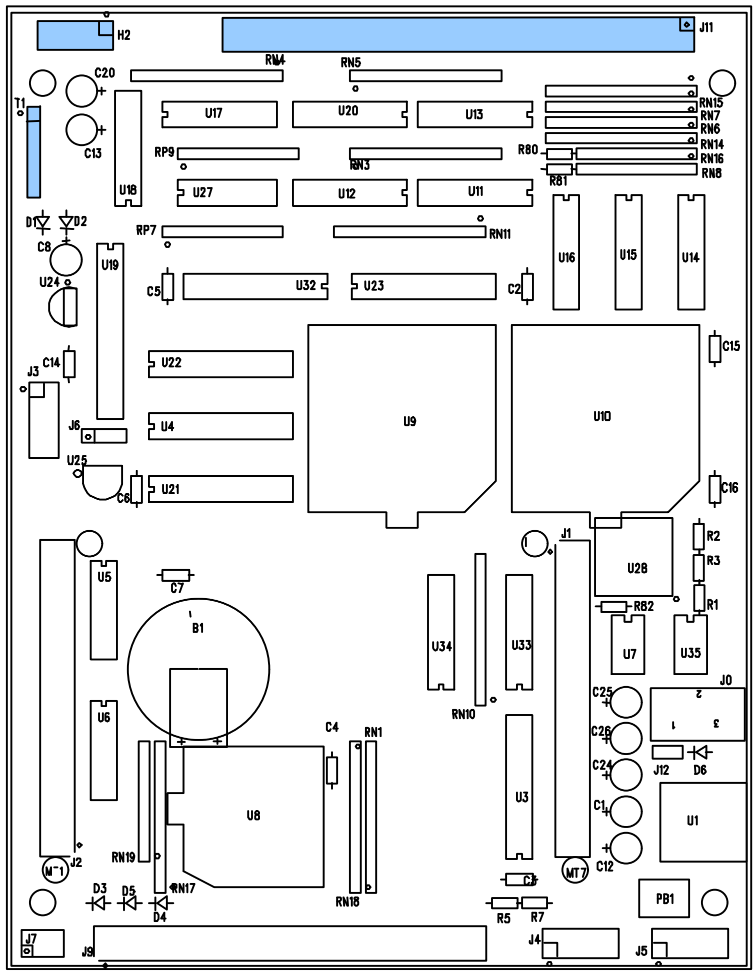


Figure 1.2 Physical layout of the MotionC

1.4 MotionC Programming Overview

Development of application software for the MotionC consists of three easy steps, as shown in the block diagram below.

STEP 1 Serial link PC and MotionC+*Engine*, program in C/C++
Debug C/C++ program on MotionC+*Engine* with Remote Debugger

STEP 2 Test MotionC+*Engine* in the field, away from PC
Application program resides in the battery-backed SRAM

STEP 3 Make application ROM or Download to Flash
Replace DEBUG ROM, project is complete

You can program the MotionC+*Engine* from your PC via serial link with an RS232 interface. Your C/C++ program can be remotely debugged over the serial link at a rate of 115,000 baud. The C/C++ Evaluation Kit (EV) or Development Kit (DV) from TERN provides a Borland C/C++ compiler, TASM, LOC31, Turbo Remote Debugger, I/O driver libraries, sample programs, and batch files. These kits also include a DEBUG ROM (*TDREM_i386EX)* to communicate with Turbo Debugger, a PC-V25 cable to the connect the controller to the PC, and a 9-volt wall transformer. *See your Evaluation/Development Kit Technical Manual for more information on these kits.*

After you debug your program, you can test run the MotionC in the field, away from the PC, by changing a single jumper, with the application program residing in the battery-backed SRAM. When the field test is complete, application ROMs can be produced to replace the DEBUG ROM. The .HEX or .BIN file can be easily generated with the makefile provided. You may also use the DV Kit or ACTF Kit to download your application code to on-board Flash.

The three steps in the development of a C/C++ application program are explained in detail in the Technical Manuals for the i386-Engine and A-Engine. Please refer to the Tutorial of the Technical Manual of the EV/DV Kit for further details on programming the MotionC+*Engine*.

1.5 Minimum Requirements for MotionC System Development

1.5.1 Minimum Hardware Requirements

- PC or PC-compatible computer with serial COMx port that supports 115,200 baud
- MotionC controller (MC1401-DAC or MC1451 version)
- *Engine* controller:
  i386-Engine controller with DEBUG ROM *TDREM_i386EX*
  or, A-Engine controller with DEBUG ROM *TDREM_AE*
- PC-V25 serial cable (RS-232; DB9 connector for PC COM port and IDE 2x5 connector for controller)
- center negative wall transformer (+9V 500 mA)

1.5.2 Minimum Software Requirements

- TERN EV/DV Kit installation diskettes
- PC software environment: DOS, Windows 3.1, Windows95, or Windows98

The C/C++ Evaluation Kit (EV) and C/C++ Development Kit (DV) are available from TERN. The EV Kit is a limited-functionality version of the DV Kit. With the EV Kit, you can program and debug the MotionC+*Engine* in Step Three and Step Two, but you cannot run Step Three. In order to generate an application ROM/Flash file, make production version ROMs, and complete a project, you will need the Development Kit (DV).

Chapter 2: Installation

2.1 Software Installation

Please refer to the Technical manual for the “C/C++ Development Kit and Evaluation Kit for TERN Embedded Microcontrollers” for information on installing software.

The README.TXT file on the TERN EV/DV disk contains important information about the installation and evaluation of TERN controllers.

2.2 Hardware Installation

Hardware installation for the MotionC consists primarily of connecting the microcontroller to your PC.

Overview

- Connect PC-V25 cable:
  For debugging (STEP 1), place the 5x2 pin header on SER0 (J3) with red edge of cable at J3 pin 1
- Connect wall transformer:
  Connect 9V wall transformer to power and plug into power jack

2.2.1 Connecting the MotionC to the i386-Engine/A-Engine

Install the *Engine* controller with the J1 (20x2) socket connector on the upper half of the J2 (dual row header) of the MotionC. Figure 2.1 and Figure 2.2 show the MotionC and the *Engine* before and after installation.

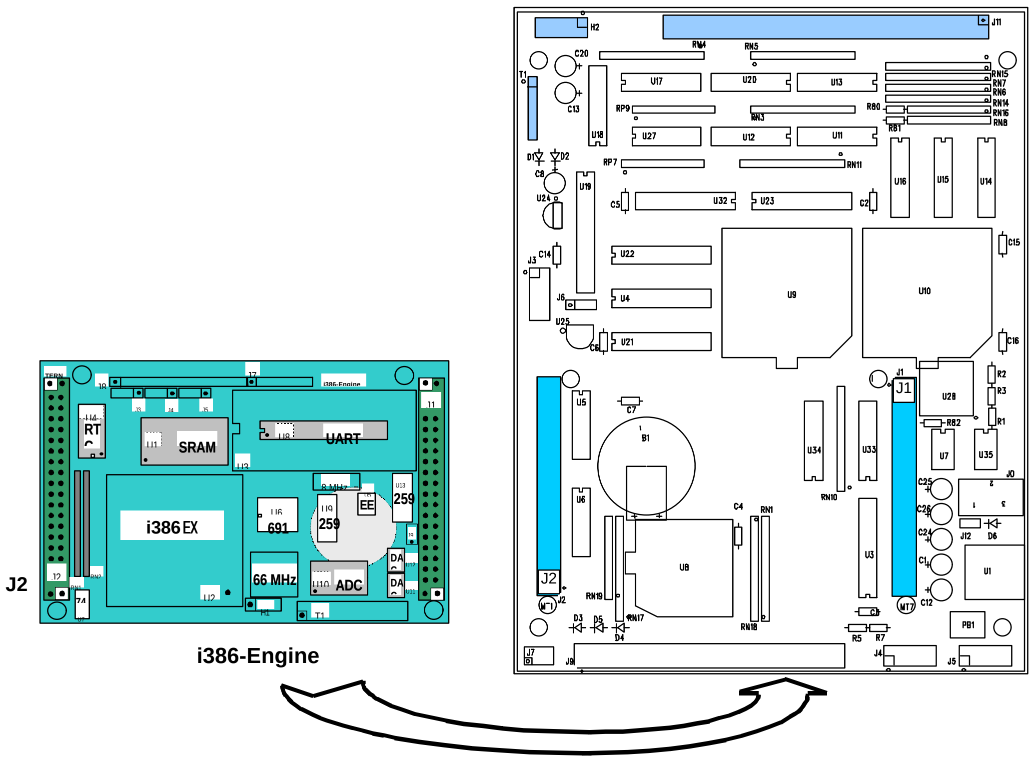


Figure 2.1 Before installing the *Engine* controller on the MotionC

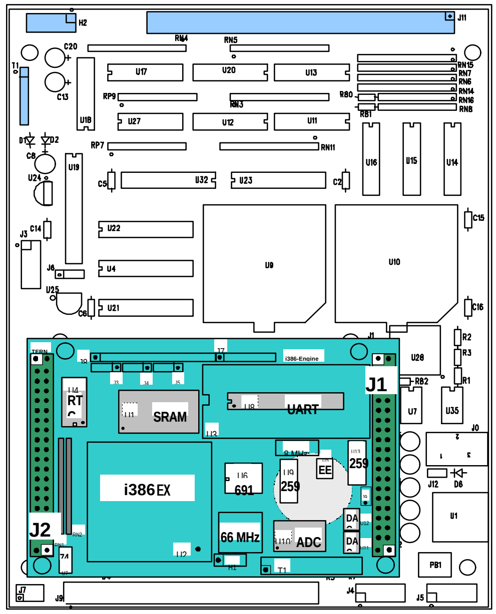


Figure 2.2 After installing the *Engine* controller on the MotionC

2.2.2 Connecting the MotionC to the PC

The following diagram (Figure 2.3) illustrates the connection between the MotionC and the PC. The MotionC is linked to the PC via a serial cable (PC-V25).

The *TDREM_i386EX / TDREM_AE* DEBUG ROM communicates through SER0 by default. Install the 5x2 IDE connector on the SER0 header (J3). ***IMPORTANT:*** *Note that the* ***red*** *side of the cable must point to pin 1 of the J3 header*. The DB9 connector should be connected to one of your PC's COM Ports (COM1 or COM2).

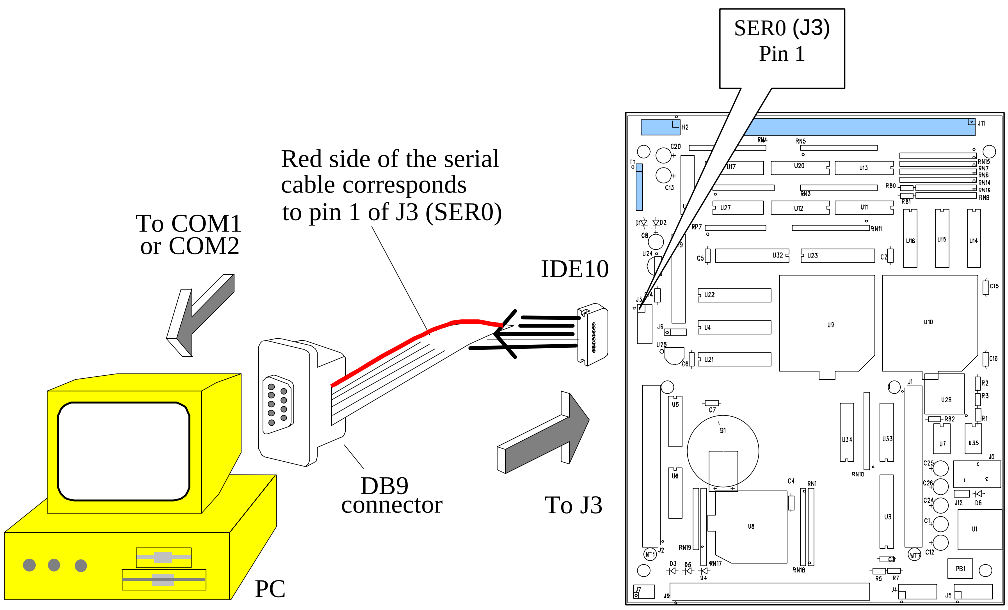


Figure 2.3 Connecting the MotionC+*Engine* to the PC

2.2.3 Powering-on the MotionC

Connect a wall transformer +9V DC output to the DC power jack.

The on-board LED should blink twice and remain on after the MotionC is powered-on or reset, as shown in Figure 2.4.

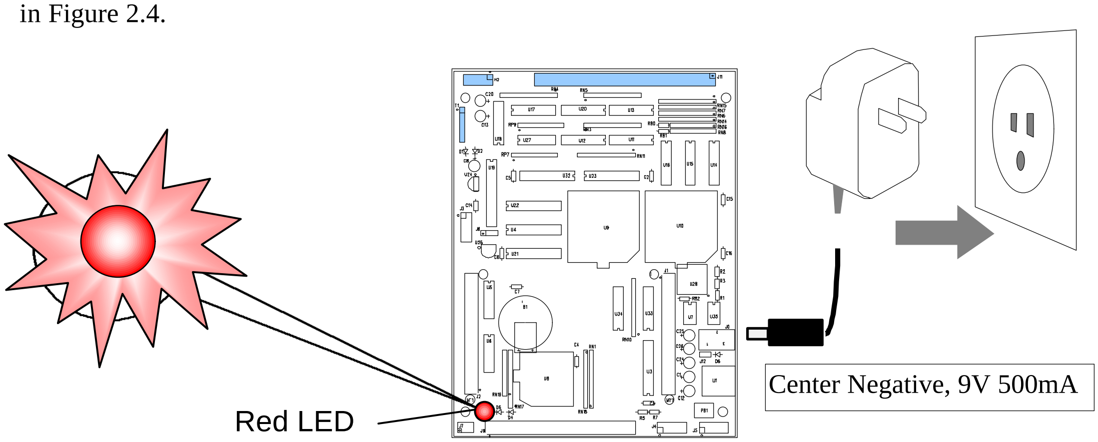


Figure 2.4 The LED blinks twice after the MotionC is powered-on or reset

Chapter 3: Hardware

3.1 i386-Engine or A-Engine

The MotionC uses an i386-Engine or an A-Engine as its host microprocessor core module. Please refer to the i386-Engine or A-Engine Technical Manual for more information.

3.2 Interface with PMD MC1401/1451 DSP chipset

The MC1401/1451 chipset is packaged in two 68 pin PLCC chips, “CP” and “I/O”. The chipset is controlled by the host i386-Engine/A-Engine via an 8-bit, bi-directional port. Communications to and from the chipset consist of packet-oriented messages. An interrupt line from the DSP-IO pin 44, is routed to J2 pin 6 of the i386-Engine /INT6, or the A-Engine /INTP1, so that the chipset can signal the host when special conditions occur, such as an encoder index pulse received.

The i386-Engine/A-Engine writes commands to the MC1401/1451 and reads results from MC1401/1451 chipset. Each command consists of a single byte, with a command code value as defined in Appendix A. MC1401/1451 manual. Data is transmitted to or from the chipset in 16-bit words.
Functions are available in **mc.lib** for the i386-Engine/A-Engine to communicate with MC1401/1451:

void mc_host_dat_wr(unsigned int dat); // host Engine writes 16-bit dat to MC
void mc_host_cmd_wr(unsigned char cmd); // host Engine writes 8-bit cmd to MC
unsigned int mc_host_dat_rd(void); // host Engine reads 16-bit dat from MC
char mc_host_rdy(void); //return 0 for MC1401/1451 IO pin 37 low, indicating host port busy
//return 1 for MC1401/1451 IO pin 37 high, indicating host port ready for cmd/data

3.3 MotionC I/O Map

The following tables list the I/O address of the MotionC, together with their Data Bits, Chip-Select Symbol and Functions.

Base I/O Address	Data Bits	Select Symbol	Function
0x??20	D0	O1	Latch D0 to PAL U3 pin 16 O1. Control LED and HV AENA1 on/off. outportb(0x??20,0); will turn LED D3 and AENA1 on
0x??40	D0	O2	Latch D0 to PAL U3 pin 15 O2. Control LED and HV AENA2 on/off. outportb(0x??40,0); will turn LED D4 and AENA2 on
0x??60	D0	O3	Latch D0 to PAL U3 pin 14 O3. Control LED and HV AENA3 on/off. outportb(0x??60,0); will turn LED D5 and AENA3 on
0x??80	D0	O4	Latch D0 to PAL U3 pin 13 O4. Control LED and HV AENA4 on/off. outportb(0x??80,0); will turn AENA4 on
0x??b0	D0-D7	/MC	Read/Write D0-7 from/to MC1401 I/O chip (U10)
0x??c0	D0-D7	/PPI	Read/Write D0-7 from/to PPI 82c55 I/O chip (U8)

0x??e4-0x??e7 /IOWR	D0	HC	Write D0 to one of the four addresses: 0xe4-0xe7, D0=1 for enabling Home Switches (J11 HMFL1-4) or D0=0 for enabling Capture Switches(H2 CAP1-4).
0xd0-0xd1 /IORD	D0-D7	/STA	Reads status of switches: FAULT1-4, HMFL1-4, +LIM1-4, and -LIM1-4. See function in mc.lib: mc_status(void) return 16 bits status of J11 for limit, home, fault switches bit 15 14 13 12 11 10 9 8 -LIM2 -LIM1 -LIM4 -LIM3 HMFL1 HMFL3 FAULT1 FAULT3 bit 7 6 5 4 3 2 1 0 +LIM2 +LIM1 +LIM4 +LIM3 HMFL2 HMFL4 FAULT2 FAULT4

3.4 Quadrature Encoder Inputs (MC1401) or Step Pulses Outputs (MC1451)

The PCB of the MotionC supports both the MC1401 and MC1451 DSP chipsets. The MC1401 supports 4 channels of Incremental Encoder inputs for motor position information. Each quadrature encoder channel consists of a square wave offset 90 degree from the other. For every channel, four position input and control signals are supported:

A channel pulses (QDAx)
B channel pulses (QDBx)
Index pulse
Home switches signal

Differential line drivers (26LS33) are used to support differential quadrature inputs. The quadrature encoders inputs are not optical isolated from digital ground (GND). They may be powered with +5V.

The MotionC with MC1451 supports 4 axis stepper control pulses outputs. MC1451 outputs steppers control pulses to the QDAx pin which was designed for MC1401 IO chip quadrature encoder phase A inputs. MC1451 also outputs direction signal at QDBx pin, which was designed for MC1401 quadrature encoder phase B inputs. The stepper pulse outputs are available on U14, U15, and U16. Please refer to Appendix B for modifications of MotionC (1451 version).

U16 pin 5=QDA1=Pulse Output of Axis 1; U16 pin 11=QDA2=Pulse Output of Axis 2;
U15 pin 3=QDA3=Pulse Output of Axis 3; U15 pin 13=QDA4=Pulse Output of Axis 4;
U16 pin 3=QDB1=Direction 1; U16 pin 13=QDB2=Direction 2;
U14 pin 5=QDB3=Direction 3; U14 pin 11=QDB4=Direction 4;

The line driver (26LS33) must NOT be installed for the MC1451. You may wire connect the pulses and direction signal to the J11 header.

There are 4 lines of high voltage driver outputs (U17, ULN2003) designed to sink up to 350 mA at 50V (designed for power amplifier enable control in a MC1401 digital servo system). You can also wire connect the pulses and direction signal to the Opto-coupler (U19 pin 2, 4, 6, 8).

In order to use high voltage drivers by the MC1451 pulse output, you must NOT install U30 (74HC259) or cutoff U30 pin 4, 5, 6, and 7.

Please refer to the MotionC schematic mc.sch for more details.

3.5 MotionC (MC1401 DAC version) analog outputs

MotionC 1401-DAC version (MC-DAC) uses a 60-pin motion interface connector J11.

The MC-DAC supports DAC output mode for the PMD1401 motor amplifier interface. A calibrated quad 12-bit voltage output DAC converter (MAX527) is used. While the 16-bit DAC output mode is used, the 12-bit DAC only uses the higher order 12 bits date from the CP chip.

The MAX527 contain four precision output buffer amplifiers providing full 12-bit performance at 1 LSB total unadjusted error without adjustments. The MAX527 has a typical 3 µs output setting time. The on-board state-machine logic provides complete circuits for latching and loading 12-bit DAC data in less than 2 µs.

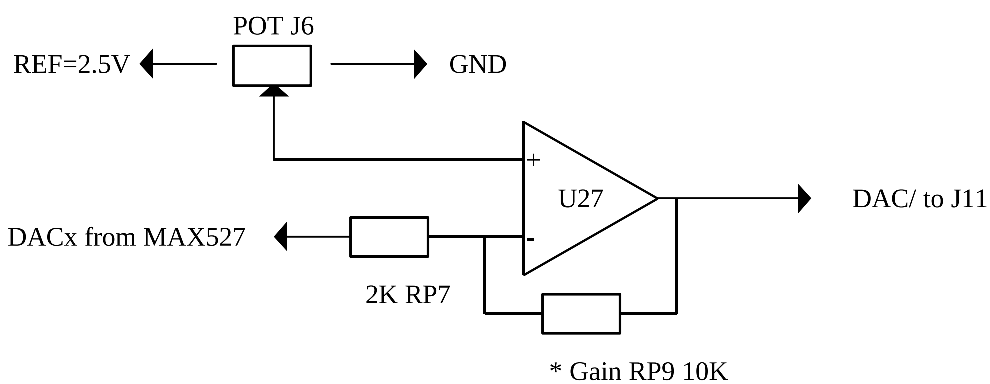
A quad amplifier buffer with user adjustable gain and offset supports varieties of motor amplifier interface.

A resistor pot is installed in J6 providing the offset voltage. User may install gain resistors in RP9 to setup necessary gains.

The DAC1(J11.43), DAC2(J11.44), DAC3(J11.29), and DAC4(J11.30) provides output voltage directly from the DAC. The PMD1401 outputs a 16-bit data word as an unsigned 16-bit number with a range of 0-65535. An output value of 0 will force the DACx to output 0 volt, a value of 32768 will force DACx to output 1.25 volt, and a value of 65536 will output 2.5V.

The DAC1/(J11.45), DAC2/(J11.46), DAC3/(J11.31) and DAC4/(J11.32) are adjusted output voltages. The default output voltages are -10V for output value of -32768, 0V for output value of 0, and +10V for output value of +32768.

See sample program ***mc_dac.c*** for detail.



3.6 Limit, Home, and Fault switches

There are 2 limit switch inputs for each axis: +LIM1 and -LIM1. There are a total of 8 limit switch inputs. There are 4 home switch inputs: HMFL1-4. There are 4 fault switch inputs: FAULT1-4. All switch inputs are pulled high to +12V and they are totally optically isolated from chipset and microprocessor digital circuits. See sample programs: **mc_h_i.c**, **mc_f_i.c**, and **mc_statu.c** for details.

3.7 Power Amplifier Control

Four lines of high voltage driver outputs (ULN2003) are designed to sink up to 500 mA at 50V. They can be used to enable external power amplifiers for driving DC motors directly. The power amplifier enable signals are optically isolated from digital circuits.

Four channels of high voltage drivers are mapped in I/O address 0x20, 0x40, 0x60 and 0x80. See sample program **mc_amp_d.c**.

3.8 Power Supplies, Digital and Analog Ground

There are two power supply systems built into the MotionC. Two power supply systems can be can be separated via J11. The field power supply +12V, -12V, and AGND are provided by user via J11 header. The +12V, -12V and AGND are used by analog output operational amplifiers, and must be well regulated. The +12VDI and GND may be an unregulated DC source from the DC wall transformer. It will generate a regulated VCC and -12V on the MotionC. In order to develop software and test the functions on the desk top, you can use the DC wall transformer only. You can add jumpers on T1 to connect +12V=+12VD, AGND=GND, and -12V=-12VD. H2 also provides a jumper for AGND=GND.

When you place the MotionC in the test field with real motors and power amplifiers, you may provide external regulated +12V, -12V, +5Vand AGND.

If you use the on board +5V VCC regulator (7805) to power the external quadrature encoders, you will need to provide additional large heat sinks to the 7805 regulator, such as mounting on a large metal standoff. If you want to provide an external regulated +5V to the MotionC via J11, you may not install the LM7805 on board. A block diagram for the MotionC power system is shown in Figure 3.1. The position of T1, H2, J11, J0 are high lighted in Appendix A: MotionC Layout.

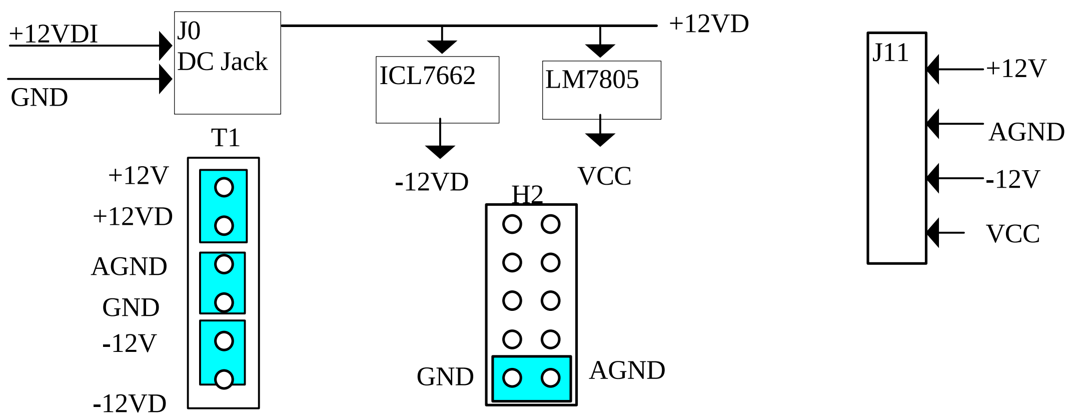


Figure 3.1 Block diagram for the MotionC power supply system.

While you are only using an external power supply to J11 +12V, -12V, and AGND, jumpers on T1 and H2 can be installed. All input signal returns of the machine home switches, capture switches, limit switches, and fault switches must tie to AGND, not GND. The external power supply and power amplifier system must be optically isolated with the i386-Engine/A-Engine and MC1401 digital system. The +5V powered quadrature encoders are not optically isolated, sharing the same GND. You need to shield the encoder differential input signal cable and keep the power noise into the digital GND.

3.9 24 I/O lines of PPI

There are 24 bi-directional I/O lines at J9 header. You may program the PPI 82C55 to place it in either input or output mode. See sample program **mc_ppi.c** for details.

3.10 RS-232 and RS-485

Two channels of RS-232 serial ports are available on the MotionC. You may use J3 SER0 for debugging and J4 SER1 for application. An RS-485 driver and header J5 supports the SCC2691 UART for networking.

3.11 Total Optical Isolation with 20 Opto-couplers

There are a total of 20 quad opto-couplers on board, to isolate the external power system and the MC1401/1451+ *Engine* digital system.

3.12 Step Pulses and Direction signals for the MotionC (1451 version)

The MotionC (MC1451 version) is a dedicated complete stepper motor controller. It can perform trajectory generation, pulse and direction signal generation for use in a wide variety of stepper-based systems. For each axis, two signals are provided which determine the desired axis velocity and direction. These two signals are the pulse signal, and the direction signal.

The pulse signal always outputs a square wave pulse train (50% duty cycle) which is proportional to the desired velocity from the trajectory generator.

The direction signal is synchronized with the pulse signal at the moment each pulse transition occurs. The direction signal is encoded such that a high value indicates a positive direction pulse, and a low value indicates a negative direction pulse.

In the standard speed mode, it can output pulses up to 48.8 K steps per second. In the high-speed mode, it can output up to 1.5625 M steps per second.

3.13 DSP Ready signal to the A-Engine/i386-Engine

The DSP IO chip pin 37 is a Hardware ready signal that indicates the DSP is busy while it is low.

The MC-DAC-99 version routes RDY signal to J2 pin 12, which is P14 for the i386-Engine or P10 of the A-Engine.

The original MC-DAC PCB routes RDY signal to J2 pin 3, which is P24 for the i386-Engine or P29 for the A-Engine. The RDY=P29 causes problems for the A-Engine at power-on or reset. Therefore the A-Engine has to be modified to use P10=J2 pin 12 on the original MC-DAC.

Chapter 4: Software

Please refer to the Technical Manual of the “C/C++ Development Kit for TERN 16-bit Embedded Microcontrollers” for details on debugging and programming tools. For software information related to the *Engine* controller, please refer to the respective manual.

4.1 Functions in MC.LIB

void ***mc_host_dat_wr***(unsigned int dat); // host C-Engine writes 16-bit dat to MC

void ***mc_host_cmd_wr***(unsigned char cmd); // host C-Engine writes 8-bit cmd to MC

unsigned int ***mc_host_dat_rd***(void); // host C-Engine reads 16-bit dat from MC

char ***mc_host_rdy***(void); // return 0 for MC1401IO pin 37 low, indicating host port busy
// return 1 for MC1401IO pin 37 high, indicating host port ready for cmd/data

mc_status(void); return 16 bits status of J11 for limit, home, fault switches

bit	15	14	13	12	11	10	9	8
	-LIM2	-LIM1	-LIM4	-LIM3	HMFL1	HMFL3	FAULT1	FAULT3
bit	7	6	5	4	3	2	1	0
	+LIM2	+LIM1	+LIM4	+LIM3	HMFL2	HMFL4	FAULT2	FAULT4

mc_fault_set(char fault, char dat);
define fault inputs active high or low in order to generate INT2 on P13.
Default FAULT1-4 active low
Examples: If mc_fault_set(3,1); then FAULT3 active high

mc_limit_en(char dat); Enable pass status of limit switches over U21 PAL CMCp030 to MC
If dat=0, PLM1-3 and NLM1-3 always low
If dat=1, PLMx = +LMxI and NLMx= -LMxI

void mc_hom_or_cap(char dat); Select Home or Capture signal to MC's HOM1-4
If dat=1, HOMx=HMxI, (PAL CMCP020)
If dat=0, HOMx=CAPxI, (PAL CMCP020)

mc_amp_enable(char amp, char dat); Enable external OPs
Examples:
If amp=1, dat=1, AENA1(J11.47) active low, sinks upto 350 mA at 50V max.
If amp=4, dat=0, AENA4(J11.34) disable, pulled high to +12V

void ***mc_HomCap_p_select***(char hc, char polar);
Define active high/low on Home or Capture signal to MC's HOM1-4
Use void mc_hom_or_cap(char dat); to
Select Home or Capture signal to MC's HOM1-4
If dat=1, HOMx=HMxI, (PAL CMCP020) HC=1
If dat=0, HOMx=CAPxI, (PAL CMCP020) HC=0
while HC=1,

while hc=1, polar=1, HM1S=1(U30), HOM1=HM1I active high, (PAL CMCP020)
while hc=2, polar=1, HM2S=1(U30), HOM2=HM2I active high, (PAL CMCP020)
while hc=3, polar=1, HM3S=1(U30), HOM3=HM3I active high, (PAL CMCP020)
while hc=4, polar=1, HM4S=1(U30), HOM4=HM4I active high, (PAL CMCP020)
while hc=1, polar=0, HM4S=0(U30), HOM4=HM4I active low, (PAL CMCP020)
while HC=0,
while hc=1, polar=1, HM1S=1(U30), HOM1=CAP1I active high, (PAL CMCP020)

void ***mc_led***(char led, char dat); MC led 0-2 control
where led=0, 1, 2
dat=0, led off. dat=1, led on.

4.2 Sample Programs

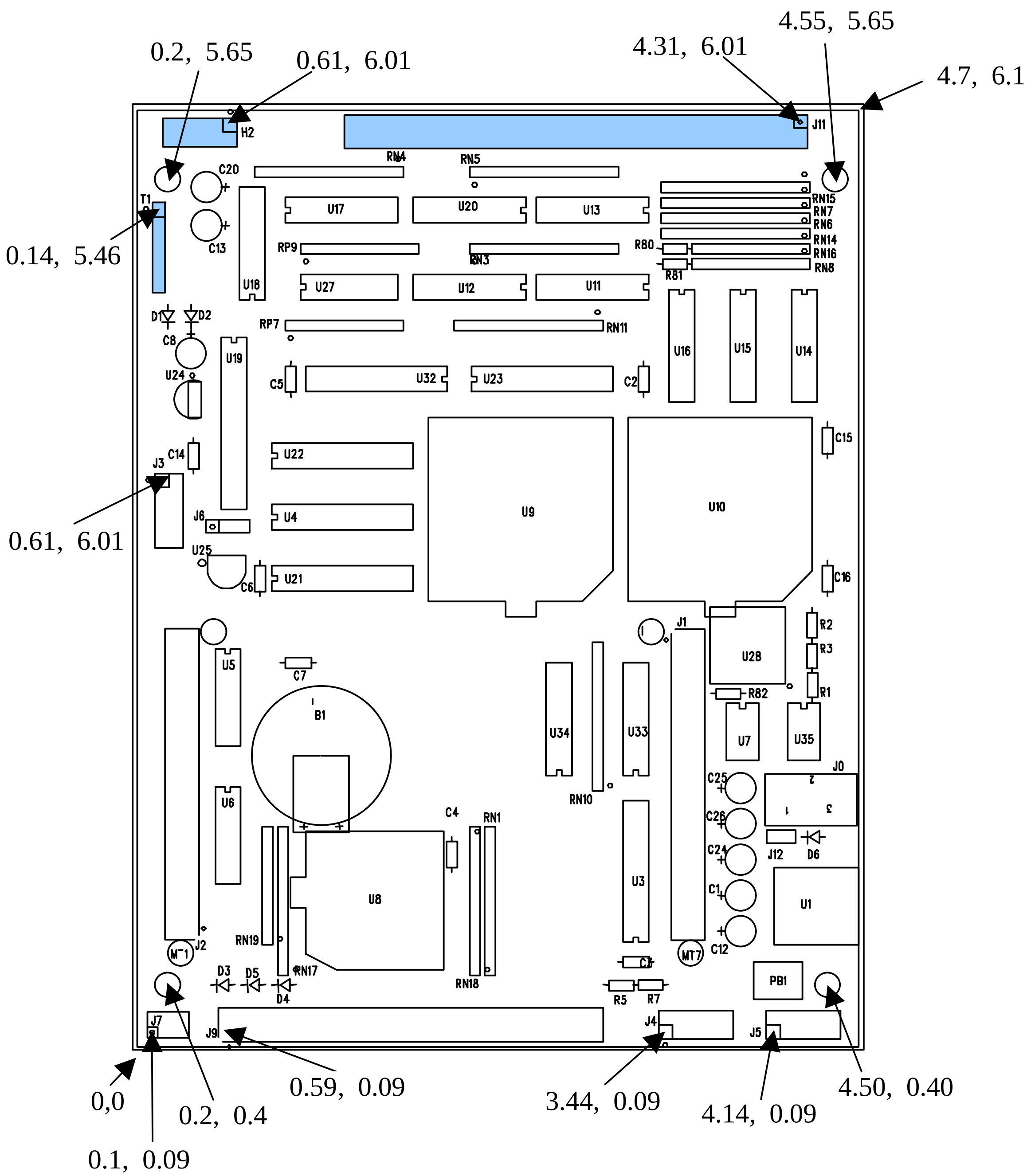
Sample programs for the MotionC are located in the samples directory at **`\samples\mc`**.

mc_pwm.c (MC1401)
mc_ppi.c
mc_led.c
mc_statu.c
mc_f_int.c
mc_h_int.c
mc_l_int.c (MC1451)
mc_step.c (MC1451)
mc_ver.c

Appendix A: MotionC Layout

The **MotionC** measures 6.1 by 4.7 inches. Its layout is shown below.

All dimensions are in inches.



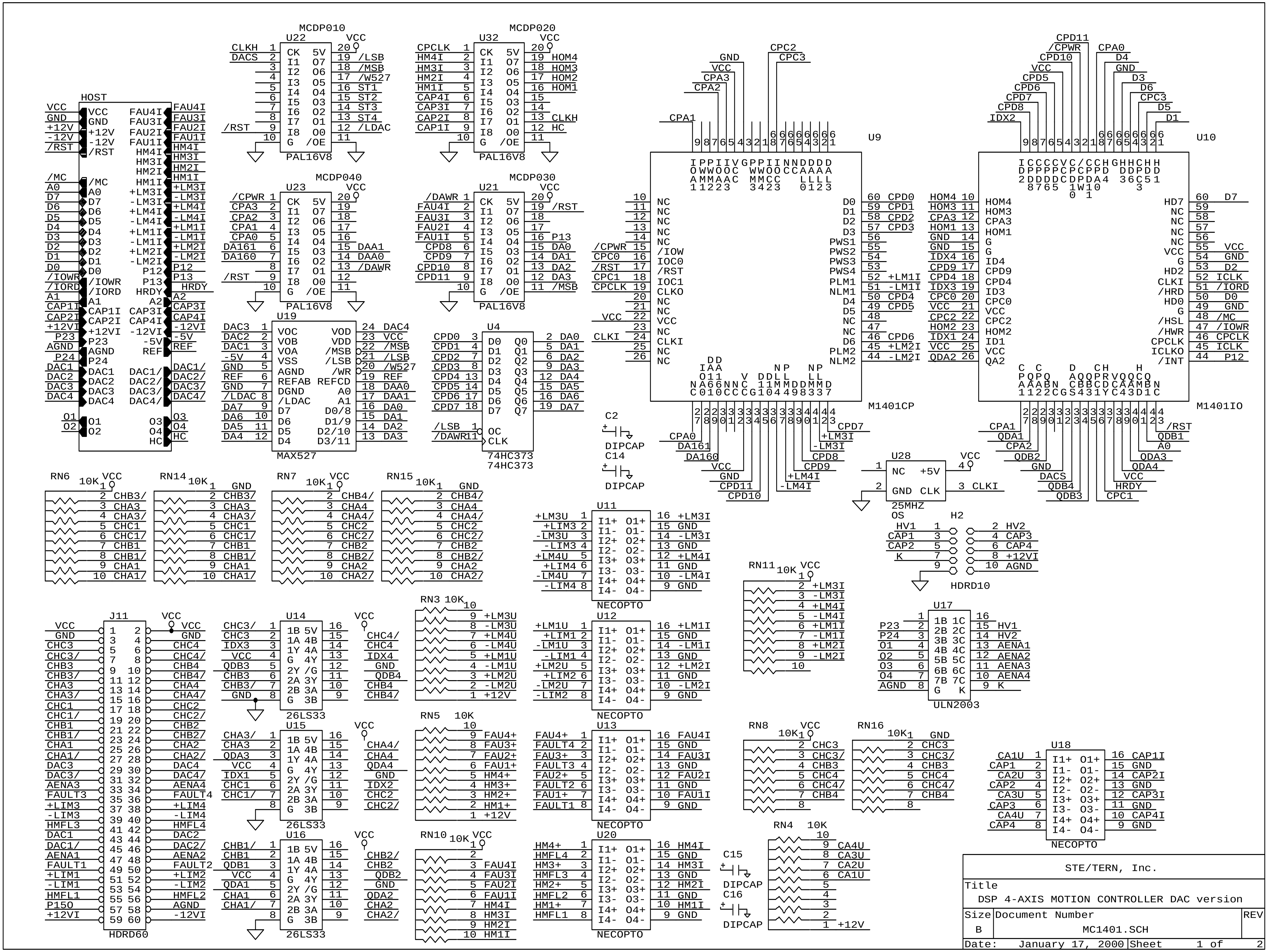
Appendix B: MotionC MC1451 version J11 modifications

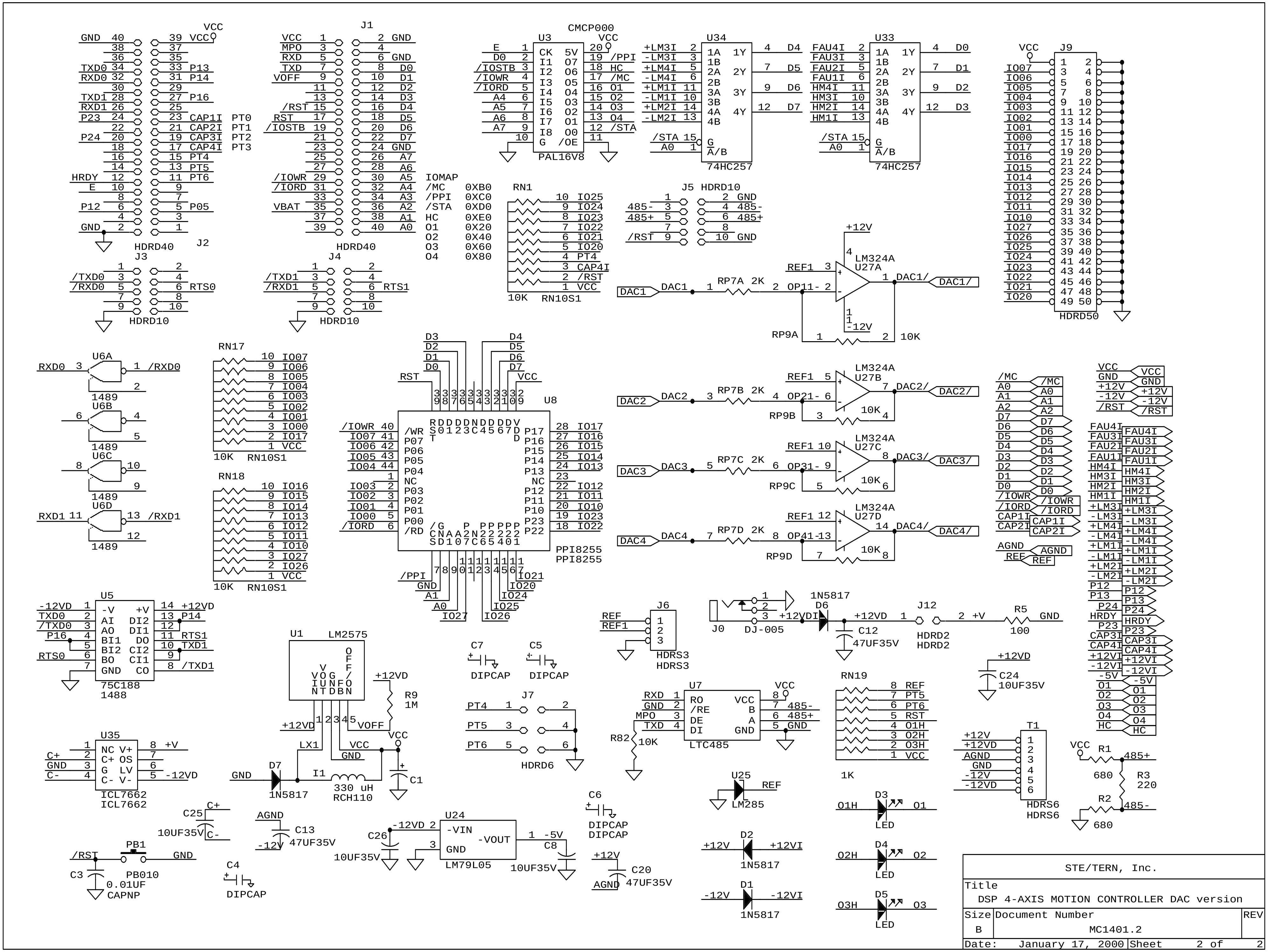
1) Add RST line from U10 pin 17(CP) to U9 pin 43(I/O).
2) Do not install U23 (Pal CMCP040), U26, U27 (LM324), U24, U25 (OPTO).
3) Jumper Connections:

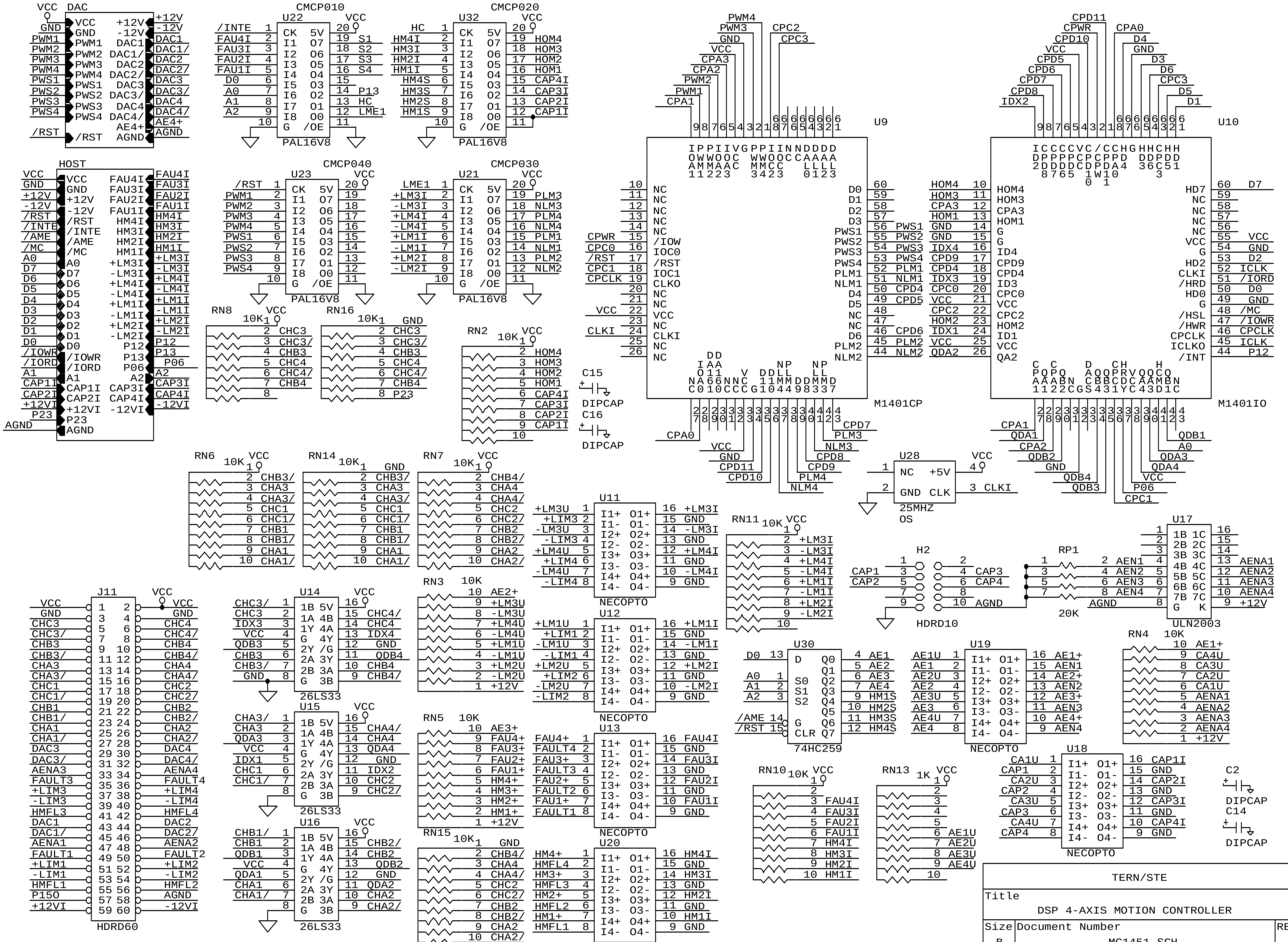
QDA1 = U16.5 = U16.6 = CHA1 = Pulse 1= J11 pin 25
QDA2 = U16.10 = U16.11 = CHA2 = Pulse 2 = J11 pin 26
QDA3 = U15.2 = U15.3 = CHA3 = Pulse 3 = J11 pin 13
QDA4 = U15.13 = U15.14 = CHA4 = Pulse 4 = J11 pin 14
QDB1 = U16.2 = U16.3 = CHB1 = Dir 1 = J11 pin 21
QDB2 = U16.13 = U16.14 = CHB2 = Dir 2 = J11 pin 22
QDB3 = U14.5 = U14.6 = CHB3 = Dir 3 = J11 pin 9
QDB4 = U14.10 = U14.11 = CHB4 = Dir 4 = J11 pin 10

4) J11 layout.

```
Vcc          1  ----o o---- 2   Vcc
GND          3  ----o o---- 4   GND
             5  ----o o---- 6
             7  ----o o---- 8
DIR3         9  ----o o----10  DIR4
             11----o o----12
PULSE3       13----o o----14  PULSE4
             15----o o----16
             17----o o----18
             19----o o----20
DIR1         21----o o----22  DIR2
             23----o o----24
PULSE1       25----o o----26  PULSE2
             27----o o----28
             29----o o----30
             31----o o----32
AENA333----o o----34  AENA4
FAULT3       35----o o----36  FAULT4
+LIM3        37----o o----38  +LIM4
-LIM3        39----o o----40  -LIM4
HMFL3        41----o o----42  HMFL4
             43----o o----44
             45----o o----46
AENA147----o o----48  AENA2
FAULT1       49----o o----50  FAULT2
+LIM1        51----o o----52  +LIM2
-LIM1        53----o o----54  -LIM2
HMFL1        55----o o----56  HMFL2
             57----o o----58  AGND
+12VI        59----o o----60  -12VI
```


STE/TERN, Inc.
Title
DSP 4-AXIS MOTION CONTROLLER DAC version
Size
B
Document Number
MC1401.SCH
REV
Date: January 17, 2000
Sheet 1 of 2


STE/TERN, Inc.
Title
DSP 4-AXIS MOTION CONTROLLER DAC version
Size B
Document Number MC1401.2
REV
Date: January 17, 2000
Sheet 2 of 2



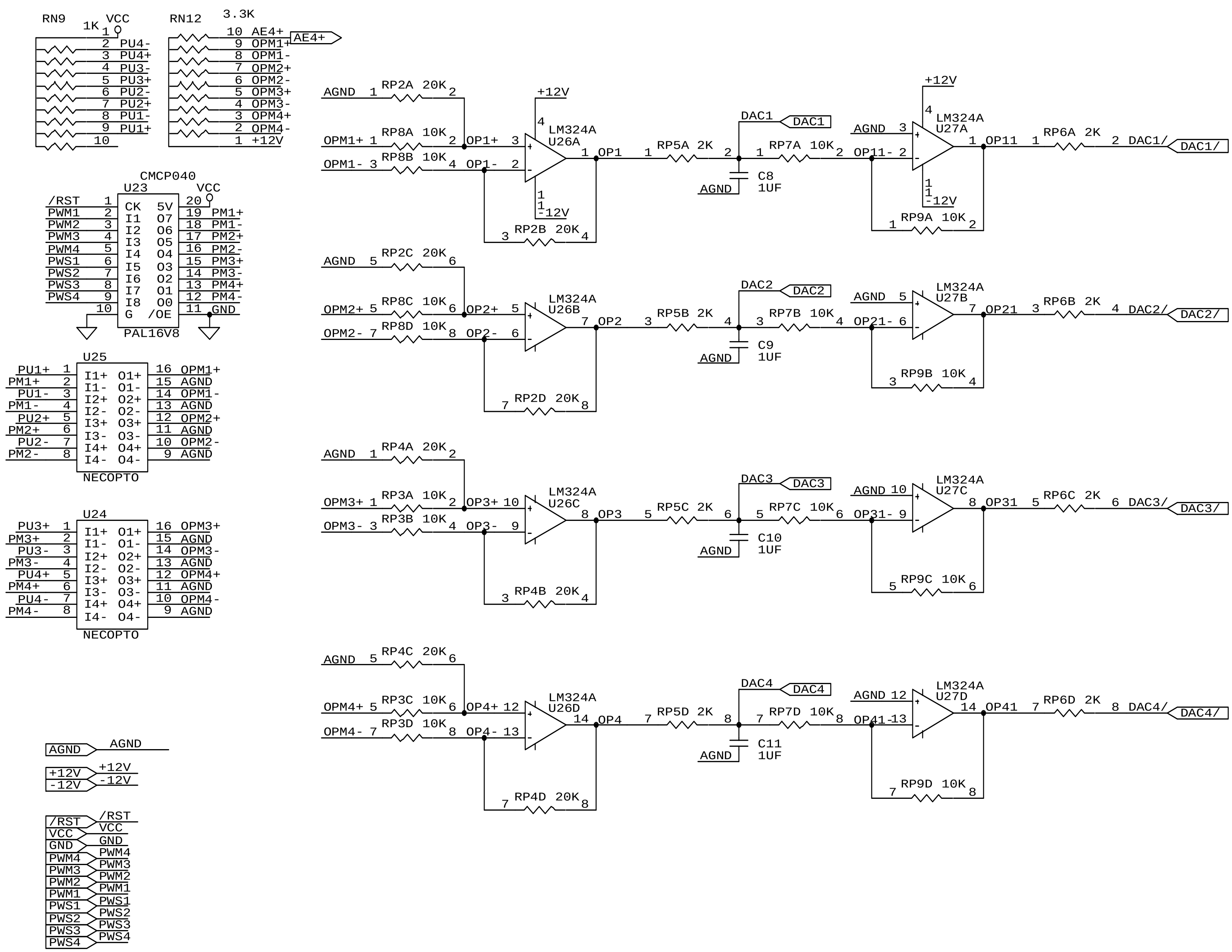
TERN/STE

Title

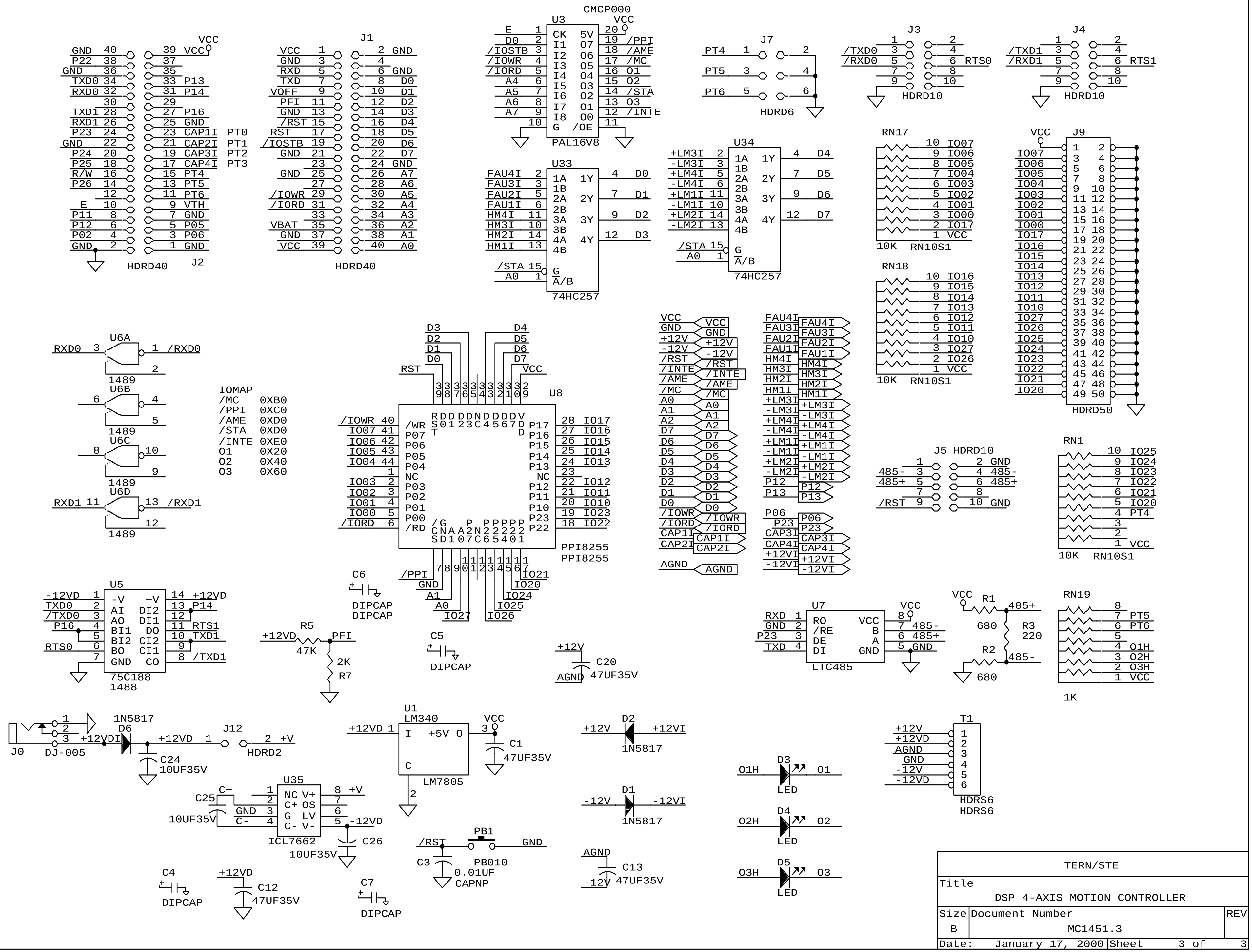
DSP 4-AXIS MOTION CONTROLLER

Size	Document Number	REV
B	MC1451.SCH	

Date: January 17, 2000 Sheet 1 of 3



tern/STE		
Title: 4-AXIS MOTION CONTROLLER DAC		
Size: B	Document Number: MC1451.2	REV
Date: January 17, 2000	Sheet 2 of 3	


IOMAP
/MC 0XB0
/PPI 0XC0
/AME 0XD0
/STA 0XD0
/INTE 0XE0
O1 0X20
O2 0X40
O3 0X60
TERN/STE
Title
DSP 4-AXIS MOTION CONTROLLER
Size Document Number REV
B MC1451.3
Date: January 17, 2000 Sheet 3 of 3