Getting Started with Your TC-GPIB (or Digital's IET11) and the NI-488.2M™ Software for the DECstation 5000

September 1992 Edition

Part Number 320397-01

Note to Digital Equipment Corporation Customers

The IET11 is the Digital Distributed Hardware (DDH) version of the National Instruments TC-GPIB interface board. The DDH product kit also includes the NI-488.2M software. Throughout this manual, all references to the TC-GPIB apply equally to the IET11.

If you have any technical problems with the IET11, contact your local Digital Service Representative.

National Instruments Corporate Headquarters
6504 Bridge Point Parkway
Austin, TX 78730-5039
(512) 794-0100
(800) IEEE-488 (toll-free U.S. and Canada)
Technical support fax: (512) 794-5678

Branch Offices:
Australia 03 879 9422, Belgium 02 757 00 20, Canada 519 622 9310,
Denmark 45 76 73 22, France 1 48 65 33 70, Germany 089 714 50 93,
Italy 02 48301892, Japan 03 3788 1921, Netherlands 01720 45761,
Norway 03 846866, Spain 91 896 0675, Sweden 08 984970,
Switzerland 056 45 58 80, U.K. 0635 523545

Limited Warranty

Note to Digital Equipment Corporation Customers

If your interface board is an IET11 unit bought from Digital Equipment Corporation, the following warranty does not apply. Warranty information for the IET11 is governed by the terms established in the Digital Equipment Corporation warranty.

The TC-GPIB is warranted against defects in materials and workmanship for a period of two years from the date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this manual is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Trademarks

Turbo488®, NAT4882™, NI-488M™, and NI-488.2M™ are trademarks of National Instruments Corporation.

Products and company names listed are trademarks or trade names of their respective companies.

FCC/DOC Radio Frequency Interference Compliance

This equipment generates and uses radio frequency energy and, if not installed and used in strict accordance with the instructions in this manual, may cause interference to radio and television reception. This equipment has been tested and found to comply with the following regulatory agencies:

- The limits for a Class A computing device, in accordance with the specifications in Part 15 of U.S. Federal Communications Commission (FCC) Rules. Operation is subject to the following two conditions:

  – This device may not cause harmful interference.

  – This device must accept any interference received, including interference that may cause undesired operation.

- The limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications (DOC).

These regulations are designed to provide reasonable protection against interference from the equipment to radio and television reception in commercial areas.

There is no guarantee that interference will not occur in a particular installation. However, the chances of interference are much less if the equipment is used according to this instruction manual.

If the equipment does cause interference to radio or television reception, which can be determined by turning the equipment on and off, one or more of the following suggestions may reduce or eliminate the problem.

- Operate the equipment and the receiver on different branches of your AC electrical system.

- Move the equipment away from the receiver with which it is interfering.

- Relocate the equipment with respect to the receiver.

- Reorient the receiver's antenna.

- Be sure that the equipment is plugged into a grounded outlet and that the grounding has not been defeated with a cheater plug.

Notice to user: Changes or modifications not expressly approved by National Instruments could void the user's authority to operate the equipment under the FCC Rules.

If necessary, consult National Instruments or an experienced radio/television technician for additional suggestions. The following booklet prepared by the FCC may also be helpful: *How to Identify and Resolve Radio-TV Interference Problems.* This booklet is available from the U.S. Government Printing Office, Washington, DC 20402, Stock Number 004-000-00345-4.

Für Bundesrepublik Deutschland
For Federal Republic of Germany
Pour la République féderale d´Allemagne

Hochfrequenzgerätezulassung und Betriebsgenehmigung

Bescheinigung des Herstellers/Importeurs

Hiermit wird bescheinigt, daß die Einrichtung in Übereinstimmung mit den Bestimmungen der DBP-Verfügung 523/1969, Amtsblatt 113/1969, und Grenzwertklasse "A" der VDE0871, funkenstört ist.

Das Zentralamt für Zulassungen im Fernmeldewesen der Deutschen Bundespost (DBP), hat diesem Gerät im Zusammenhang mit den Geräten DECstation 5000, Modell 120 und 200, der Digital Equipment GmbH eine FTZ-Serienprüfnummer zugeteilt.

Betrieberhinweis

Das Gerät wurde funktechnisch sorgfältig entstört und geprüft.

Sollten bei diesen Geräten ausnahmsweise trotzdem, z.B. im ungünstigen Fall beim Zusammenschalten mit anderen EDV-Geräten, Funkstörungen auftreten kann das im Einzelnen zusätliche Funkentstörungs-maßnahmen durch den Benutzer erfordem. Bei Fragen hierzu wenden Sie sich bitte an die örtlich zuständige Funkstörungsmeßstelle Ihres Fernmeldeamtes.

TC-GPIB Hardware Safety Information

Read this information before you install the TC-GPIB interface board.

Safety Instructions

To avoid personal injury, fire, or damage to equipment, follow these guidelines when you unpack and install your TC-GPIB board.

- Before removing the TC-GPIB board from its plastic bag, touch the plastic bag to a metal part of your computer chassis.
- Inspect the TC-GPIB board for loose components or any other sign of damage prio to installing it. Do *not* install a damaged board into your computer.
- Before you install or remove the TC-GPIB board or change the shield ground jump connection setting, turn off the power switch on the computer and unplug the powe cord.
- Ensure that the two back panel screws are securely fastened when you attach the mounting bracket of the TC-GPIB board to the back panel of the computer.

Warning: All the devices that you connect to the TC-GPIB board must be complia with the local product-safety regulations such as the UL1950, EN60950, and CSA950 standards that include the Safety Extra Low Voltage (SEL requirements.

SIKKERHEDSINFORMATIONER FOR TC-GPIB-HARDWARE

Læs nedenstående sikkerhedsinformationer før De installerer interfacekortet TC-GPIB

Sikkerhedsinformationer

Følg disse retningslinier når De pakker TC-GPIB-kortet ud og installerer det, for at undgå personskade, brand og skade på udstyret.

- Før TC-GPIB-kortet tages ud af plasticposen, skal posen bringes i berøring med e metaldel på computerrammen.
- Inspicér TC-GPIB-kortet for løse genstande eller andre tegn på beskadigelse, før installerer det. De må ikke installere et beskadiget kort, i computeren.
- Sluk altid for computeren og tag ledningen ud, før De installerer eller fjerner TC-GPIB-kortet, eller ændrer indstillingen på den afskærmede jordforbindelse.
- Sørg for at de to skruer på bagpanelet sidder godt fast, når De monterer TC-GPIB kortets fastspændingsknægt på computerens bagpanel.

Advarsel: Alle anordninger, der sluttes til TC-GPIB-kortet, skal være i overensstemmelse med lokale forskrifter for produktsikkerhed, som f.eks. standarderne UL1950-, EN60950- og CSA950, der omfatter kravene for »Safety Extra Low Voltage« (SELV).

Veiligheidsinformatie voor TC-GPIB hardware

Lees deze informatie voordat u de TC-GPIB interface-kaart installeert.

Instructies betreffende veiligheid

Teneinde lichamelijk letsel, brand of schade aan de apparatuur te voorkomen dient u zich aan deze richtlijnen te houden bij het uitpakken en installeren van uw TC-GPIB kaart.

- Breng, voordat u de TC-GPIB kaart uit de plastic zak haalt, de zak met een metal deel van uw computerrek in aanraking.

- Kijk, voordat u de TC-GPIB kaart installeert, of onderdelen op deze kaart los zitt of de kaart tekenen van beschadiging vertoont. Installeer *geen* beschadigde kaart uw computer.

- Voordat u de TC-GPIB kaart installeert of verwijdert, of de stand van de geleiderbrug afscherming-massa verandert, dient u de aan-uitschakelaar op de computer uit te zetten en de stekker uit het stopcontact te halen.

- Verzeker u ervan dat de twee schroeven van het achterpaneel goed vastgedraaid zijn, wanneer u de montagebeugel van de TC-GPIB kaart aan het achterpaneel va de computer bevestigt.

Waarschuwing: Alle apparaten die u met de TC-GPIB kaart verbindt, moeten aan de plaatselijke veiligheidsvoorschriften voldoen zoals de normen UL1950, EN60950 en CSA950, waarin de SELV (Safety Extra Low Voltage ofwel Veiligheid extra lage spanning) eisen zijn opgenome

Turvallisuusohjeet TC-GPIB laitteistolle

Lue tämä informaatio ennen kuin asennat TC-GPIB-liitäntäpiirilevyn.

Turvallisuusohjeet

Välttääksesi luokkaantumis- ja tulipalovaaraa tai laitteiston vahingoittumista, noudata näitä ohjeita kun purat TC-GPIB-levyn pakkauksesta ja kun asennat sen.

- Ennen kuin otat TC-GPIB-levyn muovipakkauksestaan, anna muovipussin koskettaa tietokoneesi asennuspohjan metalliosaa.
- Varmista ennen asennusta, ettei TC-GPIB-levyssä ole irrallisia osia ja ettei se ole muutoin vahingoittunut. *Älä* asenna vahingoittunutta levyä tietokoneeseen.
- Ennen TC-GPIB-levyn asennusta tai siirtoa tai suojamaadoitetun kytkentäjohdon kuormituksen muutosta, laita tietokone pois päältä katkaisimesta ja vedä sähköjo seinästä.
- Asentaessasi TC-GPIB- levyn kiinnitysuloketta tietokoneen takapaneeliin, varmis että takapaneelin kaksi ruuvia ovat lujasti kiinni.

Varoitus: Kaikkien TC-GPIB-levyyn asentamiesi laitteiden tulee olla paikallisten turvallisuusmääräysten mukaisia, kuten UL1950-, EN60950- ja CSA950-vaatimukset, jotka sisältävät Safety Extra Low Voltage Requirements (SELV)-määräykset.

Matériel TC-GPIB Sécurité

Veuillez prendre connaissance des mesures de sécurité suivantes avant de procéder à l'installation de la carte d'interface TC-GPIB.

Mesures de sécurité

Afin d'éviter toute blessure corporelle, incendie ou dégât de l'équipement, il faudra se conformer aux directives suivantes lors du déballage et de l'installation de la carte d'interface TC-GPIB.

- Avant de retirer la carte TC-GPIB de son sac plastique, faire entrer celui-ci en contact avec la partie métallique du châssis de l'ordinateur.

- Avant de procéder à l'installation de la carte TC-GPIB, vérifier qu'elle ne présen aucun signe de dégât visible et que tous ses composants sont bien en place. *Ne p* installer une carte endommagée dans l'ordinateur.

- Avant d'installer ou de retirer la carte TC-GPIB, ou de procéder à la modification du réglage du cavalier de protection de mise à la terre, mettre l'ordinateur hors circuit et débrancher le câble d'alimentation.

- S'assurer que les deux vis du panneau arrière sont bien serrées lors du montage d dispositif de fixation de la carte TC-GPIB au panneau arrière de l'ordinateur.

Avertissement : Tous les périphériques connectés à la carte TC-GPIB devront être conformes aux normes de sécurité des produits précisées par les réglementations locales (normes UL1950, EN60950 et CSA950, par exemple), notamment en ce qui concerne les conditions de très bass tension de sécurité (SELV).

Sicherheitshinweise für TC-GPIB Hardware

Vor der Installation der TC-GPIB Karte diese Hinweise durchlesen!

Sicherheitshinweise

Zur Vermeidung von Verletzungen, Feuer bzw. Geräteschäden sind diese Hinweise beim Auspacken und bei der Installation der TC-GPIB Karte unbedingt zu beachten.

- Vor der Entnahme der TC-GPIB Karte aus dem Plastikbeutel ein Metallteil des Computergehäuses mit dem Plastikbeutel berühren.

- Vor der Installation der TC-GPIB Karte diese auf lose Bauteile oder andere Schäden überprüfen. Eine beschädigte Karte darf *nicht* im Computer installiert werden!

- Vor dem Ein- oder Ausbau der TC-GPIB Karte bzw. vor einer Änderung der Einstellung des abgeschirmten Masse-Jumpers den Netzschalter des Computers ausschalten und das Netzkabel abziehen.

- Bei der Montage der Halterung für die TC-GPIB Karte an der Rückwand des Computers darauf achten, daß die beiden Schrauben in der Rückwand sicher befestigt sind.

ACHTUNG! Alle an die TC-GPIB Karte angeschlossenen Geräte müssen den entsprechenden Produktsicherheitsvorschriften der Normen wie UL1950, EN60950 und CSA950 entsprechen, die auch Vorschriften z Sicherheitskleinspannung (SELV) beinhalten.

INFORMAZIONI DI SICUREZZA PER HARDWARE TC-GPIB

Leggere queste informazioni prima di installare la scheda di interfaccia TC-GPIB.

Istruzioni per la sicurezza

Per evitare lesioni personali, incendi o danni alle attrezzature, seguire queste direttive massima al momento di disimballare ed installare la scheda TC-GPIB.

- Prima di estrarre la scheda TC-GPIB dal sacchetto di plastica, mettere a contatto sacchetto con una parte metallica del telaio del computer in dotazione.
- Ispezionare la scheda TC-GPIB prima di installarla, in modo da verificare l'assen di componenti allentati o di altri segni di danneggiamento. *Non* installare una scheda danneggiata nel proprio computer.
- Prima di installare o rimuovere una scheda TC-GPIB o di modificare l'impostazione del ponticello di massa dello schermo, mettere fuori tensione il computer ed estrarre la spina di alimentazione dalla presa di rete.
- Al momento di collegare la staffa di supporto della scheda TC-GPIB al pannello posteriore del computer, accertarsi che le due viti del pannello posteriore siano avvitate saldamente.

Avvertenza - Tutti i dispositivi collegati alla scheda TC-GPIB devono conformarsi a requisiti di sicurezza delle apparecchiature previsti dalla normativa loca vigente, come li standard UL1950, EN60950 e CSA950 comprendenti requisiti SELV (Safety Extra Low Voltage o tensione molto bassa di sicurezza).

SIKKERHETINFORMASJON FOR TC-GPIB-MASKINVARE

Les informasjonen nedenfor før du installerer grensesnittkortet TC-GPIB.

Sikkerhetsinformasjon

Følg disse retningslinjene når du pakker ut TC-GPIB-kortet og installerer det, slik at d unngår personskade, brann og skade på utstyret.

- Før du tar TC-GPIB-kortet ut av plastposen, skal posen bringes i berøring med en metalldel på datamaskinrammen.
- Før du installerer kortet bør du kontrollere at TC-GPIB-kortet ikke har noen løse deler eller andre tegn på beskadigelse. Du må *ikke* installere et skadet kort i datamaskinen.
- Slå alltid av datamaskinen og trekk ut ledningen før du installerer eller fjerner TC-GPIB-kortet, eller endrer innstillingen på den skjermede jordforbindelsen.
- Sørg for at de to skruene på bakpanelet sitter godt fast når du fester TC-GPIB-kortets monteringsbrakett til bakpanelet på datamaskinen.

Advarsel: Alle enheter som koples til TC-GPIB-kortet skal være i overensstemmelse med lokale forskrifter for produktsikkerhet, som f.eks. standardene UL195 EN60950 og CSA950 som omfatter kravene for «Safety Extra Low Voltag (SELV).

Medidas de Segurança para a placa TC-GPI

Leia a informação abaixo antes de instalar a placa de interface TC-GPIB.

Medidas de Segurança

Para evitar lesão pessoal, incêndio ou danificação de equipamento, observe estas instruções ao desembalar e instalar a placa TC-GPIB.

- Antes de retirar a placa TC-GPIB do envoltório de plástico, toque com ele uma parte metálica do chassis do seu computador.
- Antes de instalar, inspecione a placa TC-GPIB para verificar se não há componentes soltos ou outros sinais de danificação. Não instale placa danificada seu computador.
- Antes de instalar ou remover a placa TC-GPIB, ou de mudar de posição os conectores de ponte da blindagem de terra, desligue o computador e remova a tomada.
- Ao prender o suporte da placa TC-GPIB ao painel traseiro do computador, verifiq se estão firmemente atarraxados os dois parafusos que o prendem.

Atenção: Todos os dispositivos que você ligar à placa TC-GPIB devem obedecer ao critérios de segurança estabelecidos nos regulamentos locais, como os padrões UL1950, EN60950 e CSA950, que requerem segurança para voltagem extra baixa (SELV).

Información de seguridad para el Hardware TC-GPIB

Lea esta información antes de instalar la tarjeta de interfaz TC-GPIB.

Instrucciones de seguridad

Para evitar lesiones corporales, incendios o daños al equipo, observe estas pautas al desembalar e instalar la tarjeta TC-GPIB.

- Antes de sacar la tarjeta TC-GPIB de su bolsa de plástico, tóquela con la parte metálica del chasis de su ordenador.
- Antes de instalarla, inspeccione la tarjeta TC-GPIB para comprobar que no tenga componentes sueltos o cualquier otro tipo de daño. No instale una tarjeta dañada en su ordenador.
- Antes de instalar o retirar la tarjeta TC-GPIB o de cambiar el ajuste del cable de puente de tierra blindado, apague el interruptor de alimentación del ordenador y desenchufe el cordón de alimentación.
- Asegúrese de que los dos tornillos del panel posterior estén firmemente sujetos al colocar el soporte de montaje de la tarjeta TC-GPIB en el panel posterior del ordenador.

Advertencia: Todos los dispositivos que conecte a la tarjeta TC-GPIB deben cumplir con las normas de seguridad de los productos de acuerdo con los reglamentos locales tales como UL1950, EN60950 y CSA950 que incluyen los requisitos de Seguridad de Voltaje Extraordinariamente Bajo (SELV).

Säkerhetsinformation för TC-GPIB-hårdvar

Ta del av denna information före installation av gränssnittskort TC-GPIB.

Säkerhetsinstruktioner

Följ dessa riktlinjer vid uppackning och installation av TC-GPIB-kortet för att undvik personskada, eld eller skada på utrustning.

- För plastpåsen som omsluter TC-GPIB-kortet lätt mot en del av datorchassit som av metall innan kortet tas ut.
- Kontrollera att kortets komponenter ej är lösa och att inga andra tecken på skada förekommer innan kortet installeras. Installera *inte* ett skadat kort i datorn.
- Bryt strömförsörjningen till datorn och dra ut elkabeln ur dess uttag före installati eller avlägsnande av TC-GPIB-kortet eller före ändring av skärmjordförbindelser
- Säkerställ att de två skruvarna i datorns bakplåt är helt fastdragna när TC-GPIB-kortets fäste monteras i plåten.

Varning: Alla enheter som ansluts till TC-GPIB-kortet måste uppfylla lokala produktsäkerhetsföreskrifter av samma typ som standarderna UL1950, EN60950 och CSA950, vilka innehåller de så kallade “Safety Extra Low Voltage (SELV)”-fordringarna.

Preface

This manual contains instructions for installing and configuring the National Instruments TC-GPIB interface board and the NI-488.2M multitasking software for use with a DECstation 5000 running the ULTRIX operating system. If your interface board is an IET11 unit bought from Digital Equipment Corporation, you should refer to the section *Note to Digital Equipment Corporation Customers* on the title page of this manual. This manual is meant to be used with the *NI-488.2M Software Reference Manual* (Part Number 320351-01).

Organization of This Manual

This manual is organized as follows:

- Chapter 1, *Introduction*, contains a picture of the TC-GPIB interface board, lists the contents of your TC-GPIB kit and optional equipment, and contains instructions for unpacking your TC-GPIB.

- Chapter 2, *Installation and Elementary Programming Example*, contains instructions for installing your NI-488.2M software and the TC-GPIB interface board. This chapter also contains short programming examples using NI-488.2M routines and NI-488M functions as well as information for compiling and linking your program.

- Chapter 3, *Writing an Advanced Program Using NI-488.2M Routines*, introduces you to the NI-488.2M driver and the NI-488.2M routines. To show you how easy it is to use the NI-488.2M routines, this chapter also contains step-by-step instructions for writing an NI-488.2M program.

- Appendix A, *Changing the Shield Ground Jumper Settings*, contains information on changing the configuration of the shield ground jumper.

- Appendix B, *Installing the NI-488.2M Driver Manually*, contains instructions for installing your NI-488.2M software driver manually in the ULTRIX environment.

- Appendix C, *TC-GPIB Specifications*, contains a description of the physical, electrical, and environmental characteristics of the TC-GPIB interface board and the conditions under which it should be operated.

- Appendix D, *Customer Communication*, contains forms for you to complete to facilitate communication with National Instruments concerning our products.

Conventions Used in This Manual

Throughout this manual, the following conventions are used to distinguish elements of text:

italic	Italic text denotes emphasis, a cross reference, or an introduction to a key concept.
`monospace`	Lowercase text in this font denotes text or characters that are to be literally input from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, directories, programs, subprograms, subroutines, device names, functions, variables, and filenames, and for statements and comments taken from program code.
`italic monospace`	Italic lowercase text in this font denotes that you must supply the appropriate words or values in the place of these items.
<>	Angle brackets enclose the name of a key on the keyboard–for example, <PgDn>.
-	A hyphen between two or more key names enclosed in angle brackets denotes that you should simultaneously press the named keys–for example, <Ctrl-C>.
<Enter>	Key names are capitalized.

The word *enter* is reserved to mean that the commands immediately succeeding the word must be typed into the computer, and then executed by pressing the <Enter> key on the keyboard.

Abbreviations

The following abbreviations are used in this manual:

A	amperes
C	Celsius
°	degrees
Hz	hertz
in.	inches
kbytes	1,000 bytes
m	meters
Mbytes	1,000,000 bytes
M-	mega- (10^6)
M	megabytes of memory
µ-	micro- (10^{-6})
sec	seconds

Acronyms

The following acronyms are used in this manual:

AC	alternating current
ANSI	American National Standards Institute
GPIB	General Purpose Interface Bus
EMI	electromagnetic interference
IEEE	Institute of Electrical and Electronic Engineers
I/O	input/output
RAM	random-access memory
VAC	volts alternating current
VDC	volts direct current

Note: References in this manual to IEEE-488 and IEEE-488.2 are referring to the ANSI/IEEE Standard 488.1-1987 and the ANSI/IEEE Standard 488.2-1987, respectively, which define the GPIB.

Mnemonics

The following mnemonics are used in this manual:

CACS	Controller Active State
CIC	Controller-In-Charge
EOI	end or identify
MAV	Message Available
SRQ	Service Request

Related Documents

The following documents contain information that you may find helpful as you read this manual.

- *DECstation 5000 Hardware Operator's Guide*, Digital Equipment Corporation, Maynard, Massachusetts.
- *Writing and Porting VMEbus and TURBOchannel Device Drivers*, Digital Equipment Corporation, Maynard, Massachusetts.
- *Digital Systems and Options Catalog*, Digital Equipment Corporation, Maynard, Massachusetts.
- *DECdirect Applications Catalog*, Digital Equipment Corporation, Maynard, Massachusetts.
- *ULTRIX General Information* and *ULTRIX System and Network Management* (all volumes), Digital Equipment Corporation, Maynard, Massachusetts.
- ANSI/IEEE Standard 488.1-1987, *IEEE Standard Digital Interface for Programmable Instrumentation*.
- ANSI/IEEE Standard 488.2-1987, *IEEE Standard Codes, Formats, Protocols, and Common Commands*.

Customer Communication

National Instruments want to receive your comments on our products and manuals. We are interested in the applications you develop with our products, and we want to help if you have problems with them. To make it easy for you to contact us, this manual contains comment and configuration forms for you to complete. These forms are in Appendix D, *Customer Communication*, at the end of this manual.

Contents

Figures

Chapter 1
Introduction

This chapter contains a picture of the TC-GPIB interface board, lists the contents of your TC-GPIB kit and optional equipment, and contains instructions for unpacking your TC-GPIB.

If your interface board is an IET11 unit bought from Digital Equipment Corporation, you should refer to the *Note to Digital Equipment Corporation Customers* at the front of this manual.

The TC-GPIB is a full-function, high-performance IEEE-488.2 interface for the DECstation 5000. The TC-GPIB makes the DECstation 5000 a high-performance IEEE-488 Controller.

Figure 1-1 shows the TC-GPIB interface board.

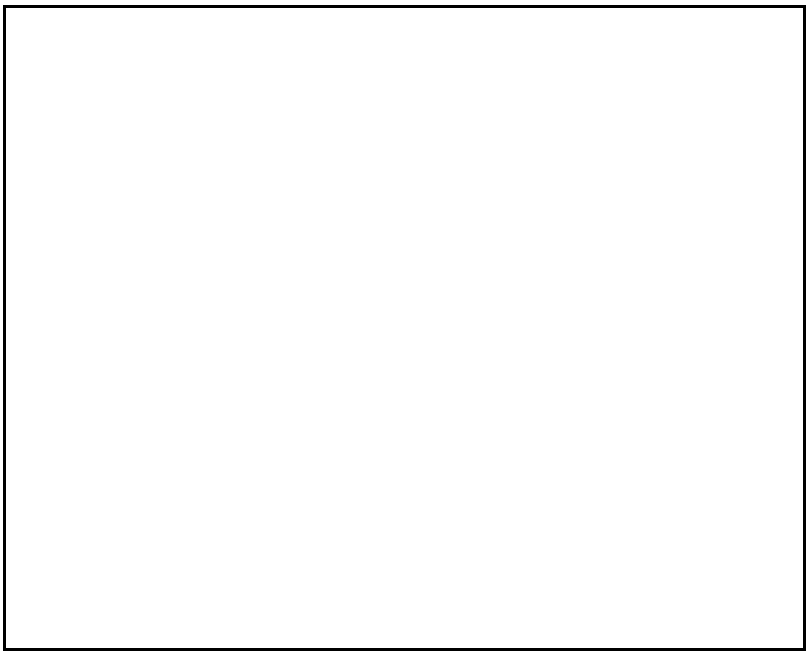

Figure 1-1. TC-GPIB Interface Board

What Your Kit Should Contain

Component	Part Number
TC-GPIB Interface Board	181060-01
Two Fixing Screws	741226-01
3.5 in. NI-488.2M TC-GPIB Distribution Disk	422772-70
TK50 NI-488.2M TC-GPIB Distribution Tape	461772-70
NI-488.2M Software Reference Manual	320351-01
Getting Started with Your TC-GPIB and the NI-488.2M Software for the DECstation 5000	320397-01
Digital Software Support Addendum (SSA)†	YG-A063A-01.A02
†This item is included in the IET11 kit only.	

Make sure each of these items is in your kit. If any item is missing, contact National Instruments.

Optional Equipment

Component	Part Number
Single-Shielded GPIB Cables*:	
Type X1 Cable – 1 m	763001-01
Type X1 Cable – 2 m	763001-02
Type X1 Cable – 4 m	763001-03
Double-Shielded GPIB Cables*:	
Type X2 Cable – 1 m	763061-01
Type X2 Cable – 2 m	763061-02
Type X2 Cable – 4 m	763061-03
* To meet FCC emission limits for this Class A device, you must use a shielded (Type X1 or X2) GPIB cable. Operating this equipment with a non-shielded cable may cause interference to radio and television reception in commercial areas.	

Unpacking Your TC-GPIB

Follow these steps when unpacking your TC-GPIB board:

1. Verify that the pieces contained in the package you received match the kit parts list given earlier in this chapter. Do not remove the board from its plastic bag at this point.

2. Your TC-GPIB board is shipped packaged in an antistatic plastic bag to prevent electrostatic damage to the board. Several components on the board can be damaged by electrostatic discharge. To avoid such damage in handling the board, touch the plastic bag to a metal part of your computer chassis before removing the board from the bag.

3. Remove the board from the bag and inspect the board for loose components or any other sign of damage. Notify National Instruments (or, for IET11 users, your local Digital Service Representative) if the board appears damaged in any way. *Do not* install a damaged board into your computer.

Software Description

The NI-488.2M software is a comprehensive package consisting of a loadable multitasking driver and programs that transform a DECstation 5000 running ULTRIX into a GPIB Controller with complete communications and bus management capabilities. The NI-488.2M package includes a C language interface, a program to test the installation of the software, a hardware diagnostic program, an interactive control program, an interactive configuration program, and an automatic installation program.

You can only order the NI-488.2M software in conjunction with the TC-GPIB. Digital customers should refer to the *Digital Systems and Options Catalog* or the *DECdirect Applications Catalog* for IET11 ordering information. National Instruments customers should contact National Instruments to order the NI-488.2M software.

The Software Support Addendum (SSA), included in the IET11 kit, lists the platforms supported by the IET11. For software licensing and warranty information, refer to the envelope in which your distribution media was shipped.

Chapter 2
Installation and Elementary Programming Example

This chapter contains instructions for installing your NI-488.2M software and the TC-GPIB interface board. This chapter also contains short programming examples using NI-488.2M routines and NI-488M functions as well as information for compiling and linking your program.

Preparing for Installation

Before installing the NI-488.2M software, consider the following:

- You must have super-user privilege to install the NI-488.2M software.
- The NI-488.2M distribution tape is in `tar` format.

Review the main files contained on the NI-488.2M distribution tape. The files are as follows:

- `gpib.o` is the NI-488.2M driver file that is loaded at system startup by the ULTRIX system.
- `cib.c` is the C language source file that contains NI-488M and NI-488.2M subroutines.
- `ugpib.h` is an include file that contains prototypes of the GPIB functions, and useful constants that you may want to use in your GPIB application. It must be included at the beginning of any file that makes GPIB function calls.
- `ibtsta` is a program you can use to test the NI-488.2M software installation.

- `ibic` is an interactive control program that executes the NI-488M functions and NI-488.2M routines you enter from the keyboard. It helps you to learn the functions or routines that you use to program instruments or other GPIB devices and to develop your application program.

- `ibconf` is a software configuration program. It is used to create device nodes and can be used to change the configuration of the NI-488.2M software driver.

- `install_gpib` is the shell script that automatically installs the NI-488.2M driver.

- `ibdiag` is a program you can use to test the TC-GPIB hardware.

Installation

Follow these steps to install your NI-488.2M software and the TC-GPIB interface board. If the following steps do not complete as described or do not contain enough detail, you should follow the directions in Appendix B, *Installing the NI-488.2M Driver Manually*.

Step 1. Install the Software

Perform the following steps to install the NI-488.2M software.

1. Log on as super-user (`root`).

2. Create a working directory (for example, `/usr/gpib`) by entering the following command:

```
mkdir /usr/gpib                    <Enter>
```

Note: For the remainder of this manual, the term *enter* is used to indicate that the user must type in the indicated line and press <Enter>.

3. Change to this working directory by entering the following command:

```
cd /usr/gpib
```

4. Copy the files to this directory by entering one of the following commands. Use Step 4a if you are using a distribution tape. Use Step 4b if you are using a 3.5 in. distribution disk.

   a. From a distribution tape, enter: `tar xvf /dev/rmt0h`

   b. From a distribution disk, enter: `tar xvf /dev/fd0`

5. Install the NI-488.2 driver using Step 5a to install automatically or Step 5b to install manually. For easy installation, use the automatic installation.

   a. Run the program `install_gpib` to install your driver automatically and create a new kernel..

   b. Refer to Appendix B, *Installing the NI-488.2M Driver Manually*, to install the driver manually.

Step 2. Install the Hardware

The following are general instructions for installing the TC-GPIB board. Consult the manual that came with your DECstation for specific instructions and warnings.

1. Shut down your system by entering the following command at the command line prompt:

   ```
   shutdown -h now
   ```

2. Turn off your computer.

3. Unplug the power cord.

4. Remove the cover of the system unit.

Note: Follow steps 5 through 8 for each TC-GPIB board you are installing.

5. Select an empty slot, and remove the slot opening cover.

6. Insert the TC-GPIB interface board into the available slot with the IEEE-488 receptacle sticking out of the opening on the back panel.

7. Screw the mounting bracket of the TC-GPIB to the back panel of the computer using the back panel screws.

8. Screw the TC-GPIB board to the posts on the motherboard at either end of the TURBOChannel connector using the two screws that were supplied with you TC-GPIB kit.

9. Replace the system unit cover.

10. Plug in the power cord.

11. Turn on your computer. The following message appears on your screen after starting your operating system:

    ```
    National Instruments IEEE-488.2 TC-GPIB
    ```

    `ib0 at ibus`*x* (for first TC-GPIB)
    `ib1 at ibus`*x* (for second TC-GPIB)

    where *x* is the corresponding TURBOchannel slot number.

The TC-GPIB interface board is now installed.

Step 3. Run ibconf

1. Run the software configuration utility `ibconf` (you must have super-user privilege). You must run `ibconf` because it creates all device nodes needed by the software.

   You can also use `ibconf` to change the GPIB parameters (refer to Chapter 2, *Installation and Configuration of NI-488.2 Software*, in the *NI-488.2M Software Reference Manual* for a complete list and description of all the parameters).

   If you have named the UNIX kernel something other than the default name (`/vmunix`), enter the full pathname of your kernel. For example, if you named your UNIX kernel `/vmultrix`, enter the following command:

   ```
   ibconf /vmultrix
   ```

2. Exit `ibconf` by typing <Ctrl-O> from the high-level map.

3. Save your changes by typing `<y>` at the prompt. (Even if you have not made any changes to the GPIB parameters, you should still type a `<y>` to have `ibconf` create the special device nodes.)

Step 4. Test the Software Installation

The software installation test `ibtsta` verifies that the software is installed correctly. It checks for a correct node `/dev/gpib0` and correct access to the device driver.

Run `ibtsta` by entering the following command:

```
ibtsta
```

If an error occurs in `ibtsta`, make sure that you have followed the instructions in this chapter correctly. If you still have problems, refer to Appendix D, *Customer Communication*, before contacting National Instruments.

If no error occurs, the NI-488.2M software is installed correctly.

Step 5. Test the Hardware Installation

Before running `ibdiag` be sure that no GPIB cables are connected to the TC-GPIB. Run `ibdiag` by entering the following command:

```
ibdiag
```

Press <Enter> when prompted, and `ibdiag` will test `gpib0`, and, if present, `gpib1`. A series of dots will be printed while the tests are running. If no errors occur, type <`q`> when prompted to quit `ibdiag`. The hardware is installed and functioning properly.

If an error does occur, an error message will be printed and you will be asked if you want to continue testing. Do not continue testing. Type <`n`> to stop testing and then <`q`> to quit. Be sure all cables have been disconnected from the TC-GPIB and run `ibdiag` again. If an error still occurs, record the error message and refer to Appendix D, *Customer Communication*, before contacting National Instruments. IET11 users must contact their local Digital Service Representative for technical support.

Programming Example

The following program is a simple C program written with NI-488.2M routines. You can easily type the whole program into the C environment and run it, because it contains only 11 lines of code, excluding comments. Or, you can inspect and run one of the sample programs that are included on the distribution tape.

This program reads a value from a Fluke 45 multimeter which is at GPIB address 6 and is connected to the DECstation 5000 through TC-GPIB board number 0 (`gpib0`). Refer to the *NI-488.2M Software Reference Manual* for specific information about each routine.

```
#include <stdio.h>
#include "ugpib.h"

main() {
    char Reading [30];

    /* Assert Interface Clear (IFC) for more than 100 µsec on
     * GPIB interface board number zero (0).
     */

    SendIFC(0);                /* initialize bus. */

    /* Selectively clear the multimeter:  address it to listen
     * (GPIB address 6 on board number 0) and send it an SDC
     * multiline command.
     */

    DevClear(0, 6);            /* clear the multimeter. */

    /* Instruct the multimeter (board number 0, GPIB address 6)
     * to reset itself (*RST); measure volts direct current
     * (VDC); select range setting number 2 (RANGE 2); select
     * trigger type number 2 (TRIGGER 2); trigger and start a
     * measurement (*TRG); send the measurement results back
     * (VAL?).  Instruct the TC-GPIB board to append a linefeed
     * character with EOI asserted as the last byte sent to the
     * meter.
     */

    Send(0, 6, "*RST; VDC; RANGE 2; TRIGGER 2; *TRG; VAL?", NLend);

    /* Read the last measured value from the multimeter (board
     * 0, GPIB address 6) into a string.  Stop reading when the
     * meter sends you the END message.
     */

    Receive(0,6,Reading,STOPend);
```

```
    /* Call the ibonl function to disable the hardware and
     * software.  This step also frees the unit descriptor.
     */

    ibonl (0,0)
    printf("%s",Reading);
    exit();
}
```

The same results can be obtained using standard NI-488M functions, as shown in the following C program.

```
#include <stdio.h>
#include "ugpib.h"

    char Reading [30];

main() {
    int Fluke45;
    Fluke45=ibdev(0,6,0,12,1,0);      /* open device */
    ibclr(Fluke45);                   /* clear device */
    ibwrt(Fluke45,
          "*RST; VDC; RANGE 2; TRIGGER 2; *TRG; VAL? ", 42);
    ibrd(Fluke45,Reading,30);
    printf("%s",Reading);
}
```

A more detailed example program using NI-488.2 M routines is given in Chapter 3, *Writing an Advanced Program Using NI-488.2M Routines*. For example programs using NI-488.2M calls, refer to Chapter 4, *NI-488.2M Software Characteristics and Routines*, of the *NI-488.2M Software Reference Manual* and for example programs using NI-488M calls, refer to Chapter 5, *NI-488M Software Characteristics and Functions*, of the same manual.

Compiling and Linking

To create an executable program, complete the following steps:

1. Compile the application program (assume the example program you just created is in a file called `acvolts.c`) by entering one of the following commands:

```
cc acvolts.c cib.o -o acvolts
        or
cc acvolts.c -lgpib -o acvolts
```

2. Run the program by entering the following command:

```
acvolts
```

Chapter 3
Writing an Advanced Program Using NI-488.2M Routines

This chapter introduces you to the NI-488.2M driver and the NI-488.2M routines. To show you how easy it is to use the NI-488.2M routines, this chapter also contains step-by-step instructions for writing an NI-488.2M program.

You can take full advantage of the ANSI/IEEE-488.2-1987 standard by using the NI-488.2M routines. These routines are completely compatible with the Controller commands and protocols defined in IEEE-488.2. The NI-488.2M routines are described more completely in the *NI-488.2M Software Reference Manual*.

The NI-488.2M routines are easy to learn and use. Only a few routines are needed for most application programs.

Interface Boards

The NI-488.2M driver supports up to two TC-GPIB interface boards. These boards are referenced by number from your application program.

- If you have installed only one TC-GPIB board in your computer, its reference number is zero (0).

- If you have installed two boards in your computer, the reference number for the first TC-GPIB board is zero (0) and the reference number for the second TC-GPIB board is one (1). (Looking from the front of your computer, the slots are numbered sequentially from right to left.)

  Refer to Chapter 2, *Installation and Configuration of NI-488.2 Software*, in the *NI-488.2M Software Reference Manual* for information about running and using `ibconf`.

Steps for Writing an NI-488.2M Program

This section demonstrates how to use the NI-488.2M routines. The example program configures a Fluke 45 digital multimeter, reads back 10 voltage measurements, and computes the average of these measurements. Error detection and reporting are handled explicitly in the program. In addition, several more of the NI-488.2M routines are used than were shown in Chapter 2. This program finds and identifies all the Listeners on the bus prior to controlling the Fluke 45 instrument.

Step 1. Preparation

The first step in writing the program is to load in the definitions of the NI-488.2M routines from a file that is on your distribution tape. Include the declaration file for the NI-488.2M C language interface as follows:

```
#include "ugpib.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Declare the subroutine `found`, which is called when the Fluke 45 is identified as a Listener on the GPIB.

```
void found (unsigned int fluke);
```

To check for errors, declare a subroutine to handle all GPIB errors.

```
void gpiberr(char *msg);
#define MAVbit 0x10 /* Position of the Message Available bit.*/
```

The following arrays are used to hold lists of IEEE-488 addresses, which are used to find all of the Listeners on the bus.

```
char    buffer[101];
int     loop,
        m,
        num_listeners,
        SRQasserted;
double  sum;
unsigned int  instruments[32],
              result[31],
              fluke,
              pad,
              statusByte;

void main() {
```

Step 2. Initialization

Initializing the IEEE-488 bus and the TC-GPIB Controller circuitry does the following:

- makes the IEEE-488 interface for each device quiescent
- makes the TC-GPIB Controller-In-Charge (CIC)
- puts the TC-GPIB into Controller Active State (CACS)

Initialization is accomplished with the `SendIFC` procedure. The first and only argument of `SendIFC` is the TC-GPIB interface board number.

```
system("clear");

SendIFC(0);
if (ibsta & ERR) {
    gpiberr ("SendIFC Error");
    exit(1);
}
```

Step 3. Find All Listeners

Create an array that contains all of the IEEE-488 primary addresses that could possibly be connected to the IEEE-488 bus. Assume that you are only attaching devices with primary addresses (0 to 30) to the bus. The end of the list of addresses must be marked with the `NOADDR` constant, which is defined in the `ugpib.h` file that was included at the beginning of this program.

```
for (loop = 0; loop <= 30; loop++) {
    instruments[loop] = loop;
}
instruments[31] = NOADDR;
```

Next find out which, if any, device or devices are attached to the IEEE-488 bus. This is easily accomplished by using the `FindLstn` procedure. The first argument is the TC-GPIB board number, the second argument is the list of instruments that was created above, the third argument is a list of instrument addresses that the procedure actually found, and the last argument is the maximum number of devices that the procedure can find–that is, it must stop if it reaches the limit.

```
    printf("Finding all listeners on the bus...\n");
    printf("\n");

    FindLstn(0, instr uments, result, 31);
    if (ibsta & ERR) {
        gpiberr("FindLstn Error");
        exit(1);
    }
```

Step 4. Identify the Instrument

Now that you have determined the addresses of the devices, you should send an identification query to each device for identification. For this example, assume that all of the instruments are IEEE-488.2 compatible and can accept the 488.2 identification query, `*IDN?`.

In addition, assume that `FindLstn` found the TC-GPIB interface board at primary address 0 (which is where it is by default). Therefore, you can skip the `0` entry in the `result` array.

```
    num_listeners = ibcnt - 1;
    printf("Number of instruments found = %d\n",
num_listeners);
    for (loop = 1; loop <= num_listeners; loop++) {
        Send(0, result[loop], "*IDN?", 5L, NLend);
        if (ibsta & ERR) {
            gpiberr("Send Error");
            exit(1);
        }
        Receive(0, result[loop], buffer, 10L, STOPend);
        if (ibsta & ERR) {
            gpiberr("Receive Error");
            exit(1);
        }
        pad = GetPAD(result[loop]);
        buffer[ibcnt] = '\0';
        printf("The instrument at address %d is a %s\n", pad,
          buffer);
        if (strncmp(buffer, "FLUKE, 45", 9) == 0) {
            fluke = pad;
            printf("**** We found the Fluke ****\n");
            found(fluke);
            break;
        }
    }
    if (loop > num_listeners)
        printf("Did not find the Fluke!\n");
    ibonl (0,0);
}
void found(unsigned int fluke) {
```

The constant NLend signals that the new line character with EOI will be automatically appended to the data.

The constant STOPend indicates that the read is stopped when EOI is detected.

Step 5. Initialize the Instrument

Now that you have found the multimeter, clear it. Initialization of the devices attached to the IEEE-488 bus is accomplished with the DevClear procedure. The first argument is the TC-GPIB interface board number. The second argument is the IEEE-488 address of the multimeter.

```
DevClear(0, fluke);
if (ibsta & ERR) {
    gpiberr("DevClear Error");
    exit(1);
}
Send(0, fluke, "*RST", 4L, NLend);
if (ibsta & ERR) {
    gpiberr("Send *RST Error");
    exit(1);
}
```

Step 6. Configure the Instrument

After initialization, the instrument is ready to receive instructions. To configure the multimeter, send device-specific commands using the Send routine. The first argument of the Send routine is the GPIB interface board number. The second argument is the IEEE-488 address of the multimeter. The third argument is a string of bytes to send to the multimeter.

The device commands in this example instruct the meter to measure volts alternating current (VAC) using auto-ranging (AUTO) and then to wait for a trigger from the Controller before starting a measurement (TRIGGER 2). The last instruction (*SRE 16) asserts the IEEE-488 Service Request signal line, SRQ, when the measurement has been completed and the meter is ready to send the result. The last argument, NLend, is a constant that is defined in ugpib.h. It instructs Send to append a linefeed character with EOI asserted to the end of the message sent to the multimeter.

The calls to actually trigger the instrument and receive the measurement are placed in a loop to repeat 10 times.

```
    Send(0, fluke, "VAC; AUTO; TRIGGER 2; *SRE 16", 29L,
NLend);
    if (ibsta & ERR) {
        gpiberr("Send Setup Error");
        exit(1);
    }
    sum = 0.0;
    for (m=0; m < 10 ; m++) {
```

Step 7. Trigger the Instrument

In the previous step, you instructed the multimeter to wait for a trigger before conducting a measurement. Now you want to send a trigger command to the multimeter. You could use the `Trigger` routine to accomplish this, but because the Fluke 45 is IEEE-488.2 compatible, you can just send it the trigger command `*TRG`. The other command, `VAL1?`, instructs the meter to send the next triggered reading to its IEEE-488.2 output buffer.

```
Send(0, fluke, "*TRG; VAL1?", 11L, NLend);
if (ibsta & ERR) {
    gpiberr("Send Trigger Error");
    exit(1);
}
```

Step 8. Wait for the Instrument to Complete the Measurement

After the meter is triggered, it takes a measurement and displays it on its front panel. When this is finished, the meter asserts SRQ.

You can detect the assertion of SRQ using either the `TestSRQ` or `WaitSRQ` routine. If you have processing that you want to do while you are waiting for the measurement, you would use `TestSRQ`. For this example, there is nothing in particular to do at this point, so you can use the `WaitSRQ` routine.

The first argument to `WaitSRQ` is the TC-GPIB board number. The second argument is a flag returned by `WaitSRQ` that indicates whether SRQ is asserted.

```
WaitSRQ(0, &SRQasserted);
if (!SRQasserted) {
    printf("SRQ is not asserted.  The Fluke is not
      ready.\n");
    exit(1);
}
```

After you have detected SRQ, poll the meter to ascertain its status. This is accomplished by using the `ReadStatusByte` procedure. The first argument for this procedure is the TC-GPIB board number, the second argument is the IEEE-488 address of the instrument, and the last argument is a variable that `ReadStatusByte` uses to store the status byte of the instrument.

```
ReadStatusByte(0, fluke, &statusByte);
if (ibsta & ERR) {
    gpiberr("ReadStatusByte Error");
    exit(1);
}
```

After you have obtained the status byte, you must check to see if the meter has a message to send. You can find this out by checking the message available (MAV) bit, bit 4, in the status byte.

```
if (!(statusByte & MAVbit)) {
    gpiberr("Improper Status Byte");
    printf(" Status Byte = 0x%x\n", statusByte);
    exit(1);
}
```

Step 9. Read the Measurement

Use the `Receive` function to read the measurement over the IEEE-488 bus. Again, the first argument is the GPIB interface board number, and the second argument is the IEEE-488 address of the multimeter. The third argument is a string into which the `Receive` function places the data bytes from the multimeter. The last argument specifies that the `Receive` function is to terminate upon receiving a byte accompanied with the `END` message. The loop is then repeated. The average value is printed at the end of the 10 measurements.

```
            Receive(0, fluke, buffer, 10L, STOPend);
            if (ibsta & ERR) {
                gpiberr("Receive Error");
                exit(1);
            }
            buffer[ibcnt] = '\0';
            printf("Reading :  %s\n", buffer);
            sum = sum + atof(buffer);
    }
    printf(" The average of the 10 readings is : %f\n",
sum/10);
}
```

The Error Handling Subroutine

The main program calls this routine whenever an error occurs. This subroutine supplies full messages for every possible error. If you want to simplify it, just use the first three lines, and the last line.

```
void gpiberr(char *msg) {

    printf ("%s\n", msg);

    printf ("ibsta = 0x%x  <", ibsta);
    if (ibsta & ERR ) printf (" ERR");
    if (ibsta & TIMO) printf (" TIMO");
    if (ibsta & END ) printf (" END");
    if (ibsta & SRQI) printf (" SRQI");
    if (ibsta & RQS ) printf (" RQS");
    if (ibsta & CMPL) printf (" CMPL");
    if (ibsta & LOK ) printf (" LOK");
    if (ibsta & REM ) printf (" REM");
    if (ibsta & CIC ) printf (" CIC");
    if (ibsta & ATN ) printf (" ATN");
    if (ibsta & TACS) printf (" TACS");
    if (ibsta & LACS) printf (" LACS");
    if (ibsta & DTAS) printf (" DTAS");
    if (ibsta & DCAS) printf (" DCAS");
    printf (" >\n");

    printf ("iberr = %d", iberr);
    if (iberr == EDVR) printf (" EDVR <UNIX Error>\n");
    if (iberr == ECIC) printf (" ECIC <Not CIC>\n");
    if (iberr == ENOL) printf (" ENOL <No Listener>\n");
    if (iberr == EADR) printf (" EADR <Address error>\n");
    if (iberr == EARG) printf (" EARG <Invalid argument>\n");
    if (iberr == ESAC) printf (" ESAC <Not Sys Ctrlr>\n");
    if (iberr == EABO) printf (" EABO <Op. aborted>\n");
    if (iberr == ENEB) printf (" ENEB <No GPIB board>\n");
    if (iberr == ECAP) printf (" ECAP <No capability>\n");
    if (iberr == EFSO) printf (" EFSO <File sys. error>\n");
    if (iberr == EBUS) printf (" EBUS <Command error>\n");
    if (iberr == ESTB) printf (" ESTB <Status byte lost>\n");
    if (iberr == ESRQ) printf (" ESRQ <SRQ stuck on>\n");
    if (iberr == ETAB) printf (" ETAB <Table Overflow>\n");

    printf ("ibcnt = %ld\n", ibcnt);
    printf ("\n");

    ibonl (0,0);

}
```

The Complete Application Program in C

The following program combines the previous steps into one program.

```
#include "ugpib.h"

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void found (unsigned int fluke);
void gpiberr(char *msg);

#define MAVbit 0x10
char    buffer[101];
int     loop,
        m,
        num_listeners,
        SRQasserted;
double  sum;
unsigned int  instruments[32],
              result[31],
              fluke,
              pad,
              statusByte;


void main() {

    system("clear");

    SendIFC(0);
    if (ibsta & ERR) {
        gpiberr ("SendIFC Error");
        exit(1);
    }

    for (loop = 0; loop <= 30; loop++) {
        instruments[loop] = loop;
    }
    instruments[31] = NOADDR;

    printf("Finding all listeners on the bus...\n");
    printf("\n");

    FindLstn(0, instruments, result, 31);
    if (ibsta & ERR) {
        gpiberr("FindLstn Error");
        exit(1);
    }

    num_listeners = ibcnt - 1;
```

```
    printf("Number of instruments found = %d\n", num_listeners);

    for (loop = 1; loop <= num_listeners; loop++) {

        Send(0, result[loop], "*IDN?", 5L, NLend);
        if (ibsta & ERR) {
            gpiberr("Send Error");
            exit(1);
        }

        Receive(0, result[loop], buffer, 10L, STOPend);
        if (ibsta & ERR) {
            gpiberr("Receive Error");
            exit(1);
        }

        pad = GetPAD(result[loop]);

        buffer[ibcnt] = '\0';
        printf("The instrument at address %d is a %s\n", pad,
          buffer);

        if (strncmp(buffer, "FLUKE, 45", 9) == 0) {
            fluke = pad;
            printf("**** We found the Fluke ****\n");
            found(fluke);
            break;
        }
    }

    if (loop > num_listeners)
        printf("Did not find the Fluke!\n");

    ibonl (0,0);

}

void found(unsigned int fluke) {

    DevClear(0, fluke);
    if (ibsta & ERR) {
        gpiberr("DevClear Error");
        exit(1);
    }

    Send(0, fluke, "*RST", 4L, NLend);
    if (ibsta & ERR) {
        gpiberr("Send *RST Error");
        exit(1);
    }
```

```
    Send(0, fluke, "VAC; AUTO; TRIGGER 2; *SRE 16", 29L, NLend);
    if (ibsta & ERR) {
        gpiberr("Send Setup Error");
        exit(1);
    }

    sum = 0.0;

    for (m=0; m < 10 ; m++) {

                Send(0, fluke, "*TRG; VAL1?", 11L, NLend);
            if (ibsta & ERR) {
                gpiberr("Send Trigger Error");
                exit(1);
            }

            WaitSRQ(0, &SRQasserted);
            if (!SRQasserted) {
                printf("SRQ is not asserted.  The Fluke is not
                  ready.\n");
                exit(1);
            }

            ReadStatusByte(0, fluke, &statusByte);
            if (ibsta & ERR) {
                gpiberr("ReadStatusByte Error");
                exit(1);
            }

            if (!(statusByte & MAVbit)) {
                gpiberr("Improper Status Byte");
                printf(" Status Byte = 0x%x\n", statusByte);
                exit(1);
            }

            Receive(0, fluke, buffer, 10L, STOPend);
            if (ibsta & ERR) {
                gpiberr("Receive Error");
                exit(1);
            }

            buffer[ibcnt] = '\0';
            printf("Reading :  %s\n", buffer);

            sum = sum + atof(buffer);

    }

    printf("The average of the 10 readings is : %f\n", sum/10);

}
void gpiberr(char *msg) {

    printf ("%s\n", msg);
```

```
    printf ("ibsta = 0x%x  <", ibsta);
    if (ibsta & ERR ) printf (" ERR");
    if (ibsta & TIMO) printf (" TIMO");
    if (ibsta & END ) printf (" END");
    if (ibsta & SRQI) printf (" SRQI");
    if (ibsta & RQS ) printf (" RQS");
    if (ibsta & CMPL) printf (" CMPL");
    if (ibsta & LOK ) printf (" LOK");
    if (ibsta & REM ) printf (" REM");
    if (ibsta & CIC ) printf (" CIC");
    if (ibsta & ATN ) printf (" ATN");
    if (ibsta & TACS) printf (" TACS");
    if (ibsta & LACS) printf (" LACS");
    if (ibsta & DTAS) printf (" DTAS");
    if (ibsta & DCAS) printf (" DCAS");
    printf (" >\n");

    printf ("iberr = %d", iberr);
    if (iberr == EDVR) printf (" EDVR <UNIX Error>\n");
    if (iberr == ECIC) printf (" ECIC <Not CIC>\n");
    if (iberr == ENOL) printf (" ENOL <No Listener>\n");
    if (iberr == EADR) printf (" EADR <Address error>\n");
    if (iberr == EARG) printf (" EARG <Invalid argument>\n");
    if (iberr == ESAC) printf (" ESAC <Not Sys Ctrlr>\n");
    if (iberr == EABO) printf (" EABO <Op. aborted>\n");
    if (iberr == ENEB) printf (" ENEB <No GPIB board>\n");
    if (iberr == ECAP) printf (" ECAP <No capability>\n");
    if (iberr == EFSO) printf (" EFSO <File sys. error>\n");
    if (iberr == EBUS) printf (" EBUS <Command error>\n");
    if (iberr == ESTB) printf (" ESTB <Status byte lost>\n");
    if (iberr == ESRQ) printf (" ESRQ <SRQ stuck on>\n");
    if (iberr == ETAB) printf (" ETAB <Table Overflow>\n");

    printf ("ibcnt = %ld\n", ibcnt);
    printf ("\n");

    ibonl (0,0);

}
```

Helpful Hint

Now that your software and hardware are installed and you have used some NI-488.2M routines, read the *NI-488.2M Software Reference Manual* for a description of all of the NI-488.2M routines available for the TC-GPIB. After reading about these functions and their capabilities, practice using them with your programmable instrument or device in an interactive environment using the `ibic` program described in Chapter 6 of the *NI-488.2M Software Reference Manual*.

Appendix A
Changing the Shield Ground Jumper Settings

This appendix contains information on changing the configuration of the shield ground jumper.

The TC-GPIB is set at the factory with the jumper in place to connect the logic ground of the TC-GPIB to its shield ground. This configuration minimizes the EMI emitted from a DECstation 5000 equipped with a TC-GPIB. However, if your application requires that logic ground be disconnected from shield ground, remove the jumper (W1) and place it across only one of the jumper pins. Jumper settings for connecting the logic ground to shield ground and disconnecting the logic ground from shield ground are shown in Figure A-1.

Note: The TC-GPIB was tested for compliance with FCC, VCCI, and VDE standards with the shield ground connected to logic ground. Removing the jumper may cause EMI emissions to exceed any or all of the applicable standards.

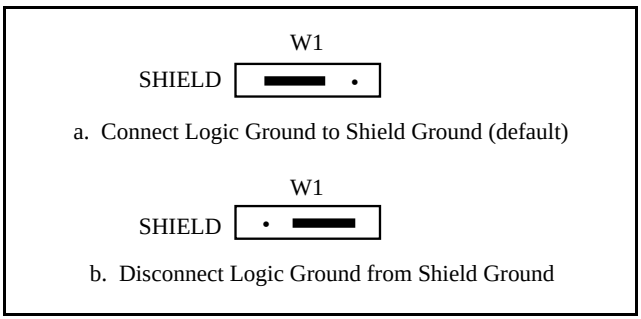

Figure A-1. Ground Configuration Jumper Settings

In the space provided here, record the new ground setting so that you will have it handy for future reference.

TC-GPIB	**New Setting**
Ground Setting:	_____

Appendix B
Installing the NI-488.2M Driver Manually

This appendix contains instructions for installing your NI-488.2M software driver manually in the ULTRIX environment.

Perform the following steps to install the NI-488.2M driver.

1. Compile the `cib.c` file in the distribution directory and turn it into a library by entering the following commands:

   ```
   cc -c cib.c
   ar r /lib/libgpib.a cib.o
   ranlib /lib/libgpib.a
   ```

   Alternately, you can add `cib.o` to an existing library or include it during the link phase of each compile operation.

2. Copy the `ibic` and `ibconf` files to a directory in the command search path (for example, `/bin` or `/usr/bin`).

3. Back up the `conf.c` file in the `/usr/sys/machine/ common` directory by entering the following command:

   ```
   cp conf.c conf.bak
   ```

4. Edit the `/usr/sys/machine/common/conf.c` file as follows:

   a. Search the file for the following statement:

      ```
      #if NIB > 0
      ```

      The section before this statement appears as follows:

      ```
      #ifdef vax
      #include "ib.h"
      #else
      #define NIB 0
      #endif vax
      ```

b. Replace `vax` with `mips` in the lines `#ifdef vax` and `#endif vax.`

c. Search for the following section in the `cdevsw` structure.

```
{  ibopen, ibclose, ibread, ibwrite, /*xx*/
   ibioctl,nodev,  nodev,   0,
   seltrue,nodev,  0        0},
```

xx in the above statement denotes the major number. Make a note of that number so you can use it in the next step.

5. Create a node for `gpib0` in the `/dev` file by entering the following command:

```
mknod /dev/gpib0 c xx 255
```

where *xx* is the major number found in step 4c.

6. Save a copy of the `tc_option_data.c` file in the `/usr/sys/data` directory by entering the following command:

```
cp tc_option_data.c  tc_option_data.bak
```

7. Edit the `/usr/sys/data/tc_option_data.c` file and search for the `tc_option` structure. Add the following line to the structure:

```
{  "TC-GPIB",  "ib",  0,  1,  'D',  0}, /*GPIB*/
```

8. Save a copy of the `files.mips` file in the `/usr/sys/conf/mips` directory by entering the following command:

```
cp files.mips files.bak
```

9. Edit the file `/usr/sys/conf/mips/files.mips` by adding the following line at the end of the file:

```
io/tc/gpib.c   optional ib device-driver Binary
```

10. Save a copy of the file *machine* (where *machine* is the name of your machine) in the /usr/sys/conf/mips directory. For example, if your machine name is National, enter the following command:

```
cp National National.bak
```

11. Edit the file /usr/sys/conf/mips/*machine* (where *machine* is the name of your machine) by searching for the device entries, which begin with device, and adding the following commands at the end of the device entries:

```
device     ib0     at ibus?    vector ibintr
```

For two TC-GPIB interface boards, enter the following commands:

```
device     ib0     at ibus?    vector ibintr
device     ib1     at ibus?    vector ibintr
```

12. In the /usr/sys/conf/mips directory, enter the following command:

```
config machine
```

where *machine* is the name of your machine.

13. Change the directory to /usr/sys/MIPS/*machine* (where *machine* is the name of your machine). This directory was created by the config command. In this directory, enter the following command to create a new vmunix file:

```
make depend all
```

14. Save the existing vmunix kernel by entering the following command:

```
mv /vmunix /vmunix.bak
```

15. Move the new vmunix to the root directory by entering the following command:

```
mv vmunix /
```

Proceed to *Step 2. Install the Hardware* in Chapter 2.

Appendix C
TC-GPIB Specifications

This appendix contains a description of the physical, electrical, and environmental characteristics of the TC-GPIB interface board and the conditions under which it should be operated. All specifications for the TC-GPIB apply equally to the IET11.

Table A-1. Electrical Characteristics

Characteristic	Specification
Turbo488 clock	20 MHz
NAT4882 clock	20 MHz
Transfer Rates GPIB Reads GPIB Writes GPIB Commands	 over 1 Mbyte/sec* over 1 Mbyte/sec* over 400 Kbytes/sec*
Power Requirement (from TURBOchannel Bus)	+5 VDC, 4.0 A maximum, 2.1 A typical
*Actual rates depend upon instrumentation capabilities.	

Table A-2. Environmental Characteristics

Characteristic	Specification
Operating Environment Ambient Temperature Relative Humidity	 0° to 40° C 5% to 90%, noncondensing

(continues)

Table A-2. Environmental Characteristics (continued)

Characteristic	Specification
Storage Environment	
Temperature	-55° to 150° C
Relative Humidity	5% to 90%, noncondensing
EMI	FCC Class A Verified

Table A-3. Physical Characteristics

Characteristic	Specification
Dimensions	5.675 in. x 4.6 in.
I/O Connector	IEEE-488 standard 24-pin

Appendix D
Customer Communication

For your convenience this appendix contains forms to help you gather the information necessary to help us solve possible technical problems, as well as a form you can use to comment on the product documentation.

By completing these forms before calling National Instruments, you will save yourself time, and our applications engineers will be able to answer your questions more accurately and efficiently. The forms contain the information that the applications engineers need from you to help solve your problem. Briefly jot down the information requested on the line after each item.

Note to Digital Equipment Corporation Customers: If you need technical support for your IET11, please contact your local Digital Service Representative.

Fax Technical Support

If you encounter any technical problems, please complete the fax and configuration forms before requesting technical support by fax. You can contact us by fax at any time at the following number: (512) 794-5678.

Telephone Technical Support

For best service by telephone, please complete the fax and configuration forms, record any error messages, and be available at your computer when you call for technical support. You can use the following numbers between the hours of 8:00 a.m. and 5:30 p.m. (central time) to call the National Instruments applications engineering department:

(512) 794-0100 or (800) IEEE-488 (toll-free U.S. and Canada)

Documentation Comments

You can use the *Documentation Comment Form* for your comments about our documents. Please mail or fax it to National Instruments.

Technical Support Fax Form

Technical support is available at any time by fax at (512) 794-5678. For best results, provide as much information as possible. Include the information from your configuration form. Use additional pages if necessary.

Name _____

Company _____

Address _____

_____

Fax (___) _____ Phone (___) _____

Computer brand _____

Model _____ Processor _____

Operating system _____

Speed _____MHz RAM _____M

Display adapter _____

Mouse _____yes _____no

Other adapters installed _____

Hard disk capacity _____M Brand _____

Instruments used _____

National Instruments hardware product model _____

Revision _____

Configuration _____

National Instruments software product _____

Revision _____

Configuration _____

(continues)

The problem is _____

List any error messages _____

The following steps will reproduce the problem _____

TC-GPIB Hardware and Software Configuration Form

Record the settings and revisions of your hardware and software on the line to the right of each item. Update this form each time you revise your software or hardware configuration, and use this form as a reference for your current configuration.

National Instruments Products

- TC-GPIB Revision: _____
- NI-488.2M Software Revision/Version Number on Distribution Medium: _____
- ULTRIX Version: _____
- Other devices in System: _____

Other Products

- Programming Language and Version: _____
- Computer Make and Model: _____
- Clock Frequency: _____
- Type of other boards installed: _____

Documentation Comment Form

National Instruments encourages you to comment on the documentation supplied with our products. This information helps us provide quality products to meet your needs.

Title: **Getting Started with Your TC-GPIB (or Digital's IET11) and the NI-488.2M™ Software for the DECstation 5000**

Edition Date: **September 1992**

Part Number: **320397-01**

Please comment on the completeness, clarity, and organization of the manual.

(continues)

If you find errors in the manual, please record the page numbers and describe the errors.

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Thank you for your help.

Name _____

Title _____

Company _____

Address _____

_____

Phone (_____) _____

Mail to: Technical Publications
National Instruments Corporation
6504 Bridge Point Parkway, MS 53-02
Austin, TX 78730-5039

Fax to: Technical Publications
National Instruments Corporation
MS 53-02
(512) 794-5678