Using the TORQ Database

Joel Hasbrouck*

1992

*Visiting Research Economist
New York Stock Exchange
11 Wall Street, 17th Floor
New York, NY 10005

and

Associate Professor of Finance
NYU, Stern School of Business
Finance Dept., Suite 9-190
44 W. Fourth St.
New York, NY 10012

The comments and opinions contained in this paper are those of the author only. In particular, the views expressed here do not necessarily reflect those of the directors, members, or officers of the New York Stock Exchange, Inc.

For assistance in compiling the TORQ database and the documentation, I am indebted to James Cochrane, Eon Griffith, Katharine Ross, James Shapiro, Jason Shell, George Sofianos, and Deborah Sosebee. None of them is responsible for any errors.

Preface

The TORQ database contains transactions, quotes, order processing data and audit trail data for a sample of 144 NYSE stocks for the three months November, 1990 through January 1991. This document covers installation, formatting and use of the data. Conceptual and institutional details concerning the data are given in a companion publication Hasbrouck and Sosebee (1992).

These data are distributed for purposes of academic research. No warranty is made that they are free of errors. The user assumes all responsibility for the consequences of any errors.

Support for the TORQ database is via electronic bulletin board. The NYSE’s academic research bulletin board can be reached at 212 656-2541, except from 4-5PM Eastern time. The board supports 300/1200/2400 baud. No direct voice support is provided.

Table of Contents

1. Introduction.

The TORQ database contains transactions, quotes, order processing data and audit trail data for a sample of 144 NYSE stocks for the three months November, 1990 through January 1991. The data are distributed on a Compact Disk Read-Only Memory (CD-ROM) that contains eight files. The most important ones are:

CT — Consolidated Transaction file. Records of the transactions and associated data.

CQ — Consolidated Quote file. Records of quote changes.

SOD — System Order Database file. Describes orders entered into the NYSE’s computer systems, and what happened to these orders.

CD — Consolidated Audit Trail file. Decomposes each transaction according to number and type of participants.

These are subset copies of internal NYSE files. The four other files on the CD-ROM contain pointers and other ancillary information. These latter files were constructed for the TORQ distribution and are not official NYSE files.

This document is organized as follows. Section 2 describes the installation procedure. Section 3 presents an overview of the TORQ data. Section 4 discusses the SELECT programs -- MS-DOS programs that you can use to build small subsets of the TORQ data for viewing or statistical analysis. Section 5 discusses the data files. Section 6 provides miscellaneous tips. Appendix A gives the file formats. Appendix B is a variable glossary. Appendix C presents a sample analysis using the TORQ data. Appendix D describes the source code of the SELECT programs.

2. Installation.

The TORQ files are distributed on a single CD-ROM. Data CD-ROMs use the same technology as do audio CDs, but are formatted differently. Another important difference is that information loss due to dust, fingerprints or general abusive treatment is much more detrimental to data CDs than their audio counterparts. Data CDs should be handled more carefully. To your PC, the CD-ROM will look like a big hard disk. You will be able to read from it, copy files from it, and use all the usual DOS commands. You will not be able to write to it.

The TORQ CD-ROM is mastered to ISO-9660 standards: it should be readable on systems other than MS-DOS PCs. The SELECT programs are specific to MS-DOS, however, and will not run on other systems without modification.

Installing the TORQ Data.

0. **Requirements:** You will need a CD-ROM reader **and** the Microsoft CD-ROM extension program (MSCDEX.EXE). This program contains instructions that let MS-DOS think that the CD-ROM reader is a normal DOS hard disk. It should be loaded in your autoexec.bat initialization file. Although many programs require this module, it is not always supplied with the CD-ROM hardware. Nevertheless, most manufacturers will make it available on request. To run the SELECT data extraction programs, you will need several megabytes of hard disk space. (The exact amount depends on how much data you plan to extract at any one time.)
1. First make sure that your CD-ROM reader is correctly installed and that you can read some non-TORQ CD-ROM. You will need to know the DOS device letter assigned to the CD-ROM drive. Look at the line in the autoexec.bat file that loads MSCDEX. There should be a parameter that looks like "/L:n". The "n" is the drive letter. If the "/L" parameter is omitted from the MSCDEX command line, the system defaults to the next available drive letter. For example, if you have one hard disk "C", the CD-ROM reader will be assigned to letter "D". On my computer, the CD-ROM reader is assigned to drive "H", and this will be used in all examples.
2. The procedure is easiest if you are running in DOS, i.e., not in Windows or any other command shell. With a non-TORQ CD-ROM in your reader, type at the DOS prompt "DIR H:<cr>" (I will use <cr> to indicate a carriage return or enter.) You should see a directory listing of the files on the CD-ROM. If you get an error message, it means that your CD-ROM reader is not correctly installed.
3. Repeat this exercise with the TORQ CD-ROM. After placing the TORQ CD in the reader and typing "DIR H:<cr>", you should see:

```
 Volume in drive H is
 Directory of H:\

torq         <DIR>     11-06-92   4:24p
torqprog     <DIR>     11-06-92   4:35p
        2 file(s)          0 bytes
                           0 bytes free
```

4. Now type "DIR H:\TORQ<cr>". You should see:

```
      Volume in drive H is TORQ01
 Directory of H:\TORQ

.            <DIR>     11-06-92   4:24p
..           <DIR>     11-06-92  11:46p
cd       bin  72172045 08-08-92  12:40p
cq       bin  34447975 08-08-92  12:46p
ct       bin  22594319 08-08-92  12:27p
name     asc      8209 10-10-91   5:34p
sod      bin 146024543 08-08-92  12:23p
symdat   asc   1016065 07-15-92  12:05p
symdat   bin    498960 07-15-92  12:07p
symdat   lst   1519487 07-15-92  12:07p
       10 file(s)  278281603 bytes
                           0 bytes free
```

5. Verify that you can access the TORQ files. The smallest one is NAME.ASC, which contains the symbols, names and CUSIPs of the firms in the database.

   Type "TYPE H:\TORQ\NAME.ASC | MORE<cr>".

The first few lines should look like this:

```
AC 01854810ALLIANCE CAPITAL MANAGEMENT L.P.  UNITS
ACN00511310ACUSON CORPORATION
ACS02490510AMERICAN CAPITAL CONVERTIBLE SECURITIES INC
ADU02342610AMDURA CORPORATION
AL 01371610ALCAN ALUMINIUM LIMITED  HOLDING CO.
ALL01991210ALLSTATE MUNICIPAL INCOME TRUST III   SBI
```

When you have seen enough, you can hit Ctrl-C a few times to get back to the DOS prompt.

At this point, the TORQ files can be accessed by your programs.

Installing the SELECT Programs.

The SELECT programs can be used to construct small, easily-readable subsets of the data. They are not strictly necessary to the successful use of the TORQ data: at this point, the data may be read directly by your programs. Installation of the SELECT programs is highly recommended, however, for two reasons. First, many users of the data may have neither the need nor inclination to do their own programming. The SELECT programs may be sufficient. Second, for those users who will be doing their own programming, the Fortran source code of the SELECT programs will be a useful template in writing their own code.

Installation

0. The SELECT programs are now distributed on the CD-ROM with the data. If you are reinstalling the programs over those distributed with the earlier release of TORQ, skip to step 3.

1. First, get to the root directory of your hard disk (i.e., the "C>" prompt).

2. Create a directory named TORQ using the command:

```
C:\> MKDIR TORQ
```

3. Connect to the TORQ directory:

```
C:\> CD \TORQ
```

4. Copy the programs with the command:

```
C:\TORQ\> XCOPY H:\TORQPROG\*.*
```

5. Modify the TORQFILS.INI file, if necessary. This is a character (ASCII) file that contains the locations of the TORQ files. As shipped, it looks like this:

```
d:\torq\name.asc                   name file
d:\torq\symdat.bin                 symbol/date file
d:\torq\cq.bin                     consolidated quote
d:\torq\ct.bin                     consolidated trade
d:\torq\sod.bin                    system order database
d:\torq\cd.bin                     consolidated audit file
```

If your CD-ROM drive is assigned some letter other than D:, you will need to modify this file. Edit it so that it points to the correct device. For example, if your CD-ROM reader is assigned the letter E:, then the file should be modified to look like:

```
e:\torq\name.asc                   name file
e:\torq\symdat.bin                 symbol/date file
e:\torq\cq.bin                     consolidated quote
e:\torq\ct.bin                     consolidated trade
e:\torq\sod.bin                    system order database
e:\torq\cd.bin                     consolidated audit file
```

(The verbiage to the far right (beginning in column 31) is "comment" information, and is not read by the programs.)

4. Test a SELECT program. Connect to the TORQ subdirectory (type "CD \TORQ<cr>"). Then type "SEL1 <return>". You should see:

```
sel1 v. 1.1 command line:
sel1 AC  AC  901101 901101 TQSA LIST
Input parameters:
Retrieving symbols from AC         to AC
Retrieving dates from      901101 to      901101
Selecting: TQSA      ; Output format is: LIST
retrieving data for AC  . . .
retrieving data for      901101
         15 transactions found.
         30 quotes found.
         19 audit trail records found.
         62 system order database records found.
sel1ct.asc contains any transaction records
sel1cq.asc contains any quote records
sel1cd.asc contains any audit records
sel1sd.asc contains any system order database records
```

The file SEL1CT.ASC can be examined by typing:

"TYPE SEL1CT.ASC | MORE<cr>"

The first few lines should look this:

```
sym  tdate     ttim ex     price    siz cond g127   tseq
AC  901101 10:39:06  N  13.00000    100         0   1587
AC  901101 10:39:36  M  13.00000    100         0      0
AC  901101 10:39:38  M  13.00000    100         0      0
AC  901101 10:39:41  M  13.00000    100         0      0
AC  901101 10:41:38  N  13.00000    300         0   1591
AC  901101 11:52:07  N  13.00000    200         0   1592
AC  901101 13:40:15  N  13.00000    200         0   1595
AC  901101 13:46:27  N  12.87500    800         0   1598
AC  901101 13:55:33  N  13.00000    500         0   1600
AC  901101 13:58:52  N  13.00000    500         0   1601
AC  901101 14:25:07  N  13.00000    500         0   1602
AC  901101 14:57:26  N  13.00000    200         0   1603
AC  901101 15:10:10  N  13.00000    100         0   1605
AC  901101 15:20:50  N  13.00000    200         0   1606
AC  901101 16:02:03  N  13.12500    700         0   1611
```

The SELECT programs are now installed.

3. The TORQ Database: An Overview.

This section summarizes the properties of the TORQ dataset, and should be read by all users of the data. The companion publication Hasbrouck and Sosebee (1992) covers institutional and conceptual details of the trading process. More detailed information on the files is given in section 5 and the appendices of the present paper.

The TORQ database covers 144 firms from November, 1990 through January, 1991. The firms were selected as follows. In the summer of 1990, there was a random draw on the population of listed companies. The draw was stratified by equity capitalization: 15 firms from the lowest equity capitalization decile; 15 from the second lowest, and so on. Data collection on the 150 firms began in September, but because data needs evolved over time, several iterations were necessary before a final set of files and variables could be decided upon. (Due to the difficulty of restoring older archived files, it was easier to pull off data from the daily stream than to go back and reconstruct.) The final data collection began on November 1, 1990, by which time six of the original securities were no longer listed.

The four principal files comprising the TORQ database (CT, CQ, CD and SOD) are close replicas of their counterparts in the NYSE’s internal information system. Certain recoding was necessary to make the data usable, but this has been kept to a minimum. The data are, for the most part, exactly as they appear on the NYSE’s systems.

The four principal files are self-contained in the sense that each record of each file contains a ticker symbol, date and time. However, for purposes of directly accessing particular firms and days, it is useful to have pointers to the data arranged by symbol and date. The name and symbol/date files contain these pointers. These files were not taken from the NYSE’s information systems: they were constructed solely to facilitate use of the TORQ database.

Together the four files give an unprecedented picture of trading activity. Nevertheless, caution should be exercised in linking the files and combining information across the files. The information systems from which the files are drawn are diverse and fragmented. Time stamps in particular are not uniform: there a number of "clocks" throughout the system and they may not be perfectly synchronized. This is an especially important consideration when integrating quote and transaction data. See Hasbrouck and Sosebee (1992).

4. The SELECT Programs.

The TORQ database is large and complicated. The SELECT programs provide a means for users to obtain small, manageable and easily readable subsets of the data. The user specifies the symbols, dates and data desired on the command line that runs the SELECT program. (If these are not supplied, defaults are used.) The program then builds one or more files that contain the selected data. These constructed files are character (ASCII) files that may be printed or loaded into the user’s favorite text editor. (The original TORQ files on the CD-ROM are binary.)

There are two SELECT programs. SEL1 constructs subsets of the four principal TORQ files (CT, CQ, SOD and CD). Two data formats are available. DATA gives the formatted data and nothing else. This option is most useful for importation into other statistical programs. LIST formats the file with variable names. This option is easier for casual viewing. SEL1 produces one file for the transactions, another for the quotes, another for the orders, etc..

The second selection program, SEL2, builds a single ASCII file that contains everything that happened for a given symbol/day in time-sequenced order. SEL2 is useful for getting a overview of trading activity. For casual users, SEL2 may be more useful than SEL1.

Both selection programs expect to find the TORQFILS.INI file in the current directory. All files are built in the current directory.

Running SEL1

At the command prompt, SEL1 is invoked as follows:

```
SEL1 symbol1 symbol2 date1 date2 retcode format
```

The parameters are:

symbol1 first ticker symbol you wish to retrieve (default AC).

symbol2 last ticker symbol (default AC). symbol2 must be greater than or equal to symbol1.

date1 first date you wish to retrieve (yymmdd, default 901101).

date2 last date (default 901101). date2 must be greater than or equal to date1.

retcode Retrieval code. Some combination of the letters T, Q, S and A.
These letters indicate:
T transactions
Q quotes
S SOD
A audit trail
For example, specifying TQSA (the default) will retrieve everything. Specifying TA will retrieve transactions and audit trail info only.

format LIST The data are printed with identifiers. This format is suitable for viewing or printing (default).
DATA The data are printed with one line per observation with no identifiers. This format is useful for importing into statistical packages. The layouts of these files are given in Appendix A.

All of the paramters have defaults. If you wish to specify a parameter, you must specify all preceding parameters. See Section 2 for a demonstration of SEL1.

Running SEL2

SEL2 builds a single file that contains data from the four principal TORQ files. SEL2 is intended to give an overview of what was happening in the market. At the command prompt, it may be invoked by the line:

SEL2 symbol1 symbol2 date1 date2 retcode time1 time2

The first five parameters are the same as for SEL1. The remaining ones are:

time1 First time to retrieve (hhmmss past midnight).
Default: 000000.

time2 Last time to retrieve. Default: 23:59:00. time2 must be greater than or equal to time1.

The time defaults should not require modification unless you are requesting a particularly voluminous amount of information. (The program will print a message if it is swamped.) The default for the retrieval code is TQ (trades and quotes). Start with these, adding audit and SOD data as you become more familiar with what’s going on.

When you run SEL2 without any parameters (i.e., type "SEL2<cr>") you should see something like this:

```
sel2 v. 1.1 command line:
sel2 AC  AC  901101 901101 TQ   000000 235900
```

```
Input parameters:
Retrieving symbols from AC         to AC
Retrieving dates from      901101 to      901101
Selecting: TQ
Retrieving times from 000000 to 235900 (hhmmss past midnight)
retrieving data for AC  . . .
retrieving data for      901101
         15 transactions
         30 quotes
         45 events found
output data written to file sel2.asc.
```

As indicated, the data are written to the file SEL2.ASC. The first part of this file is given overleaf. (Some sequencing information has been stripped to fit on the page.) The quotes and trades are listed in the correct time sequence. The first item is a quote posted at the New York Stock Exchange ("N"). The integers printed after the bid and offer prices are the sizes of these quotes (the number of 100-share round lots for which the quotes are good). Quotes from the Midwest (M), Boston (B) and Philadelphia Exchanges (X) follow. The first transaction was at 10:39:06 (100 shares on the NYSE at $13/share).

```
quote at  9:30:44. cq record is:
AC  901101  9:30:44 N bid  12.87500  32 ofr  13.12500   5

quote at  9:30:47. cq record is:
AC  901101  9:30:47 M bid  12.75000   1 ofr  13.25000   1

quote at  9:30:51. cq record is:
AC  901101  9:30:51 B bid  12.75000   1 ofr  13.25000   1

quote at  9:30:52. cq record is:
AC  901101  9:30:52 X bid  12.75000   1 ofr  13.25000   1

transaction print at 10:39:06. ct record is:
AC  901101 10:39:06 N   13.00000    100 cond:  g127:  0

transaction print at 10:39:36. ct record is:
AC  901101 10:39:36 M   13.00000    100 cond:  g127:  0

transaction print at 10:39:38. ct record is:
AC  901101 10:39:38 M   13.00000    100 cond:  g127:  0

transaction print at 10:39:41. ct record is:
AC  901101 10:39:41 M   13.00000    100 cond:  g127:  0

quote at 10:40:13. cq record is:
AC  901101 10:40:13 M bid  12.75000   2 ofr  13.12500   2

transaction print at 10:41:38. ct record is:
AC  901101 10:41:38 N   13.00000    300 cond:  g127:  0

transaction print at 11:52:07. ct record is:
AC  901101 11:52:07 N   13.00000    200 cond:  g127:  0

quote at 13:36:15. cq record is:
AC  901101 13:36:15 N bid  12.87500  32 ofr  13.00000  10

quote at 13:36:17. cq record is:
AC  901101 13:36:17 B bid  12.75000   1 ofr  13.12500   1

quote at 13:36:21. cq record is:
AC  901101 13:36:21 X bid  12.75000   1 ofr  13.12500   1

quote at 13:40:03. cq record is:
AC  901101 13:40:03 N bid  12.87500  32 ofr  13.00000   8

transaction print at 13:40:15. ct record is:
AC  901101 13:40:15 N   13.00000    200 cond:  g127:  0
```

SELECT Usage Notes.

1. The SELECT programs read the TORQFILS.INI file to find out where the main data files are located. Make certain that the TORQFILS.INI file is in the current directory and that it is pointing to the right device

letter.

2. SEL2 uses workspace in the current directory to build scratch datasets, which it then merges. If you are running out of space, moderate your data request.

3. CD-ROM drives can be slow. If you have the disk space to spare, consider copying some of the TORQ files onto your hard disk. If you do this, you should change TORQFILS.INI so that it points to your hard disk.

5. File Descriptions.

This section provides detailed documentation on the TORQ files. Record layouts are given in Appendix A. A variable glossary is given in Appendix B. All files are sorted by ticker symbol, date and time.

5.1 The CT File (CT.BIN)

The consolidated transaction file contains one record per transaction. The transaction time (TTIM) reflects the time at which the information entered the consolidated transaction system. See Hasbrouck and Sosebee (1992) for a description of the reporting process and issues surrounding the time stamping.

5.2 The CQ File (CQ.BIN)

The consolidated quote file contains one record per quote change. A quote remains in effect until it is changed. The quote time (QTIM) reflects the time at which the information entered the consolidated quote reporting system.

5.3 The SOD File (SOD.BIN).

The system order database is a complicated database that documents the entry and processing of an order. It covers the SuperDOT system, OARS (the Opening Automated Report System) and ITS (the Intermarket Trading System). A preliminary analysis of these data is given in Harris and Hasbrouck (1992).

On a given day, the SOD file for that date (TDATE) will have at least one record for every order for which "something" was done (e.g., order entry, execution, cancellation, etc.). The SOD file on any given day does not contain all pending orders. For example, if there is a limit order that was submitted far away from the market, it may remain in the system unexecuted, uncanceled and perhaps forgotten. There are no records of these orders. Accordingly, the SOD file cannot be used to build up a full picture of the limit order book at a point in time.

A simple order.

In the simplest of cases, all the information about an order will be contained in one record. The ODATE field gives the date on which the order was entered. Usually OTIME gives the time. However, if the order was carried over from a previous day (e.g., a pending limit order) OTIME is set to zero. To determine if the order is a buy or sell, or if it was tick sensitive, examine the OSIDE field. OCOND reflects other order conditions, including whether the order was a market or limit order. If a limit order, the limit price is in LMTPR. The size of the order is in OSHRS. The time-in-force (day order or good-til-canceled) is given in TIF.

If the order was executed, in part or in toto, the execution price will be in EXECPR and the time at which the report was generated will be in RTIME. The report time is sometimes a useful indication of when the transaction really took place. It generally precedes the print time of the transaction (TTIM on the CT file). The quantities that went to each of up to four counterparties are in CONQTY1-4 and partial descriptions of the contraparties are in CONTRA1-4. If the order is canceled, CXLQTY is nonzero.

Stop Orders.

A stop order (not to be confused with a "stopped order") is one which is triggered when the last trade price hits a certain level. A stop sell order, for example, is triggered when the last trade price is at or below the stop price (the STPPR field). Stop orders may be either market or limit.

Stopped Orders.

A stopped order (not to be confused with a "stop order" or with a "trading halt") is an order for which a price is guaranteed, but a possibility of betterment remains. For example, the quotes are 20 bid to 20 1/4 offered. A market order to buy arrives. The specialist may stop the order at 20 1/4. This means that the buyer is guaranteed a fill at 20 1/4, but might do better. For example, if a market sell order arrives while the buy is stopped, the specialist will generally cross the orders at 20 1/8. If an order is stopped, this will be noted by a ’3’ or ’4’ in the ATYPE field. The time will be in ATIME; APRICE will contain the price at which the order was stopped. As with all executions, EXECPR will contain the actual price at which the order was reported.

Multiple record sets (IND and INDTOT).

Often, not all of the information relevant to an order will fit in a single record. A common occurrence, for example, is that an order is submitted, and executed in part over the course of the trading session. The single order thus gives rise to multiple executions and multiple reports. When this happens, the records will be arranged in consecutive sequence, with INDTOT giving the total number of records in the set, and IND indicating the relative position in the set. The multiple record set so described is contained on a single day: the set does not include records arising when actions were taken on other days. In extreme cases, an order may have multiple sets on multiple days.

Other useful information.

To see whether the order is part of a program or index arbitrage trade, check ACCTYP.

BOOK and FLRIND describe the order routing. FLRIND reflects the point of entry for the order. A "0" is by far the most common flag, indicating that the order came through the Common Message Switch (CMS). A "2" indicates that it was entered by the specialist’s clerk on the display book. A "1" indicates that the order was entered by one of the specialist’s staff on a Texas Instrument Personal Computer (TIPC, "tipsy").

BOOK indicates the path taken by the order. A "0" indicates that it went from CMS to the specialist’s post. A "1" indicates that it went from CMS to the broker’s booth, and the broker handled it manually. A "2" indicates that it went to the booth, but at the discretion of the booth personnel, it was automatically forwarded to the post.

Example

The following SOD record is taken from the file SEL1SD.ASC as constructed by running SEL1 with the default parameters (section 2). A market sell order for 200 shares was received at 11:23. The specialist stopped it at 11:24 at a price of 12 7/8. The execution took place at 11:51 at a price of 13. The buyer was a system limit order.

```
symbol  tdate  odate    otime order ind indtot flrind book omni
   AC  901101 901101 11:23:45     2   1      1      0    0    1
oside ocode ocond oshrs     lmtpr     stppr acctyp tif oqual
  SEL    DM     0   200    .00000    .00000      I   1    0
atype    atime    aprice cxlqty    rtime rshrs    execpr lvsqty clrcode
    3 11:24:27  12.87500      0 11:51:53   200  13.00000      0
 conqty1 contra1 conqty2 contra2 conqty3 contra3 conqty4 contra4
     200    LMT        0               0               0
```

5.4 The Audit File (CD.BIN)

The audit file is in a sense an expanded version of the transaction file. It provides information on the number and type of parties to a trade.

Example

The following record is taken from SEL1CD.ASC and corresponds to the SOD order discussed in the last section. The buyer was a system limit order ("L1"). Both buyer and seller were individual investors. Note that the transaction print time is 14 seconds after the report time.

```
symbol  tdate     ttim     price    siz  seqnum subseq cond corr ex origsiz
   AC  901101 11:52:07  13.00000    200    1592      1        0   N     200
btype buyacct   buytim buycomp selcomp   seltim selacct stype
   L1       I 11:51:53     200     200 11:51:53       I    D1
```

5.5 The Name File (NAME.ASC)

The name file is an ASCII file containing 144 records, one for each security, with ticker symbols, firm names and CUSIPs. The name file is not a standard NYSE file. It was constructed solely to accompany the TORQ data set.

5.6 The Symbol/Date File (SYMDAT.BIN)

The Symbol/Date Map file contains one record for each firm for each of the 63 dates covered in the sample. Each record contains pointers to the CT, CQ, SOD and CD files. These pointers indicate where the data start and stop for that firm/date. An ASCII version of the file is given in SYMDAT.ASC. A formatted listing of the file is in SYMDAT.LST. Like the name file, the symbol/date map file is not a standard NYSE file.

6. Tips and Suggestions.

Linking the Files.

Users desiring to test models that relate orders, transactions and quotes will almost certainly want to combine information from the four principal TORQ files. Due to the limitations of the data, this is not an endeavor that should be undertaken casually. There are no definitive pointers, for example, that link an order in the SOD file with the resultant transaction print in the CT file. Instead, the connections must be built by the user. Here are some guidelines.

1. In linking SOD records across days (necessary when submission, and/or cancelation span multiple days), the following should be noted. Orders carried over from previous days generally have an order time (OTIME) of zero. They can be matched to the original orders by using SYMBOL, ODATE, OCOND, OQUAL, OSIDE and TIF.
2. To link the transaction file with the audit file, match on SYMBOL, date and the transaction sequence number for NYSE trades.
3. To link a SOD record with the audit file, match first on SYMBOL, date and times. The report time (RTIME) on the SOD record appears to match the BUYTIM or SELTIM on the audit record. If multiple SOD records and audit records have the same times, try matching on volume. The CONQTY fields in the SOD file can be matched to the BUYCOMP and SELCOMP fields in the audit file. This problem arises at the open.
4. In mapping the SOD file to the transaction file, it is easier to map from the SOD file to the audit file, and from the audit file to the transaction file (as described above).

Reading the Files Directly in Your Own Programs.

The data files may be read directly by Fortran, SAS and many other packages. Each record on each file is self-contained. That is, each record contains a ticker symbol, date and time. There is no need to pick these things up from auxiliary files. (The exceptions are the full company names and CUSIPs: these are only present in the name file.) It is therefore quite possible to perform an analysis by reading a single file through from beginning to end. (Were you interested solely in spreads, for example, the CQ file would suffice.)

It is often useful, however, to read the files using direct access. The name and symbol/date files may be useful in this process. The second record in the name file, for example, contains the ticker symbol, CUSIP and name for the second company (Acuson). Each symbol has 63 records in the symbol/date file (one record for each date). Acuson’s records in the symbol/date file are records 64 through 126. Each of these records contains pointers to the first and last observations in the CT, CQ, SOD and audit files. Both SEL1 and SEL2 perform direct access on the files. If you wish to implement this in your own code, you might use these programs as templates. (See appendix D.)

Appendix A: File Formats

There are two main groups of files. The original TORQ files are those which reside on the CD-ROM. The SELECT ("SEL1") files are those which are built by the selection program SEL1 operating under DATA format. Variable descriptions here are abbreviated. Fuller explanations are given in the variable glossary (Appendix B).

The original TORQ files

These are all fixed length records with no record separators (like carriage returns or line feeds). Variables are stored as either ASCII characters, 2-byte integer values or 4-byte integer values. (Integers are stored with the low-order byte first. This is standard for most MS-DOS software packages. IBM mainframes and Sun workstations, however, store the high-order byte first. If you are exporting the binary data to one of these systems, you will have to swap the bytes of all integer variables.) There are no real (floating point) variables. Prices are stored as 4-byte integers in units of $1/256. Dates are stored as 4-byte integers, YYMMDD. Times are 4-byte integers, seconds past midnight. NULL fields are fields which are blank or zeros in the files.

Name File (NAME.ASC)					
Variable	Type	Length	Start	Stop	Description
SYMBOL	C	3	1	3	Symbol
CUSIP	C	8	4	11	CUSIP
NAME	C	44	12	55	Firm name

Symbol/Date Map File (SYMDAT.BIN)					
Variable	Type	Length	Start	Stop	Description
SYMBOL	C	3	1	3	Symbol
TDATE	I	4	4	7	Date
CT1	I	4	8	11	-> first CT record
CT2	I	4	12	15	-> last CT record
NCT	I	4	16	19	Number of CT records
CQ1	I	4	20	23	-> first CQ record
CQ2	I	4	24	27	-> last CQ record
NCQ	I	4	28	31	Number of CQ records
CD1	I	4	32	35	-> first CD record
CD2	I	4	36	39	-> last CD record
NCD	I	4	40	43	Number of CD records
SD1	I	4	44	47	-> first SOD record
SD2	I	4	48	51	-> last SOD record
NSD	I	4	52	55	Number of SOD records

C: Character; I: Integer; NULLxx: blank; all positions are bytes.

Consolidated Transaction (CT) File (CT.BIN)					
Variable	Type	Length	Start	Stop	Description
COND	C	1	1	1	Condition code
EX	C	1	2	2	Exchange
G127	I	2	3	4	See glossary
PRICE	I	4	5	8	Transaction price
SIZ	I	4	9	12	Transaction volume
SYMBOL	C	3	13	15	Ticker symbol
TDATE	I	4	16	19	Trade date
TSEQ	I	4	20	23	Transaction sequence number
TTIM	I	4	24	27	Transaction time
NULL01	I	4	28	31	

Consolidated Quote (CQ) File (CQ.BIN)					
Variable	Type	Length	Start	Stop	Description
SYMBOL	C	3	1	3	Ticker symbol
TDATE	I	4	4	7	Quote date
QTIM	I	4	8	11	Quote time
BID	I	4	12	15	Bid price
OFR	I	4	16	19	Offer (ask) price
BIDSIZ	I	2	20	21	Bid size
OFRSIZ	I	2	22	23	Offer size
MODE	I	2	24	25	See glossary
QSEQ	I	4	26	29	Quote sequence number
EX	C	1	30	30	Exchange
NULL02	C	1	31	31	

C: Character; I: Integer; NULLxx: blank; all positions are bytes.

System Order Database (SOD) File (SOD.BIN)					
Variable	Type	Length	Start	Stop	Description
ACCTYP	C	1	1	1	Account type
APRICE	I	4	2	5	Admin price
ATIME	I	4	6	9	Admin time
ATYPE	C	1	10	10	Admin type
BOOK	C	1	11	11	See glossary
CLRCODE	C	2	12	13	Clearance code
NULL06	I	2	14	15	
CONQTY1	I	4	16	19	Contra quantity 1
CONQTY2	I	4	20	23	Contra quantity 2
CONQTY3	I	4	24	27	Contra quantity 3
CONQTY4	I	4	28	31	Contra quantity 4
CONTRA1	C	4	32	35	Contra 1
CONTRA2	C	4	36	39	Contra 2
CONTRA3	C	4	40	43	Contra 3
CONTRA4	C	4	44	47	Contra 4
CXLQTY	I	4	48	51	Cancel quantity
NULL10	I	2	52	53	
EXECPR	I	4	54	57	Execution price
NULL04	C	4	58	61	
FLRIND	C	1	62	62	See glossary
IND	I	2	63	64	This record # in set
INDTOT	I	2	65	66	Total # records in set
LMTPR	I	4	67	70	Limit price
LVSQTY	I	4	71	74	Leaves quantity
OCODE	C	2	75	76	Order code
OCOND	C	1	77	77	Order condition
ODATE	I	4	78	81	Order date
OMNI	C	1	82	82	Omnibus code
OQUAL	I	2	83	84	Order qualification

C: Character; I: Integer; NULLxx: blank; all positions are bytes.

System Order Database (SOD) File (SOD.BIN)					
Variable	Type	Length	Start	Stop	Description
ORDER	C	1	85	85	Valid order flag
OSHRS	I	4	86	89	Number of shares in order
OSIDE	C	3	90	92	Order side
OTIME	I	4	93	96	Order time
NULL05	C	1	97	97	
RSHRS	I	4	98	101	Number of shares executed
RTIME	I	4	102	105	Time of execution (report)
STPPR	I	4	106	109	Stop price
SYMBOL	C	3	110	112	Ticker symbol
NULL07	I	2	113	114	
TDATE	I	4	115	118	SOD file date
TIF	C	1	119	119	Time in force

Audit (CD) File (CD.BIN)					
Variable	Type	Length	Start	Stop	Description
BTYPE	C	2	1	2	Buyer’s type
BUYACCT	C	1	3	3	Buyer’s account type
NULL08	I	2	4	5	
BUYCOMP	I	4	6	9	Buyer’s submitted quantity
BUYTIM	I	4	10	13	Buyer’s submitted time
COND	C	1	14	14	Condition
CORR	C	2	15	16	Correction
EX	C	1	17	17	Exchange
ORIGSIZ	I	4	18	21	Original size
PRICE	I	4	22	25	Price
SELACCT	C	1	26	26	Seller’s account type
NULL09	I	2	27	28	
SELCOMP	I	4	29	32	Seller’s submitted quantity

C: Character; I: Integer; NULLxx: blank; all positions are bytes.

Audit (CD) File (CD.BIN)					
Variable	Type	Length	Start	Stop	Description
SELTIM	I	4	33	36	Seller’s submitted time
SEQNUM	I	4	37	40	Sequence number
SIZ	I	4	41	44	Size
STYPE	C	2	45	46	Seller’s type
SUBSEQ	I	2	47	48	Subsequence
SYMBOL	C	3	49	51	Symbol
TDATE	I	4	52	55	Date
TTIM	I	4	56	59	Transaction time

C: Character; I: Integer; NULLxx: blank; all positions are bytes.

The SEL1 DATA files.

These files are produced by the SEL1 SELECT program under DATA format. The files are ASCII; a record is terminated with carriage return/linefeed. The "Types" given below are character (C), integer (I) or floating (F). (Floating point numbers are written with a decimal point.) In the SEL1 files, prices are given in dollars with five places after the decimal point (not in $1/256 as in the original TORQ files). Times are given as seconds past midnight.

SEL1CT.ASC File					
Variable	Type	Len	Start	Stop	Description
SYMBOL	C	3	1	3	Ticker symbol
TDATE	I	6	4	9	Trade date
TTIM	I	6	10	15	Trade time
EX	C	1	16	16	Exchange
PRICE	F	10	17	26	Trade price
SIZ	I	7	27	33	Trade volume
COND	C	1	34	34	Condition code
G127	I	3	35	37	See glossary
TSEQ	I	6	38	43	Trade sequence number

SEL1CQ.ASC File					
Variable	Type	Len	Start	Stop	Description
SYMBOL	C	3	1	3	Ticker symbol
TDATE	I	6	4	9	Quote date
QTIM	I	6	10	15	Quote time
EX	C	1	16	16	Exchange
BID	F	10	17	26	Bid price
OFR	F	10	27	36	Offer (ask) price
BIDSIZ	I	6	37	42	Bid size
OFRSIZ	I	6	43	48	Offer size
MODE	I	5	49	53	See glossary
QSEQ	I	7	54	60	Quote sequence number

C: Character; I: Integer; F floating point; positions are bytes.

SEL1CQ.ASC File

Variable	Type	Len	Start	Stop	Description

SEL1SD.ASC File

Variable	Type	Len	Start	Stop	Description
SYMBOL	C	3	1	3	Ticker symbol
TDATE	I	6	4	9	SOD file date
ODATE	I	6	10	15	Order date
OTIME	I	6	16	21	Order time
ORDER	C	1	22	22	Valid order flag
IND	I	4	23	26	This record # in set
INDTOT	I	4	27	30	Total # records in set
FLRIND	C	1	31	31	See glossary
BOOK	C	1	32	32	See glossary
OMNI	C	1	33	33	Omnibus code
OSIDE	C	3	34	36	Order side
OCODE	C	2	37	38	Order code
OCOND	C	1	39	39	Order condition
OSHRS	I	6	40	45	Number of shares in order
LMTPR	F	10	46	55	Limit price
STPPR	F	10	56	65	Stop price
ACCTYP	C	1	66	66	Account type
TIF	C	1	67	67	Time in force
OQUAL	I	4	68	71	Order qualification
ATYPE	C	1	72	72	Admin type
ATIME	I	6	73	78	Admin time
APRICE	F	10	79	88	Admin price
CXLQTY	I	6	89	94	Cancel quantity
RTIME	I	6	95	100	Time of execution (report)
RSHRS	I	6	101	106	Number of shares executed
CLRCODE	C	2	107	108	Clearance code
EXECPR	F	10	109	118	Execution price
LVSQTY	I	6	119	124	Leaves quantity

C: Character; I: Integer; F floating point; positions are bytes.

SEL1CQ.ASC File					
Variable	Type	Len	Start	Stop	Description
CONQTY1	I	6	125	130	Contra quantity 1
CONTRA1	C	4	131	134	Contra 1
CONQTY2	I	6	135	140	Contra quantity 2
CONTRA2	C	4	141	144	Contra 2
CONQTY3	I	6	145	150	Contra quantity 3
CONTRA3	C	4	151	154	Contra 3
CONQTY4	I	6	155	160	Contra quantity 4
CONTRA4	C	4	161	164	Contra 4

SEL1CD.ASC File					
Variable	Type	Len	Start	Stop	Description
SYMBOL	C	3	1	3	Symbol
TDATE	I	6	4	9	Date
TTIM	I	6	10	15	Transaction time
PRICE	F	10	16	25	Price
SIZ	I	6	26	31	Size
SEQNUM	I	6	32	37	Sequence number
SUBSEQ	I	5	38	42	Subsequence
COND	C	1	43	43	Condition
CORR	C	2	44	45	Correction
EXCD	C	1	46	46	Exchange
ORIGSIZ	I	6	47	52	Original size
BTYPE	C	2	53	54	Buyer’s type
BUYACCT	C	1	55	55	Buyer’s account type
BUYTIM	I	6	56	61	Buyer’s submitted time
BUYCOMP	I	6	62	67	Buyer’s submitted quantity
SELCOMP	I	6	68	73	Seller’s submitted quantity
SELTIM	I	6	74	79	Seller’s submitted time
SELACCT	C	1	80	80	Seller’s account type
STYPE	C	2	81	82	Seller’s type

C: Character; I: Integer; F floating point; positions are bytes.

Appendix B: Variable Glossary.

The following table describes the variables found in the TORQ files. For some variables, the list of permissible values is incomplete: this documentation only attempts to describe the most common or relevant values. The positions of the variables in the files are given in Appendix A.

Variable Name	Description
ACCTYP	Account type. Prog Index Indiv. Proprietary Code Trade Arb. Invest. /Agency A No No No Agency C Yes No No Proprietary D Yes Yes No Proprietary I No No Yes J Yes Yes Yes K Yes No Yes P No No No Proprietary U Yes Yes No Agency Y Yes No No Agency Note: "Proprietary" refers to a member's own trades. "Agency" trades are those in which the member is representing another party (possible another member). [Character, 1 byte; SOD.]
APRICE	Price associated with an admin (administrative message), in $1/256. See ATYPE. [Integer, 4 bytes; SOD.]
ATIME	Time of admin (administrative message) response associated with order (seconds past midnight). Zero if there is no admin.
ATYPE	Admin (administrative message) type code. 0 No admin 1 Cancel acknowledged 2 Too late to cancel 3 Stopped stock (see APRICE) 4 Stopped at opening (no price) 5 Price correction (see APRICE) 6 Nothing done 7 Price is correct (see APRICE) 8 Limit is out of range 9 Confirm delete [Character, 1 byte; SOD.]
BCOMPQTY	Number of shares compared for buyer. Note: "compared" means that buyer and seller agreed on the terms of the trade. [Integer, 4 bytes; CD.]

Variable Name	Description
BID	Bid price in $1/256. [Integer, 4 bytes; CQ.]
BIDSIZ	Bid size (number of shares bid for at BID) in 100 share lots. (NB: this does not necessarily represent the full amount bid for. More stock may be in the limit order book at this price, or the specialist may be willing to buy more at this price.) [Integer, 2 bytes; CQ.]
BOOK	Order routing method. 0 Post routing. 1 booth routing. 2 booked. [Character, 1 byte; SOD.]
BTYPE	Buyer’s type. D1 DOT trade, system side D2 DOT trade, crowd side, compared D3 DOT trade, crowd side, uncompared I1 ITS trade, NYSE executing market I2 ITS trade, NYSE committing market L1 LMT trade, system side L2 LMT trade, crowd side, compared L3 LMT trade, crowd side, uncompared O1 OPN trade, system side O2 OARS trade, crowd side blank indicates not available. [Character, 2 bytes; CD.]
BUYACCT	Buyer’s account type. See ACCTYP. [Character, 1 byte; CD.]
BUYTIM	Trade time submitted by buyer (in seconds past midnight). [Integer, 4 bytes; CD.]
CD1, CD2	Starting and ending record pointers for symbol/date in CD file. Zero if no audit records on date for symbol. [Integer, 4 bytes; SYMDAT.]
CLRCODE	Clearance code. ’ ’ regular way ’C ’ cash ’ND’ next day ’nn’ number of seller’s days [Character, 2 bytes; SOD.]
COND	Sale condition. blank regular way (5-day settlement) O opening trade, out of sequence C cash sale N next day settlement A acquisition D distribution R seller’s option Z sold sale (reported out of sequence) L sold last (reported late, but in sequence) For the CD file only: * The audit algorithm determined that a trade took place at the indicated time (TTIM), but there was no corresponding tape print.

Variable Name	Description
	1 same as ’*’ for a cash sale. 2 same as ’*’ for a seller’s option trade. 3 same as ’*’ for a next day trade [Character, 1 byte; CT, CD.]
CONQTY1-CONQTY4	Contra quantity (shares). These fields indicate the number of shares on the other side of the transaction for up to four counterparties. The price is EXECPR. (If there is more than one execution price, there will be multiple SOD records.) [4 fields, each of which is a 4-byte integer; SOD.]
CONTRA1-CONTRA4	Contra mnemonic. This gives information about the counterparties. DOT DOT system LMT LMT system OPN OARS (Open Automated Report System) OARS OARS ITSx (where x is an exchange code, A, B, C, M, P, N, T or X [see EX]). This indicates that the contra side of the trade was a commitment conveyed on the ITS. Note: An executed OARS order is always assigned the contra OPN. A non-OARS order that executes at the open is assigned the contra OARS (e.g., a limit order left over from the previous day). [4 fields, each of which is a 4-byte character; SOD.]
CORR	Correction code. ’0 ’ original trade, not corrected ’1 ’ original trade, corrected ’7 ’ original trade, entered in error, canceled. ’8 ’ original trade, canceled. [Character, 2 bytes; CD.]
CQ1, CQ2	Starting and ending record pointers for symbol/date in CQ file. Zero if no quotes on date for symbol. [Integer, 4 bytes; SYMDAT.]
CT1, CT2	Starting and ending record pointers for symbol/date in CT file. Zero if no transactions on date for symbol. [Integer, 4 bytes; SYMDAT.]
CUSIP	CUSIP number. (NB: these were collected by hand and have not been independently verified.) [Character, 8 bytes; NM.]
CXLQTY	Quantity canceled (shares). [Integer, 4 bytes; SOD.]
EX	Exchange. B Boston C Cincinnati I ITS control center M Midwest N NYSE O Instinet P Pacific T NASD X Philadelphia [Character, 1 byte; CT, CQ, CD.]

Variable Name	Description
EXECPR	Execution price, in $1/256. [Integer, 4 bytes; SOD.]
FLRIND	Point of entry indicator for orders. 0 entered via CMS 1 entered from a TI/PC 2 entered from a display book [Character, 1 byte; SOD.]
G127	NYSE trades only. G127 is interpreted as a 3-digit integer. If the tens digit is 4, 6 or 7, the trade was reported to the CTS from a Display Book. If the hundreds digit is 1 or 3, the trade is a "Rule 127" trade (a block cross outside the quotes). [Integer, 2 bytes; CT.]
IND	Record counter within a SOD set. See INDTOT. [Integer, 2 bytes; SOD.]
INDTOT	Total number of records comprising a SOD set. See IND. [Integer, 2 bytes; SOD.]
LMTPR	Limit price, in $1/256. [Integer, 4 bytes; SOD.]
LVSQTY	Leaves quantity in shares. This gives the number of shares left in the order. I.e., it is OSHRS less fills and cancellations. [Integer, 4 bytes; SOD.]
MODE	Quote mode. 1 depth on bid side 2 depth on offer side 3 closing. (NB: the sizes on the closing quotes are not meaningful.) 4 News dissemination 5 Fast trading 6 depth, bid and offer 7 order imbalance 8 closed market maker (NASD) 9 nonfirm quote 10 opening quote 11 news pending 12 regular (NASD open) 13 trading halt due to related security 14 trading halt in view of common 15 order influx 16 no open/no resume [Integer, 2 bytes; CQ.]
NAME	Security name. [Character, 44 bytes; NM.]
NCD	Number of audit records in CD file for symbol/date. [Integer, 4 bytes; SYMDAT.]
NCQ	Number of quotes in CQ file for symbol/date. [Integer, 4 bytes; SYMDAT.]
NCT	Number of transactions in CT file for symbol/date. [Integer, 4 bytes; SYMDAT.]

<table>
<tr><th>Variable Name</th><th>Description</th></tr>
<tr><td>NSD</td><td>Number of records in SOD file for symbol/date. [Integer, 4 btyes; SYMDAT.]</td></tr>
<tr><td>OCODE</td><td>Order code.
First character:
D,R Day
G,N Good-till-canceled (GTC)
X,C Day replaces CXL
Y,K GTC replaces CXL
(Note: "replaces CXL" means that the order replaces a previous order that was canceled.)
Second character:
B Nonstandard order
M Market
L Limit
S Stop
P Stop limit
Special combinations:
Q simple cancel
[Character, 2 bytes; SOD.]</td></tr>
<tr><td>OCOND</td><td>Order condition.
0 market
1 limit
2 stop
3 stop limit
4 market at close
9 limit or better
[Character, 1 byte; SOD.]</td></tr>
<tr><td>ODATE</td><td>Order date, YYMMDD.
[Integer, 4 bytes; SOD.]</td></tr>
<tr><td>OFR</td><td>Offer (ask) price in $1/256.
[Integer, 4 bytes; CQ.]</td></tr>
<tr><td>OFRSIZ</td><td>Offer size (number of shares offered at OFR) in 100-share lots. (NB: this does not necessarily represent the full amount offered. More stock may be in the limit order book at this price, or the specialist may be willing to sell more at this price.)
[Integer, 2 bytes; CQ.]</td></tr>
<tr><td>OMNI</td><td>Omnibus type code. If "6", the trade was "busted," (i.e., annulled).
[Character, 1 byte; SOD.]</td></tr>
<tr><td>OQUAL</td><td>Order qualifications as requested by the member firm.
<table>
<tr><th>Value</th><th>All or None</th><th>Try to Stop</th><th>Not Held</th><th>Do Not Incr.</th><th>Do Not Reduce</th></tr>
<tr><td>0</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>1</td><td>x</td><td></td><td></td><td></td><td></td></tr>
<tr><td>2</td><td></td><td></td><td>x</td><td></td><td></td></tr>
<tr><td>3</td><td></td><td>x</td><td></td><td></td><td></td></tr>
<tr><td>4</td><td></td><td></td><td></td><td>x</td><td></td></tr>
<tr><td>5</td><td></td><td></td><td></td><td></td><td>x</td></tr>
<tr><td>6</td><td>x</td><td></td><td>x</td><td></td><td></td></tr>
<tr><td>7</td><td>x</td><td>x</td><td></td><td></td><td></td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Variable Name</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>8</td><td>x</td><td></td><td></td><td>x</td><td></td></tr>
<tr><td>9</td><td>x</td><td></td><td></td><td></td><td>x</td></tr>
<tr><td>10</td><td></td><td>x</td><td>x</td><td></td><td></td></tr>
<tr><td>11</td><td></td><td></td><td>x</td><td>x</td><td></td></tr>
<tr><td>12</td><td></td><td></td><td>x</td><td></td><td>x</td></tr>
<tr><td>13</td><td></td><td>x</td><td></td><td>x</td><td></td></tr>
<tr><td>14</td><td></td><td>x</td><td></td><td></td><td>x</td></tr>
<tr><td>15</td><td>x</td><td>x</td><td>x</td><td></td><td></td></tr>
<tr><td>16</td><td>x</td><td></td><td>x</td><td>x</td><td></td></tr>
<tr><td>17</td><td>x</td><td></td><td>x</td><td></td><td>x</td></tr>
<tr><td>18</td><td></td><td>x</td><td></td><td>x</td><td></td></tr>
<tr><td>19</td><td></td><td>x</td><td></td><td></td><td>x</td></tr>
<tr><td>20</td><td>x</td><td>x</td><td></td><td>x</td><td></td></tr>
<tr><td>21</td><td>x</td><td>x</td><td></td><td></td><td>x</td></tr>
<tr><td>22</td><td>x</td><td>x</td><td>x</td><td>x</td><td></td></tr>
<tr><td>23</td><td>x</td><td>x</td><td>x</td><td></td><td>x</td></tr>
<tr><td>99</td><td>Other</td><td></td><td></td><td></td><td></td></tr>
</table>
[Integer, 2 bytes; SOD.]</td></tr>
<tr><td>ORDER</td><td>Order indicator. If "0", the record represents a copy of the original order. (This often occurs when an order is executed or canceled on a day after its original submission.) If this field is not equal to "0", the record represents the intial submission of an order.
[Character, 1 byte; SOD.]</td></tr>
<tr><td>ORIGSIZ</td><td>Original CTS trade size in shares. (See SIZ.)
[Integer, 4 bytes; CD.]</td></tr>
<tr><td>OSHRS</td><td>Order size, in shares.
[Integer, 4 bytes; SOD.]</td></tr>
<tr><td>OSIDE</td><td>Order side.
BUY buy
BY- buy-minus
SEL sell
SL+ sell-plus
SST sell short
SSX sell short exempt (from uptick rule)
CXL cancel
[Character, 3 bytes; SOD]</td></tr>
<tr><td>OTIME</td><td>Order time (seconds past midnight).
[Integer, 4 bytes; SOD.]</td></tr>
<tr><td>PRICE</td><td>Trade price, in $1/256.
[Integer, 4 bytes; CT, CD.]</td></tr>
<tr><td>QSEQ</td><td>Quote sequence number. (Applies only to NYSE quotes.)
[Integer, 4 bytes; CQ.]</td></tr>
<tr><td>QTIM</td><td>Quote time (seconds past midnight.) [Integer, 4 bytes; CQ.]</td></tr>
<tr><td>RTIME</td><td>Report time (seconds past midnight).
[Integer, 4 bytes; SOD.]</td></tr>
<tr><td>SCOMPQTY</td><td>Number of shares compared for seller. Note: "compared" means that buyer and seller agreed on the terms of the</td></tr>
</table>

Variable Name	Description
	trade. [Integer, 4 bytes; CD.]
SD1, SD2	Starting and ending record pointers for symbol/date in SOD file. Zero if no SOD records on date for symbol. [Integer, 4 bytes; SYMDAT.]
SELACCT	Seller’s account type. See ACCTYP. [Character, 1 byte; CD.]
SELTIM	Trade time submitted by seller (seconds past midnight). [Integer, 4 bytes; CD.]
SEQNUM	For the NYSE, this is the MDS sequence number. For other exchanges, this is the CTS sequence number. (Cf. TSEQ.) A zero in this field indicates an unmatched print. [Integer, 4 bytes; CD.]
SIZ	CTS trade size in shares. (In the CD file, this may be adjusted by the audit algorithm.) [Integer, 4 bytes; CT, CD.]
STPPR	Stop price for stop and stop limit orders, in $1/256. [Intger, 4 bytes; SOD.]
STYPE	Seller’s type. See codes for BTYPE. [Character, 2 bytes; CD.]
SUBSEQ	Subsequence number. In the CD file, transactions with multiple buyers and/or sellers have multiple records. The first record in the set has SUBSEQ=1; the second has SUBSEQ=2, etc. Unmatched comparison records contain a zero. Unmatched CTS records contain a one. [Integer, 2 bytes; CD.]
SYMBOL	Ticker symbol. [Character, 3 bytes; CT, CQ, SOD, CD, NM, SYMDAT.]
TDATE	Date of record, YYMMDD. CT date of transaction CQ date of quote CD date of transaction SOD date of SOD file that contained the record. (The SOD file for a given day contains only records that were entered/canceled/executed/ADMIN’ed on that day. It does not contain pending orders on which nothing was done.) SYMDAT date for which record pointers are supplied. [Integer, 4 bytes; CT, CQ, CD, SOD, SYMDAT.]
TIF	Time in force for an order. 1 day order 2 good till canceled 3 opening 4 good till canceled 5 immediate or cancel 6 fill or kill 7 day [Character, 1 byte; SOD.]

Variable Name	Description
TTIM	Transaction time. (Note: this is the time stamp from the Consolidated Trade System.) [Integer, 4 bytes; CT, CD.]

Appendix C: An Analysis Using TORQ Data

This appendix presents an example using the TORQ trade and quote data. The data for the first ticker symbol (AC) are pulled off the database into ASCII files using the SEL1 program. Next a SAS program reads the files and performs statistical analyses. The analysis implemented here is the vector autoregression and random-walk decomposition techniques discussed in Hasbrouck (1991, a, b). The SAS code that generated the output is supplied on the TORQ program disk. After installation, you should find it in \TORQ\SRC\VAR.SAS.

This example does not completely reproduce the analysis of the cited papers. In the interests of simplicity, only one trade variable is used and overnight returns are not accorded any special handling.

Although SAS is capable of reading the TORQ files directly, it is easier to extract the trade and quote data using the selection program SEL1. To pull off trades and quotes for the first ticker symbol for all the days in the sample, the command used was:

```
SEL1 AC AC 901101 910131 TQ DATA
```

The trades and quotes are placed in the two files SEL1CT.ASC and SEL1CQ.ASC. These files could be exported to wherever your preferred statistical software happens to reside. This particular analysis was done using the PC version of SAS. The edited and annotated output is given on the next pages. Framed numbers are used for cross-reference with the discussion.

The program VAR.SAS first reads the data files, constructs SAS datasets and computes some summary statistics. See . Next, the program performs two parallel analyses. The first is for quote-midpoints; the second is for transaction prices. The procedures are identical for the two cases. (In fact, they are executed using calls to the same macro.) Each analysis uses two variables:

r_t 100 x log return (based on either transaction prices or quotes).

xs_t a signed trade variable.
xs = +(trade volume in round lots)$^{1/2}$ if the transaction price is above the midpoint of the prevailing quote.

xs_t = -(trade volume in round lots)$^{1/2}$ if the transaction price is below the midpoint of the prevailing quote. xs_t is set to zero if the transaction price is at the midpoint.

A single trade variable is used to keep the analysis simple. The square-root transform was chosen for its concavity.

The general form of the vector autoregressions is:

$$\begin{aligned} r_t &= \sum_{i=1}^{5} A_i r_{t-i} + \sum_{i=0}^{5} B_i xs_{t-i} + v_{1,t} \\ xs_t &= \sum_{i=1}^{5} C_i r_{t-i} + \sum_{i=1}^{5} D_i xs_{t-i} + v_{2,t} \end{aligned} \qquad \text{(C.1)}$$

The coefficient estimates are given in . To get the vector moving average (VMA) form of the model, it is necessary to result to a little trickery. Strictly speaking, SAS does not invert a VAR. It can compute forecasts, however, and this ability can be used to construct the VMA. The VMA coefficients may be computed from the forecasts subsequent to unit shocks on the variables. This program computes the the moving average representation for the r_t equation only:

$$r_t = \sum_{i=0}^{10} AA_i v_{1,t-i} + \sum_{i=0}^{10} BB_i v_{2,t-i} \qquad \text{(C.2)}$$

The MA computation is truncated at lag 10, and the AA and BB coefficients are printed at .

In a random-walk decomposition model, the price is viewed as the sum of two components: $p_t=m_t+s_t$, where p_t is a price variable (transaction price or quote midpoint), m_t is assumed to follow a random-walk with increments w_t ($m_t=m_{t-1}+w_t$) and s_t is a zero-mean covariance stationary process. The cited papers note that

$$Var(w_t) = (\Sigma AA_i) Var(v_{1,t}) (\Sigma AA_i) + (\Sigma BB_i) Var(v_{2,t}) (\Sigma BB_i) \qquad \text{(C.3)}$$

(Since xs_t is included in the r_t equation, the covariance matrix is diagonal.)

The coefficient sums are reported in the table at 3 as WAA and WBB; the variance is reported at . The relative measure of trade informativeness proposed in the RFS paper is

$$R_w^2 = \frac{\left(\Sigma BB_i\right) Var(v_{2,t})\left(\Sigma BB_i\right)}{Var(w_t)} \tag{C.4}$$

This is also reported at .

Under the Beveridge-Nelson identification restriction, s_t may be represented as

$$s_t = \sum_{i=0}^{9} SAA_i \, v_{1,t-i} + \sum_{i=0}^{9} SBB_i \, v_{2,t-i} \tag{C.5}$$

where

$$SAA_j = -\sum_{k=j+1}^{10} AA_k \quad and \quad SBB_j = -\sum_{k=j+1}^{10} BB_k \tag{C.6}$$

Next,

$$Var(s_t) = \sum_{i=0}^{9} SAA_i^2 Var(v_{1,t}) + \sum_{i=0}^{9} SBB_i^2 Var(v_{2,t}) \tag{C.7}$$

Hasbrouck (1991b) suggests taking Var(s_t) for transaction prices as a summary statistic of market quality. This number for transaction prices is given at the end of the listing at . It corresponds to a standard deviation of .2%. In other words (loosely speaking), the standard deviation of the discrepancy between the actual transaction price and the implicit random-walk ("efficient") price is .2%.

```
Stats for Trades             10:26 Wednesday, January 8, 1992   1

N Obs  Variable     N  Minimum  Maximum    Mean  Std Dev
--------------------------------------------------------
 1413  TDATE     1413   901101   910131  904344     4281
       TTIM      1413    34208    57873   45517     7319
       PRICE     1413    12.88    17.37   15.38    1.106
       SIZ       1413    100.0    75000   791.7     2211
       G127      1413        0    40.00   23.72    19.66
       TSEQ      1413        0     3142    1466    651.8
--------------------------------------------------------

Stats for Quotes             10:26 Wednesday, January 8, 1992   2

N Obs  Variable     N  Minimum  Maximum    Mean  Std Dev
--------------------------------------------------------
 1270  TDATE     1270   901101   910131  903920     4135
       QTIM      1270    34207    58781   45683     7867
       BID       1270    12.88    17.25   15.12    1.136
       OFR       1270    13.00    17.50   15.30    1.144
       BIDSIZ    1270    1.000    170.0   23.26    29.45
       OFRSIZ    1270    1.000    140.0   18.29    18.56
       MODE      1270    3.000    12.00   11.31    2.241
       QSEQ      1270    695.0     3148    1606    398.9
--------------------------------------------------------
```

```
Quotes: Summary Stats (main.varq)                                3
                                 10:26 Wednesday, January 8, 1992

N Obs  Variable     N  Minimum  Maximum    Mean  Std Dev
--------------------------------------------------------
 1890  R         1889   -2.391    2.708  0.0138    0.366
       XS        1890   -27.39    10.20   0.153    2.103
--------------------------------------------------------
```

```
Quotes: Estimation of Bivariate VAR (main.varq)                  4

MODEL Procedure
SUR Estimation

Nonlinear SUR Summary of Residual Errors

            DF    DF
Equation Model Error        SSE       MSE R-Square Adj R-Sq

R           11  1873  205.5212   0.10973   0.1879   0.1836
XS          10  1874      8298   4.42795   0.0071   0.0024

Nonlinear SUR Parameter Estimates

                          Approx.       ’T’   Approx.
Parameter    Estimate     Std Err     Ratio  Prob>|T|

A1         0.00917173     0.02311      0.40    0.6915
A2           0.016865     0.02314      0.73    0.4662
A3          -0.012121     0.02314     -0.52    0.6005
A4          -0.022193     0.02315     -0.96    0.3378
A5          -0.015666     0.02310     -0.68    0.4977
B0           0.073977   0.0036364     20.34    0.0001
B1         0.00940287   0.0040261      2.34    0.0196
B2        -0.00065929   0.0040386     -0.16    0.8703
B3        -0.00239973   0.0040388     -0.59    0.5525
B4         0.00324216   0.0040384      0.80    0.4222
B5         0.00283399   0.0040360      0.70    0.4826
C1          -0.239185     0.14667     -1.63    0.1031
C2          -0.329525     0.14678     -2.24    0.0249
C3          -0.154319     0.14697     -1.05    0.2938
C4          -0.138918     0.14700     -0.94    0.3448
C5           0.055473     0.14672      0.38    0.7054
D1           0.074922     0.02552      2.94    0.0034
D2           0.043708     0.02564      1.71    0.0884
D3           0.041731     0.02564      1.63    0.1038
D4           0.053097     0.02562      2.07    0.0384
D5          -0.015343     0.02564     -0.60    0.5496


Number of Observations       Statistics for System
Used               1884       Objective      1.9889
Missing               1       Objective*N      3747
```

```
Quotes: Moving Average Rep of r Equation (main.varq)             8


OBS  I    AA        BB      WAA      WBB      SAA       SBB

  1  0  1.00000  0.073977 0.90028 0.096980  0.099724 -0.023004
  2  1 -0.00852  0.014315 0.00000 0.000000  0.091202 -0.008689
  3  2 -0.01101  0.002752 0.00000 0.000000  0.080187 -0.005937
  4  3 -0.02916 -0.000736 0.00000 0.000000  0.051030 -0.006673
  5  4 -0.03576  0.004941 0.00000 0.000000  0.015271 -0.001732
  6  5 -0.01541  0.001454 0.00000 0.000000 -0.000141 -0.000278
  7  6 -0.00146  0.000001 0.00000 0.000000 -0.001605 -0.000277
  8  7 -0.00001  0.000144 0.00000 0.000000 -0.001612 -0.000133
  9  8  0.00043  0.000216 0.00000 0.000000 -0.001186  0.000083
 10  9  0.00090 -0.000054 0.00000 0.000000 -0.000289  0.000030
 11 10  0.00029 -0.000030 0.00000 0.000000  0.000000  0.000000
```

```
Quotes: Disturbance Variance-Covariance Matrix (main.varq).

OBS        R          XS

 1      0.10973    -0.00000
 2     -0.00000     4.42795
```

```
Quotes: Results of Random-Walk Decomposition (main.varq).

OBS       VS          VW         VWX        R2W

 1     .0060663    0.13058    0.041646    0.31893
```

```
Trades: Summary Stats (main.vart)                              11
                                 10:26 Wednesday, January 8, 1992

N Obs  Variable     N  Minimum  Maximum    Mean  Std Dev
--------------------------------------------------------
 1413  R         1412   -1.905    2.317   0.019    0.608
       XS        1413   -27.39    7.141   0.200    2.403
--------------------------------------------------------

Trades: Estimation of Bivariate VAR (main.vart)                12
                                 10:26 Wednesday, January 8, 1992

MODEL Procedure
SUR Estimation

Nonlinear SUR Summary of Residual Errors

            DF    DF
Equation Model Error        SSE       MSE R-Square Adj R-Sq

R           11  1396  295.9501   0.21200   0.4324   0.4283
XS          10  1397      7992   5.72083   0.0200   0.0137


Nonlinear SUR Parameter Estimates

                          Approx.       ’T’   Approx.
Parameter    Estimate     Std Err     Ratio  Prob>|T|

A1          -0.185455     0.02676     -6.93    0.0001
A2          -0.061201     0.02724     -2.25    0.0248
A3          -0.036452     0.02725     -1.34    0.1812
A4         0.00073850     0.02717      0.03    0.9783
A5          -0.026055     0.02430     -1.07    0.2839
B0           0.147812   0.0051504     28.70    0.0001
B1          -0.061610   0.0065166     -9.45    0.0001
B2        -0.00592037   0.0067401     -0.88    0.3799
B3        -0.00561752   0.0067392     -0.83    0.4047
B4         0.00447836   0.0067208      0.67    0.5053
B5           0.009664   0.0063674      1.52    0.1293
C1           0.203153     0.13891      1.46    0.1438
C2          -0.266113     0.14132     -1.88    0.0599
C3          -0.140889     0.14149     -1.00    0.3195
C4           0.099084     0.14111      0.70    0.4827
C5          -0.027450     0.12625     -0.22    0.8279
D1           0.094896     0.03376      2.81    0.0050
D2           0.106184     0.03490      3.04    0.0024
D3           0.016895     0.03501      0.48    0.6294
D4          -0.052300     0.03488     -1.50    0.1340
D5           0.042573     0.03306      1.29    0.1980


Number of Observations       Statistics for System
Used              1407       Objective     1.9851
Missing              1       Objective*N     2793

Trades: Moving Average Rep of r Equation (main.vart)           16
                                 10:26 Wednesday, January 8, 1992
```

```
OBS  I    AA       BB      WAA      WBB      SAA      SBB

  1  0  1.00000  0.14781 0.75609 0.084711  0.24391  0.063102
  2  1 -0.15543 -0.07056 0.00000 0.000000  0.08848 -0.007455
  3  2 -0.08604 -0.00006 0.00000 0.000000  0.00244 -0.007518
  4  3 -0.01304 -0.00632 0.00000 0.000000 -0.01060 -0.013843
  5  4  0.03741  0.00266 0.00000 0.000000  0.02680 -0.011179
  6  5 -0.03799  0.01117 0.00000 0.000000 -0.01118 -0.000007
  7  6  0.01360  0.00055 0.00000 0.000000  0.00241  0.000543
  8  7 -0.00319 -0.00057 0.00000 0.000000 -0.00077 -0.000030
  9  8  0.00011  0.00017 0.00000 0.000000 -0.00066  0.000142
 10  9 -0.00018 -0.00041 0.00000 0.000000 -0.00084 -0.000264
 11 10  0.00084  0.00026 0.00000 0.000000  0.00000 -0.000000

Trades: Disturbance Variance-Covariance Matrix (main.vart).    17
                                 10:26 Wednesday, January 8, 1992

OBS        R          XS

 1      0.21200    -0.00000
 2     -0.00000     5.72083

Trades: Results of Random-Walk Decomposition (main.vart).      18
                                 10:26 Wednesday, January 8, 1992

OBS       VS          VW         VWX        R2W

 1     0.039712    0.16225    0.041052    0.25302
```

Appendix D: The Source Code for the SELECT Programs

The SELECT installation procedure places the source code for the SELECT programs in the subdirectory \TORQ\SRC. The SELECT programs were written in Fortran and compiled with the Microsoft Fortran Compiler v. 5.1. I generally follow the practice of placing all COMMON blocks in separate files (with the .CMB extension) and including them in the source code (with the Fortran "INCLUDE" statement) whenever I need them.

The files are:

TORQFILS.CMB This file contains common blocks that describe the TORQ data files. The layout of the common blocks corresponds exactly to the layout of the files. TORQFILS.CMB is frequently included in SEL1 and SEL2 subroutines.

TORQSUBS.FOR Subroutines used by the SELECT routines to do case and format conversion.

SEL1.FOR Source code for SEL1.

SEL1.CMB Common blocks used in SEL1.

SEL2.FOR Source code for SEL2.

SEL2.CMB Common blocks used in SEL2.

VAR.SAS A SAS program to perform statistical analysis. See Appendix C.

A note on variable names: because standard Fortran does not allow for two variables to have the same name, I have modified certain variable names in the common blocks. SYMBOL, for example, is found in all TORQ files. To keep the variable names unique, it is SYMBOLCT in the CT file, SYMBOLCQ in the CQ file, etc..

References

Lawrence Harris and Joel Hasbrouck, 1992, "Market vs. Limit Orders: SuperDOT Evidence on Order Submission Strategy," (NYSE working paper, Research and Planning Section).

Joel Hasbrouck and Deborah Sosebee, 1992, "Orders, Trades, Reports and Quotes at the New York Stock Exchange," (NYSE working paper, Research and Planning Section).

Joel Hasbrouck, 1991, "Measuring the Information Content of Stock Trades (Journal of Finance, 46, 1 (March, 1991), 179-208).

Joel Hasbrouck, 1991a, "The Summary Informativeness of Stock Trades: An Econometric Analysis" Review of Financial Studies, 4 (1991), 3.

Joel Hasbrouck, 1991b, "Assessing the Quality of a Security Market: A New Approach to Transaction Cost Measurement," forthcoming, Review of Financial Studies.