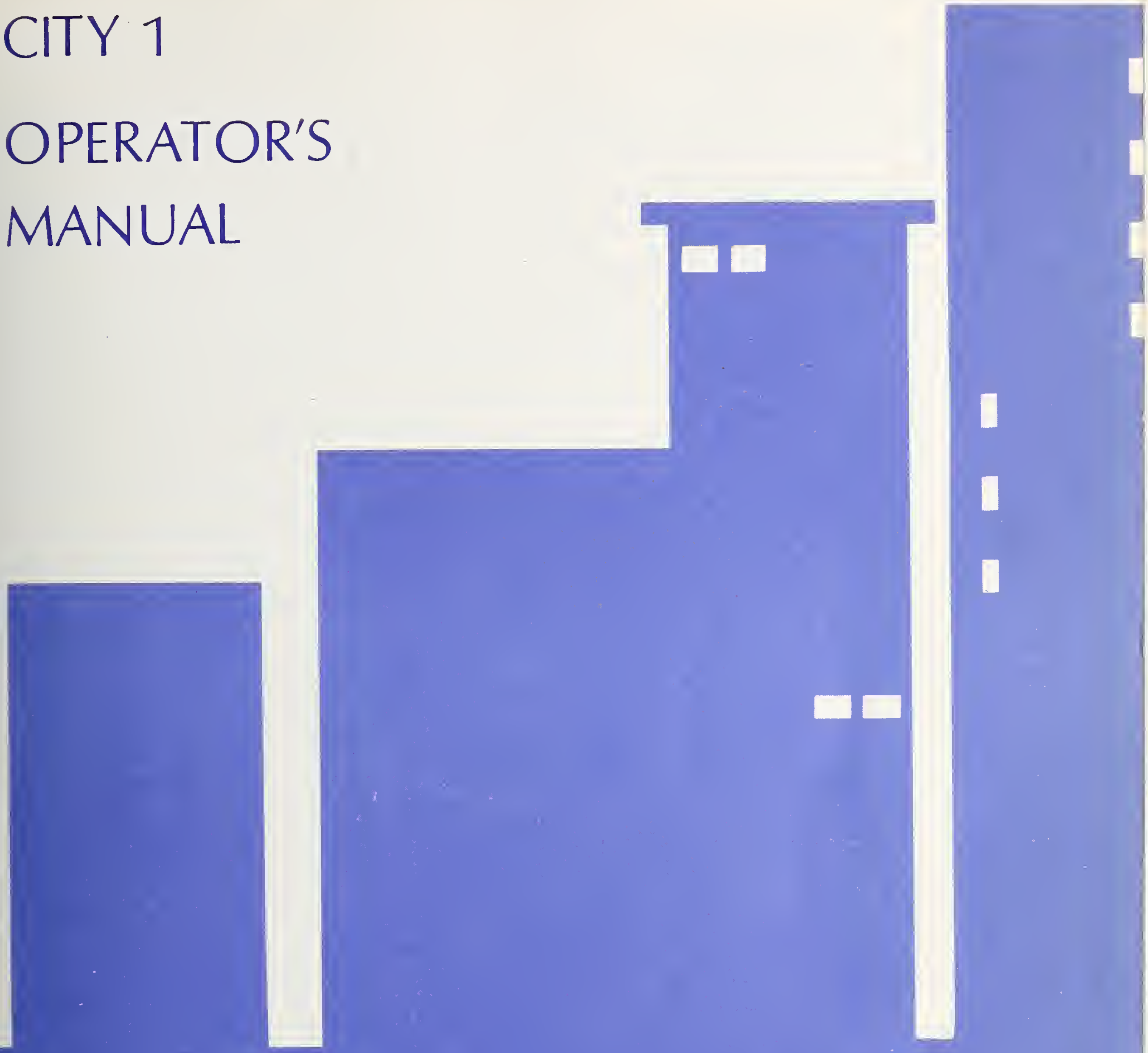
NBSIR 73-108

COM-74-10701-2

24

CITY 1

OPERATOR'S MANUAL



CITY GAMES

National Technical Information Service

NTIS
National Technical Information Service
U.S. DEPARTMENT OF COMMERCE
P.O. Box 1551
Springfield, Va. 22151

Price of City Games
Computer Files Includes
Related Manuals

CITY I

OPERATOR'S MANUAL

written by John E. Moriarty

Technical Analysis Division
Institute for Applied Technology
National Bureau of Standards
Washington, D. C. 20234

October 1973

Final Report

Sponsored by
National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22151

TABLE OF CONTENTS

I. INTRODUCTION

City I is an operational simulation game in which participants make economic, government and social decisions affecting a hypothetical metropolitan area. Through the use of a computer, the simulated urban system responds to the participants' decisions as any real city would. Each player in City I is assigned to a team which shares an economic and governmental role. The interrelated decisions made by teams will guide the way the simulated city changes in composition and size.

This Computer Operator's Manual is designed to be a reference manual for the computer operator. Although examples are given for a test program, the design of the games assumes that the operator's directions are given by the Game Director during actual game play.

It is not necessary that the operator know how to play the game or that the computer be located at the game play site. Game details are outlined by the Game Director (See Director's Manual) and the players respond utilizing information supplied by the Director and Players' Manual. The primary role of the computer operator is therefore to actually operate the computer, receiving playing instructions from the Game Director and to advise the Game Director of any operating difficulties that might arise during operations. If additional information is desired about the actual game play, it is suggested that the operator acquire the Director's and Players' Manuals. These Manuals are designed as reference manuals and contain instructions for gaming operations along with card formats and command instructions.

II. MODEL PROCESS

A. The Computer

It is assumed throughout this manual that the operator knows how to cold start the 1130, change carriage tapes, and load and clear the card reader. These are simple operations, and can be learned from anyone familiar with the 1130.

City I must be run on an IBM 1131 Central Processor with at least 8K core and single disc storage and the following accessories:

- 1442 mod 6 or 7 card reader punch or 2501 card reader and 1442-5 card punch

- 1403 printer or 1132 printer

The binary decks enclosed in the City I package are already adapted to this particular user's machine configuration.

Output takes approximately 30 minutes on the 1403 printer and one hour on the 1132. The time required for output increases as the city grows.

B. Game Sequence

Each round of City I, representing one year in the life of the simulated area, takes about two hours of players' time. When they have completed their decisions, those decisions are coded for computer input and punched on cards. After the information on the cards is put into the computer, the computer operates on the new data, relating it to the previous conditions in the city, and produces output containing detailed information about the new status of the city. The computer outputs represent yearly reports on city conditions. After output is printed, the players spend another two hours making decisions for the next year.

These processes might not be performed consecutively. For example, players might spend two hours of one day making decisions. The next day, the director codes and punches the cards. Two days later the decisions are processed and output is printed. If the players were meeting only once each week, they would have a few days to examine their output before meeting with the rest of the group.

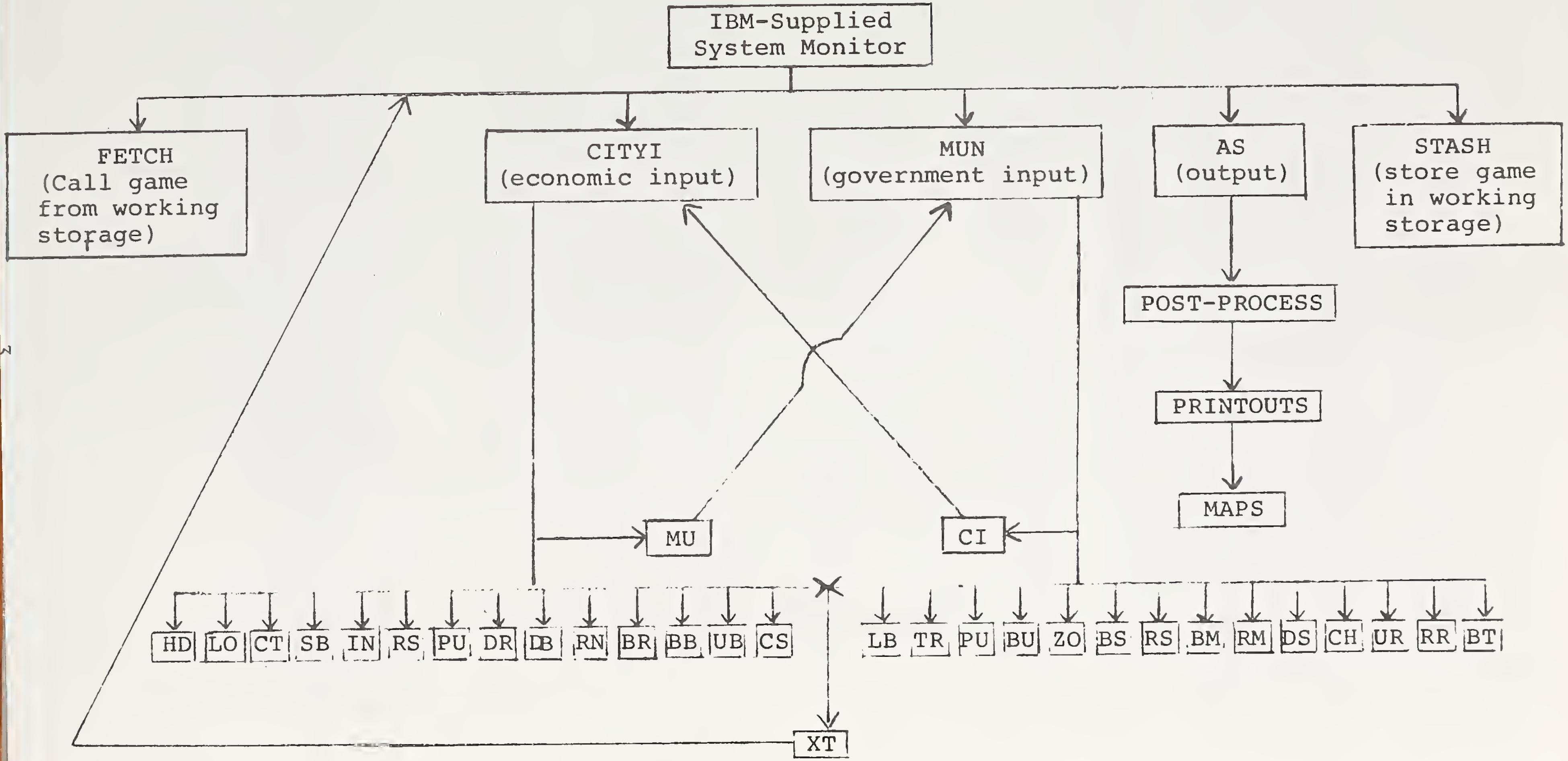
C. Program Design[1]

Figure 1 illustrates the design of the City I program. There are five main programs which can be called directly through the IBM monitor. Any one can be called by a //bXEQbname command. For example: //bXEQbCITYI calls the economic decisions input programs.

[1] All commands to the computer begin in Column 1. Blank spaces are indicated by a "b".

FIGURE I

CITY I PROGRAM



All main programs, except CITYI and MUN, return to the IBM monitor when they are finished. Another main program can then be called. At any time the program will return to the IBM monitor if Interrupt Request is hit on the console. But, such an action garbles the contents of core and should be avoided when a program is operating.

CITYI and MUN are the main input programs. Each contains several programs which must be specifically addressed in order for particular decisions to be processed. These input programs are called with a dollar sign preceeding the name.[2] For example, bXEQbCITYI $SB calls the program that accepts decisions to grant subsidies. When all subsidies have been input, a card with two asterisks returns control to CITYI, and another CITYI program can then be called.

Control can be returned to the IBM monitor through both CITYI and MUN by the special code $XT when either of the two main programs is in control.

The following example shows a call from the IBM monitor to two CITYI input programs and a return to the IBM monitor.

```
//bXEQbCITYI
$SB
Subsidy  decision cards (any number)
**
$CT (another type of economic decision)
decision cards
**
$XT
```

Because operators may want to transfer from CITYI to MUN often during input, there are special programs that connect the main programs directly and allow the transfer without requiring a return to the IBM monitor. From MUN to CITYI the program is $CI; its opposite is $MU. In the previous example, $XT could have been replaced by $MU and followed directly by $RM, for example.

The command //bXEQbAS is used after all decisions have been processed. It automatically calls the post-process programs and produces output for the players. FETCH is usually used only at the very beginning of input and STASH is usually used only at the end of output.

[2] If the decisions are entered through the console, omit the dollar sign. The end of a line is specified by hitting the EOF button (EOF).

Note: The entire program is well protected against typing errors, so if any are made, nothing is affected. Just try again.

III. GENERAL INPUT PROCEDURE

A. Preparing the Computer for Input

First, the disc containing the City I program must by put on the machine. Turn the file on and wait for the FILE READY light to glow on the console. Make sure that all console switches are down. Then, cold start the disc by putting a cold start card and a blank card in the reader, turning the reader and printer on, and hitting IMMEDIATE STOP, RESET, and LOAD, in that order, on the console. If typewriter control is desired, put a //bTYPE card in the reader, followed by a blank. (A blank should always be the last card in the reader.) Hit START on the reader and PROGRAM START on the console.

Copy the data base on which the decisions are to be run into the game area. (See Retrieving Data from Storage-III).

B. Processing Decisions

After a round is in the Game Area, data switches must be set for input of the new round's transactions.

All switches except the following should be down:

Switch 10: Down - Messages regarding transaction success or failure appear on the printer.

Up - Messages printed on console typewriter. This is useful when decisions are not on cards.

Switch 11: Down - Residential growth restrictions in effect.

Up - Growth restrictions omitted.

Switch 13: Down - Typewriter prints messages specifying what information must be given for each transaction. An operator unfamiliar with all the programs and not using card input should have this switch down.

Up - Messages not printed.

Switch 14: Down - Information is to be input through the console typewriter.

Up - Information is to be input through the card reader.

If control is at the console keyboard and decisions are to be input by cards, make sure that Switch 14 is up and type //bXEQbCITYI or //bXEQbMUN, whichever applies to the first decision. Hit EOF on the console and START on the reader. The first card in the reader should be a $ card, i.e., a program within whichever main program name was typed on the console. A blank card must follow the last typed card in the reader.

The computer will process the cards in the reader until all cards have been read or a data error is encountered. The program pauses when it finds a data error. Either find the error and correct it or ignore it, in which case the decision with the error won't be processed. To get the program restarted, hit START on the console.

After all decisions have been processed, return control to the IBM monitor. (See Program Design.)

IV. RETRIEVING DATA FROM STORAGE (before input)

When a data base is to be copied into the game area in order that operations may be performed on it, the name under which it was originally copied out must be known. Occasionally the operator may forget what name he assigned to it. If the data was copied on cards, the name can be found in the last columns. If it was stored in the Users' Area, it can be found by typing:

```
//bDUP
*DUMPLET
```

This prints the names of programs and data bases stored. The list of data base names has DDF in the format column in LET.

If the data is to be copied from cards, type:

```
//bDUP
*STOREDATAbbCDbbWSbbXXXXXbYYYY
```

XXXXX is the name of the data base. YYYY is the number of binary cards in the deck. This number is punched on the last card of the deck.

If the data base is to be copied from the Users' Area, type:

```
//bDUP
*DUMPbbbbbbbUAbbWSbbXXXXX
```

Whether the data was copied from cards or from the Users' Area, the next step is:

```
//bXEQbFETCH
```

When this process is complete, the data is in the game area, ready for input.

The methods of copying data are illustrated in Figure 2.

Figure 2

TRANSFER OF DATA

From: \ TO:	Game Area	Users Area	Working Storage	Cards
Game Area			//bXEQbSTASH (When this finishes copying in the game, it prints STORE OR DUMP N SECTORS (N = 2 digit number of Working Storage sectors used by the data)	
Users Area			1) //bDUP 2) *DUMP cc. 13 UAbbWSbb Name stored under	1) //bDUP 2) *DUMP cc. 13 UAbbCDbb Name 3) blank cards
Working Storage	//bXEQbFETCH	1) //bDUP 2) *STOREDATAbb WSbbUAbb Name cc. 28-30 sector count		1) //bDUP 2) *DUMPDATA cc. 13 WSbbCDbbName cc. 28-30 sector count 3) blank cards
Cards		1) //DUP 2) *STOREDATAbb CDbbUAbbName cc. 27-30 count of number of binary cards (on last card of binary deck) 3) Binary Deck	Same as from cards to Users Area except WS is punched instead of UA	

V. INPUT CODES AND FORMATS

A. Ordering the Data

The order in which decisions are processed by the computer can be very important for their success or failure. Player decisions are not related to each other until post-process. The input programs only check that each line of information is meaningful, e.g., that the typed coordinates are in fact on the board, and that the attempted transaction is legal. For example, a construction decision will be rejected if the team constructing does not have sufficient funds. If the team made a decision to borrow money to finance the construction, the loan decision would have to be processed first in order for the construction to be accepted by the computer.

Input programs change the data files whenever the computer accepts a decision. When an economic team or government department makes a decision which involves an expenditure or receipt of money, its cash account is changed accordingly by the input program. So, if for some reason the operator wishes to rerun an entire set of inputs, he must call a clean data base from the Users' Area or cards.

In general, because the operator does not always know whether the order of decision inputs is important for a particular round, all government decisions should be processed before economic ones are. The suggested order is:

1. Government Sector

   a. Budget
   b. Land purchases
   c. Road construction
   d. Terminal construction

2. Economic Sector

   a. Loans, cash transfers, subsidies
   b. Land purchases
   c. Demolition
   d. Construction*

NOTE: Dollar amounts, when typed for the computer, are not preceded by a dollar sign. None of the digits in one number should be separated by commas. For example, $1,200 must be expressed as 1200.

*Construction of low-income residences should precede any low-income demolitions.

B. Summary of MUN Input Programs

Call Letters	Function	Format
LB	Load Budget	Appropriations for PW, SC, HY, PZ/1000
TR	Tax Rates & Base	(1) Tax rates for property, sales, income in tenths of percents. (2) Tax base for property, sales, income/1000
PU	Purchase Land	Location, bid/100, purchaser, seller, 1 if part square, 1 to assure purchase, (Two-letter codes for department names: Highway, HY; School, SC; Public Works & Safety, PW; Planning, PZ)
BU	Build Utilities	Location of plant, 1 (build) or 0 (add to existing plant) /squares to be serviced (divided by slashes and end with /**)
ZO	Change Zoning	Location, new zoning code, 1 if more than 5 changes are to be processed this round.
BS	Build Schools & Change School Employment	Location, number high, medium employees, 0 (build) or 1 (change employment)
RS	Renovate Schools	Location, new value ratio
BM	Build MS & Change MS Employment	Location, number medium, low employees, 0 (build) or 1 (change employment)
RM	Renovate MS	Location, new value ratio
DS	Demolish MS or School	Type (MS or SC), location

Call Letters	Function	Format
CH	Change Public Salaries	Department code (PW or SC), new salaries for low, medium or high/100
UR	Build and Upgrade Roads	Segment location, new type
RR	Renovate Roads	Segment location, new value ratio
BT	Build Terminals	Intersection location, new type

C. Summary of City I Input Programs

Call Letters	Function	Format
HD	Game Heading	Heading (maximum 80 characters)
LO	Loan	Team borrowing, amount/100, lender (OU or team), (if team) term, interest rate in tenths of percents.
CT	Cash Transfer	Team giving (team name or OU), team receiving (team name or OU), amount/100
SB	Subsidy	Team receiving, amount/100
IN	Outside Investment	Team, amount/100, SP (Speculative) or BC (Conservative)
RS	Restore Slum	Owner, location
PU	Land Bid or Land Transfer	Location, bid (in hundreds), purchaser, seller (if seller is county, type CI), 1 to buy part or -1 to buy all parkland on parcel, 1 to assure purchase or -1 to assure bid failure
DR	Demolish Residences	Owner, location, new level.

Call Letters	Function	Format
DB	Demolish Businesses	Owner, location, new level,
RN	Renovate	Team, location, new value ratio
BR	Build Residences	Team, location, number new R's, class (1 = low, 3 = high)
BB	Build Businesses	(1) Team, location, building type (two letter code), level (2) Salaries to low, medium, high employee/100 (3) Charges to LI1, HI1, PG1, PS1/10 (3) Charges to low, middle, high/10
UB	Upgrade Businesses	Team, location, new level
CS	Change Prices or Salaries	(1) Location, 1 to change prices, 2 to change salaries (2) New charges to LI1, HI1, PG, PS/10 (2) New salaries to low, medium, high/100

VI. SAMPLE TRANSACTIONS WITH THE COMPUTER

Below is a listing, from cold start to beginning of output, of all transactions with the computer if the decisions used as examples in the next section were put in as one round's decisions and the procedures described were followed.

```
     Console:  All switches down
Cards in card reader: Cold start card
                      //bTYPE
                      blank card

     Hit on card reader:  START
     Hit on console:      IMMEDIATE START
                          RESET
                          LOAD

     Control is sent to console typewriter.
Type on console:      //bDUP           EOF
                      *DUMPbbbbbbUAbbWSbbSAMPL      EOF
                      //bXEQbFETCH     EOF
```

The round called SAMPL (fictional round stored in Users' Area) is now in the game area.

Set data switches for card input: Switches 13 and 14 up, others down.

Put input cards in card reader. The cards are:

```
//bXEQbCITYI
$LB
102,132,97,22
**
$TR
29,15,20
7875,74,452
**
$PU
9014,10,HY,F,1
**
$BU
9735,0/9432/9430/**
$BU
7713,0/7620/7622/**
$BU
10331,1/10430/10630/**
```

```
$ZO
7412,10
**
$BS
8418,400,300,0
7412,200,400,1
**
$RS
7412,95
**
$BM
10238,400,300,0
9636,500,200,1
**
$RM

9636,97
**
$DS
SC,7412
MS,9636
**
$CH
SC,O,52,104
**
$UR
8413,1
8613,2
**
$RR
7413,97
7613,97
**
$BT
8317,1
**
$CI
$HD
*****ECONOMICS 104 CLASS *****
**
$LO
B,250,OU
C,340,D,5,60
**
$CT
E,F,46
OU,G,850
B,OU,600
**
$SB
```

```
A,45
**
$IN
H,120,BC
C,-30,SP
C,30,BC
**
$RS
D,7412
**
$PU
9214,50,E,OU
10032,12,G,CI,1
9216,50,A,OU,0,1
**
$DR
B,7414,1
**
$DB
H,7416,0
**
$RN
B,7412,95
D,7618,100
**
$BR
A,10038,2,2
F,9438,2,3
**
$BB
A,10038,BG,2
26,51,100
2700,1900,450
E,10034,HI,1
25,50,100
D,9830,PG,3
26,52,104
90,160,315
**
$UB
G,7014,3
E,9836,2
**
$CS
9834,2
27,54,100
7810,1
2100,1700,400
**
blank card
```

The CITYI monitor is in control. If any cards require correction, and just as a precautionary measure, now return input control to the console.

Set data switches for return to console control: All switches down. Cards in card reader:

```
$XT
blank card
```

```
Hit on card reader:  START
Hit on console:      PROGRAM START
```

Input control is now at the console and the IBM monitor is in control.

To start post-process, type:

```
//bXEQbAS     EOF
```

After all maps are printed, to store the results of the round in the Users' Area, type:

```
//bXEQbSTASH   EOF
//bDUP         EOF
(Typewriter prints:  STORE OR DUMP
042 SECTORS)
*STOREDATAbbWSbbUAbbSAML2bb042  EOF
```

Turn off the disc. Remove it when disc unlocks.

VII. OUTPUT PROCEDURE

When all transactions for a round have been entered and output is desired, the following switches must be set (and all others should be down):

```
Switch 1:  Down - Economic output prints in alphabetical
                  order by team.

           Up   - Economic output prints by team in a
                  random order.

Switch 2:  Down - Economic output printed in detail.

           Up   - Detailed economic output omitted.

Switch 3:  Down - Economic summary printed.

           Up   - Economic summary not printed.

Switch 5:  Down - Prints New Town Councilman Report.

           Up   - New Town Councilman Report not printed.
```

Type through console or process from card:

`//bXEQbAS` Then hit EOF button EOF.

The computer will then be occupied for at least 30 minutes. First, it runs assessments and depreciation, printing several pages of output. Then for 10 to 15 minutes nothing is printed while it runs the employment, commercial, school, municipal service, and terminal allocators. Then it begins to print the full output. Economic is first, followed by the summary output and then the government portions.

After government output, the console typewriter prints the following message:

```
PLEASE ADJUST THE PRINTER
CARRIAGE TAPE WITHOUT CHANNEL 12 PUNCH
SPACING OF 8 LINES/INCH
```

The program pauses while the operator changes printer tapes and sets switches. The printer carriage tape which is used for normal output must be replaced by the map tape. The normal tape has a 1-punch in Channels 1 and 67 and a 12-punch in Channels 62 and 128. The cut is at 132. The map tape has a 1-punch in Channel 1 and is cut at 88. When this is complete, set the following data switches (if a switch is down, the related operations are not performed):

RECEIVE

If All Switches Are Down: 1 status map, 1 change map, 1 highway map, and 1 new road map are printed. No matter which switches are set, at least one of each of these maps is printed.

If Switch 7 Is Up: The typewriter prints the name of each map before it is printed. If this switch is up before regular output finishes, the typewriter prints a list of the functions of the map data switches before it prints the message to adjust the carriage tape.

Switch 8: The computer repeats whatever set of maps is indicated by switches 10, 11, 13, & 15.

Switch 9: If this is put up while a map is printing, the printer stops the map and begins printing the next map.

Switch 10: Up to 11 copies of the complete status map are printed. To print fewer than 11, put this switch down when the desired number of maps have been printed.

Switch 11: Up to 11 copies of the change map are printed. The procedure is the same as that for the status map.

Switch 12: All roads are printed on the change map. Usually, only new roads are printed.

Switch 13: A map of each team's land and developments is printed.

Switch 14: The map printing begins at the top of a page, not in the normal fashion of spacing.

Switch 15: Up to 11 copies of the highway map are printed. Procedure is the same as that for Switches 10 & 11.

Usually the only switch up during mapping is 10, since status maps are the most useful.

After the data switches are set, hit PROGRAM START on the console, and the computer will execute the instructions. When all mapping is finished, hit INTERRUPT REQUEST and the program will return to the IBM monitor. Change the carriage tape.

Store the data. (See Copying Data)

Turn the disc off. When the DISC UNLOCK light glows, the disc can be removed from the machine.

VIII. COPYING DATA (END OF OUTPUT)

The record of a round may be kept in one of three areas on the disc, or on cards:

1. Game Area - This is where the record must be kept while information is being processed for a round, i.e., all transactions occur here. Only one round's record can be in the Game Area at a time.

2. Working Storage - A record of a single round can be stored here if no operations are to be performed on the data.

3. Users Area - This is the library area where the records for several rounds may be stored, each with a different five-letter name. From 10 to 15 rounds can be stored, depending on the size of the records.

4. Cards - The record for a round can be cut on a binary deck, thus eliminating the necessity for storing the round in the Users Area.

In order to insure that a round's decisions are processed on the data base created by the previous post-process, a copy of the data base should be stored in the Users Area and/or on cards.

To store a data base, put in the following cards[3]:

```
//bXEQbSTASH
```

This copies the data from the Game Area into Working Storage. When the process is finished, the console typewriter prints[4]:

```
STORE OR DUMP n SECTORS
```

[3] It is probably easier to type all of the storage commands through the console, but cards can be used.

[4] If the typewriter prints a message to the effect that there is not enough room for a copy of the data, a round can be deleted from the Users' Area. Type:

```
//bDUP
*DELETEbbbbbbbbbbbbXXXXX
```

XXXXX is the name of the data which is to be deleted.

N is the number of sectors of Working Storage used by the data.

In order to store the data in the Users' Area, enter:

```
//bDUP
*STOREDATAbbWSbbUAbbXXXXXbbYYY
```

XXXXX is the five-character name of the data base and is assigned by the operator. It is the unique identification for the data base being stored. YYY is a three space name of the number of sectors, the "n" printed by the console after STASH. Each space must have a digit (e.g., 039, not 39).

When this process has completed, a copy of the data which is still in the Game Area is also in the Users' Area.

Another copy of the data can be dumped to cards from the Users' Area.

```
//bDUP
*DUMPbbbbbbbUAbbCDbbXXXXX
```

XXXXX is the name assigned by the operator to the data being called from the Users' Area. Make sure that there is a stack of blank cards (200 to 300) in the card punch.

BIBLIOGRAPHIC DATA SHEET

1. Report No.

2.

3. Recipient's Accession No. COM-74-10701-2

4. Title and Subtitle
CITY GAMES -
CITY I
Operator's Manual

5. Report Date

6.

7. Author(s)
John E. Moriarty

8. Performing Organization Rept. No.

9. Performing Organization Name and Address
National Bureau of Standards
Department of Commerce
Washington, D. C. 20234

10. Project/Task/Work Unit No.

11. Contract/Grant No.
956.2106

12. Sponsoring Organization Name and Address
National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22151

13. Type of Report & Period Covered

14.

15. Supplementary Notes

16. Abstracts

City I is an operational simulation game in which participants make economic, government, and social decisions affecting a hypothetical metropolitan area. The game is run on an IBM 1131 Central Processor with at least 8K core and single disc storage.

This manual describes the methods of operating the Central Processor relative to the game along with the special coding required to change and/or update the computer program and core mapping.

It is assumed throughout this manual that the operator knows how to cold start the 1130, change carriage tapes, and load and clear the card reader.

17. Key Words and Document Analysis. 17a. Descriptors

City; computer; directors; economic; games; government; metropolitan; players; sectors; simulation; social; urban.

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement This manual is available only with the lease of the CITY I Games system on IBM 1130 Disk Cartridge - COM-74-10701.

19. Security Class (This Report)
UNCLASSIFIED

20. Security Class (This Page
UNCLASSIFIED

21. No. of Pages
24

22. Price