SECTION

DATE ISSUED | DATE REVISED

ALERT II.

DATA PROCESSING

MANUAL OF STANDARDS

Section Number	Section Title
01	INTRODUCTION TO STANDARDS MANUAL
02	ORGANIZATION AND ENVIRONMENT OF COMPUTER SYSTEMS DIVISION
03	SECURITY PROCEDURES AND STANDARDS
04	FINAL DOCUMENTATION STANDARDS
05	SYSTEMS DEVELOPMENT AND DESIGN STANDARDS
06	PROGRAMMING STANDARDS AND AIDS
07	OPERATIONS STANDARDS
08	PROJECT MANAGEMENT STANDARDS
09	DEPARTMENT CODE CATALOG

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

DATE ISSUED | DATE REVISED

TABLE OF CONTENTS

01 INTRODUCTION TO STANDARDS MANUAL

INDEX NUMBER

01

01 INTRODUCTION TO STANDARDS MANUAL

1.0 GENERAL INFORMATION

The Data Processing Manual of Standards was established as a means of improving communications, of reducing duplication and ambiguity, and of minimizing difficulties caused by personnel turnover. The Standards Manual was designed to regiment clerical tasks and documentation within data processing and was not meant in any way to control the individualistic effort that should prevail in systems design and program development.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
INTRODUCTION TO STANDARDS MANUAL

DATE ISSUED | DATE REVISED

2.0 FUNCTIONS OF STANDARDS

Some of the functions of standards and their benefits are as follows:

2.1 Communications - The adoption of standards makes communications easier and more consistent. The communication of instructions for a job by word-of-mouth, or hastily written memos can often result in one or more of the following:

a. Rewriting and/or recompiling of programs due to a misunderstanding of job definition.

b. Additional test time often resulting in missed deadlines.

c. Logic errors in the completed object deck that may pass unrecognized for several months, thus compounding the errors.

All of the above mentioned problems can be reduced, if not eliminated, by conforming to pre-established standards of job specification, input/output layouts, block diagramming, coding, and testing. Standards not only greatly benefit direct communications between Management and Programming on immediate jobs, programs, etc., but also are a great aid in indirect communications. If all programs released to operations are in the same format, conform to standard labels, degree of comment, etc., it is much easier for machine operators to set up the job, check out and restart machine halts, and particularly debug latent program errors without the help of the original programmers. Standards also are of a great benefit to people other than the original programmers in working with a program and, when necessary, in making any revisions and modifications. This benefit is particularly evident when considering the turnover of personnel, in which case the original programmers of many jobs may no longer be with the Department or even in town.

2.2 Time Savings - Adhering to standards saves considerable effort. It further defines to the employee exact responsibilities and therefore he can better plan and utilize his time. In the absence of standards, people normally take the time to develop consistent approaches for themselves. This is particularly true in the use of individual subroutines rather than standardized macros or in individual testing procedures which often leave the program not fully tested prior to release to operations.

The greatest time-savings of all, however, are realized in the debugging phase of a program where additional programmers may be called in to render assistance on a program or where a programmer is assigned to finish another individual's program. It will take considerably less time for the new programmer to become acquainted with the program logic and coding if pre-established standards of labeling, commenting, and coding are utilized.

2.3 Aid to Management - Establishment of standards provide management with criteria to measure progress. It helps supervisors schedule, estimate completion time in order to meet deadlines, evaluate what has already been accomplished and, most important, what remains to be done. A program pending file, consisting of job specifications write-ups, can be reviewed weekly by management. By so doing management is continually aware of programming - in process and can act to alleviate potential problem areas before they develop.

2.4 Identification of Revisions

The following procedure will be used to identify revisions to this manual. The cover sheet should be filed at the front or back of the Standards Manual.

a. The cover sheet will be dated but will not necessarily be the same date as appears on the revised or added pages.

b. The cover sheet will note which Sections or pages have been added, deleted and/or modified.

c. Pages added in a revision will be noted by two asterisks preceding the page number.

d. Pages that have been added to or modified will have an asterisk in the left margin of the paragraph affected.

e. All modified pages will contain the revision date in the upper right corner of the page.

f. Pages inserted will have the same number as the page they follow, plus an alphabetic character. Example - Page 5A will follow page 5 and precede page 6.

g. Issue date is the date the page was first issued.

TABLE OF CONTENTS

02 ORGANIZATION AND ENVIRONMENT OF COMPUTER SYSTEMS DIVISION

1.0 ORGANIZATIONAL RESPONSIBILITIES - COMPUTER SYSTEMS DIVISION

1.1 The Computer Systems Division has three basic responsibilities.

a. To provide all automation tasks required to support the informational requirements of Kansas City, Missouri Police Department. The priority sequence of providing information is first to the officers on duty in the streets of Kansas City, Missouri and, as time and resources permit, in the form of administrative reports.

b. To provide on-line telecommunications information services, as furnished to our department, to metropolitan and regional agencies involved in the Criminal Justice Process.

c. To provide assistance in automating systems, for criminal justice agencies, who express a desire to combine efforts for a total Criminal Justice System.

1.2 The Computer Systems Division is composed of seven operating units.

a. On-Line Telecommunications Unit
b. Management Information Unit
c. Operating Systems & Research Unit
d. Criminal Justice Unit
e. Computer Operations Unit
f. Data Control Unit
g. Data Processing Unit

2.0 ON-LINE TELECOMMUNICATIONS UNIT

A Senior Systems Analyst will function as a supervisor in charge of the On-Line Telecommunications Unit as follows:

a. The conceptual design and programming of all applications related to FASTER transaction processing description which are primarily written for police usage or benefit.

b. Design and programming of all telecommunication actions and interfaces.

c. The control and maintenance of accurate documentation related to these systems.

d. Preparation of written instructions for use by field units in the proper use of the telecommunications system.

e. The availability of qualified technicians to service the telecommunications system as required 24 hours a day.

f. The accurate recording of job codes for authentic accountability of machine time.

g. Accurate utilization of Software Specialists according to established standards when efforts are primarily for police department benefit.

h. Maintenance of the ALERT System according to rules and procedures established by M.U.L.E.S.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
ORGANIZATION AND ENVIRONMENT
OF
COMPUTER SYSTEMS DIVISION

DATE ISSUED | DATE REVISED

3.0 MANAGEMENT INFORMATION UNIT

A Senior Systems Analyst serves as Supervisor in charge of all personnel assigned to the Unit, and answers directly to the Manager of Computer Facilities. The functions of the unit in general are outlined as follows:

a. The conceptual study, design, programming, implementation and maintenance of all applications related to management systems, criminal systems or any other systems assigned for development by the Computer Facilities Manager. Generally speaking, projects assigned to this Unit are for use of the Kansas City, Missouri Police Department and do not involve outside agencies, nor do they involve on-line telecommunications.

b. The development of these applications may include systems which interface through teleprocessing terminals. In such cases, the responsibility relates only to the Assembler language routines. The actual link into the system, priority of action by Main Task and the transmission of data to and from the system are the responsibility of the On-Line Telecommunications Unit.

c. The accurate documentation of all operational systems and programs assigned to the Unit.

d. The maintenance of all the Source Data Collection Systems (excluding the teleprocessing involved).

e. Handling of all special requests (for programs, printouts, modifications, etc.) that are generated by command personnel from within the Kansas City Police Department.

f. Maintenance of the LEMRAS and PLANTRAN systems.

g. Control and maintenance of Payroll programs.

h. Maintenance and control of Modus Operandi programs.

4.0 OPERATING SYSTEMS & RESEARCH UNIT

A Senior Systems Analyst, as the Unit Supervisor, is responsible for

a. Maintenance of S360/S370 Operating System and computer supervision.

b. Establishment of standards related to all facets of EDP Systems.

c. The development of procedures and forms related to computer operations.

d. The periodic review of documentation to insure conformance to established standards.

e. The conduction of periodic training classes related to new techniques or correction of malpractices in programming.

f. Establishment of programs for continued technical education.

g. Evaluation of EDP hardware and software.

h. Writing and publication of a Standards Manual.

i. Clearance of and documentation of all modifications to Faster monitor and other modifications to IBM software systems.

j. Specialized research into technical problems.

k. Prepare technical report write-ups describing problems and recommended solutions.

l. Review of all malfunction reports and publication of a monthly summary.

m. Inspection into areas to determine compliance with standards as may be directed by the Manager.

5.0 CRIMINAL JUSTICE UNIT

A Senior Systems Analyst will function as the Unit Supervisor, with the following responsibilities:

a. Supervision and control of assigned personnel on projects for which they were funded.

b. Coordination with all agencies in the region who are involved in the Criminal Justice Process, in the development of automated systems on the police computer.

c. The design and development of automated projects with the criminal justice family.

d. Designs, programs and implements automated systems for the Municipal Court and ASAP System.

e. The conceptual design and programming of applications primarily for courts, prosecution, and rehabilitation usage.

f. Utilization of Software Specialists according to established standards when efforts are primarily for non-police benefit.

6.0 COMPUTER OPERATIONS UNIT

Under direction of the Manager of Computer Facilities, the Computer Operations Supervisor is responsible for the efficient utilization of personnel and material resources.

Functions

a. Insure that all computer operators are trained to operate the S360 and S370 Components.

b. Insure that the computer room is manned by properly qualified operators throughout the 24-hour day - 7-day week period.

c. Direct the Operations Scheduler to insure that all disk-tapes are properly labeled, identified, and that the retention dates are posted on a weekly basis.

d. Direct the Operations Scheduler in the maintenance of the Tape/Disk Library.

e. Insure that all operators are trained to handle "Error Recovery Routines" and "Restart Procedures" for the Telecommunications System.

f. Insure that procedures are available to operators to promptly advise IBM Field Engineering of suspected malfunctions of computer equipment.

g. Direct the Operations Scheduler to Schedule workflow to facilitate production of EDP report runs and to coordinate test time with system personnel.

h. Advise Systems Personnel of program operations difficulty.

i. Insure that Operations personnel are thoroughly indoctrinated in all aspects of computer room security and fire protection procedures.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
ORGANIZATION AND ENVIRONMENT
OF
COMPUTER SYSTEMS DIVISION

DATE ISSUED | DATE REVISED

j. Insure that rules for 'No Smoking' and 'No Consuption of Liquids or Eating' are strictly enforced at all times, and that security precautions are complied with.

k. Insure that all computer operations time is accurately accounted for.

l. Coordinate flow of work between operator shifts to assure continuity of operations.

m. Insure that humidity/temperature requirements are within authorized limits and take precautionary safeguards when the limits are exceeded.

n. Maintains adequate supply of cards and paper.

INDEX NUMBER
02-6.0

7.0 DATA CONTROL UNIT

7.1 The Data Control Supervisor reports directly to the Manager of Computer Facilities, functioning as a working supervisor of this Unit, responsible for managing the personnel and material resources of the Unit as follows:

Function

a. Receives all source documents in which EDP or punched card applications are involved, reviews for accuracy and codes into proper format for key punching or "on-line" updating.

b. Maintains, in proper filing sequence, code sheets or copies of source documents for future reference as may be needed.

c. Develops necessary audit check list or documentation to be used by clerks in checking accuracy of source information, as well as reports produced by EDP or punched card application.

d. Advises programmer or analyst of any discrepancies noted on EDP output listings.

e. Researches and resolves rejected transactions from computer update runs and re-enters the correct transactions into subsequent update actions.

f. Coordinates with Warrant Control to insure agreement of warrants in the Real Time Files.

g. Works in close relationship with Systems personnel, to insure all actions which affect the EDP Reporting System are properly coordinated with interested agencies.

h. Must maintain thorough understanding of all reporting systems which apply to the Police Department.

i. Develops necessary written documentation to be used in checking input data and output products for accuracy and consistency.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
ORGANIZATION AND ENVIRONMENT
OF
COMPUTER SYSTEMS DIVISION

DATE ISSUED | DATE REVISED

j. Resolves transactions rejected by the computer and reintroduces the corrected transactions for final update actions.

k. Posts the computer ALERT number (CCN) to source code documents.

l. Files retention copy of all computer output runs.

m. Prepares update W/W transactions in the proper sequence for computer update runs.

n. Responsible for furnishing interpretation of technical instructions of the FBI (NCIC) computer systems to personnel of the Police Department and to Law Enforcement personnel in the Metropolitan Kansas City Area.

7.2 It is the policy of this division to channel the flow of data/reports through one control point to insure smooth and harmonious processing of Information and the prompt preparation of reports.

This is a full time task and the incumbent will not be assigned other coding duties unless all primary tasks outlined herein are completed.

a. Coordinates with the Operations Scheduler all changes in status of information contained in the Reports Distribution listing.

b. In coordination with the Operations Scheduler, insures the timely preparation of all reports produced by the Computer Systems Division.

c. Distributes all reports according to the official reports distribution list.

d. Performs basic review of reports listing for accuracy before release to requesting agency.

e. Maintains suspense on due date by agency to insure the timely receipt of all reports due in.

INDEX NUMBER
02-7.0

f. Receives reports, separates carbon, and arranges in proper sequence before release to the using agency.

g. Arranges for letters of transmittal which may accompany reports to using agency.

h. Arranges for addressing of envelopes for mailing of reports.

i. Maintains suspense on due dates and coordinates entry of data for reports to the Computer Operations Unit.

j. Sets up list of all reports due in and due out.

k. Reminds Computer Systems Division Units of due dates of data due in for processing, reports due out.

8.0 DATA PROCESSING UNIT

The supervisor of the Data Processing Unit will function as a working supervisor, reporting directly to the Manager of Computer facilities.

a. Maintenance of Data Processing punched card files as specified in Dentention-Records Division memorandum dated October 17,1963.

b. Receive source data from Data Control section and return to proper destination after keypunching or on-line entry action is completed.

c. Receive and process all voided traffic tickets. They are then entered on-line daily, and a transmittal is printed and copies distributed to the Traffic Unit and Municipal Court, for ticket accountability.

d. Key punch and verify all accident data for monthly accident statistical reports.

e. Key punch and verify cards for the Records Unit Civil Index file on accidents and miscellaneous reports.

f. Enter on-line traffic data for the maintenance of Municipal Court docket system.

g. Enter on-line parking tickets for daily transmittal of ticket accountability.

h. Enter on-line dispatcher incident cards for monthly workload report.

i. Enter on-line offense reports for statistical, location and civil index information. The statistical portion will create a file for all monthly and annual F.B.I. reports.

j. Enter on-line arrest, received daily from Data Control, These are entered as soon as possible after arrest, and source data is returned to Data Control.

k. Key punch and verify case assigment cards for Investigations Division for monthly detective workload report.

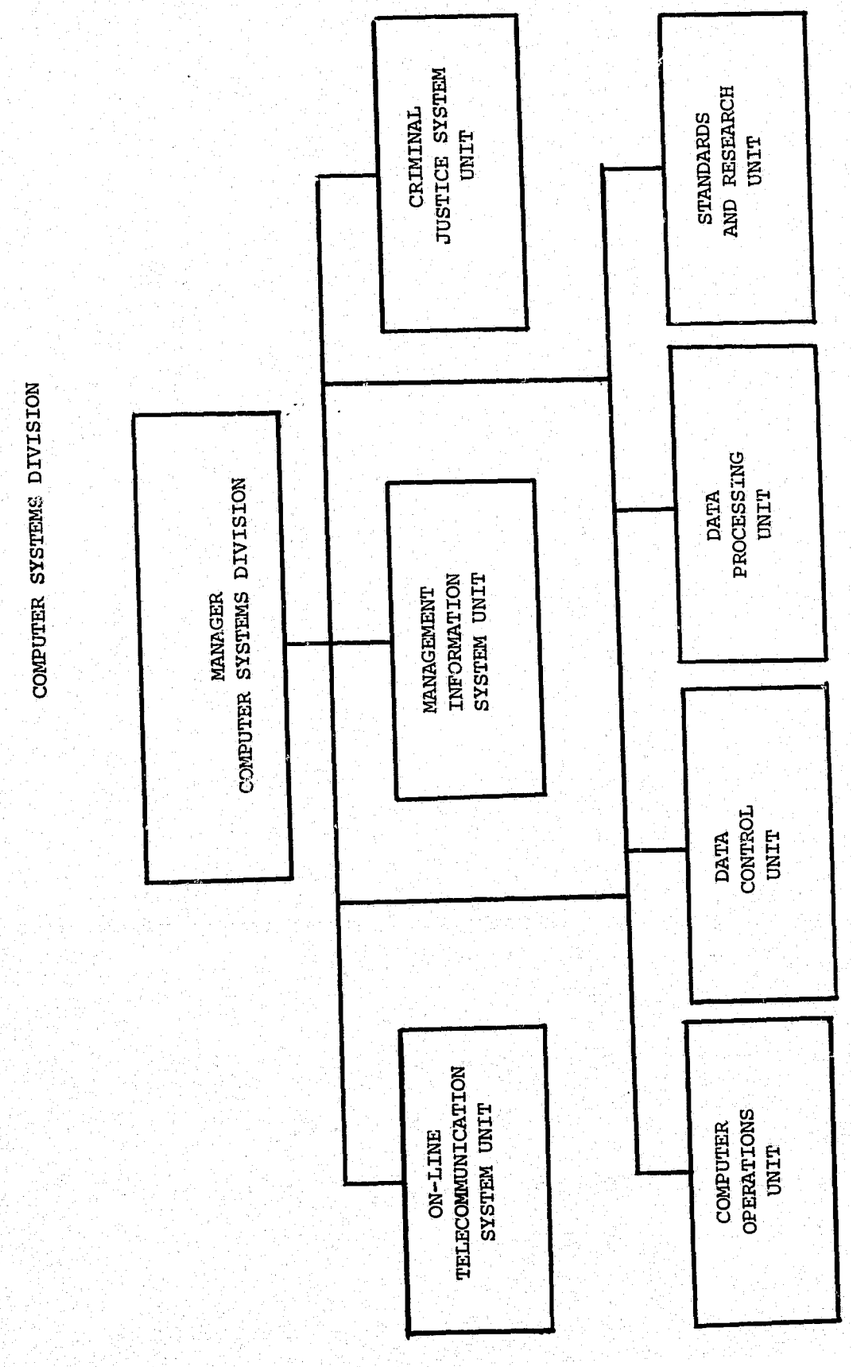
l. Cards punched and verified for Computer Systems Division personnel workload for monthly report to the Chief's office, computer programs card requirements for computer operations cards punched.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
ORGANIZATION AND ENVIRONMENT
OF
COMPUTER SYSTEMS DIVISION

DATE ISSUED | DATE REVISED



INDEX NUMBER
02-8.0

TABLE OF CONTENTS

03 SECURITY PROCEDURES AND STANDARDS

1.0 INTRODUCTION

Criminal Justice Information Centers are a prime target of revolutionaries. These revolutionaries and liberal civic groups are searching for an understanding of the capabilities of the computer in order to spot its vulnerability - not in order to interrupt it as a power supply but rather to compromise security by disguise, alteration of data, unauthorized use, access and dissemination of data to embarrass administrators and to present a false image to the public of computer operations.

Summarily, Criminal Justice Information Systems are vulnerable in three specific areas:

A. Overt attempts at actual sabotage of the computer complex.

B. Disguised attempts to alter on-line criminal files.

C. Disguised attempts to embarrass administrators by falsified incidents in the name of invasion of privacy and/or dissemination of data to those who do not have a 'need to know'.

As a consequence, the following policy statements with regard to agency operation security, agency system security and regional system security have been established for guidance and protection of data files and the public.

2.0 CENTER PERSONNEL

All personnel employed by the Criminal Justice Information Systems Center will be required to:

2.1 Take a Personality and Character battery of tests and to receive a grading which indicates that they:

a. Have high integrity.

b. Will not easily divulge confidential information.

c. Are extremely concerned about the propriety of information.

d. Are orderly, organized and will file reports in their proper places as a matter of behavior (not inclined to leave material laying around).

2.2 Repeat the test each year.

2.3 Sign a document in which they agree to take a Polygraph test at any time that Police Management request.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SECURITY PROCEDURES AND STANDARDS

DATE ISSUED | DATE REVISED

3.0 SAFEGUARDING AND CONTROLLING THE CRIMINAL JUSTICE INFORMATION SYSTEM

3.1 Criminal Justice Systems Operations

Criminal Justice Information is of a nature which reflects an individual's alleged violation against the rules established for an orderly society. A final determination of guilt or innocence is the responsibility of the courts. The final determination of the courts then becomes a matter of public record. The arrest information is a matter of record and for use only within law enforcement and criminal justice agencies involved in the criminal justice process. Information concerning intelligence subjects, outstanding warrants, parole status, arrests and convictions are stored in the computer real time files. In order to provide the necessary safeguards concerning the dissemination of this sensitive criminal justice information the following policies are established:

A. Personnel of the regional computer system's division will not disseminate criminal justice information. This is the responsibility of the line and auxiliary support agencies.

B. Only personnel acting in an official department capacity will be allowed to operate terminal devices.

C. Personnel operating the terminals will insure that only "test" records are reviewed by visitors to the Department.

D. When an employee is in doubt as to whether specific information should be released or not he should not release it but rather consult with his immediate superior before action is taken. It shall be the policy of ALERT that no information shall be entered in the on-line files unless a source document is received from one of the agencies of the department which authorizes the entry of information or if that information is entered by the agency itself.

3.2 It shall be the policy of the KCMOPD that no information shall be entered in the on-line files unless a source document is received from one of the agencies of this Department which authorizes the entry of the information.

In the case of organized crime or militant groups, that information can be accepted only from the Organized Crime Unit. Data changes received by phone are not to be accepted for entry into the computer. Either a written document or receipt of a message over the ALERT network is required before any of this category of information is updated on-line in the future.

3.3 The Computer System Division prints out a considerable amount of data containing information, which, by law and by Department policy, may only be released to persons acting in legitimate law enforcement capacity.

The following policies are to be followed in preparation of such programs:

a. Printouts containing Warrant/Pickup, Arrest/Parole and Intelligence Subject data to contain a statement on the top and bottom lines of each page,

"Restricted Data - For KCMOPD or for Metropolitan Law Enforcement Official Use Only."

b. Periodic Printout of Warrants for Municipal Court,

"Restricted Data - For Municipal Court and KCMOPD Use Only."

c. Printouts to be discarded must be delivered to the Building Maintenance Unit, where such matter will be destroyed periodically.

3.4 Kansas City Regional Law Enforcement Telecommunications Network

a. Information exchanged over the network involves official FBI and other privileged area law enforcement information. This information must be processed and safeguarded in such manner that only personnel on official law enforcement duties may have access to such information.

b. Information contained in the Computer is only an abstract of a subject's record; therefore, the contents must be used only for legitimate law enforcement or agencies involved in the Criminal Justice Process operations. Request for record checks by any other agency or private concern must be made against official police documents and records.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SECURITY PROCEDURES
AND
STANDARDS

DATE ISSUED | DATE REVISED

c. Computer printouts produced for other agencies involved in the Criminal Justice Process will contain the heading - Restricted Information - For (Name of Criminal Justice Agency).

d. Data security is a most vital element of our operations. No other responsibility is as great as that of exercising proper management control over the entire spectrum of security and confidentiality of automated information systems. Each employee of the Computer Systems Division must be cognizant of the great responsibility we all bear with regard to maintaining the quality and confidentiality of computer-based information systems.

INDEX NUMBER
03-3.0

4.0 SECURITY - COMPUTER COMPLEX

4.1 The purpose of this procedure is to implement security control of the computer room complex until Department General Orders can be implemented. The use of common sense, good judgment, courtesy, and firmness should apply at all times.

a. All visitors (not members of this department) will be required to sign the Security Register after identifying themselves. These personnel are to be escorted by Division personnel at all times.

b. There are some exceptions. Local IBM representatives who are well known and Municipal Court personnel who have daily business in Data Control are to be authorized in without signing in.

c. During non-duty hours, 4:00 p.m. - 7:00 a.m., Monday through Friday, all holidays and weekends, all personnel not assigned the Computer Systems Division will sign the Security Register after establishing identification and reason on the fourth floor.

d. Visitors not previously announced will not be allowed in during non-duty hours unless a clearance slip has been previously given by the Division Manager.

e. Operations personnel are held responsible for determination of admittance or non-admittance during non-duty hours, meeting Security Log requirements and escort when authorized in the complex.

f. The Security Log maintained at the Receptionist's desk will be used during normal duty hours. The Security Log maintained in Data Control will be used during non-duty hours.

g. Under no circumstances will visitors be allowed to carry briefcases or suitcases through the computer room complex. Inspection of briefcases or packages brought to the fourth floor is authorized when suspicion is involved. If refused, visitor may be asked to leave the floor.

h. Division Manager, or in his absence, the duty officer in the Communications Center will be notified if difficulties are encountered in complying with these directives during non-duty hours.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SECURITY PROCEDURES AND STANDARDS

DATE ISSUED | DATE REVISED

i. Computer Operations Center duty personnel are to refuse entry to the computer room to all persons except the following:

Computer Systems Division personnel.

Persons escorted by Computer Systems Division personnel.

Persons escorted by Chief of Police, Assistant Chief of Police, or Operations Bureau Commander.

IBM Engineers personnally known to the operators, or those who can properly identify themselves as having official reason for being in the computer room.

j. All Division personnel are issued keys for entry to the fourth floor. Under no circumstances should Division personnel allow anyone to enter with themselves unless the party is clearly known to have business in this area.

k. The Senior Computer Operator on duty will perform the following duties at 5:15 p.m. each work day during the week.

Insure the ladies' and men's rest rooms are searched for any objects which appear to be left in a suitcase, satchel, package or sack container. The waste paper container should also be checked.

Male and female members may be asked to perform this task in respective rest rooms. The Duty Operator will be advised when this action is completed, if performed by another person.

The Duty Operator will check to insure the Public Information Office, Data Control and Data Systems Division doors are locked at all times other than during normal week working days. The Stairway Door will also be checked for locked condition at 5:15 p.m., even though it should be locked at all times. Keys are provided for all doors on the 4th floor.

INDEX NUMBER
03-4.0

If any strange object referred to above is observed, Duty Operations should immediately phone the Communications Center, Ext. 280. In any such eventuality, all personnel should immediately move to the most distant area from the object when calling the Dispatch Center.

Computer Operators are to be especially alert to any object left unattended in the 4th floor hallway at all times.

Under no circumstances, should you attempt to tamper with or try to identify the object once, in your judgement, it appears to be a suspicious or foreigh object.

l. The Duty Operator will sign a daily log to be maintained in the Division Manager's office, certifying that all actions outlined in this Section are completed.

m. This procedure is a precautionary measure only. Be alert but no one need be alarmed. Many area buildings have implemented similar policies as a result of the times we now live in.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SECURITY PROCEDURES
AND
STANDARDS

DATE ISSUED | DATE REVISED

5.0 FIRE PROCEDURES

5.1 It is imperative to establish a standard procedure in case of fire, to insure protection of personnel and equipment. The following is the set procedure under the fire protection plan.

a. Determination

If fire can be determined to be minor and controllable use Step b.

If fire is identifiable as either uncontrollable or near uncontrollable follow Step c.

b. Action to be taken: (minor or controllable fire)

Power off system at CPU.

Power off system at main circuit panel in computer room.

Determine if fire can be put out by CO^2 extinguisher.

Use CO^2 extinguisher if necessary.

Notify dispatchers with cause and action taken in fire alarm.

Notify Fire Department if significant damage; to inspect cause and estimate damage.

Notify Computer Supervisor, Mr. Bockelman.

c. Action to be taken: (Major fire)

Power off system at CPU.

Power off system at main circuit panel located in computer room.

Depress Local fire alarm located next to A/C control panel.

Evacuate computer room and adjoining areas.

Use stairwell if accessible.

Use fire escape East side fourth floor
if stairwell inaccessible.

5.2 The FIRE-ALERT Ionization Smoke/Fire Detection System consists of the following units:

Eight ceiling mounted detectors

Four under floor mounted detectors

Three return air duct mounted detectors

One alarm panel mounted on South pillar

One manual alarm pull switch mounted on South pillar

One alarm bell mounted on South pillar

One alarm light mounted in 1st floor Dispatcher's office of Communications Unit.

The basic principle of this system is that at the very beginning of any fire, minute particles are released into the air. These particles are present long before visible smoke or high temperature associated with combustion. The ionization detector is specifically designed to sense these particles and sound an alarm before life or property are needlessly damaged or destroyed.

Annual cleaning of the collector plate is required by the operator to remove dust particles. If residue adheres, remove collector plate (2 screws); clean collector plate and concealed radium tip detector element with alcohol coated swab.

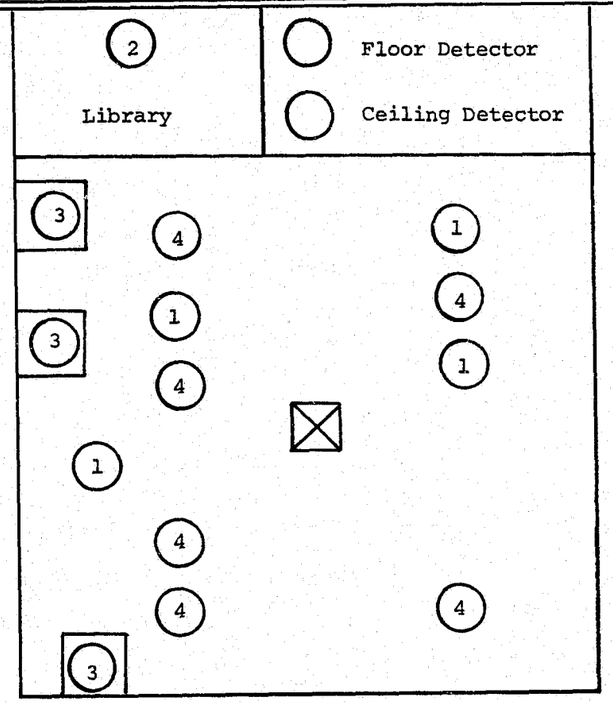
Service instructions and wiring diagrams are stored inside the alarm panel cabinet.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SECURITY PROCEDURES
AND
STANDARDS

DATE ISSUED | DATE REVISED



NOTES:

1. Each detector contains a neon lamp visual indicator.
2. The trouble light gives indication of failure in the system. An internal buzzer sounds in the monitor panel when the trouble light indicates detection of an internal system failure.
3. The normal-silence switch turns off the internal buzzer.
4. The reset switch turns off the fire alarms in the communications center and the computer room above the monitor panel and the visual alarm indicator.
5. Enter details in log each time the alarm sounds.
6. Cycle power off on computer in event of possible equipment damage.
7. Use portable extinguishers and/or call FIRE DEPARTMENT as circumstances dictate.

INDEX NUMBER
03-5.0

6.0 LOSS OF POWER

The purpose of this procedure is to serve as a guideline to duty personnel in determining proper procedures for loss of power and/or in case of fire.

In most instances power loss is due to thunderstorms or transformer failures. Past experience has shown this loss to be limited to only a few seconds.

a. Indications

In most instances CPU alarm checks will be displayed along with one or more peripherals not functioning.

b. Action to be taken

In most cases resetting the circuit breaker on the individual unit, then resetting the circuit breaker on the control unit will place the equipment back on-line. Most circuit breakers are readily accessible and are simple on and off switches labeled: CB1, CB2, etc.

Following is a list of all I/O peripherals with respective control unit.

Control Unit	I/O
2841	2311
2314	2314
2821	2540, 1403
2803	2401
2703	2740
2848-II	2260-I
2848-III	2260-II

If resetting both circuit breakers is unsuccessful then try resetting the circuit breakers on the main power control box located on the north wall, FE Room. If this action does not have any effect, contact: Operations Supervisor.

TABLE OF CONTENTS

04 FINAL DOCUMENTATION STANDARDS

1.0 BASIC CONCEPTS

Documentation is the device which is used to facilitate communication between those individuals responsible for the planning, implementation and maintenance of applications to be processed on EDP equipment. Standards are necessary if all persons involved in an application are to communicate easily and effectively. In effect, the standards define a common language understood by all parties involved.

The idea that documentation is a separate task to be accomplished in the last phase of application implementation seems to have gained broad acceptance. Much of the value of documentation is lost if it is done "after the fact". The proper approach to documentation is to make it a part of every phase of the application, from the initial planning to the final implementation, and to conscientiously carry it over into subsequent maintenance procedures.

Each data processing application represents a capital investment on the part of the Police Department, and it is the responsibility of each of us to see that this investment is adequately protected with good documentation.

2.0 THE AUDIENCE AND THEIR NEEDS

Listed below are four groups requiring documentation and some of the reasons why it is needed.

2.1 Supervision

a. To provide a record of how we spend our time and the Police Department's money.

b. To allow removal or addition of personnel with minimum detrimental effect on the project.

c. To monitor the progress of an application and evaluate the effectiveness of the individuals involved.

2.2 Other Programmers

a. To provide language independent backup to the program coding.

b. To allow the program logic to be studied at the block diagram level.

c. To allow program modification to be accomplished with a minimum of time.

d. To determine if a program can be easily modified to handle a similar application.

2.3 Operations

a. To run a program without having to consult with programmers.

b. To better schedule the use of data processing equipment.

2.4 Customer

a. To allow customer to take a more active role in checkout.

b. To provide wider dissemination of information concerning program capabilities.

c. To provide information concerning the system inputs and outputs.

DATA PROCESSING

MANUAL OF STANDARDS

3.0 RESPONSIBILITIES AND PROCEDURES

Although the documentation of systems is the responsibility of each individual, the ultimate responsibility for adequate overall documentation rests with Programming Supervision.

It is recognized that the amount of detail required in documentation is a function of the complexity of the system. For example, some programs might require only a Detailed Block Program. Other programs will be so complex as to require a General as well as a Detailed Block Diagram. Supervision must then require documentation consistent with the degree of complexity of each individual program.

Since documentation is primarily a communication tool, its effectiveness is directly proportional to the degree of standardization obtained. Therefore, it is the responsibility of each individual to implement the standards described in this manual.

A data processing system may be defined as being a collection of operations and procedures required to accomplish a specific data processing objective. These operations and procedures involve many data processing activities:

Keypunch
Processing
Equipment Scheduling
Clerical Editing, Balancing, and Salvaging
Distribution of Outputs

System documentation is concerned with factors that affect more than one of these activities. Program documentation is concerned with factors that affect only one program.

Documentation of applications will consist of the standard set of items described in Sample 5.1. Standard techniques to be used in the preparation of these items are described in this Section.

When the programming task has been completed, the documentation is assembled in the order shown in Sample 5.1 and presented to Programming Supervision for approval.

4.0 SYSTEM DOCUMENTATION

4.1 System Title Page

Enter the following information:

System Name
User Department and job number(s)
Names and titles of all personnel involved in the development of the system
Date Completed
Date Operational
Approval of Data Systems Manager

An example of the same number illustrates this system title page.

4.2 Purpose of System: Give a brief description of the objectives of the system. Explain briefly the interaction of the programs within the system. An example of the same number illustrates the purpose write-up.

4.3 Letter(s) of Definition: Include all letters and/or agreements that set forth task specifications and commitments.

4.4 System Flow Chart: This chart will reflect the flow of system inputs through all steps taken in the Data Systems Division to process the data. An example of the same number shows the type of detail that should be included in a system flow chart.

4.5 Source Documents: Include samples of all source documents from which system inputs are derived.

4.6 Reports: Include sample copies of the first and the last page of each report generated by the system. Also prepare a description of computer report form to describe the report elements completely.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
FINAL DOCUMENTATION STANDARDS

DATE ISSUED | DATE REVISED

4.7 Input/Output Layouts:

a. All system inputs and outputs will be formatted on the appropriate form. All fields will be labeled. Field names on layouts should be the same as those used in the diagrams, coding, and glossary.

b. The following layouts will be entered where applicable:

Card - for each type of card input/output
Tape - for each type of tape record
Report - for each report
Disk - for each type of disk record
CRT - for each type video display (inquiry or mask)

Examples of these layouts are included following this section.

4.8 Scheduling and Control Data:

a. List all programs in the order that they should be run.

b. List all source inputs with dates of their expected arrival in this department. If the task is recurring, enter the dates expected during each reporting period. Enter expected volume of each source input.

c. List all report names with the date each is expected by the user. If the task is recurring, enter the date expected during each reporting period.

4.9 Glossary: List and define all system parameters that are not defined where used. This includes all constants, areas, file names, record names, and all input/output fields shown on layouts.

INDEX NUMBER
04-4.0

EXAMPLE 4.1

EXAMPLE OF SYSTEMS DOCUMENTATION

SECTION	
(SYSTEM NAME)	
DATE ISSUED	DATE REVISED
(DATE OPERATIONAL	

SYSTEM TITLE PAGE

Names and titles of personnel involved during development:

- (Systems Analyst)
- (Systems Supervisor)
- (Using Department Liason)
- Etc.
- Etc.

DATE SYSTEM COMPLETED: _____

APPROVAL: (DATA SYSTEMS MANAGER) _____

(USER DEPARTMENT)

INDEX NUMBER
(JOB NO.)

OFFENSE REPORTING SYSTEM

The Offense Reporting System was developed to provide timely, statistical information concerning all offenses reported to police within the city limits of Kansas City, and in the cities and counties in the Region covered by the ALERT System. Information is provided the user departments in the form of standard periodic computer-generated reports, and also upon reasonable request for specific offense information to be generated by the computer. The regularly scheduled offense statistical reports and FBI Uniform Crime Reports are prepared on a weekly, monthly or annual basis. All other reports are produced on an as required or as requested basis.

The input data for the offense system is collected from standard police department reports that are filled out by police officers as a result of a reported offense. The reports used by the Kansas City, Missouri Police Department fall into four classifications:

A. Offense Report (P.D. Form 189).

B. Crime Against Person Report (P.D. Form 339).

C. Motor Vehicle Report (P.D. Form 280).

D. Supplementary Report (P.D. Form 184).

After completion the above reports are checked by the reporting officers' immediate supervisor and if approved are forwarded to the Kansas City Police Department Report Review Section where the reports are checked for completeness and final classification. The approved reports are then forwarded to the Data Control Unit for final coding of the data for on-line entry. Some of the statistical information is coded on a form titled Computer Data Entry Sheet (P.D. Form 110) to facilitate easier entry by the on-line terminal operators. After complete coding the forms are sent to the Data Processing Unit for on-line entry through Remote CRT Terminals. The data being entered is subject to on-line primary edits and any errors encountered are returned to the CRT Screen in the form of asterisks. The operator may then make the corrections and re-enter the information.

A required field in any offense entry is the address of occurrence, and during the on-line entry of offense information the data is passed through a census tract and block lookup. This is performed by loading into a key the street number and street name of the address of occurrence and reading an on-line file containing the census tract and block code corresponding to this address. This information is added to the offense data already entered and then all of the data is formatted into the necessary records

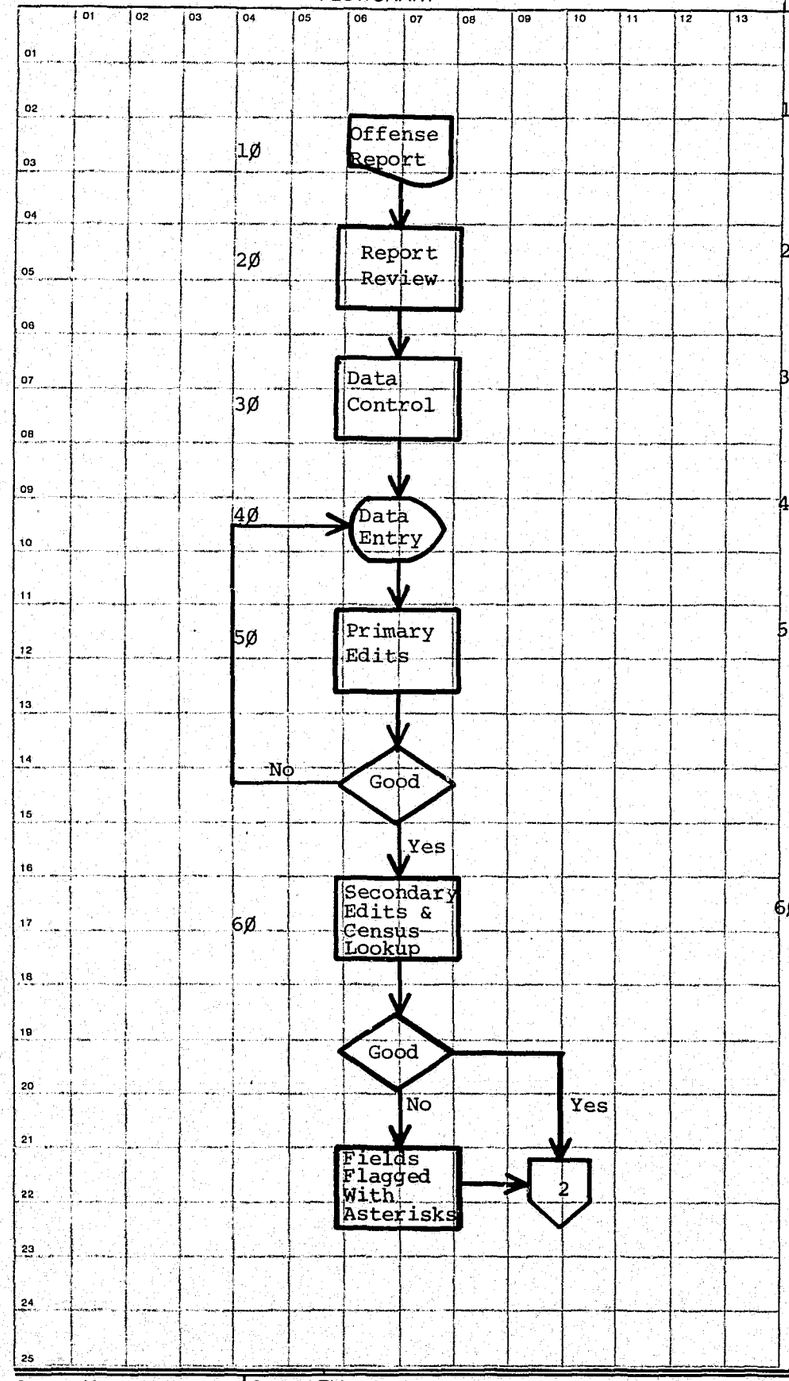
which are subsequently written on the master file, general index file, and name index file.

At the end of each month, a program is run that reads the general index and master files and creates a monthly offense tape which is used as input to most of the offense report programs. The monthly tape provides a comprehensive list of statistics on the victim's name and address, the particulars of how the offense occurred, stolen and recovered property information, location of offense information, and various identifying codes such as the originating agency, case report number, report date, officer's serial number, etc. After the monthly report programs have been run, the monthly tape is merged with the prior month's year-to-date tape to create an updated year-to-date offense tape. This tape has the same record format as the monthly tape and is used as input to several annual offense reports. The year-to-date offense tape is kept indefinitely as a permanent history file.

SYSTEM FLOWCHART

SYSTEM FLOW CHART — Page 1 of 4

FLOWCHART



DESCRIPTION

1Ø An offense is reported to Police and an offense report is written describing the details of the offense.

2Ø Offense report is routed to the Report Review Section for review and final classification.

3Ø Offense report is routed to the Data Control Unit for coding of the data for on-line entry.

4Ø Offense report and code sheet routed to the Data Processing Unit for on-line data entry.

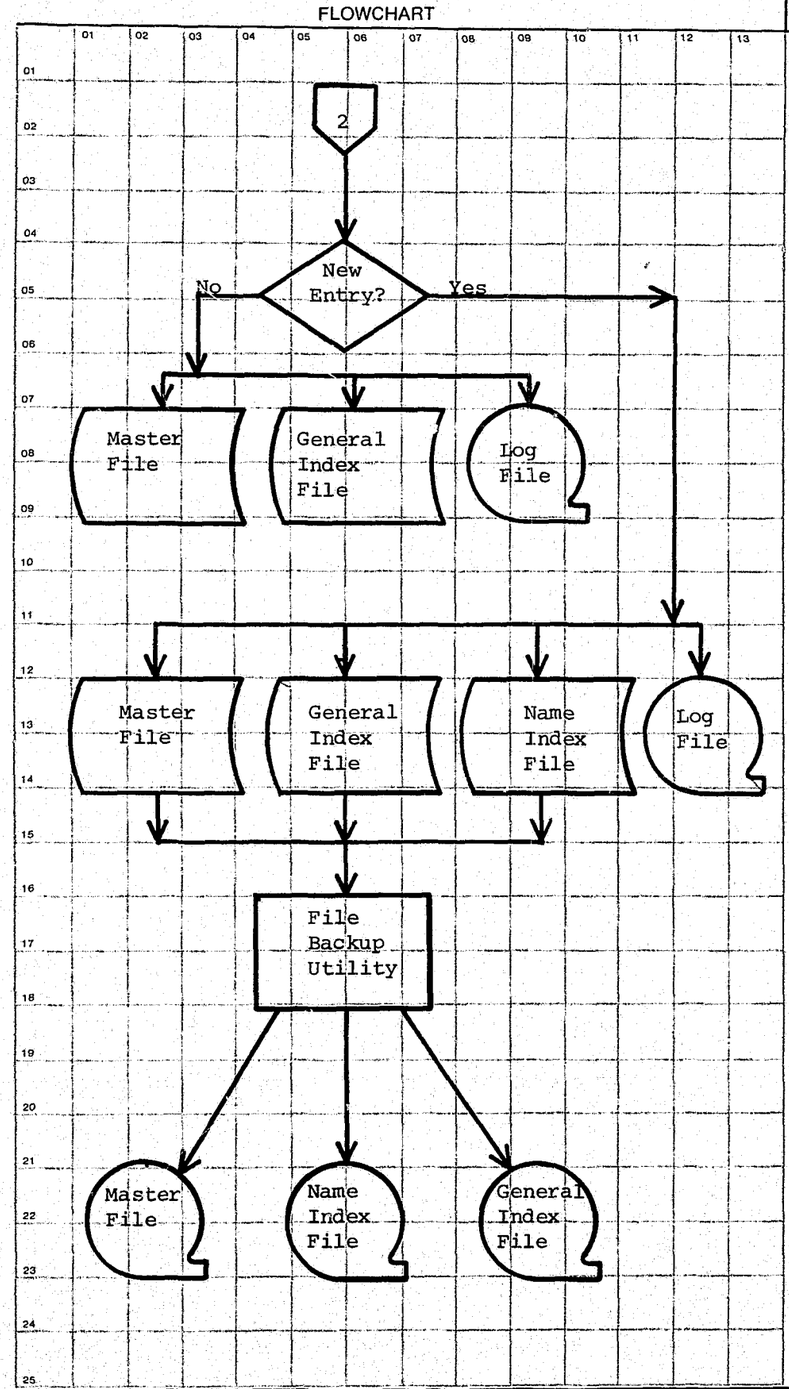
5Ø Data entries are passed through primary edits. Rejects are re-displayed on 226Ø for correction.

6Ø Data entries are passed through secondary edits and census tract/block lookup. Errors are flagged in the record for logging. Data is passed on for file update.

System No.	System Title: OFFENSE REPORTING SYSTEM		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

SYSTEM FLOW CHART — Page 2 of 4

FLOWCHART



DESCRIPTION

7Ø One of the required data fields on any offense entry is the name of the victim (persons or business). At this point the computer makes a determination as to whether the victim of this offense has been previously entered into the computer files or rather that this is an entirely new entry.

75 If the offense being processed concerns a previously entered victim, new offense information is added to the Master File and the General Index File. The transaction is also entered on the Log File.

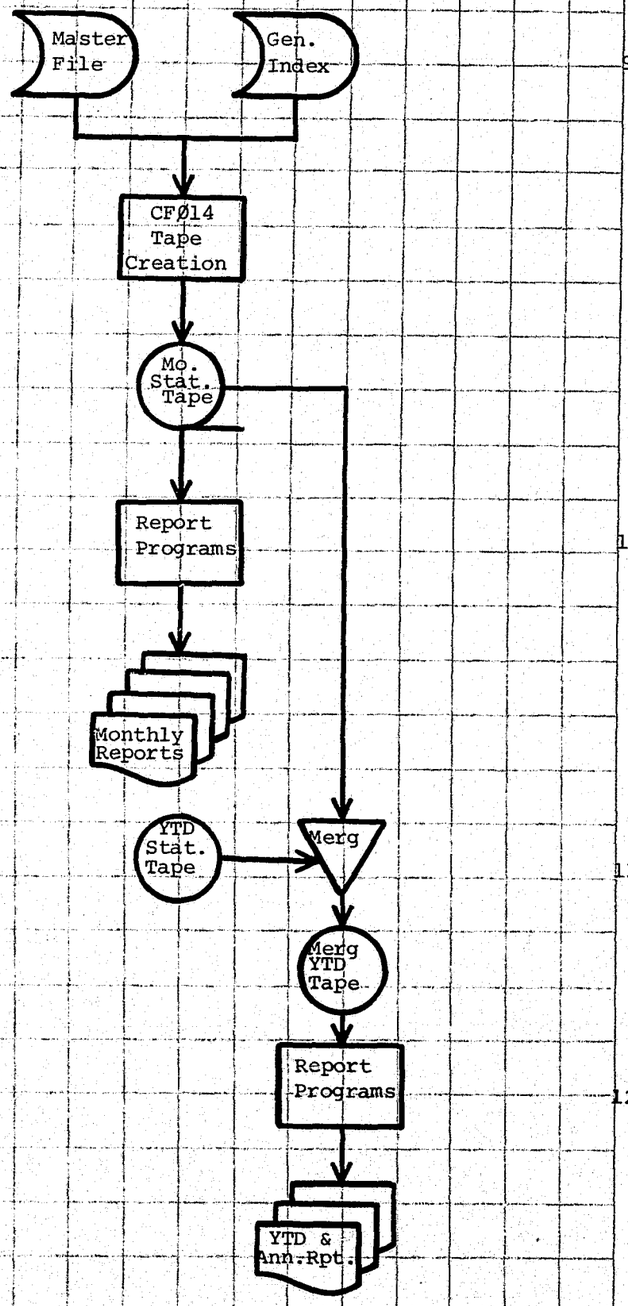
80 If the entry being processed is entirely new, information is added to the Master File, General Index File and the Name Index File. The transaction is entered on the Log File.

85 For security purposes as well as input to other programs, the Master File, General Index File, and Name Index File are "backed up" on tape.

System No.	System Title: OFFENSE REPORTING SYSTEM		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

SYSTEM FLOW CHART — Page 3 of 4

FLOWCHART



DESCRIPTION

9Ø Monthly statistical tape created.

1ØØ Monthly reports produced.

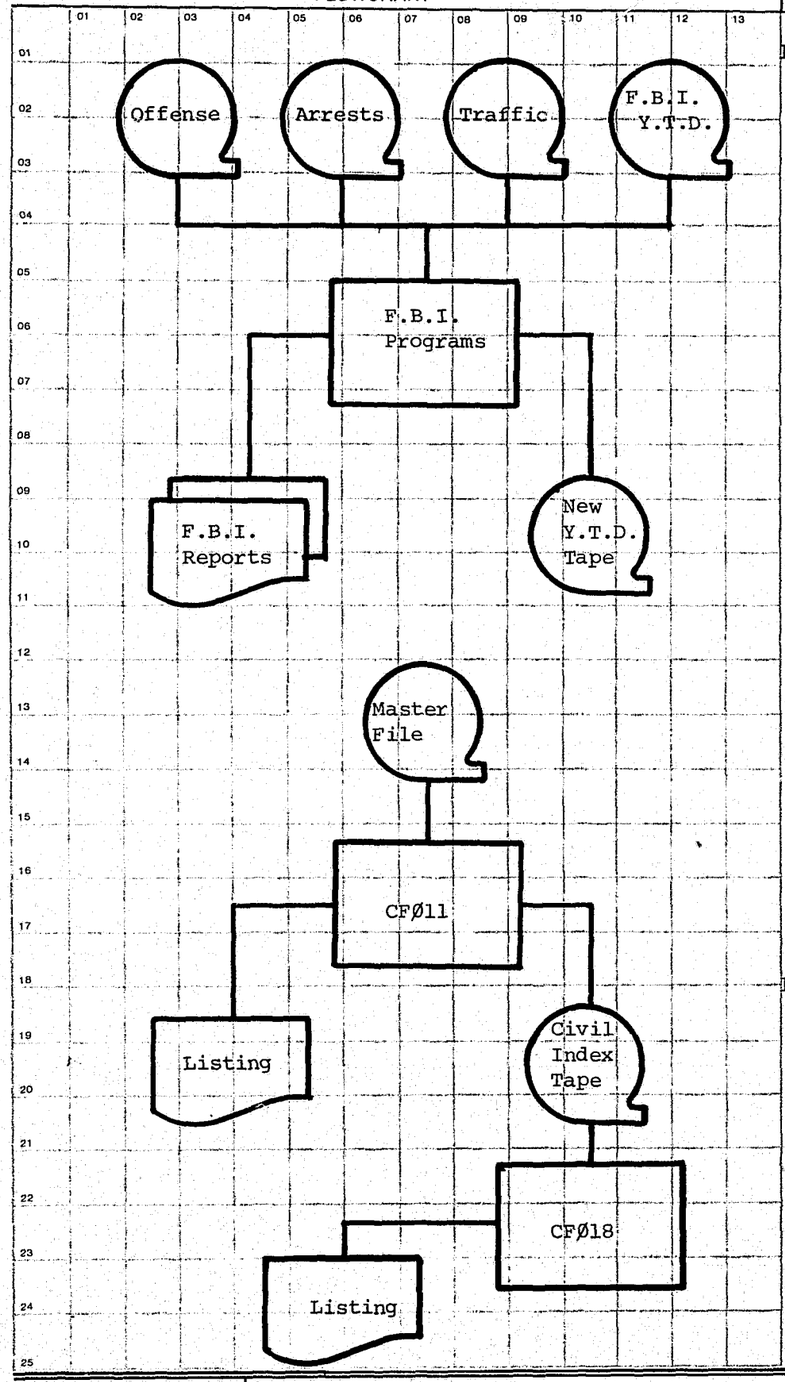
11Ø Monthly and YTD tapes merged.

12Ø YTD and annual reports produced.

System No.	System Title: OFFENSE REPORTING SYSTEM		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

SYSTEM FLOW CHART — Page 4 of 4

FLOWCHART



DESCRIPTION

13Ø Although the majority of the offense report program use the tape created by CFØ14, there are four programs that use other input sources. CFØØØ and CFØØ1 combine to produce the monthly and the year-to-date FBI Report, and use as their input source four tapes; monthly offense tape, monthly arrest tape, monthly traffic ticket tape, and the year-to-date FBI tape. The output of the FBI programs consists of the FBI reports and a new year-to-date FBI tape.

14Ø The other two programs that deviate from the normal input source are: CFØ11 and CFØ18. CFØ11 uses as input the backup tape for the Master File and produces a listing and a Civil Index Tape of selected offenses. This tape is used as input to CFØ18 which creates a second listing.

System No.	System Title: OFFENSE REPORTING SYSTEM		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

SECTION EXAMPLE 4.1

DATE ISSUED	DATE REVISED

SYSTEM SOURCE DOCUMENTS

FORM 180 (REV. 6-72) KANSAS CITY, MISSOURI POLICE DEPARTMENT

APPROVED BY _____

OFFENSE REPORT

CASE REPORT # _____

REPORT REFERRAL FORM ISSUED ☐ YES ☐ NO DATE _____ TIME _____

OFFENSE _____ PLACE OF OCC _____

DATE & OF OCC _____ BEAT OF OCC _____

IF OFFENSE COMMITTED IS LISTED BELOW, MARK APPROPRIATE BOX(ES):

☐ ROBBERY OR ATTEMPT ☐ BURGLARY OR ATTEMPT ☐ RESIDENCE ☐ NON-RESIDENCE

☐ ARMED-ANY WEAPON ☐ FORCIBLE ENTRY ☐ DAY ☐ DAY

☐ STRONGARM-NO WEAPON ☐ UNLAWFUL ENTRY-NO FORCE ☐ NIGHT ☐ NIGHT

☐ ATT. FORCIBLE ENTRY ☐ UNKNOWN ☐ UNKNOWN

SCENE PROCESSED ☐ YES ☐ NO

PROCESSED BY ☐ EVIDENCE TECHNICIAN ☐ REPORTING OFFICER

LATENT PRINTS ☐ YES ☐ NO

OTHER ITEMS OF EVIDENCE ☐ YES ☐ NO

VICTIM (FIRM) _____ TYPE OF BUSINESS _____

ADDRESS _____ PHONE _____

STREET # | DIRECTION | STREET NAME AND CLASSIFICATION | ROOM #

VICTIM

LAST NAME	FIRST NAME	INIT.	JR/SR	RACE	SEX	DATE OF BIRTH	STATE OF BIRTH

HEIGHT	WEIGHT	EYES	HAIR	SOCIAL SECURITY NUMBER	STREET #	DIRECTION	STREET NAME AND CLASSIFICATION	APT.#

CITY	STATE	ZIP CODE	PHONE	BUSINESS/SCHOOL ADDRESS	BUSINESS PHONE

PERSON DISCOVERING/ REPORTING

LAST NAME	FIRST NAME	INIT.	JR/SR	RACE	SEX	DATE OF BIRTH	STATE OF BIRTH

HEIGHT	WEIGHT	EYES	HAIR	SOCIAL SECURITY NUMBER	STREET #	DIRECTION	STREET NAME AND CLASSIFICATION	APT.#

CITY	STATE	ZIP CODE	PHONE	BUSINESS/SCHOOL ADDRESS	BUSINESS PHONE

RELATIONSHIP

HOW ATTACKED (HOW ENTERED-FROM WHERE STOLEN) _____

MEANS OF ATTACK (DESCRIPTION OF WEAPON OR TOOLS USED) _____

DESCRIPTION OF VEHICLE USED _____ VEHICLE TOWED ☐ NO ☐ YES TOW-IN # _____

QUANTITY	DESCRIPTION OF PROPERTY	SERIAL NUMBER	ESTIMATED VALUE

RECOVERED PROPERTY YES ☐ NO ☐	ESTIMATED VALUE OF →	CURRENCY	JEWELRY	FURS	CLOTHING	MISCELLANEOUS	TOTAL

SUSPECT 1

LAST NAME	FIRST NAME	INIT.	JR/SR	RACE	SEX	EST. AGE/D.O.B.	MONIKER

HEIGHT	WEIGHT	BUILD	HAIR	EYES	COMPLEXION	HAT	SHIRT	TIE	COAT	TROUSERS

	YES	NO
IN CUSTODY		
PICK UP BROADCAST		

OTHER CLOTHING	GLASSES	JEWELRY	OTHER NOTICEABLE CHARACTERISTICS (SPEECH, TATTOOS, SCARS, LIMP, ETC.)

ADDRESS	APT.#	CITY	STATE	BUSINESS/SCHOOL ADDRESS

SUSPECT 2

LAST NAME	FIRST NAME	INIT.	JR/SR	RACE	SEX	EST. AGE/D.O.B.	MONIKER

HEIGHT	WEIGHT	BUILD	HAIR	EYES	COMPLEXION	HAT	SHIRT	TIE	COAT	TROUSERS

	YES	NO
IN CUSTODY		
PICK UP BROADCAST		

OTHER CLOTHING	GLASSES	JEWELRY	OTHER NOTICEABLE CHARACTERISTICS (SPEECH, TATTOOS, SCARS, LIMP, ETC.)

ADDRESS	APT.#	CITY	STATE	BUSINESS/SCHOOL ADDRESS

VICTIM ☐ WITNESS ☐

LAST NAME	FIRST NAME	INIT.	JR/SR	RACE	SEX	DATE OF BIRTH	STATE OF BIRTH

HEIGHT	WEIGHT	EYES	HAIR	SOCIAL SECURITY NUMBER	STREET #	DIRECTION	STREET NAME AND CLASSIFICATION	APT.#

CITY	STATE	ZIP CODE	PHONE	BUSINESS/SCHOOL ADDRESS	BUSINESS PHONE

☐ CLEARED BY ARREST OF: _____ COLOR _____ SEX _____ DOB _____ G.O.S./JUV. RPT. # _____

☐ EXCEPT CLEARED: NO PROSECUTION _____ OTHER(SPECIFY) _____

REPORTING OFFICERS: _____ SERIAL(S) _____ BEAT REPORTING _____

CASE NO._____

KANSAS CITY MISSOURI POLICE DEPARTMENT
UNFOUNDED REPORT

_____ SUPERVISOR APPROVAL

_____ REPORTING OFFICER'S NAME & SERIAL NO.

CLASSIFICATION OF INCIDENT AS ORIGINALLY RECEIVED OR DISPATCHED.

☐ HOMICIDE	☐ ROBBERY	☐ VEHICULAR
☐ RAPE	☐ BURGLARY	☐ RECOVERED PROPERTY
☐ ASSAULTS	☐ LARCENY	☐ BICYCLE
☐ ANIMAL BITE	☐ STOLEN AUTO	☐ OTHER _____

NAME OF COMPLAINANT_____

COMPLAINANT'S ADDRESS_____ PHONE NO._____

NAME OF PERSON CONTACTED IF NOT COMPLAINANT_____

ADDRESS_____ PHONE NO._____

LOCATION DISPATCHED TO_____

DATE:_____ TIME: _____

☐ INVESTIGATION PROVES THE CRIME/INCIDENT DID NOT HAPPEN OR WAS NOT ATTEMPTED. (STATE BRIEFLY IN NARRATIVE SECTION THE CIRCUMSTANCES TO JUSTIFY THIS CONCLUSION.)

☐ UNABLE TO LOCATE CRIME OR INCIDENT THAT REQUIRES REPORT.

☐ UNABLE TO LOCATE ANY PERSON WHO CALLED POLICE.

☐ CRIME OR INCIDENT OCCURRED IN ANOTHER JURISDICTION. (STATE BRIEFLY IN NARRATIVE SECTION WHAT THE COMPLAINANT WAS ADVISED TO DO, AND IF OUTSIDE AGENCY NOTIFIED, THE NAME, RANK/TITLE, AND OUTSIDE ORGANIZATION THAT WAS CONTACTED.)

☐ ANOTHER CASE NUMBER HAS ALREADY BEEN ISSUED FOR THIS CRIME OR INCIDENT. THE ORIGINAL CASE NUMBER WAS_____.

NARRATIVE/COMMENTS_____

_____

_____

_____

_____

COMPLAINANT ACKNOWLEDGES COMPLAINT IS UNFOUNDED _____
SIGNATURE

FORM 229 P. D. (6-69)

EXAMPLE 4.1

APPROVED BY

_____

TYPED BY_____

SUPPLEMENTARY REPORT
POLICE DEPARTMENT
KANSAS CITY MISSOURI

COMPLAINT #_____

OFFICERS REPORTING

_____

_____

CONCERNING:_____ TYPE OF ORIGINAL REPORT | DISTRICT REPORTING | DATE OF THIS REPORT

DATE AND TIME OF OCCURRENCE | PLACE OF OCCURRENCE | DISTRICT OF OCC.

VICTIM (IF FIRM, NAME AND TYPE OF BUSINESS) | HOME ADDRESS | BUSINESS ADDRESS

ADDITIONAL DETAILS

FORM 184 P.D. (REV. 9-63) IF ADDITIONAL SPACE IS NEEDED USE CONTINUATION REPORT

EDP RECORD LAYOUT

0				5				10				15				20				25				30				35				40

41				45				50				55				60				65				70				75				80

81				85				90				95				100				105				110				115				120

121				125				130				135				140				145				150				155				160

COMMENTS:

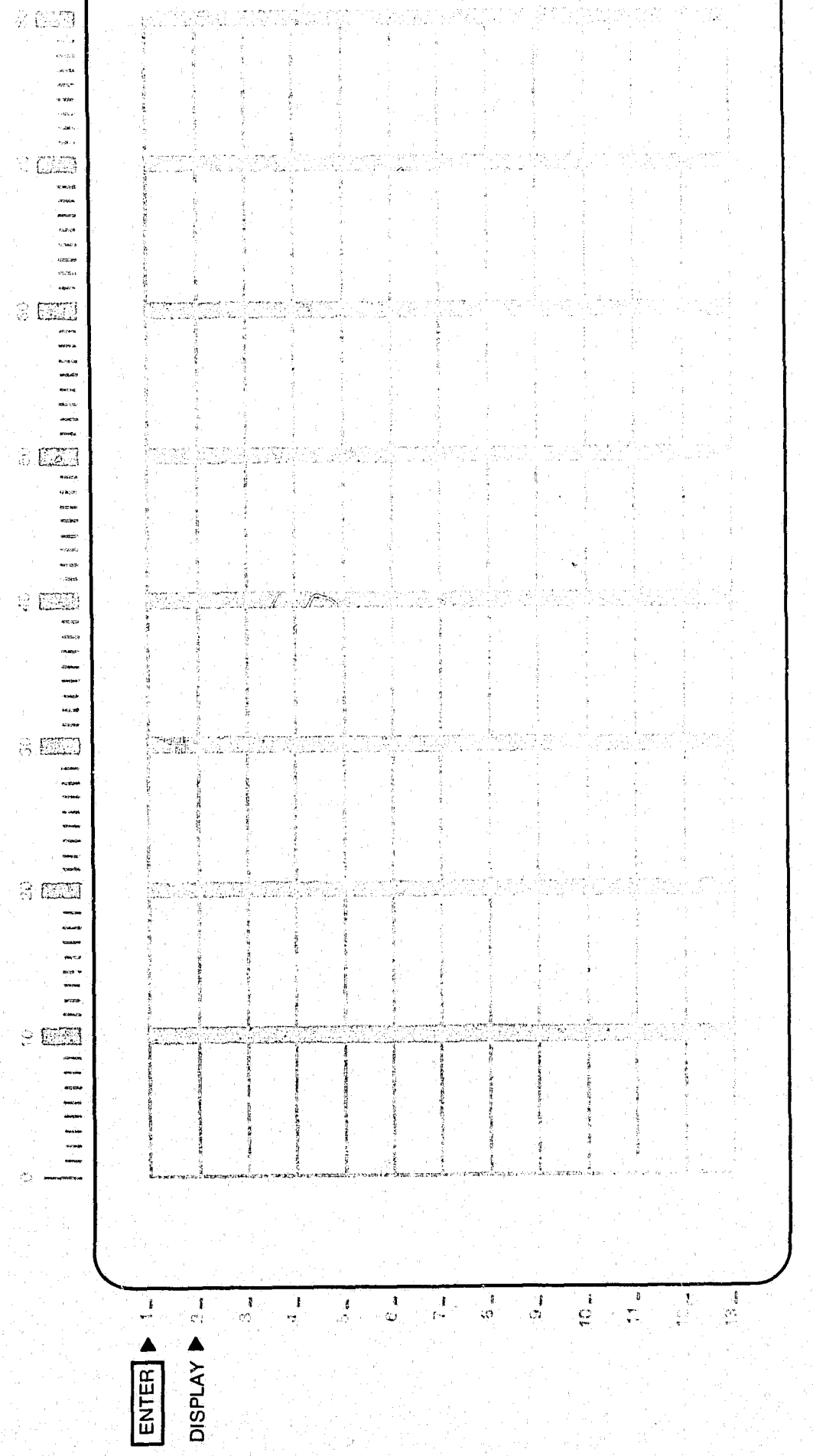
EXAMPLE 4.1

CRT ENTRY EXAMPLE FOR: ☐ INQUIRY ☐ UPDATE

NAME:

ENTER CALL CODE:

DISPLAY MASK:



FIELD DEFINITIONS

FIELD	FIELD NAME		SIZE	TYPE CHARACTERS

INDEX NUMBER

4.10 Scheduling and Control Data:

a. List programs in order of running.

b. List source inputs, expected date of arrival and expected volume.

c. Summarize report names, frequency of issue (daily, weekly, monthly, etc.) and date expected.

5.0 PROGRAM DOCUMENTATION

5.1 Program Title Page: Enter the following information on a separate title page for each program:

Program Name and Program Title
Programmer Name(s)
Date Completed
Approval of Unit Supervisor

The exact format is depicted in example

5.2 Table of Contents: Enter a completed Table of Contents form for each program. Enter an "X" in the circle beside each item that is being included in the program documentation. The format to be used is shown in example 5.3.

5.3 Program Description

a. Describe briefly the purpose of the program and the processing that takes place in the program.

b. Describe any balancing or control procedures that are contained within the program. Explain any output generated by out-of-balance conditions.

c. In general, the program description should be used to clarify, supplement, and extend the program logic shown in the Detailed Block Diagram. Any file names, field names, etc. used in the description should be the same as those used in the diagrams, coding, layouts, and glossary.

5.4 General Block Diagram: This diagram is to be included to clarify the logic of large and/or complex programs. It should show the interaction of the logical subsections of a program. The level of detail is described in the example of the same number as this paragraph.

5.5 Detailed Block Diagram: This diagram will show all processing and calculations required to produce the specified outputs. All paths that the program logic might follow are displayed graphically. Each closed subsection of a program is diagrammed separately. Machine produced block diagrams are acceptable.

DATA PROCESSING

MANUAL STANDARDS

SECTION
FINAL DOCUMENTATION STANDARDS

DATE ISSUED | DATE REVISED

5.6 Program Listing: Include a complete current assembly or compiler listing with the Link map.

5.7 Input/Output Layouts: Although Input/Output Layouts are part of the System Documentation, the layouts appropriate to an individual program may be duplicated and included with the Program Documentation for convenient reference.

5.8 Register and Switch Assignment Log: Include a completed Register and Switch Assignment Log.

5.9 Tables: Include a description of each table used by each program and its method of input to the program.. Designate responsibility for updating or changing tables.

5.10 Control Cards: Enter a layout of all control cards.

5.11 Computer Operations Run Sheet: A copy of the Computer Operations Run Sheet will be completed and entered for each program. The exact form to be used is depicted in example of the same number as this paragraph.

5.12 Console Input: Enter a layout of any console input.

5.13 Restart Procedures: Include complete directions for restarting of each program because of machine errors, etc.

5.14 Test Data: Include listings of all inputs to, and outputs from the final test run for each program option.

5.15 Carriage Tape: Enter instructions for making the carriage tape if other than standard.

PROGRAM TITLE: FBI REPORT - RETURN A & B

DATE OPERATIONAL: January 16, 1972

PURPOSE: CFØØ1 prints the Monthly FBI Report Return A, and the year-to-date FBI Report Return B. The input to this program is in the form of the two output tapes created by CFØØØ. If Return A is desired the input to CFØØ1 must be the Monthly tape created by CFØØØ. Also the CFØØ1 Control Card must contain the letter "A" in Column 1. If Return B is desired the input must be the year-to-date merged tape from CFØØØ. The Control Card must then contain the letter "B" in Column 1. Output is listing.

- ⊗ Program Description
- ⊗ General Block Diagram
- ⊗ Detailed Block Diagram
- ○ Register and Switch Assignment Log
- ○ Description of Tables
- ⊗ Program Listing
- ○ Test Data
- ⊗ Input/Output Layouts
- ⊗ Control Card Layouts
- ○ Console Input
- ⊗ Computer Operations Run Sheet
- ○ Carriage Tape
- ⊗ Distribution Sheet
- ○ Restart Procedures

I. PROGRAM NARRATIVE

Total arrests of adults and arrests of juveniles (excluding traffic arrests) are printed from arrest tape. Part I offenses are broken down into the following categories from the offense tape:

A. Number of offenses reported (by type offense)
B. Number baseless or unfounded (by type offense)
C. Number actual offenses (by type offense)
D. Number of offenses cleared by arrest (by type offense)
E. Number of arrests of juveniles (by type offense)
F. Value of property stolen (currency, jewelry & precious metals, furs, clothing, locally stolen automobiles, misc.)
G. Value of property recovered (currency, jewelry & precious metals, furs, clothing, locally stolen automobiles, misc.)
H. Number of actual offense & value of property stolen (robbery, burglary, larceny, autotheft)
I. Number of automobiles recovered.

Traffic tape provides the following information:

A. Hazardous violations
B. Other violations
C. Parking (except meter) violations
D. Parking meter violations
E. Total traffic
F. DWI
G. Hit & Run
H. Arrests, citations or custody at accidents

II. DETAILED DESCRIPTION

Files are opened, current date is moved to headers, type return is moved from control card to headers. If Originating Agency Code is other than that of the Kansas City, Missouri Police Department, it is moved from control card containing specific agency.

HEADERS, MOVE-ZEROS-IN move zeroes to counters and switches.

READ-ARREST-TAPE reads the arrest tape, checks month if Return A, check age for juvenile or adult and adds to counter (JUVENILE-ARREST or ADULT-ARREST).

READ-OFFENSE-TAPE moves zeroes to subscripts, reads file, compares ORI, compares month if Return A, compares for original, supplemental, stolen property or recovered property only, edit clearance codes.

PROGRAMMING DOCUMENTATION

SECTION EXAMPLE 4.2

OFFENSE PROGRAMS

DATE ISSUED January 16, 1973

DATE REVISED

FRIZLE compares code in file (OFFENSE-CODE2 or OFFENSECODE2) with literals to move actual values into subscripts or ignore record if no match.

ADD-TO-FIRST-COUNTERS and ADD-FOR-LARCENY-FIRST add or subtract from subscripted counters (COUNT-PAGE1 and COUNT-PAGE2-4A) for Part 1 Offenses to be counted as number reported, number found to be baseless or unfounded, number of actual offenses, number cleared by arrest, and of those cleared by arrest, number of Juveniles (COUNT-PAGE 1). Also adds for actual offenses for Robbery, Burglary, Larceny and Auto theft to appear later in the report (COUNT-PAGE 2-4A).

ADD-SUBS-CHANGE moves a blank into the add-subtract field of stolen property and recovered property if a zero now appears in that field.

STOLEN-RECOVERED, SUB-RECV-STOLEN, ADD-RECV-AUTO, SUB-RECV-AUTO, and ADD-AND-GO add or subtract the amount of stolen or recovered property for each of the six fields (currency, jewelry, furs, clothing, autos, miscellaneous) to appropriately subscripted counter (COUNT-PAGE2A).

COUNT-ACTUAL-VALUES, CHECK-BURGLARY, CHECK-LARCENY, CHECK-AUTOS, ADD-NATURES, ADD-TO-EACH-COUN, STOLEN-ADD, AND SUB-OUT adds or subtracts from counters for number of actual offenses and amount of stolen for Robbery, Burglary, Larceny and Auto Theft (number of actual offenses--COUNT-PAGE 2-4A; amount of stolen property--COUNT-PAGE 2-4B).

RECV, ADD-AUT, and SUB-AUT add or subtract from counter (COUNT-PAGE 2-4A) for type recovered auto:

1. locally stolen and locally recovered
2. locally stolen and recovered by other
3. total stolen locally recovered
4. stolen out of town, recovered locally

LARCENY-SEPARATE counts larceny's under $5, $5 - $49.99, and those $50 and over by amount stolen to counter (COUNT-PAGE 2-4B).

READ-TRAFFIC-TAPE reads traffic files, checks ORI, and month if Return A, and traffic ordinance and/or type.

ADD-TO-CTR adds 1 to appropriate counter whenever data found (COUNT-PAGE 2-4A).

START-TO-PRINT, PRINT-PAGE 1 moves and prints data for first page of report.

MOVE-IN-STOLEN-RECOVERED, MOVE-IN-TRAFFIC moves and prints data for second page of report.

PROGRAMMING DOCUMENTATION

SECTION EXAMPLE 4.2

OFFENSE PROGRAMS

DATE ISSUED	DATE REVISED
January 16, 1973	

CLASSIFY-PRINT and LAST-LINE-PRINT move and print data for the third and fourth page of report. If there are additional agencies remaining to be processed branch is to beginning of Procedure Division.

STOP-JOB closes files and stop run occurs.

C-CONSTANT SECTION moves and prints all constants on printout referring to any totals.

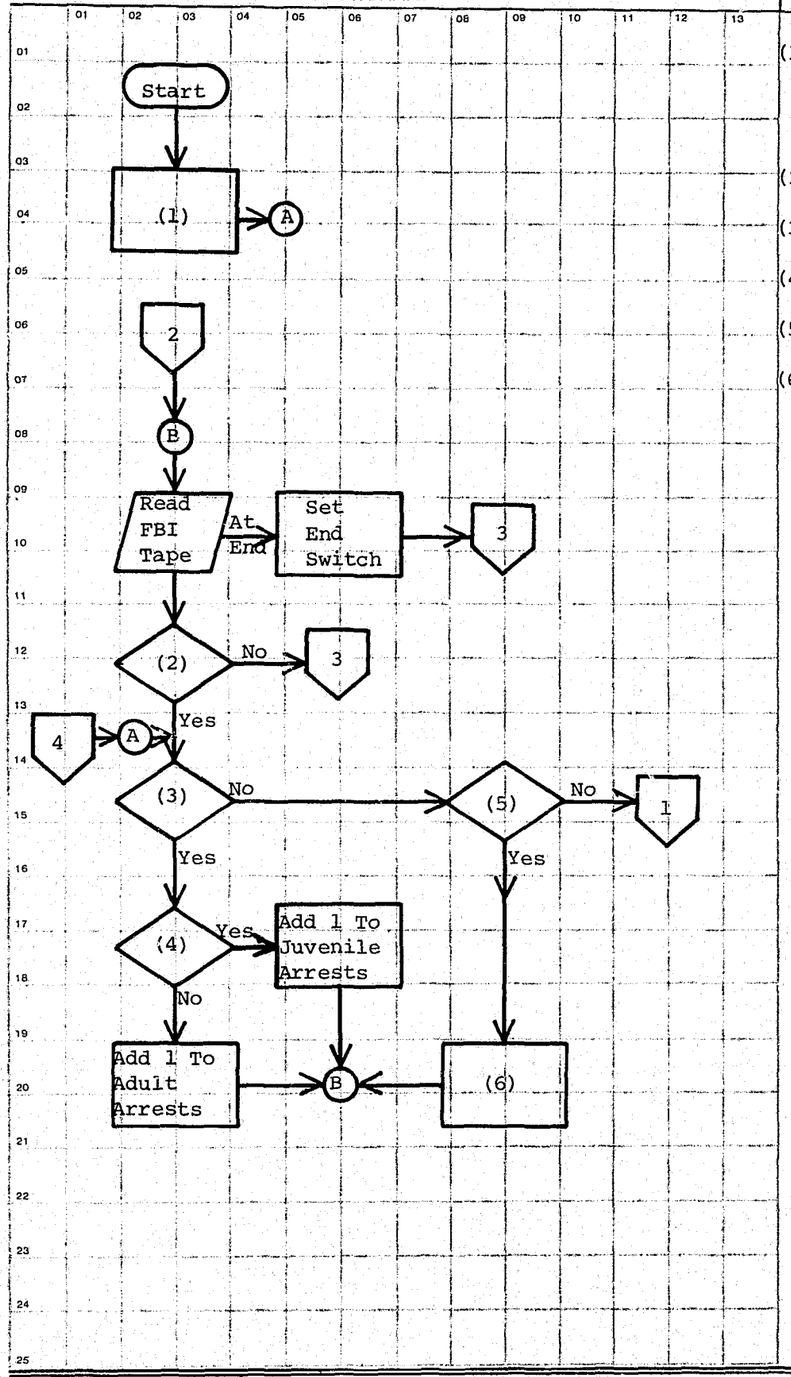
END-OF-PAGE SECTION prints standard information at bottom of pages and necessary headers.

INDEX NUMBER
CFØØ1

EXAMPLE 4.2
Page 1 of 2

SYSTEM FLOW CHART

FLOWCHART



DESCRIPTION

(1) Open Files, Set Counters etc. Read First Tape Record and Save ORI Code.

(2) ORI = Saved ORI?

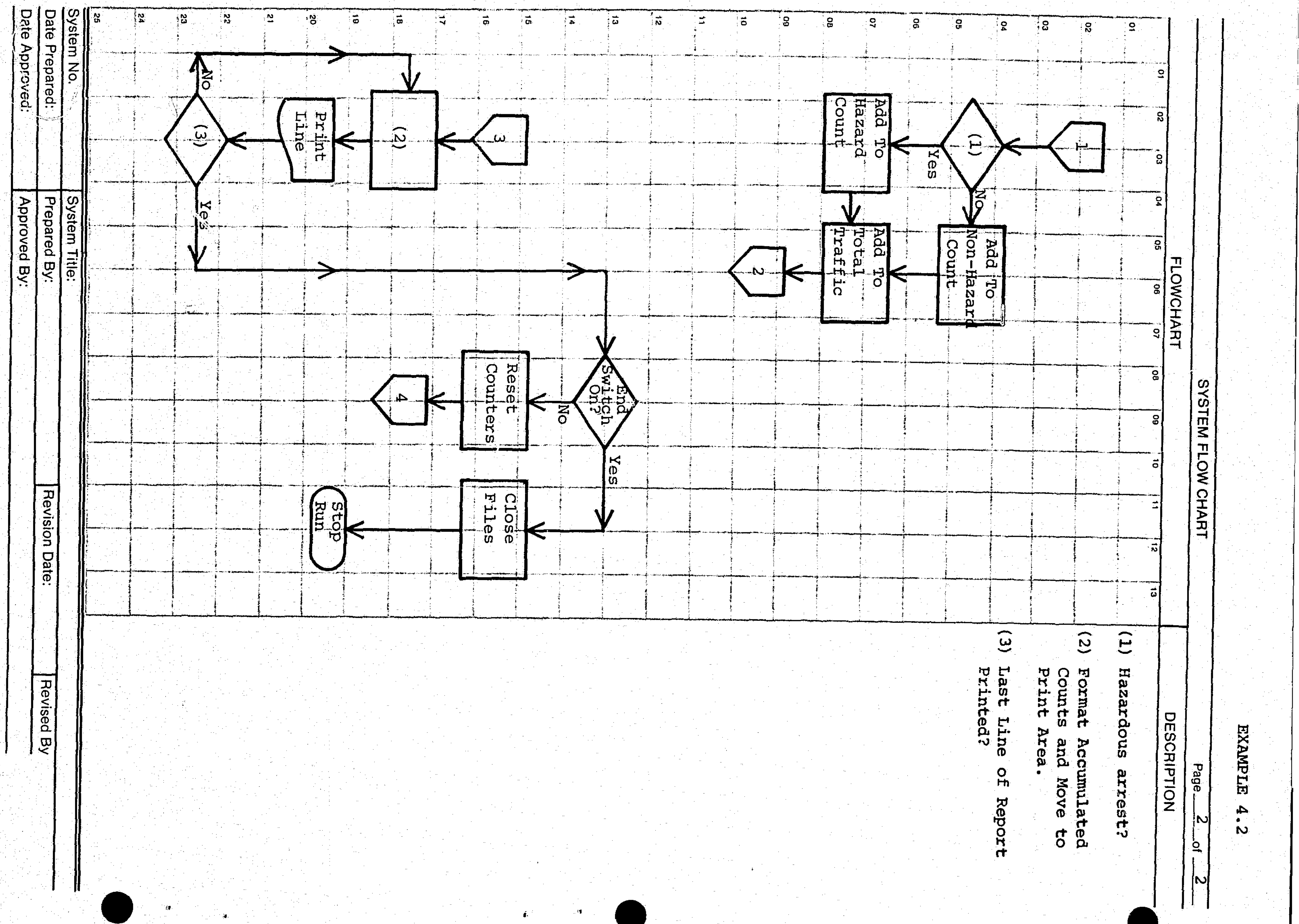
(3) Arrest Record?

(4) Age<18?

(5) Offense Record?

(6) Add to Correct Counter Based Upon Offense Type.

System No.	System Title:		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		



E D P RECORD LAYOUT

FBI INPUT TAPE - YEAR-TO-DATE - CFØØØT2

RECORD KEY							CRIME			ATTACKED				OCCURRED			REPORTED		CLEARANCE				OFNDR							
ALERT #	RECORD TYPE 60	WHICH OFFENSE	TYPE REC Ø1	# OF TYPE	ORIG AGCY	CASE REPORT #	OFFENSE CODE	NATURE CODE	MOTIVE CODE	PROPERTY	HOW	WEAPON / MEANS	INJ TO VIC	DATE	HOUR	DAY OF WK	DATE	HOUR	HOW CLRD	UNIT	DATE	ADD/SUB CD	RACE	SEX	AGE	FILLER	OFFICERS SERIAL NUMBER	REPORT DATE	0	ORIG AGCY

RECORD KEY							CRIME		STOLEN PROPERTY INFORMATION: CURRENCY			JEWELRY			FURS			CLOTHING			AUTOS			MISC.							
ALERT #	RECORD TYPE 60	WHICH OFFENSE	TYPE REC 2	# OF TYPE	ORIG AGCY	CASE REPORT #	OFFENSE CODE	NATURE CODE	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	FILLER	OFFICERS SERIAL NUMBER	REPORT DATE	0	ORIG AGCY

RECORD KEY									CHARGE		ARR.									ARR. OFFICER							PREVIOUS RECORD													
ALERT #	TYPE 4Ø	NO. OF ARRESTS THIS PERSON	RECORD COUNT ØØØ	ORIG AGCY	ARREST NUMBER	ARREST DATE	ARREST HOUR	ARREST CHARGE	NATURE	AGE	RACE	SEX	NARCOTICS INVLD	JUV. COURT	JUV. PREVIOUS	JUV/PAROLE PROB.	HELD FOR PROS.	UNIT	SERIAL NO. 1	SERIAL NO. 2	ORDINANCE ARRESTS	ORDINANCE CONVICTIONS	MILITANT ACTIVIST	TRAFFIC ARRESTS	TRAFFIC CONVICTIONS	MISDEMEANOR ARRESTS	MISDEMEANOR CONVICTIONS	FELONY ARRESTS	FELONY CONVICTIONS	ARREST COUNT	DISP. FLAG	JACKET NO.	REPORT DATE	A	ORIG AGCY				

RECORD KEY							STATISTICS									COURT		DISPOSITION																
ALERT #	TYPE (45)	TIC TYPE	SEQ	RCD (ØØ)	ORIG AGCY	CASE REPORT #	TICKET NO.	ORD. CODE	DATE OCC.	TIME OCC.	OFF. SER. NO.	UNIT	DAY OF WEEK	RES. CD.	DISP	COURT DATE	CT TM	DISP DATE	DISP TP	MONTHS	DAYS	AND/OR	FINED	PAR/PROB LENGTH	PAR/PROB	TRAF SCHL	ASAP RESULTS	ASAP REFUSED	FILLER	RCD TYPE	REPORT DATE	T	ORIG AGCY	

RECORD KEY							CRIME		RECOVERED PROPERTY INFORMATION: CURRENCY			JEWELRY			FURS			CLOTHING			AUTOS			MISC.							
ALERT #	RECORD TYPE 60	WHICH OFFENSE	TYPE REC 3	# OF TYPE	ORIG AGCY	CASE REPORT #	OFFENSE CODE	NATURE CODE	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	TYPE CODE	VALUE	ADD/SUB	FILLER	OFFICERS SERIAL NUMBER	REPORT DATE	0	ORIG AGCY

307 P.D. (7-69)

EXAMPLE 4.2

DESCRIPTION OF COMPUTER REPORT OR LISTING

DATE	ID

☐ NEW ☐ REVISION—SHOW WHY IN 'COMMENTS'

TITLE OF REPORT OR LISTING

FBI RETURN A - OFFENSES KNOWN TO POLICE CFØØ1

PURPOSE OR FUNCTION IT SERVES

THIS REPORT IS DESIGNED TO MEET THE MONTHLY REPORT OBLIGATIONS TO THE FEDERAL BUREAU OF INVESTIGATION UNIFORM CRIME REPORTING SYSTEM.

ORIGINATES FROM (SHOW COMPUTER RUN AND/OR MAIN FILE FROM WHICH DATA IS DEVELOPED AND SPAN OF TIME COVERED OR AGE OF DATA)

INFORMATION IS EXTRACTED FROM THE MONTHLY FBI TAPE

NO. COPIES	FREQUENCY ISSUED
	☐ DAILY ☐ WEEKLY ☒ MONTHLY ☐

DESIGN FORMAT APPROVED BY	DATE	RELEASE PERIOD

COPY DISTRIBUTION

	SENT TO	RETENTION	DISPOSITION
1	REQUESTING AGENCY (2)		
2	FILE (1)		
3			
4			
5			
6			

COMMENTS

DETAILED EXPLANATION OF DATA (WHEN PRINTED CAPTIONS ARE NOT SELF EXPLANATORY)

THIS REPORT DISPLAYS IN ARRAY FORM:

a. OFFENSES REPORTED OR KNOWN TO POLICE.
b. UNFOUNDED, FALSE OR BASELESS COMPLAINTS.
c. NUMBER OF ACTUAL OFFENSES.
d. NUMBER OF OFFENSES CLEARED BY ARREST.

FOR THE FOLLOWING OFFENSES:

a. CRIMINAL HOMICIDE.
b. FORCIBLE RAPE.
c. ROBBERY.
d. ASSAULT.
e. BURGLARY.
f. LARCENY.
g. AUTO THEFT.
h. TOTAL.

PAGES 2, 3 AND 4 OF CFØØ1 REPRESENT VALUES OF PROPERTY STOLEN AND RECOVERED WITH REGARD TO THE OFFENSES ENUMERATED IN PAGE 1.

CONTINUE ON REVERSE SIDE

EXAMPLE 4.2

DEC 1972 44370 404 MOKPD00 1 4 L 510600 9

KANSAS CITY MISSOURI POLICE DEPARTMENT
RETURN A
RETURN OF OFFENSES KNOWN TO THE POLICE

CLASSIFICATION OF OFFENSES	OFFENSES REPORTED OR KNOWN TO POLICE	UNFOUNDED FALSE OR BASELESS COMPLAINTS	NUMBER OF ACTUAL OFFENES	NUMBER OF OFFENSES CLEARED BY ARREST TOTAL	NUMBER OF OFFENSES CLEARED BY ARREST UNDER 18
1. CRIMINAL HOMICIDE	7	1	6	7	0
A. MURDER & NONNEGLIGENT MANSLAUGHTER	9	5	4	4	0
B. MANSLAUGHTER BY NEGLIGENCE					
2. FORCIBLE RAPE TOTAL	16	0	16	7	0
A. RAPE BY FORCE	14	0	14	6	0
B. ASSAULT TO RAPE - ATTEMPTS	2	0	2	1	0
3. ROBBERY TOTAL	222	1	221	50	13
A. ARMED - ANY WEAPON	149	1	148	36	10
B. STRONGARM - NO WEAPON	73	0	73	14	3
4. ASSAULT TOTAL	181	0	181	131	12
A. GUN	36	0	36	21	1
B. KNIFE OR CUTTING INSTRUMENT	33	0	33	27	3
C. OTHER DANGEROUS WEAPON	36	0	36	25	2
D. HANDS, FEET, ETC. - AGGRAVATED	12	0	12	10	0
E. OTHER ASSAULTS - NOT AGGRAVATED	64	0	64	48	6
5. BURGLARY TOTAL	835	22	813	225	98
A. FORCIBLE ENTRY	626	7	619	172	72
B. UNLAWFUL ENTRY - NO FORCE	170	14	156	36	22
C. ATTEMPTED FORCIBLE ENTRY	39	1	38	17	4
6 LARCENY - THEFT (EXCEPT AUTO THEFT	1,105	13	1,092	281	113
A. $50 AND OVER IN VALUE	490	4	486	65	17
B. UNDER $50 IN VALUE	615	9	606	216	96
7. AUTO THEFT	345	33	312	71	57
GRAND TOTAL	2,720	75	2,645	776	293

TOTAL ARRESTS FOR ALL OFFENSES EXCEPT TRAFFIC -
ADULTS 1920 JUVENILES 472

DATE JANUARY 10, 1973

PREPARED BY_____

CHIEF_____

DECEMBER 1972

KANSAS CITY MISSOURI POLICE DEPARTMENT

EXAMPLE 4.2

6.0 SORT AND MERGE

6.1 Enter only the following items for Sort/Merge runs in addition to System Documentation.

a. Program Title Page

b. Table of Contents

c. Program Description

d. Detailed Block Diagram of any modification to Sort/Merge Utility Routines

e. Program Listing of any modification to Sort/Merge Utility Routines

f. Output/Input Layouts

g. Control Card Layouts

h. Computer Operations Run Sheet

i. Halt List if there are any halts in modification made to Sort/Merge Utility Routines

j. Test Data

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
FINAL DOCUMENTATION STANDARDS

DATE ISSUED | DATE REVISED

7.0 OPERATIONS SUPPORT DOCUMENTATION

Effective utilization of computer equipment depends largely on Operators and Programmers ability to understandably communicate with each other. This is done primarily through the use of Run Sheets and other special guidance necessary to set up and complete specific jobs or program runs.

The following standard is established for this installation to allow operations and other personnel to understand the general functions performed in each production program and to be aware of any special instructions.

7.1 Programmers are responsible for completing the K.C.P.D. computer operations Run Sheet as depicted in example 5.11 for each program set up for processing. One copy to be furnished the Operations unit, one copy filed as documentation with the program and one copy to the Systems Analyst.

7.2 In addition to this Run Sheet the appropriate Job Control cards as needed, will be furnished along with the Run Sheet.

7.3 Whenever changes necessitate the revision of the above documents it is the programmer's responsibility to insure revised instructions are distributed to all concerned.

Explanation of Computer Operations Run Sheet

- Preparation and use of Computer Operations Run Sheet

A "run sheet" is to accompany each test or operational job and/or job step submitted to operations and is to be filled out as follows:

JOB NAME	Same as on Job card.
JOB CLASS	Same as on Job card for OS, not used for DOS.
TITLE	Title of the program.
TYPE OF RUN	Check appropriate circle.
STEP NAME	Same as on Job Step card for OS, not used for DOS.
OUTPUT CLASS	Same as on SYSOUT cards for OS, not used for DOS.
PROGRAMMER	Name of person responsible for the program.
ALTERNATE PROGRAMMER	Name(s) of alternates to the person responsible for the program.
FREQUENCY	How often the program is to be run (applies to operational programs only).
SEQUENCE	Sequence number of a run sheet within a series and the total number of run sheets in the series.
CARRIAGE TAPE	Check appropriate circle.
FORM NUMBER	Form number of special forms or indicator for special printer set-up.
ORIGINAL DATE	Date the program was submitted to Operations.
REVISED DATE	Date of the latest change to the run sheet.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

I/O	"I" for an input data set,"O" for an output data set. "W" for a work data set being used as both an input and an output.
VOLUME NUMBER	Volume number of tape or disk pack to be mounted.
UNIT	Hardware address of unit. Used for OS only if a specific unit is required. Does not apply to On-Line Mater File, General Index, Name Index or Payroll Master File for DOS.
LABEL RETENTION	Number of retention days or the expiration date for an output file.
DISPOSITION	Disposition of a file after the job or job step has been completed.
COMMENTS	Any other information necessary to aid in locating or identifying a data set.
SPECIAL INSTRUCTIONS	Any information necessary for the successful running of the job or job step, including the name of any special forms for output and the carriage tape to be used if it is not Standard.

EXAMPLE 4.3

K.C.P.D. COMPUTER OPERATIONS RUN SHEET

JOB	TITLE	PROGRAMMER	ALTERNATE PROG.
CFØØØ	FBI TAPES	HARMON	BAILEY

TYPE OF RUN	FREQUENCY	SEQUENCE	ORIGINAL DATE	REVISED DATE
⊗ PROD. ○ TEST	MONTHLY	1 OF 3	5/19/72	7-10-72

CARRIAGE TAPE ⊗ STANDARD ○ OTHER (see spec. instr.) 1 PART PAPER

I/O	360 UNIT ADDR.	REEL OR DISK NUM.	FILE-ID	COMMENTS	LABEL RETEN.	DISPOSITION	370 UNIT ADDR.
I	28Ø		CUØØØT1 MM/YY ARRESTS			LIB	48Ø
I	281		CFØ14T1 MM/YY			LIB	481
I	282		CDØ1ØT1 FOR MM/YY			LIB	482
I	135	79	CBØ79D1 CODE FILE			LIB	331
W	133		SORT			SCRATCH	155
0/I	28Ø		CFØØØT1 FBI MM/YY		2 MOS.	TO CFØØ1	48Ø
I	281		CFØØØT2 FBI YTD Ø1-MM/YY			LIB	481
0	282		CFØØØT2 FBI YTD Ø1-MM/YY		2 MOS.	TO CFØ17	482

SPECIAL INSTRUCTIONS:

ONE CONTROL CARD REQUIRED. TO BE PREPARED BY OPERATIONS. IN COLUMNS 1 & 2 PUT NUMBER OF MONTH OF DATA WANTED. (EXAMPLE: PUT Ø5 IN COLUMNS 1 & 2 IF MONTH OF MAY IS WANTED). IN COLUMNS 7Ø-8Ø PUT 'CFØØØCTLØØ1'.

ALL TAPES (EXCEPT INPUT CFØØØT2 FBI YTD) TO BE SAME MONTH AS INDICATED ON CONTROL CARD. INPUT CFØØØT2 FBI YTD TO BE MONTH PRIOR TO MONTH INDICATED ON CONTROL CARD.

DO NOT MOUNT INPUT CFØØØT2 YTD WHEN RUN IS FOR JANUARY'S DATA. WHEN CUØØØT1 UNLOADS, MOUNT SCRATCH TAPE ON '28Ø' (OR '48Ø'). WHEN CFØ14T1 UNLOADS, MOUNT CFØØØT2 FBI YTD ON '281' (OR '481'). WHEN CDØ1ØT1 UNLOADS, MOUNT SCRATCH TAPE ON '282' (OR '482').

Form 190 (Rev. 1-72)

TABLE OF CONTENTS

05 SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

1.0 SYSTEMS PLANNING AND FEASIBILITY STUDY

1.1 INTRODUCTION

In the most general sense, computerization of information processing is usually accomplished at any one of a number of levels in a hierachy. Briefly, the levels are:

Systems
Applications
Computer programs or runs

A system is made of applications which in turn are made up of computer programs.

Ideally, to accomplish the most significant benefits from automation, computerization should be developed at the system level and a system should be defined as broadly as practical. Organizationally this means that computerization plans should be developed at the Administration level so that the objectives of the Administration can be carried out with the minimum amount of duplication of effort.

An Administration should be able to relate every computerization project to an overall Administration Information Systems Plan.

Such a plan should include:

a. Systems Objectives

b. Present Status of

(1) Information Systems
(2) Structure
(3) Equipment
(4) Personnel
(5) Documentation
(6) Operating levels
(7) Strengths-weaknesses service/information
(8) Environmental factors
(9) Expenditures

c. Projected Performance Requirements

d. Conceptual Plan

(1) Information structure
(2) DP equipment
(3) DP personnel

e. Action Plan

(1) Organization and staffing
(2) Detailed project plans
(3) Project priorities and schedule
(4) Resource estimates (people, equipment and money)
(5) Action Responsibility

f. Plan Costs and Benefits (Projected 5-10 years)

(1) Quantifiable
Investment
Operating costs
Cost displacement
Cost avoidance
Other
(2) Non-Quantifiable
Increased service/information
Better service/information
Other

g. Major Plan alternatives considered

There are too many variables involved in the development of such a broad based systems plan for it to be covered by this standard. Consequently, the emphasis of the sections is at the project level. Adherence to the standards presented in the manual will facilitate the integration of independently developed systems into complete Information System.

1.2 OBJECTIVES

As used in this standard, a systems planning and feasibility study is conducted to determine a technical, operational and economical method to accomplish management's data processing systems objectives. The Systems Planning and Feasibility Study includes almost all of the activities involved in the development and implementation of a system. The objective of the study is to carry out the activities only to the level

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

of detail necessary to define one or more systems plans for management's approval or selection for project continuation to the next phase. The selection should be based on the systems ability to meet management's objectives within the constraints of the following feasibilities:

Technological Feasibility - The system is possible within the limits of available time and technology.

Operational Feasibility - If the system is successfully developed, there are people with the appropriate responsibility and authority in the organization who are committed to making the system work.

Economic Feasibility - The benefits provided by the system are equal to or greater than the total cost of providing them.

1.3 PROCEDURE FOR CONDUCTING

A system planning and feasibility study is normally conducted in sequence from problem definition through problem analysis, new system requirements, and new system design, to final technical, operational, and economic analysis. The problem definition must always be the first activity, but the other activities are not always to be conducted in the precise sequence in which they are outlined below. Throughout the study, all practical management and technical alternatives should be considered in order to insure the development of the most pragmatic system.

a. Problem Definition

This problem definition under guidance of management should clearly establish the scope of the study, the requirements to be met, the objectives to be accomplished, and the areas, functions, or systems to be examined. It is essential that the problem definition be jointly developed by the systems analyst(s) and the user subject matter specialist(s) and that it receive approval by all pertinent levels of user management. It should be made as specific and comprehensive as possible before any further detailed work is undertaken. Problem definition must include clearly defined concepts of how management desires to:

(1) Build its information pyramid (progressive report summarization for upward management levels) (e.g. types of exception reports, statistical reports...etc.).

(2) Limit information turn-around times at each management level:

- Mode of communication between physical locations.
- Batch of processing possible vs. real time processing dictated.

(3) Location of operations (i.e., centralized vs. decentralized operations or combinations of both).

Definitions of the above management concepts must be obtained and set forth as part of the problem definition. These decisions frequently have the greatest economic impact on the proposed systems. Such concepts must be approved by user management before the analysis and design work begins. These managing concepts become the design parameters and serve as the basis for determining types of computer and communications equipment required, frequency of operations (realtime, batched daily, batched weekly, etc.), and locations of operating facilities and staffs. Alternative designs may be required to provide management with the necessary information upon which to base its final selection of an operating concept.

b. Analysis of Existing System

The method to be used in examining the present system depends to a large extent on the indicated objective.

Usually, even if a new system is being designed to provide a function not currently being performed an analysis is needed of the current functions with which it would have to interact. Unless the current system user is in a position to state that existing data elements and logic in the present system will not be required by the new system, the detailed study described in the next paragraph should be made. If the purpose of the study is the improvement of an existing system, a step-by-step study of the present system is necessary in order to determine the feasibility of the proposed change.

Such a study should include the preparation (at a minimum) of a general narrative description of the present system. Copies of all significant documents should be obtained to assist in the analysis and included as part of the present system documentation.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

A glossary of terms used in the operation and their meaning should be started and added to during the course of the study. A list of abbreviations of systems terms and lists of significant codes should also be developed. Prior to starting this analysis, the analyst team should formulate a concept for the future steps of evaluating technical feasibility.

If it is decided that a computer program will be used for simulation, then the particular simulator program should be selected at this time. All codes and descriptions used in the systems planning and feasibility study must be adapted initially to the simulator requirements. Many of the forms described in the subsequent systems development phases will be helpful in accomplishing the analysis and documentation essential to this phase.

c. Determine Input and Output Requirements for Proposed System

A determination must be made as to the output requirements and the availability of the required input data. Where and how is the input created? What processing steps must the input data go through to reach the point where it can be used as input to the DP System?

This part of the study may raise problems in coding, source data automation and data transmission, each of which may have to be examined in detail. Full consideration of source data automation techniques is extremely important. Consideration should also be given to input and output controls for assuring the validity of the data. The end products should be clearly defined as to the type (e.g. punched cards, reports, etc.), volume, and when, where and by whom they are required. Record descriptions should be prepared for each input and output indicating in detail the principle fields and the expected maximum size of the record(s). The Records Layout Worksheet can be used for this purpose. Important data fields should not be overlooked. For example, in the design of a master address file, the "ZIP" code should be made part of the master address record. Data Processing Input and Output Volume Requirements should be determined. The File Layout Sheet may be used to record this information. Input from or output to other automated systems must be fully coordinated, in detail, at this point. All responsible parties must be made aware of cross system impact of proposed

INDEX NUMBER
05-1.0

future design or maintenance changes. A formal control point for clearing such proposals must be established. All control decisions must be documented in detail with copies to impacted systems.

d. Develop Systems Flow Chart

The analyst should prepare a system flow chart of the proposed system(s) at the functional level and where necessary at the program level. Associated functional program run narratives are to be developed. Included in this aspect of the feasibility system design is the method of CONVERTING the current system (if one exists) to the proposed system. Design considerations should also be be given to RESTART and FALL BACK (BACKUP) functions.

e. Technical Analysis

A careful timing analysis of the proposed system(s) is required in order to determine if the system(s) meet the user's requirements as to when, where, and by whom the end product is required. If there is more than one proposed system, the respective technical merits of each need to be compared.

Timing estimates must include both manpower as well as equipment timings to perform the required functions. The results should be placed in a technical analysis report identifying each task, the critical factor(s) determining timing and the estimated timings.

f. Operational Analysis

The operational feasibility of the project is determined by a combination of user and data processing management review of the technical design (system flow charts, associated inputs/outputs, system narrative and sytem timing) together with an analysis of the human factor implications in getting operating personnel to function as required in order for the proposed system to function successfully.

g. Benefits

A list should be made of all benefits that result from the proposed system and that are not a direct operational savings that would be identified on the Economic Evaluation form.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

The list should include quantifiable and non-quantifiable benefits. Wherever possible, it is desirable to quantify benefits in dollars.

A Benefits form and associated procedure can be used to list these benefits.

h. Management Summary

A Management Summary form should be completed as part of the Systems Planning and Feasibility Study report that is presented to management for their review and approval. This form condenses the salient factors of the study into a compact package, so that high level management, who do not have enough time to review the entire report, can get an overview of the study.

i. Systems Planning and Feasibility Study Report

The functions to be performed during the systems planning and feasibility study should be summarized in a formal Systems Planning and Feasibility Study Report that will serve as the basis for project funding. A suggested table of contents for the report follows:

(1) Abstract of Feasibility Study
    Brief description of system
    Recommendation

(2) Management Summary
    Requirements for study
    Duration of study
    Participants
    Departments outside of Data Processing involved

(3) Description of the Proposed New System
(This chapter will provide guidelines to the design team for the Problem Analysis and New System Requirements phase).

- Functional Narrative
- Functional Flow Chart
- Inputs to the System
- Outputs from the System
- Master Files
- Special hardware and performance requirements

INDEX NUMBER
05-1.0

(4) Developing the New System

For each phase

- Manpower estimates
- Schedule estimates

Technical (programming, hardware, etc.) and Operational (user-oriented) difficulties that have to be resolved.

Areas of investigation for Problem Analysis and New System Requirements phase.

(5) The Present System

- Deficiencies

(6) The Proposed System

- Technical Analysis
- Operational Analysis
- Operational Costs
- Economic value of additional functions
- Benefits

(7) Glossary of Terms

(8) List of Abbreviations

(9) System Code List

1.4 MANAGEMENT REVIEW AND APPROVAL

A presentation should be prepared for users and general management review of the Systems Planning and Feasibility Study. The management review decision to stop further action or to proceed to the comprehensive Problem Analysis and New System Requirement phases should be documented together with approved action.

- Short descriptions are required of each rather than the detail required during the Problem Analysis and New Systems requirements phase.

1.5 EQUIPMENT SELECTION

If additional or new equipment is required for the system, equipment requirements and specifications must be prepared in accordance with the guidelines set forth in Equipment Procurement and Performance Standards.

2.0 PROBLEM ANALYSIS AND NEW SYSTEM REQUIREMENTS

2.1 INTRODUCTION

The Problem Analysis and New System Requirements phase in the system development cycle is quite similar to the initial System Planning and Feasibility Study phase. The most significant difference is the depth of detail to which the tasks are pursued. In the first phase investigation, analysis and design was detailed enough to develop a feasible and acceptable system design and plan for implementation. In this phase, all the problem details and requirements must be identified, analyzed, and spelled out. A second pass must be made at the new system design and the system plan upgraded so that all parties, both user and data processing personnel, understand what is to be done, how it is to be accomplished, what the final product will look like, what the economic of the project and the finished operating product will be, and what resources and schedule are required to complete the tasks.

2.2 PROBLEM ANALYSIS

a. Scope

The system analyst, in order to design a new or revised system which will meet the requirements of the user, must understand how and why the present system operates as it does. This understanding is necessary even though the new system may be radically different from the present system. The system analyst must know the constraints or restrictions upon the present system which are beyond the control of the organization to change, and therefore must be allowed for in the design of the new system. The constraints or restrictions may be documented in the Design Constraints. The present system must be thoroughly defined and documented in order to determine the benefits and economics associated with a systems change.

b. Activities of Problem Analysis

The activities of the Problem Analysis are largely identifying documents and files, and determining what people do to those documents and files. It is important that the system analyst understand the scope of the proposed system, and that he focus his attention upon the current documents, files,

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

and activities which are to be affected by the new system. The following is a partial task list of those activities which might be performed where applicable:

(1) Operations to be survey or documented

(a) Input - analyze

- Volume
- Origin
- Frequency
- Format
- Size of record
- Handling exceptions

(b) Process - analyze

- Time Estimate
- Frequency
- Volume of data

(c) Output - analyze

- Frequency
- Distribution
- Urgency
- Action
- Amount of output
- Format
- Handling exceptions

(d) Files - analyze requirements

- Record types
- Length
- Format
- Volume
- Organization
- Identification
- Maintenance and updating
- Access
- Activity
- Sorting

INDEX NUMBER
05-2.0

(e) Procedures - analyze

- What is unnecessary and can be eliminated
- What can be simplified
- What can be combined

(2) Resources required

(a) Personnel

- Organizational chart
- Present skills

(b) Equipment

- Computer system
- Peripheral equipment

(c) Software

- Programming systems
- Monitor and operating systems
- Application programs

(d) Facilities

- Space
- Air conditioning and humidity control
- False Floor
- Power requirements
- Fire Proof storage vaults or cabinets

(3) Desired or required characteristics (limitations)

(a) Legal
(b) Timing - cycle and access (if real-time, minimum terminal time)
(c) Quality
(d) Form
(e) Interrelationships with other activities
(f) Applications ranking in priority sequence
(g) Management information
(h) Appropriateness of organizational structure
(i) Visual records versus periodic reports

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

(j) Budget
(k) Integrated data processing
(l) Adherence to Statewide ADP Plan
(m) Ability to consolidate programs with other organizational units

(4) Outline history of present procedures

c. Techniques

(1) Interview

The techniques used in developing the above information are common to most systems surveys. Information about present operations is usually obtained by interview. The interview is usually directed first to the supervisor to solicit his cooperation and to gain an understanding of the general picture of the operation. Detailed information is then obtained from personnel actually performing the work. The procedure for obtaining this information includes the following steps:

(a) Interview each employee successively in line with the information flow.

(b) Obtain a copy of each form or report originated by the employee and trace its flow from origination to final disposition.

(c) Account for all functions including an estimate of the volume of work and time required to perform functions.

(d) List all items of equipment, machines or other devices.

(e) Obtain some measure of efficiency of personnel and equipment.

(f) Determine workload distribution (i.e., Is work load similar from day to day or does it reach a peak at month's end?).

(g) Verify the accuracy of all information with the supervisor.

INDEX NUMBER
05-2.0

(2) Questionnaire

The use of questionnaires for fact gathering provides a means of contacting large numbers of individuals and obtaining answers to a wide variety of questions. Questionnaires must be carefully worded to obtain the responses that will contribute to the effectiveness of the study.

(3) Records Reviewing

An important means of fact gathering is the review of currently used records, documents, organizational charts, and other printed material which tend to provide documented evidence of existing procedures. Work sampling techniques and observation of actual working activities also provides a source of information that may contribute to a greater understanding of actual procedures and practices.

(4) Analysis

It is important that after a survey of the present system is completed the data be thoroughly analyzed by the system analyst to identify areas where technical considerations preclude immediate successful computer applications. The prompt elimination of these applications leaves more room for detailed consideration of those which have greater potentials. Then the analyst should critically analyze the facts to develop an effective solution to any systems problem or inefficiency which may have been discovered. During the fact gathering phase it has been determined what is done, where, when, by whom and how. In analyzing the facts the question of "why?" is now essential. Why is a particular activity done at all? Why by that person? Why in that manner? Why in that department? In analyzing the facts the question "why?" should be applied to every activity in a process or every procedure and element of an operation. After each phase an operation is critically analyzed and steps toward improvement of an existing system may be developed by applying the basic concepts of work simplification which are as follows:

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

(a) What steps can be eliminated?
(b) What steps can be combined?
(c) What steps can be changed?
(d) What steps can be simplified or improved?

The solution to, or improvement of a system problem must always be considered in relationship to the operating requirements or objectives of the user structure.

d. Documentation

Information obtained in the Problem Analysis phase should be recorded in a manner to facilitate its analysis. The following is a suggested list of information to be developed and standard and suggested forms to be utilized:

(1) System objectives and abstract
(2) Current organization chart
(3) Current operating flow chart
(4) Narrative of current system
(5) Job Description Form
(6) Input/output Document
(7) Record Layout Sheet
(8) List of reports being produced
(9) Reports distribution list
(10) File layout sheet
(11) Design constraints
(12) Flow chart of forms distribution
(13) Personnel workload utilization
(14) Glossary of terms
(15) Economical Evaluation Worksheets
(16) Economical Evaluation Summary

2.3 NEW SYSTEM REQUIREMENTS

a. Objective

The principal objective of the New System Requirements is to define the new system from the point of view of the user organizations and to make these organizations thoroughly aware of how they will be affected by the new system. Whatever the potential benefits of the system, it is the USER who must finally cause all actual benefits to be realized.

Only when he has a complete understanding of what he must do and what the system will do for him can the user be expected to attest that he will achieve the benefits defined. If the user fully understands the system but is unwilling to accept responsibility for achieving the benefits, it is probable that the proposed benefits will never be achieved and the decision to implement the system should be reconsidered.

b. Defining the New System

The definition of the system from the user's point of view is largely the identification of what the user will do and receive when the system is in operation. Major time requirements of the phase usually involve defining inputs which need to be provided and outputs which are to be produced. The controls which the user will provide and manual processes which he must perform are also identified. Finally, the activities of the phase include the formal identification of benefits of the system of other than direct dollar operational savings, so that the user will understand exactly what he is expected to achieve.

c. Analyst's Requirements

During the New System Requirements, the system analyst works at a level of great detail with respect to the system design. The specific content of each input document and output report should be defined. Each user organization function in the new system must be identified in sufficient detail to allow knowledgeable estimates of time requirements. The level of greatest detail at which the analyst works in the New System Requirements effort is in the complete specification of all data elements in the system. This includes not only input elements but also elements generated within the computer system together with the rules for their generation. It is, of course, the constant requirement of the system analyst that he define the system within the capability of the equipment and personnel, which must make it work. Therefore critical path diagrams, bar charts, work plans and related project control techniques should be utilized by the system analyst to develop and schedule resource requirements. A reassessment of the project at this point should be made by updating the Economic Evaluation Worksheets to develop an updated Economic Evaluation Summary. The requirement of defining the new

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

system within the capability of the proposed equipment and personnel, which must make it functional, can be met in this phase in terms of only general definitions of the necessary computer processes.

d. New System Data Specifications

The New System Requirements effort creates end products which completely define the system so far as each user organization is concerned. The specific end items which are required for the following project phases include complete data specifications (Element ID's, Data ID's, or Data Descriptions) for each data element, input or report contents for each input or output document, and all external controls and constraints.

(1) Element ID's

Precisely describe the content and characteristics of the data fields required as input or in reports of the new system. The data element should be checked to the actual document being considered for input for the purpose of consistency and completeness.

(2) Data ID's

Identify precisely, groupings of data elements which logically or physically occur together in some storage media and/or input/output document. The analyst should define the content and uses of the data group.

(3) Data Description

Technically describes the content and characteristics of the data required as input to the new system.

e. Documentation

Clear, consistent documentation is essential to a well-organized, efficient and easily maintained data processing system. Documentation requires the same high degree of accuracy and completeness that a programmer applies to his program. Without adequate documentation, a system is not complete and future analysis or revisions are virtually impossible. The dependency of all elements of the organization

INDEX NUMBER
05-2.0

upon documentation cannot be underestimated. Documentation has proven to be the only effective channel through which the analyst can communicate with those involved in making the system run properly.

A suggested list of information to be developed, standards, and suggested forms to be utilized should include:

(1) System objectives and abstract
(2) New organization chart (if applicable)
(3) Flow Chart of the New Manual System
(4) Narrative of new system
(5) Input/Output Document
(6) Output report layout
(7) List of reports to be produced
(8) Input/Output files
(9) Special hardware and performance requirements
New Flow Chart of forms distribution
(10) Design Constraints
(11) Summary of Controls
(12) Management Summary
(13) Glossary of terms
(14) Systems Code List

f. Management User Review

The completion of the document detailing the benefits of the system leads to a critical review point. It is at this time that the user organizations must approve the system requirements as defined, and the benefits as identified. The Management Summary serves as a useful and necessary document for giving non-technical management the major features and benefits of the system in a concise orderly manner.

The degree of success of a management presentation is closely related to the time spent in planning and preparation. One should avoid the natural tendency to delay documenting plans for the systems presentation program until the last possible moment. The following is a rough outline of the information which should be conveyed to management:

(1) Purpose of the Presentation
(2) History of the System Study

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

(3) Assignment Criteria Established for Analyst
(4) System Design Goals
(5) Problem Analysis
(6) New System Requirements
(7) Benefits and Improvements
(8) Installation Requirements
(9) Costs and Implementation Schedule
(10) Summary and Action Requested

One of the major purposes of the New System Requirements phase is to insure that the user's needs will be satisfied by the system. It is through this critical review at the end of the phase that the analyst achieves this assurance. After formal approval by the users, the analyst can proceed with the knowledge necessary to design a successful system.

INDEX NUMBER
05-2.0

3.0 SYSTEM DESIGN AND PROGRAM SPECIFICATIONS

3.1 INTRODUCTION

The System Design and Program Specification is an in-depth expansion of the system recommended in the Problem Analysis & New System Requirement Phases. Here, the overall system is expanded to more manageable subsystems, functions, programs and procedures. It is a refinement of the gross feasibility design into a detailed system design. It not only includes the on-line and off-line data processing details, but also the development of detailed specifications of control and checking features. Such features include input data validation, batch totals, audit trails, and other internal and external control measures to insure validity, accuracy, and precision of the system when it becomes operational.

In this section a minute analysis of the proposed system (that system concept approved in the New System Requirements) and all information required to program the system is developed. The finished product from use of this section should be reviewed by user management and finalized. Thereafter, Program Specifications are developed, hopefully with no change in the System Design. If subsequent changes in System Design are necessary as a result of management review, user management should be officially advised as to the effect these changes have on:

- Schedule to implement
- Resources required to implement

Therefore, the end product of the detailed design must specify in finite detail all requirements of the systems from which the data processing system can be programmed. The Detailed System Design and Program Specification steps specify not only what is to be achieved but how it is to be achieved.

3.2 DETAILED SYSTEM DESIGN

a. Objective

The previous section on Problem Analysis and New System Requirements dealt primarily with detailed fact finding and a

CONTINUED

1 OF 3

critical examination of the facts. In systems design, the analyst must develop a plan(s) for processing data. The facts already learned during the previous Problem analysis step through the user will determine what plan or plans will be most practical.

b. Possibilities in System Design

Systems Design is the creation in part or in full of a new scheme for processing data. The design must not only meet the problems discovered during problem analysis, but also the information demands specified by both management and operating personnel during the New System Requirement step. These demands are controlling factors in the final design. Approaches to systems design may be placed in the following general categories:

(1) Management Systems Design where the decision rules by management and the information required to implement them are open for examination and change. The maximum freedom is allowed the systems designer since none of the elements of the system - inputs, processing, or outputs - are fixed.

(2) Data System Redesign fall into two types:

- New equipment is used or data set media are changed which require altering the processing. This is internally generated redesign where inputs and outputs remain virtually unaltered.

- User stimulated redesign which results in modification of an existing system. There are varying degrees of changes relating to inputs, outputs and processing.

(3) Simplification is the traditional non-automated approach. This technique involves designing new forms that are easier to prepare and use, eliminating useless data, planning for more efficient flow of data, consolidating files, and division improvements to existing processing techniques. This approach may be simple and relatively economical but the level of

analysis from information produced by the system is rarely improved.

c. Selection of a Design Approach

(1) Many factors must be considered when selecting the most suitable design. Some are:

- The needs of the user
- Availability of appropriate software and/or hardware
- Time limitations
- Financial considerations
- Receptivity of management to proposed changes

(2) Each system under study must be recognized as unique and there is no quantifiable formula by which the the above factors can be easily related. Therefore all concerned must use considered judgment with each problem.

(3) The processing methods to be considered by the systems analyst in selecting a design approach are:

- Sequential Process - That type of process that operates from beginning to end in a pre-defined and rigid order, and on data structured in a similar manner.

- Random-Processing - That type of processing that operates without predetermined sequence. The significant aspect is that, because there is no sequence, all data that might be used during the process must be available at all times.

- Batch Process - That type of process that operates on all relevant data at a pre-determined time; usually in large masses and infrequently run. Batch processing may be either sequential or random.

- Real-Time Processing - The processing of individual inputs at the time of occurrance, rather than the most favorable time. Highly critical in any real-time process is the necessity for completing action on a specific element of input data within a specific time period.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

- Time Sharing - The use of single computer sys-systems (not necessarily a single central processor) by many working on different system applications terminals to the central processor at the same time. The terminals may be directly coupled to the central processor but normally are remotely located.

- Multi-Programming - The use of a single central processor by more than one operating program at the same time without significant degradation of individual program running time.

- Conversational Mode - The direct interaction of man-machine through a dedicated peripheral device that provides for two-way transmission of conversation type information.

NOTE: These processing methods have been presented in brief and generalized form. It is not the intent to completely stipulate the full scope and meaning of any of them in this section. These processing methods deal with tools or techniques that are inherent in any data processing system.

d. Planning the Detailed System Design

(1) Approvals

Approval of the previous steps (System Planning & Feasibility Study, Problem Analysis & New System Requirement) constitute the authority for entering the Detailed System Design step and provide the basic foundation upon which this step is built.

(2) Assigning Responsibility

Regardless of the size of the system or application, a data processing project leader and a user(s) project leader(s) need to be designated and assigned administrative and technical responsibility for the project. To facilitate the transition from the previous system development step to the Detailed System Design, it is desirable to assign the same persons

who worked on the previous steps, to work on the Detailed System Design. This will lessen the orientation impact and facilitate continuity.

(3) Work Plan

The initial task step of the designated project leaders is to develop a detailed project plan for this step. The leaders must determine in detail exactly what must be accomplished and develop a work plan on how it can best be accomplished. This involves all the elements of project planning. The importance of a well thought out detailed plan is never overstressed. It is essential that both data processing and user(s) management be kept advised of the plan and subsequent progress.

(4) Related Systems Considerations

In planning the detail systems design, it is essential that consideration be given to the requirements of all groups who have an interest in the system. This is particularly important when data from one system is to be interfaced with another system. In processing data through a system, all reasonable checks should be made to insure the validity of the data being processed. If the data is to be interfaced with another system, all efforts should be made in the initial processing to insure that the data meets, not only the edit criteria for the present system, but all editing requirements for any system with which the data is to be interfaced. Therefore, prior to the establishment of editing specifications, discussions should be held with all parties having an interest in using the data in order to establish all editing requirements.

e. General Guidelines

The culmination of Detailed Systems Design, regardless of the design approach selected, is the preparation of a comprehensive, but concise System Description of the proposed system. The System Description contains the following elements:

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

(1) Scope of the system

- Description
- Organizations supplying information
- Organizations receiving information

(2) General description of system

- Objectives
- Functions to be performed
- Methods employed
- Changes to current system

(3) Systems model - a graphic depiction of how the proposed system would work indicating samples of actual transactions and files. The system model is frequently abstracted for use as a presentation device to familiarize interested parties with the proposed system.

(4) Inputs

- Contents
- Source
- Frequency and average volume

(5) Outputs

- Specimen forms
- Distribution
- Frequency and volume

(6) Data sets

- Summary of contents
- Volumes
- Media of storage

(7) Priorities and schedules

(8) Summary of volumes - average, peak

(9) Applications system flowchart of all computer runs and document flow. The system flowchart graphically describes

the sequence in which all the programs in the system are to be performed. It should illustrate how the input requirements of the system are combined with the data files to produce the required output. Further, the chart should be so organized as to keep together all the processing which is performed on a common time scale; e.g., daily, weekly, monthly, etc. The system flowchart is intended to depict every computer program and file which will be required by the system.

(10) List of programs to be written

(11) Notes on conversion of master files and/or file establishment

(12) Summary of controls

(13) System development plan

- Time required for balance of System Development effort (Schedule)
- Staff requirements

(14) Estimates of costs

- Balance of System development effort
- Operation

(15) Date application will be "frozen"

f. Documentation

Documentation is necessary in order to successfully communicate the results of the Detail System Design effort. The input-output specifications and the functional computer process charts, together with data specifications prepared in previous phases, define the system completely. These create a "System Specification Package" which should be used as input to:

(1) A critical management review of the Detailed System Design.

(2) The program specification effort following management approval of the design.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

The following is a check list of standard and suggested forms to be utilized:

- Summary System Description
- System, Program, Job and File Numbering Scheme
- Updated System Flowchart
- Updated Program Run Narratives Drafts
- Updated Record Layout
- Updated File Layout
- Updated Glossary of Terms
- Updated System Codes Lists
- Updated List of Cards and Forms and Their Usage
- Updated Summary of Controls
- Data conversion Instructions with Sample Source Input Forms
- System Development Plan. Critical path diagram and associated bar charts used to develop resource requirements and associated schedules.

g. Management Review

The review by management at the end of the Detailed System Design effort is critical. It is the most meaningful time for management review of the proposed system and the most timely point for a functional and technical review of the process. Normally the functional review comes first. This insures that the new system meets user requirements. If the system design is functionally approved, then a technical review is held. Its purpose is to assure that the detail system design is technically sound and efficient. Once programming has begun, it is usually too late to make the system more efficient.

When the specific computer process has been selected for approval, the costs of the proposed system both one-time operating, becomes much easier to estimate than before. These better cost estimates, usually lead to revised savings estimates, which may vary significantly from the cost-benefit estimates which caused the initiation of this project. It should be noted that while some money has been spent on the project, the major amount is still unspent and officially uncommitted, until management gives its approval. Therefore, it is imperative that management be given a final review and an opportunity to finally approve the project based upon an improved understanding of the system, its associated costs, and the resulting net benefits.

The following documentation should be presented to user(s) management for its review and final approval:

(1) Summary Systems Description

(2) System Flowchart

(3) Program Run Narrative Drafts

(4) File Layout

(5) Record Layout

(6) Glossary of Terms

(7) Data Conversion Instructions with Sample Source Input Forms

(8) Benefits

(9) System Development Plan

3.3 PROGRAM SPECIFICATIONS

a. Introduction

The purpose of a program specification is to clearly and concisely document a routine to be coded for execution by a computer. The routines documented by the specifications must satisfy the input, output, and processing requirements of the system as given by the Detailed System Design and New System Requirements phases of the project. The specification writer's function is assumed to be primarily communication, not invention or design. The most fundamental question to be answered before beginning a set of program specifications is that of level of detail. As indicators of level of detail for specifications, the following guidelines apply:

(1) The completed program specification package is the key item of program documentation. Program compilation listings are secondary documentation. Questions of program logic are resolved by reference to the specifications package.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

(2) The programmer may have limited knowledge of the system being programmed and may have no detailed background in the application.

(3) The turnover of a program specification package to a programmer is accompanied by a briefing session in which the specification writer explains the specification in detail. The specification is not intended to stand alone. The specification writer, or someone with equivalent system knowledge, is available for questions and clarification during the entire programming phase. The program specifications package is a machine and language dependent document. The specification writer must be continually aware of combined hardware-software capabilities of the computer system for which the program is intended.

b. Organization of a Program Specification Run Narrative

A good program run narrative is organized like a good program - it is modular. A narrative for a complex update could have:

(1) A "mainline" section explaining in outline what the program must do, together with master and transaction file information.

(2) An update section which will specify how each transaction type affects master record fields.

(3) A report section which will specify how the update report is prepared.

(4) A fourth section which might specify how a special summary file is created.

(5) A final section which will define control totals required and their method of computation.

The arguments for modular specification writing are identical to those for modular programming. A carefully sectioned specification is easy to review and once accepted, easy to modify. In addition, if the specification writer is an experienced programmer, the modules of the specification narrative will be carried over into the coding, thereby increasing the value of the specification as program documentation.

c. The Program Specification Package

In addition to his primary goal of communication the necessary program design information to the programmer, the specification writer is also concerned with speed and conciseness in specification preparation. This section lists a number of items to be used for program specification. Since each program developed is unique, the specific format of each will differ. A program specification package should contain each of the following six items, however, the detailed content of each item is a function of the individual program being specified.

- Purpose of Program
- Detailed Program Run Narrative
- Program Run Chart
- English Language Program Flowchart
- File Layout and Record Layout Sheets
- Testing Information

(1) Purpose of Program

This section summarizes the major functions performed by the program. The problem, for which a programmed solution is being developed, is briefly described here.

(2) Detailed Program Run Narrative

The Detailed Program Run Narrative describes the coding requirements in terms of five major classes:

- Input and output data
- Method and processes
- Restrictions
- Error control and recovery
- Calling sequence (if a subroutine)

All information necessary to code the program should be included in the narrative in adequate detail.

(3) Input and Output Data

All input and output data are defined in the following:

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

- Item Name
- Item symbol
- Tag assignment
- Source (input data) or destination (output data)
- Bit designations
- Number of bits (or words)
- Scaling
- Accuracy
- Data grouping or intervals

The file and record layout sheets provide the majority of the needed I/O data referenced above. The narrative refers to them as required for processing.

(4) Methods and Processes

The methods and processes define the particular set of machine instructions executed by a program. Every program contains machine instructions that comprise one or more of the following methods:

- Computation

  If computation is performed by the program, the mathematical method should be prescribed in the form of equations or other mathematical representations.

- Control

  Any control which must be exercised by the program, in addition to whatever may be provided by an operating system, should be completely spelled out.

- Processing

  A general processing method requires more descriptive text than is necessary for a computational explanation. Complete program functions are outlined and detailed, stating solutions in definite terms. The methods of programming the problem must be clearly defined in detail. In addition to any applicable mathematical representations of the program solution, descriptive materials may be used to further clarify the methods used. Adequate detail in both mathematical representation and descriptive material may explain why one specific mathematical presentation is used rather than another.

(5) Restrictions

Some limitations imposed on the program are:

- Internal Core Storage: The actual memory locations and/or size available.
- External Storage: The actual number of external storage devices and/or portions thereof.
- Specific Registers: These items may be specified as not available or available only with special processing considerations.
- Data limits restrict the quantity of data handled by a particular program. The data tables, with the practical limits of length indicated, are an example of data limits.
- Execution time is specified as a limit under Restrictions if a program must be executed in less than a given period of time. This is a pertinent factor in real-time problems.
- Systems of tag designations for program and memory sections are often used in large-scale program development. If a system of tag designation is used, it is specified and described in a separate category of restrictions.
- Index register use is described by indicating the index registers totally unavailable for use if other than standard under the operating system being used, those available for use without storing and restoring. Not only are the types of index registers described, but descriptions should also be made of the values located in index registers on entry to and exit from the program.

(6) Error Control and Recovery

When an error is encountered, the program must contain a functional description of the action to be taken. The design of the hardware and operating system being used has some bearing on how error control and recovery are handled. The requirements of the problem to be solved is another major factor. In addition, existing programming conventions relative to error control and recovery techniques should be considered. To be included are such pertinent instructions as setting recovery addresses, computing and sorting backup/restart information, safeguarding decisions, getting error flags, printing error messages, etc.

(7) Calling Sequence

If the program being described in the specification is a subroutine, the procedure for utilizing the subroutine must be described. Included will be the necessary preliminaries to using the subroutine such as proper set-up of its inputs and transferring control to and from it. The description should also include the location and format of results generated by the subroutine.

(8) Program Run Chart

Depicts graphically, using flowchart symbols, the file inputs and outputs for the particular program being specified.

(9) English Language Program Flowchart

Detail Program Logic Flowcharts are time-consuming to prepare, unwieldy to modify, and as a rule, are not able to inform the programmer, as to what the program, as a whole, is actually supposed to do. Intricate logic is much better specified by a decision table or some other tabular method. As a specification tool, flowcharts perform best when their use is limited to showing the overall modular structure of a complex program, as part of the specification's introductory section. An English language flowchart showing the logical approach to the program should be included with detailed flowcharting left to the programmer. Only the overall data flow and major decisions are outlined in the English language flowchart.

(10) File Layout and Record Layout Sheets

As stated in the section on Detailed Program Run Narrative, the above items supply the basic I/O data for the program. Associated system codes and glossary of terms are also to be provided to the programmer. The narrative describes how the data is to be processed.

(11) Testing Information

Complete information defining any testing needed to insure program performance exactly as specified should be included. This includes all necessary input data to verify all paths of the program and the corresponding required outputs. Any special conditions to be considered should also be specified.

d. Writing the Program Narrative

Program Narrative specification writing begins with a thorough study of the System Design Manual, which serves as a basis for all specifications. From the design manual are obtained:

- Summary System Description
- System, Program, Job and File Numbering Scheme
- Updated System Flowchart
- Updated Program Run Narratives Drafts
- Updated Record Layout
- Updated File Layout
- Updated Glossary of Terms
- Updated System Codes Lists
- Updated List of Cards and Forms and Their Usage
- Finalized Samples of Output Cards, Forms, and Output Report Layouts
- Updated Summary of Controls
- Data Conversion Instructions with Sample Source Input Forms
- Benefits
- Systems Development Plan. Critical path diagram and associated bar charts used to develop resource requirements and associated schedules

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

In preparing each narrative the specification writer:

(1) Studies the data from the Systems Design Manual to determine exactly what processing must occur to obtain required outputs from inputs.

(2) Confers with the systems analyst responsible for the design if further information is required.

(3) Decides upon the subsections (modules) into which the specification will be organized.

(4) Writes the specifications required for each subsection. In general, each subsection will be based upon one central narrative supplemented by references to one or more supporting items (e.g., file layout, record layout, decision table, output report layout, etc.). During the specification process the specification writer mentally programs each subsection, including in the narrative explicit instructions for all error conditions which may arise during the processing. In many cases these instructions are arrived at after discussion with the system designer.

(5) Reviews the entire narrative for coherence and logical presentation. The completed narrative must read as a unified whole. It should not appear to the programmer as a disconnected set of processing functions.

(6) Delivers the specification package (e.g., the narrative, together with the supporting documents) to the system designer for review.

e. The Program Specification Package

(1) This package provides:

- The vast majority of information upon which the programmer develops his program.

- The vast majority of the required final program documentation.

INDEX NUMBER
05-3.0

(2) This package should consist of:

- Program Run Chart
- Program English Language Flowchart
- Program Narrative
- File Layout(s)
- Record Layout(s)
- Output Record, Card, Special Form Layout(s)

It is assumed that the programmer has access to all other required system design information such as:

- Code List
- Glossary of Terms
- Summary of Controls
- System Flowchart

(3) Technical Review

The "Program Specification Package" should be reviewed by manager of systems design for approval and then assigned to a programmer for the development of the actual program.

4.0 SYSTEM DESIGN STANDARDS

4.1 INTRODUCTION

System Design Standards are guidelines to be used in the design and implementation of the new systems. It is expected that these guidelines will provide a basis for efficient, logical programming, efficient operation, and a basis from which maintenance and operation changes can be implemented. Systems that do not adhere to the Standards will cause disruption and inefficient use of the Department's resources. It is expected that all system design analysts will adhere to these system design standard guidelines wherever possible. If, in a special case, it is in the judgment of a systems analyst better to not adhere to the specific standards, written permission should be obtained from the Department Manager.

4.2 TELEPROCESSING STANDARDS

a. Update and entry TPDs are limited to 17K. This is the size of the overlay areas in those tasks that handle file maintenance transactions. Inquiry TPDs are limited to 10K.

b. SIMLATing (calling TPDs) cannot expect a record to be in its I/O area after SIMLATing to another TPD which does file I/O of its own on the same file.

c. All TPDs are tested using the background tester until the final tests which are made on-line against test files, or the backup real time files, during the 3:00 A.M. to 6:00 A.M. test period.

d. Any tests that change or add records to the file are followed by a utility program to dump the affected file so it may be checked for accuracy.

e. Cylinder indexes are kept in core in order to speed up responses.

f. Dummy records are added to the General index in the areas where a high volume of adds occur. These dummy records are then rewritten as needed. This saves overflow chaining which is time-consuming.

g. Transaction codes that generate a series of transactions give up control after each transaction and route the remainder of the input data back through the input queue. This is done to avoid one input transaction tying up the system for an excessive length of time.

h. In all cases where possible a key which is sequential in nature is inverted so as to cause a more random distribution over the file.

i. Priorities - all transactions are assigned a certain priority relative to all other transactions in the system. These priorities insure a 10-second or less response 95% of the time to operational inquiries from field officers. Maintenance inquiries from in-house terminals always have a lower priority.

j. All transactions require a radio number or 'test' following the inquiry code.

k. All transactions are logged to tape to provide a history of activity on the network. The log also provided information for rebuilding the files.

4.3 BATCH PROCESSING STANDARDS

a. Size limited to 54K. This was due to the ANS COBOL compiler needing 54K and we chose to use ANS COBOL because of the additional facilities it offers over D COBOL. Our objective was to keep the TP partition to a size that would allow a 54K background partition.

b. Limited to three tape drives.

c. Internal Sort was used as much as possible to reduce the number of jobs to run and to reduce the number of times files were handled manually.

d. Source decks were kept on disk using DOS Utilities to keep them updated and ANS COBOL Source Statement Library facilities for compiling and cataloging them.

e. Programs are cataloged in a Core Image Library instead of executing them from an object deck, therefore no decks were punched from COBOL compiles.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
SYSTEMS DEVELOPMENT AND DESIGN STANDARDS

DATE ISSUED | DATE REVISED

f. Programs that changed the on-line files put a message out to the operator to stop allowing updates from TP and waited for a response from the operator before continuing. They also put a message out to the operator when they completed so that updates could again be allowed from TP.

g. Programs that added records to an on-line file had to assure that updates from TP were not taking place and they had to put messages out to the operator to close the file for TP before the Batch program closed the file and open it again after the Batch program had closed the file.

NOTE: The procedures in f and g are necessary because inquiries are operational as much of the time as possible and DOS allows updating ISAM files from different partitions at the same time.

4.4 EQUIPMENT UTILIZATION STANDARDS

a. CPU interfaces must operate at a minimum of 2400 band.

b. All terminals must be buffered.

c. All CRTs must have a printer available for use in printing desired information.

d. Channel separation on high activity files.

e. If possible arrange files so that one module of a 2314 device is left vacant for hardware maintenance purposes.

f. Execute TP program in Foreground 1.

g. System logging should be on a tape instead of disk. There is less I/O involved and the data is placed where it can be permanently saved.

h. Busy files should not share the same module.

i. Utilize disks for sorts.

INDEX NUMBER
05-4.0

TABLE OF CONTENTS

06 PROGRAMMING STANDARDS AND AIDS

1.0 GENERAL INFORMATION

1.1 Preface

The various approaches and techniques that can be used to program a computer operation will normally vary in direct relationship to the experience and background of the personnel involved. This allows each programmer certain latitude in the use of advanced programming techniques and encourages the self-improvement of each programmer's technical capabilities. However, it is this programming latitude that most frequently creates the many time-consuming problems associated with future program maintenance and/or modification for implementation at other locations. Consequently, it is essential to future programming economics that the certain techniques be selected as standards; discourage undesirable or unusually difficult techniques and require thorough documentation of each detail contained in all its programs.

1.2 Purpose

The purpose of this programming standards section is to describe the techniques which have been selected as standards and to describe the documentation required for each program. Compliance with the standards contained in this manual will result in improved efficiencies whenever program modifications are prepared by other than the original programmer, and permit program implementation at other locations or agencies with a minimum of reprogramming effort.

1.3 Objectives

The objectives of this Programming Standard manual can be summarized as follows:

a. Uniform programming methods
b. Complete and meaningful documentation
c. Better reprogramming and program modification control
d. Efficient utilization of scarce programming resources

The issuance of programming standards is not intended to discourage the use of advanced programming techniques or restrict

original thinking. Instead, it is hoped that this guide will facilitate the efficient exchange of data and techniques between data processing personnel and thus improve the technical capabilities of all concerned.

1.4 Use of the Program Specifications

The Programming Specifications as developed by the systems analyst, is the complete description and documentation for the entire operation or Project to be programmed. Therefore, the programmer should recognize that the programming specifications are his sole source of input. If there are any questions or inadequacies, they must be resolved with the specification writer. By the same token, he must also have the opportunity to review and reject the programming specifications if they are inadequate. If the programmer accepts the programming specifications, he must then be responsible for implementation of this system as designed. The first step in evaluation of the programming specifications is determination that the standards of documentation have been adhered to by the system analyst. The programming method specifies six tasks to be performed by the programmer. These are listed below with reference to appropriate standards:

a. Task 1 - Develop Detail Program Flow Charts

The programmer's first task is to develop a detail flow chart for all complex processing. High level flow charts are required as a standard for documentation, while detail flow charts are discouraged. Therefore detail flow charts are only to be prepared when necessary.

An alternate to complex detail flow charts is the use of decision tables.

b. Task 2 - Coding

The programmer prepares instruction sequence (coding) from the decision table and/or logic flow chart.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS
AND AIDS

DATE ISSUED | DATE REVISED

c. Task 3 - Desk Checking

The programmer is responsible for manually verifying the logic and coding of the program as defined in the program specifications.

d. Task 4 - Program Preparation

Program preparation includes the following activities:

(1) Key Punch code.

(2) Removal or correcting of errors which occurred during preparation of machine readable form.

(3) Production of an error-free compilation or assembly.

e. Task 5 - Test Data Preparation

In this task, a set of data is developed specifically to test the adequacy of a computer program.

f. Task 6 - Program Testing

The purpose of this task is to locate and correct any errors in the computer program. This is the final task before turnover to the systems analysts for system test.

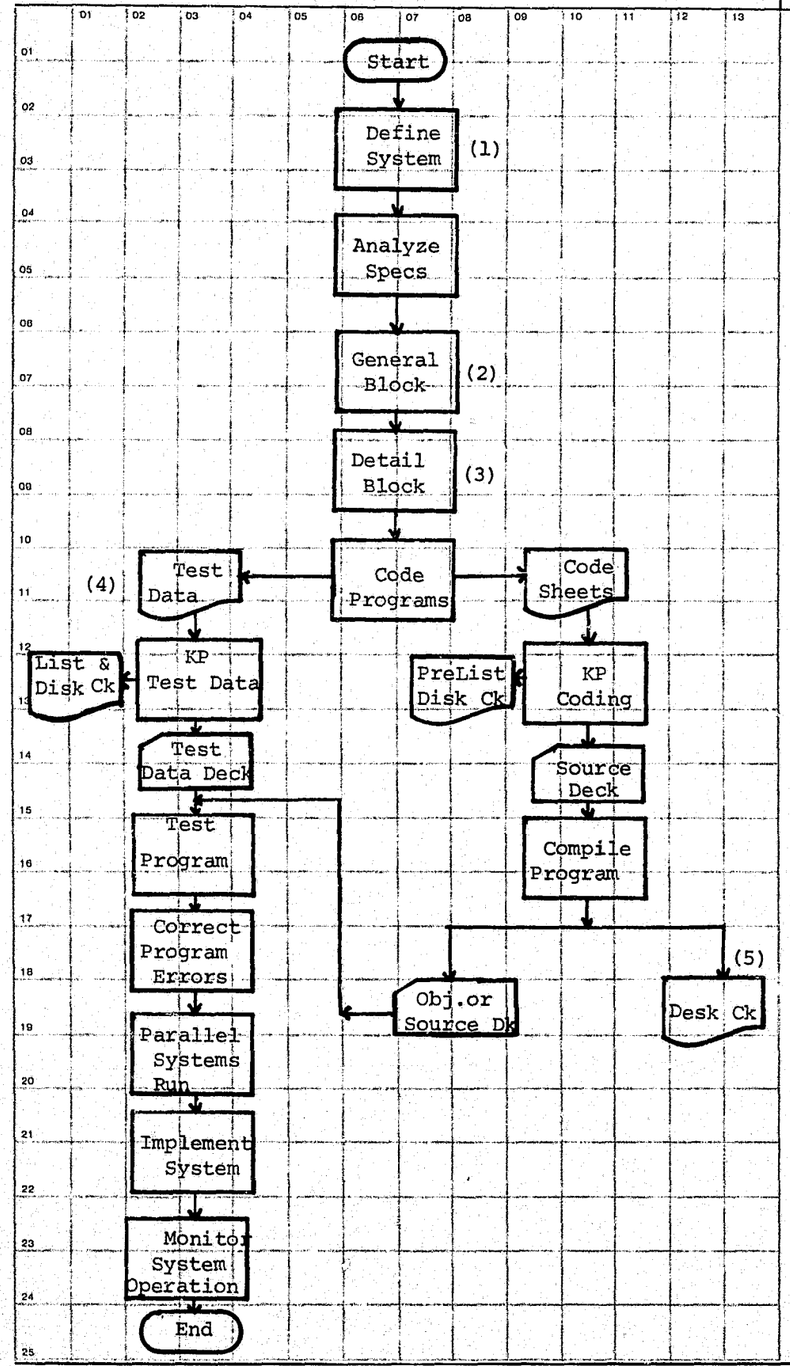
These six tasks are usually performed in an iterative fashion. That is, the majority of the work for each task is performed in the six-step sequence as shown. However, iterations back to earlier tasks to correct subsequently discovered situations is the rule, rather than the exception.

INDEX NUMBER
06-1.0

SYSTEM FLOW CHART

Page ____ of ____

FLOWCHART



DESCRIPTION

(1) Develop Systems Specifications

(2) General Block Diagrams for Each Program

(3) Not required if Coding can be Accomplished without this step

(4) Develop Test Data Sheets from Representation Source

(5) Correct Compiled program

System No.	System Title: System/Program Development Flow		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

INDEX NUMBER 06-1.0

2.0 NAMING CONVENTIONS

2.1 Program Name, Data Set Name, and Report Name

a. Before programs are written, programming personnel will obtain a program name from their Unit Supervisor. This requirement is necessary to maintain a program control inventory and prevent possible duplication of assigned program names.

b. The use of a five-character program name allows the ability to append alphabetical suffixes to the five-character name to represent various data sets and reports in a program. Thus, compatability and direct association are achieved between a program and the output it creates.

c. Use of the five-character program name also applies to utility programs, including sorts, when used for a specific system application since the User control instructions would be designed to meet the needs of the specific application.

2.2 Program Name Construction (Five Characters)

a. The first character represents the Data Processing Division Unit responsible for the program. Characters to be used are, 'C' for Management Information Unit, 'J' for Criminal Justice System Unit, 'R' for On-Line Telecommunications Systems Unit, and 'S' for Operating Systems and Research Unit.

b. The second character is a code representing the System for which the program is being written.

c. The third, fourth and fifth characters are for numerical sequence within Unit.

2.3 Data Set Name Construction (First Seven Characters)

a. An alphabetical suffix is added to the program name to designate the type of output; 'D' for disk, 'T' for tape, 'L' for printer and 'P' for punched card.

INDEX NUMBER 06-2.0

The seventh character is used for numerical sequence within type of output.

b. Data Set Name for TLBL Cards may be up to seventeen characters, for DLBL Cards up to forty-four characters.

2.4 Report Name Construction and Use

a. The first seven characters of Data Set Name are to be used as the Report Name. The Report Name is to be printed as part of a heading line at the top of each page of the report. For standardization purposes, the left side of the first heading line is suggested for placement of the Report Name.

2.5 Character Writing Conventions

In order to reduce the amount of written text on the flowchart or decision table, the use of certain symbols is permitted.

+	plus, add	#	number
-	minus, subtract	;	compare, ratio of
±	plus or minus	∑	summation of
÷	divide	@	at
.	multiple, times	∴	therefore
=	equal	HI	high
≠	not equal	LO	low
<	less than	EQ	equal (used with HI & LO)
≮	not less than	π	pi, 3.1416
>	greater than	Ø	numeric zero
		Z	letter 'z'

3.0 PROGRAMMING FOR OPERATOR EFFICIENCY

3.1 Systems Interruptions for Operator Instructions

Pause cards must not be used in background programs as a means of communicating with the operator. Another control card (STOP) must be used if operator communication is necessary. The format of the STOP card is: STOP in card columns 1-4 followed by at least one blank column followed by comments up through card column 72.

3.2 Sequence Numbering of Source Decks

All source decks must be sequence-numbered. Sequence numbers for COBOL decks are to be in card columns 1-6, for assembly language decks in card columns 77-80.

4.0 FLOWCHARTING (HANDDRAWN FLOWCHARTS)

4.1 Program Flowcharting Standards

a. Flowcharting Techniques -- Adherence to the flowcharting techniques outlined in IBM Manual C20-8152 will be required at all times.

b. Flowcharting Worksheets -- All program flowcharts will be drawn up on IBM Form No. X20-8021-2. Exceptions to this rule must be approved by the programming manager.

c. Flowcharting Template -- All program flowcharts will be drawn up using IBM Flowcharting Template X20-8020.

d. Flowchart Labeling -- Entry point labels, those other than constants, shall be labeled as follows:

(1) First character: Always use a "P".

(2) Second and third characters: This represents the flowchart page number.

(3) Other characters: On form number X20-8021-2 the blocks are numbered with one alphabetic character and one numeric character. These two characters will be the last two positions of a label. For example, the label PlØH3 represents block H3 on flowcharting worksheet page #1Ø.

4.2 System Flowcharting Standards

a. System Flowcharting techniques outlined in IBM Manual C20-8152 will be adhered to at all times.

b. Form 1218-067 will be used for all System Flowcharts.

c. All system flowcharts will be drawn using IBM Flowcharting Template X20-8020.

d. Each page of the flowchart will be assigned a section number. The first page will be section #10; the second page will be section #20, etc., increasing by 10 each time to allow for insertion.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

e. Each block on the chart that represents some sort of processing is assigned a step #10; the second block step #20, etc., increasing by 10 each time to allow for insertion.

f. Each step will be described in the space provided on the right side of the form.

g. Each step drawn will be properly described:

(1) EDP Programs Blocks:

Steps - Enter machine number and program name in the block.

Description - Enter estimated running time. If this is a standard utility program, enter "UTILITY PROGRAM #XXXXX" and estimate running time.

(2) Keypunch Block:

Block - Enter "KEYPUNCH"
Description - Enter keypunch job number and estimate number of cards to be punched.

(3) Data Control Block:

Block - Enter "Data Control"
Description - Enter detailed editing, balancing, and/or salvaging instructions.
Estimate volume.

h. Input/Output Symbols

(1) Magnetic Tape and Disk -- Enter file name, unit, record length and blocking factor.

(2) Report - Enter report name and title

(3) Source Document - Enter form number and title

(4) Card - Enter file name, title (Form number and/or report number also, if applicable)

i. Chart Continuity and Clarity

(1) Insufficient space -- If there is not enough space within the symbol to include both the title and number, leave the number outside the symbol. Be careful to leave no doubt as to which symbol the number is associated with. Use an arrow to indicate the association where necessary.

(2) Off-page connectors -- If output from a step on one page is input to a step on another page, the symbol showing the disposition of the output from the first step should contain the section number(s) and step number(s) of the second step. The symbol showing the source of the input to the second step should contain the section number and the step number of the first step.

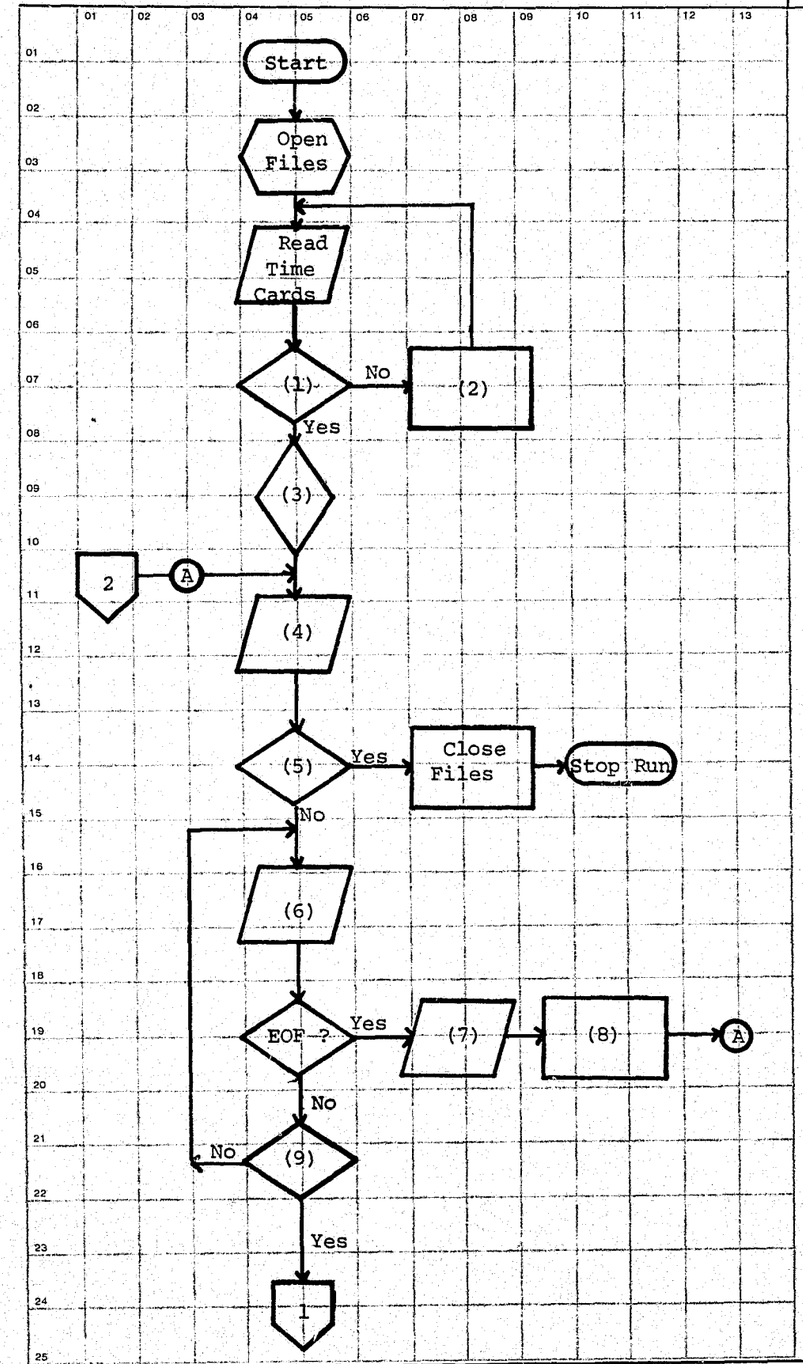
(3) Output used in another system -- If some output is to be used as input to a step in another system, the disposition symbol should contain the system name as well as the section number and step number within that system. If the section number and step number is not known, enter the program name along with the system name in the disposition.

(4) Output distribution -- Do not enter distribution on the chart. Simply indicate that the output is routed to Data Control.

EXAMPLE OF HIGH LEVEL FLOW CHART

SYSTEM FLOW CHART

FLOWCHART



DESCRIPTION

(1) End of File?

(2) Release Time Record to Sort

(3) Sort Time Records on Employee Number

(4) Return a Sorted Time Record

(5) End of File?

(6) Read Employee Master File

(7) Display "Master Record Not Found"

(8) Close and Reopen Master File At Start

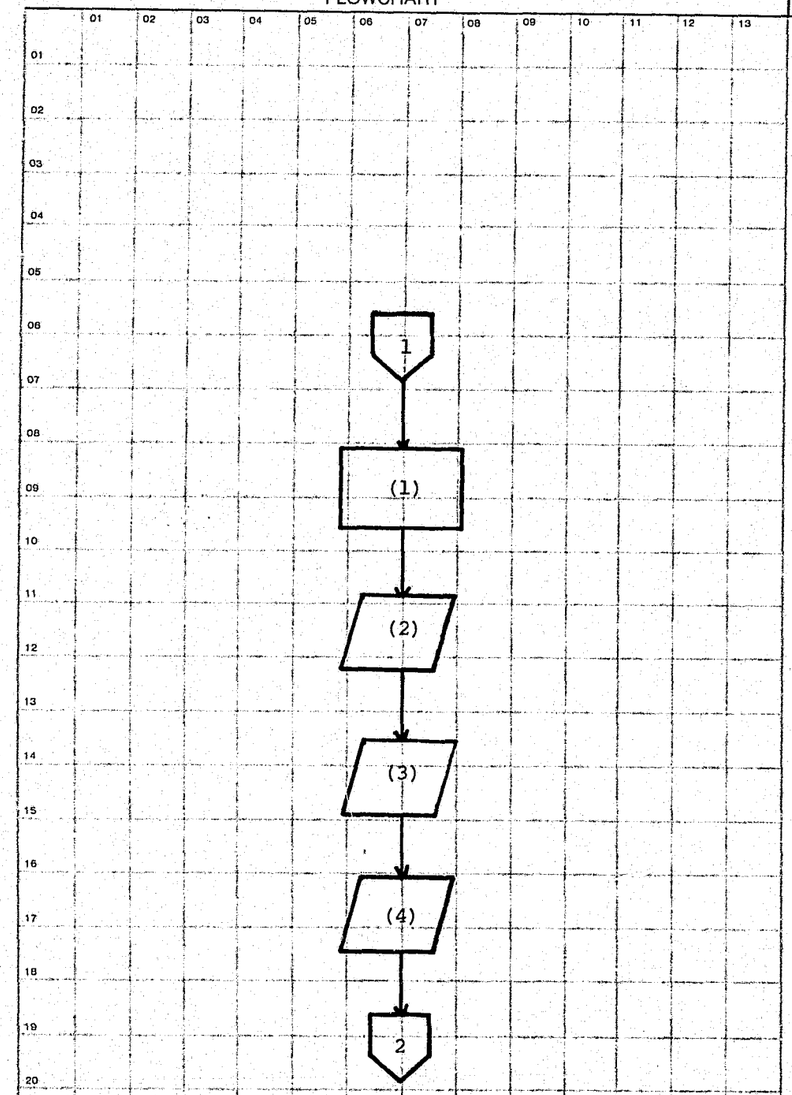
(9) Does Master File Record Equal Time Record?

System No.	System Title: Example of English Language Flow Chart		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

SYSTEM FLOW CHART

Page 2 of 2

FLOWCHART



DESCRIPTION

(1) Compute Payroll From Time Record and Master File Data

(2) Write Ledger Records

(3) Update Master File Year-to-Date Payroll

(4) Write Paycheck On Printer

System No.	System Title:		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

INDEX NUMBER 06-4.0

5.0 THE USE OF DITTO FOR UTILITY PROGRAMS

a. There is available a method of using DITTO through the use of control cards. In using DITTO we can remove the decision requirements from the console operator by preparing control cards and submitting the job in a Normal Job Stream Environment. This will speed throughput of jobs and lessen the possibility of mistakes.

b. The control card operation is suggested for sampling file data, card to card and card to printer dittos, etc. It should not be used for lengthy DITTOs such as a whole tape to printer.

c. To denote control card operation the user must submit between the Job card and Exec card a // UPSI 1 card. DITTO will test the communications region to determine whether control card or console operation is desired. Depressing the interrupt key at any time overrides the control card operation.

d. All required control card information can be contained on one card. Each control card is treated as a new operation. The control card format is as follows:

```
CC 1-7    $$DITTO
CC 10-12  Function code (see table)
CC 16     Parameter 1,.....,Parameter N
```

e. Parameters are in standard key word format. Each parameter must be separated with a comma with no imbedded blanks. A blank stops the card scan. Lower case letters denote user-supplied information. Parenthesis ( ) denote optional parameters. Refer to the "Parameter Requirements" for a listing of parameters associated with each function.

- FUNCTION CODES

Code	Alternate	Description
CCU	CC	Card to Card
CCS	--	Card to Card with sequence numbers and deck name identification
CPU	CP	Card to Printer in character format
CHU	CH	Card to Printer in character and vertical hexidecimal format

INDEX NUMBER 06-5.0

Code	Alternate	Description
CTU	CT	Card to Tape Unblocked
CTR	--	Card to Tape Reblocked
TCR	TC	Tape to Card unblocked or blocked
TPU	TP	Tape to Printer unblocked in character format
TPR	--	Tape to Printer reblocked in character format
THU	TH	Tape to Printer unblocked in character and vertical hexidecimal format
THR	--	Tape to Printer reblocked in character and vertical hexidecimal format
TTU	TT	Tape to Tape (1 to 99 files)
TFA	--	Tape to Printer using Type A forms control
TFD	--	Tape to Printer using Type D forms control
WTM	--	Write Tape Mark
REW	--	Rewind Tape
RUN	--	Rewind and Unload Tape
FSF	--	Forward Space tape file
BSF	--	Backspace tape file
FSR	--	Forward Space tape record
BSR	--	Back Space tape record
ERG	--	Erase Record Gap
DDU	DD	Disk Dump. Disk to printer in character and vertical hexidecimal format unblocked

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

Code	Alternate	Description
DDR	--	Disk Dump with reblocking of data records into logical record length
SDU	SD	Disk Dump for split cylinder files
SDR	--	Disk Dump for reblocked for split cylinder files
EOJ	--	End of Job

- PARAMETER REQUIREMENTS

CC 1-7	CC 10-12	C 16
$$DITTO	DDU	INPUT = SYSnnn,BEGIN = ccchh,END = ccchh
$$DITTO	DDR	INPUT = SYSnnn,BEGIN = ccchh,END = ccchh, RECSIZE = nnnnn
$$DITTO	SDU	INPUT = SYSnnn,BEGIN = ccchh,END = ccchh
$$DITTO	SDR	INPUT = SYSnnn,BEGIN = ccchh,END = ccchh, RECSIZE = nnnnn
$$DITTO	TPU	INPUT = SYSnnn (,NBLKS = nnnn)
$$DITTO	TPR	INPUT = SYSnnn,RECSIZE = nnnnn (,NBLKS = nnnn)
$$DITTO	THU	INPUT = SYSnnn (,NBLKS = nnnn)
$$DITTO	THR	INPUT = SYSnnn,RECSIZE = nnnnn (,NBLKS = nnnn)
$$DITTO	TFA	INPUT = SYSnnn
$$DITTO	TFD	INPUT = SYSnnn
$$DITTO	TCR	INPUT = SYSnnn
$$DITTO	CTU	OUTPUT = SYSnnn
$$DITTO	CTR	OUTPUT = SYSnnn,BLKFACTOR = nn.
$$DITTO	TTU	INPUT = SYSnnn,OUTPUT = SYSnnn
$$DITTO	WTM	OUTPUT = SYSnnn

INDEX NUMBER
06-5.0

CC 1-7	CC 10-12	C 16
$$DITTO	REW	OUTPUT = SYSnnn
$$DITTO	RUN	OUTPUT = SYSnnn
$$DITTO	FSF	OUTPUT = SYSnnn
$$DITTO	BSF	OUTPUT = SYSnnn
$$DITTO	FSR	OUTPUT = SYSnnn,NBLKS = nnnn
$$DITTO	BSR	OUTPUT = SYSnnn,NBLKS = nnnn
$$DITTO	CPU	
$$DITTO	CHU	
$$DITTO	CCU	
$$DITTO	CCS	DECKTYPE = xxx,DECKNAME = xx...x
$$DITTO	EOJ	

- PARAMETER DEFINITIONS

Parameter	Description
INPUT=SYSnnn	Logical input device
OUTPUT=SYSnnn	Logical output device
BEGIN=ccchh	Lower disk extent
END=ccchh	Upper disk extent
NBLKS=nnnn	Number of tape blocks
RECSIZE=nnnnn	Logical record size
BLKFACTOR=nnn	Output blocking factor (CTR)
DECKTYPE=xxx	CCS Decktype
DECKNAME=xxxxxxxx	CCS deckname (Ø-8) characters Decknames which are less in length than required will be left justified and padded with blanks

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

- EXAMPLES OF CONTROL CARD JOBSTREAMS

Example 1: The following is a typical jobstream for a card to card DITTO.

```
// JOB DITTO
// UPSI 1
// EXEC DITTO
$$ DITTO CCU
DATA CARDS (WITHOUT/* OR /&)
/*
$$ DITTO EOJ
/*
/&
```

Example 2: The next DITTO jobstream would be used to skip the standard label on a tape then unblock the records for printing on SYSLST. Upon completion the tape is rewound and unloaded.

```
// JOB DITTO
// UPSI 1
// EXEC DITTO
$$ DITTO FSF OUTPUT = SYSØ1Ø
$$ DITTO THR INPUT = SYSØ1Ø,RECSIZE = ØØØ81,
   NBLKS = ØØØ5
$$ DITTO RUN OUTPUT = SYSØ1Ø
$$ DITTO EOJ
/*
/&
```

INDEX NUMBER
06-5.0

Example 3: The following jobstrean could be used to dump from a disk cylinder 18 head Ø.

```
// JOB DITTO

// UPSI  1

// ASSGN SYS Ø1Ø, X'143'

$$DITTO DDU INPUT = SYSØ1Ø,BEGIN = 1Ø8ØØ,END = Ø18ØØ

$$DITTO EOJ

/*

/&
```


DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

6.0 CODING SHEET FORMAT - COBOL

a. All program coding should be done on a standard coding sheet, supplied by the hardware vendor. Each sheet should have the heading completed in case a batch becomes separated during key-punching. The heading should include:

- Program name
- Programmer name
- Date sheet was completed
- Sequential page number

b. All coding should be in black pencil to allow for corrections by eraser. The last five lines of each sheet should be left blank to permit insertions after the original coding is complete.

c. All lines of program code should have a sequence number in columns 1-6 and a unique identification code in columns 73-8Ø. The sequence number must consist of only numeric characters and is constructed as follows:

- Columns 1-3 Page number
- Columns 4-6 Line number

d. The line numbers for each page should start at Ø1Ø and be incremented by 1Ø. Insertions made at the bottom of a page must be given a line number to correspond to the place where the instruction is to be inserted. In addition, the place where the insertion is to go shall be marked with an arrow in the left hand margin.

e. The identification code should be constructed as follows:

Card Columns	Contents
73 - 78	Six character program number
79 - 80	Two digit module number, if necessary*

*NOTE: The module number is used only if the program is composed of a calling routine and one or more subroutines. The calling routine should have module number Ø1; each subroutine should be numbered sequentially starting with Ø2.

6.1 Identifiers

a. General

The requirements of a standard labeling system are:

(1) Uniqueness (general terms such as LOOP, NEXT, GO, START, or BEGIN should be avoided).

(2) Description of type (instruction and data labels should not be confused).

(3) Indication of sequence within the program.

(4) Compatibility with the programming system.

In addition, the standard labeling system facilitates interchange of programs between groups by curbing use of individuals' names, local humor, and complicated abbreviations.

b. File Identification numbers consist of two items:

(1) Identification of the program or operation which generates the file.

(2) A file code which indicates the type of file and its sequence number.

c. Labels

Within programs or modules, labels describing data areas or instruction areas, should be prefixed by a one-character code according to the system indicated below:

(1) An Accumulator, counter, switch, or any other area which is modified.

(2) A block number, branch point or other instruction label.

(3) Any constant.

(4) Data.

DATA PROCESSING
MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

Descriptive names or three digit sequential serial numbers should be used after the prefix. Some examples follow:

Label	Meaning
BENDIT	A branch point named "ENDIT"
ADAYS	A counter named "DAYS"
CMASK	A constant named "MASK"
DØØ4	Data area 4
BØ17	Branch point 17

6.2 Standard Internal File Labels

a. Every family of computer or software operating systems generally has unique label types. The labels generated by one manufacturer's software support package are often not acceptable to another software package. For example, tape labels generated by IBM's 1401 IOCS, are not in the same format as those generated by IBM's System 360 DOS. Thus it is difficult to establish universal standard formats for internal file labels. Instead, the standard label formats supplied by the hardware manufacturer should be used in all cases. As a minimum, the header label should provide:

(1) Identification of type of label (e.g., standard header label).

(2) File identifier.

(3) File generation/version number.

(4) Creation date.

(5) Expiration date.

b. The trailer label should indicate, as a minimum, End of Volume or End of File and a file record or block count.

INDEX NUMBER
06-6.0

c. Prior to the start-up of a new system, it should be the responsibility of the systems analyst and the operations section to decide for each file what label checking will be specified.

d. In general, all files except for "work files" should have standard header and trailer labels. Work files (also called "scratch files") are files which are written and read by the same program without ever being used by another program. In other words, the contents of a work file does not have to be saved for further use after the program which created it has been run.

7.0 GENERAL PROGRAMMING RULES

7.1 Programming Do's and Don't's

DO all editing in the initial program of a series.
DO use any available macro libraries.
DO use the subroutines or modular concept of programming.
DO use utility programs.
DO use the standard identifiers for I/O files.

DON'T punch cards on-line if off-line equipment is available.
DON'T print large volume reports on-line, generate print files if off-line or multiprogramming facilities are available.
DON'T use small blocking factors on tape if larger ones can be used.
DON'T use the console typewriter for messages that cannot be acted upon by the console operator. In particular, don't print control totals on the console typewriter. EOJ message is permissable.

7.2 Mainline Conventions

a. The mainline routine makes all decisions governing the flow of the data to the proper processing routines. That is, no processing routine can direct data flow to another processing routine on the same level. However, lower levels of processing routines may be entered from a higher processing routine if required.

b. All common areas should be defined as part of the main program. It is feasible to define a common area which is shared by two processing routines but which is not defined in the mainline program. For example, a computation routine might create a table called RESULTS which is accessed by a different routine which summarizes the data stored in RESULTS. RESULTS would have to be defined in the computation routine and in the summary routine. It would not necessarily be defined in the mainline routine since the mainline routine never uses RESULTS. This procedure is not good programming practice.

7.3 Size Limitation

Modules should be small enough so that they may be conveniently understood, quickly written, and thoroughly tested. Time should be spent planning the structure of the program rather than on writing large sophisticated modules.

7.4 Re-Entrant Modules

A module is said to be re-entrant when it makes no stores into itself and can be used at any time by more than one program perhaps during an interrupt. Any answers, switches to be set, work areas, etc., can be stored in registers or in the calling module. Whenever possible, assembly language modules should be re-entrant. Generally, compilers do not offer the programmer a choice of making COBOL or FORTRAN modules re-entrant. The status of module (whether or not re-entrant) will be noted on the module description at the beginning of the module assembly listings.

7.5 Processing Routine Conventions

a. Ideally, a separate processing routine (i.e., module) should be created for each logical segment of the program. It should accomplish one task in its totality. Each processing routine is complete within itself, with its own defined areas when such areas are for the exclusive use of that routine. No decisions made outside the segment should determine the processing within a segment and likewise, no decision within a segment should determine the processing outside the segment.

b. Each routine is designed so that it is, in effect, a closed (out of line) subroutine. Control is transferred to the processing routine from the mainline routine, and when the routine has performed its function it sends control back to the mainline routine. A processing routine may transfer control to a multiple-use subroutine. When that routine has performed its function, it transfers control back to the processing routine. Input or output functions that affect only one processing routine may be performed by that routine. All segments must contain their own housekeeping to assure non-interference with other segments. Any results of processing to be passed back to higher level modules should be stored either in an area in the higher level module, or in registers.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

7.6 Aids to Operation

a. All programs using preprinted forms should include a forms alignment routine which prints a line, stops to allow the operator to check and/or adjust the alignment and either prints another line or exits depending on instructions from the console operator. All programs using the printer should start and end with a "skip to top of page" instruction.

b. It may be necessary to terminate programs which generate punched cards with an instruction to punch a blank card in order to run the last blank card into the stacker.

c. Tape programs should include a rewind instruction at the beginning and end of the program unless the tape is to be used in read-backwards mode.

d. If decisions are to be typed in by the operator, always try to cover all possibilities. Do not assume that if the answer is not YES, it must be NO. It could have been YET, or YEA, or any of a number of other spellings, indicating operator error.

e. If the decision is to be communicated through the setting of console switches, always try to record the result of the key setting for logging purposes.

f. When establishing a loop, always reset the loop at the beginning, not at the end. In addition, when two loops are used, one inside the other, be sure that the index of the outside loop has the smaller range.

g. Never assume that storage has been cleared (set to blanks or zeros) at the beginning of the program. Always clear the necessary areas during initialization. Always set counters and switches to their initial conditions during the housekeeping operation.

h. When matching up a code number compare on the whole number, not just part of it.

i. When setting up a series of branches make sure that the most likely branches are tested first.

INDEX NUMBER
06-7.0

7.7 Program Block Comments

Extensive comments should be made prior to the start of each program block. These comments should define what the block is done, and, if applicable, how it goes about doing it. These comments are retained as part of the master source program listing, and become an invaluable aid in helping others to modify and maintain the program. Comments, next to or preceding each line of source program coding, are also a valuable maintenance tool and should be used extensively.

7.8 Checkpoint/Restart Procedures

Checkpoint and restart procedures make it possible in the event of an error or interruption, to begin processing from the last checkpoint rather than from the beginning of the run. These techniques should be included wherever possible, in applications which require 30 minutes or more of processing since heavy machine scheduling and deadlines make complete reruns uneconomical and/or impractical.

7.9 Revisions and Modifications to Programs

Past experience has indicated that many programs require subsequent modification due to changing conditions or because of implementation at other locations. Consequently, all programs should be written in a straightforward manner. Sophisticated techniques and complex routines, which do not really lend themselves to modification, are not necessarily equated with program efficiency or programming ability and should be avoided except when meaningful system processing time can be saved or when required because of core limitation. This concept is even more prevalent with the conversion to third generation systems. Because of the increased flexibility of the third generation assemblers and compilers over previous software, adherence to straightforward programming becomes a necessity if the department is to reach its goal of producing efficient, well-documented programs that are relatively easy to implement and to modify.

7.10 Card Procedures

a. Programs which require a date card can be classified as either; Type 1 (has data cards as input), or Type 2 (does not have data cards as input).

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

(1) Date cards for Type 1 programs will be submitted to Operations with the data cards. Keypunch and Data Control procedures will contain instructions for preparing the date card and making sure the date card is placed properly in the deck of data cards.

(2) Date cards for Type 2 programs will be prepared by the Computer Operations Unit. The SPECIAL INSTRUCTIONS section of the Computer Operations Run Sheet will contain instructions for preparing the date card and its placement in the Job Control deck.

7.11 Preparation of DOS Job Card

a. DOS Job cards are to be prepared as follows:

C/C 1-2 //
C/C 3 Blank
C/C 4-6 JOB
C/C 7 Blank
C/C 8-12 Program Name
C/C 13 Code for benefiting agency (refer to Codes Section)
C/C 14 Code for system concerned
C/C 15 Code for run category
C/C 16 Blank
C/C 17-72 Comments, comments should not contain any remarks concerning job set-up or run time.

7.12 Control Card Procedure - Program Control Cards

a. This procedure does not apply to programs obtained from other agencies, or to programs that do not require control cards.

b. Card columns 7Ø-8Ø are to contain the following:

C/C 7Ø-74 Program ID (same as on the // Job Card).
C/C 75-77 The letters CTL.
C/C 78-8Ø Sequence number (must be numbered consecutively starting with ØØ1).

c. Card columns 1-69 (except on date cards) should be fully utilized to keep the number of control cards to the minimum. Date cards should only contain information that changes from run to run of the program.

d. If a date card is used it must be sequence number ØØ1.

e. Control cards supplied by Data Control and control cards prepared by Computer Operations must have lower sequence numbers than programmer-supplied control cards.

f. The total number of control cards required for a program and the control card source must be entered in the SPECIAL INSTRUCTIONS section of the Computer Operations Run Sheet as well as the format of any control cards which are to be prepared by Computer Operations.

g. Computer operators are responsible for correct placement of control cards they put in the job control deck (control cards supplied by Data Control or prepared by computer operations).

h. If the control cards are not in sequence or if the required control cards are not read in, the program should display an appropriate message on the console and terminate. Computer operator action after program termination for these conditions is:

(1) When the message indicates an out-of-sequence condition, put the control cards in sequence and be sure they are all present before restarting the program.

(2) When the message indicates the required control cards were not read in, locate the missing control cards (if possible) and put them in sequence before restarting the program. If the missing control cards cannot be located notify the person responsible for the program.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

8.0 SPECIFIC COBOL PROGRAMMING STANDARDS

It is important to realize that COBOL is a programming language that is as sensitive as any other. It's English language format is deceiving in this regard. A thorough knowledge of the language is necessary to the development of efficient programs in COBOL as it is in any other language. It is important, therefore, that all Data Processing personnel contributing to program development, including the preparation of data divisions, must be trained in the COBOL language. In addition, the conventions set forth in this section of the manual should be adhered to for all systems written in COBOL.

8.1 General Rules

a. Only one clause should be entered on each line. In this way any clause can be amended without difficulty. Sentences should be short. This generally results in better code being generated and greatly improves diagnostic messages.

b. A subset of COBOL may be specified by the Programming Supervisor. Statements not in the approved subset should not be used without permission. Coding should be simple. Sophisticated coding is not desirable when using COBOL. Maximum clarity for those who subsequently may have to modify the program should be the aim. For instance, "IF" statements should not be tested within other "IF" statements. Arithmetic statements are generated better if kept simple. Data division entries should be entered with each clause on a separate line as specified above. In this way standard clauses like "USAGE COMPUTATIONAL" can be prepunched in bulk and inserted manually into the source program, eliminating key punch errors.

8.2 Object Program Efficiency

Adherence to the course of action that follow should result in a more efficient object program.

a. When possible, in using conditional and "MOVE" statements:

(1) Establish fields that are multiples of computer words (when using a word machine).

(2) Make sending and receiving fields of uniform size.

(3) Keep decimal points aligned in sending and receiving fields.

(4) Avoid moving group items with mixed class and/or usage. (A redefine at the proper level can be used to prevent this.)

b. Minimize the number of relationship tests where:

(1) Either value is subscripted.

(2) An alphabetic or alphanumeric operand is signed.

(3) Fields with different class and/or usage are compared.

(4) Comparing fields with unlike signs.

8.3 Coding Techniques

Recommendations and background information are provided to aid the COBOL programmer in optimizing his object code execution and/or core usage. These guidelines should be evaluated for their applicability and their relative efficiency should be weighed for each production program to be coded. Listed below are the topics covered here. Where appropriate, illustrations are given. A quick reference may be made by examining just the illustration explaining the topic of interest.

CONTENTS

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

CONTENTS (CONCLUDED)

a. Characteristics of Numeric Data

Usage	Type of Data	Allignment Required	Converted In Arithmetic
Display	External Decimal/ Floating Point	No	Yes
Computational	Binary	Yes	Yes/No
Computational-1	Short Precision Floating Point	Yes	No
Computational-2	Long Precision Floating Point	Yes	No
Computational-3	Internal Decimal	No	Yes/No

ILLUSTRATION FOR 'a'

The efficient definition of data used in arithmetic statements is an essential ingredient of an efficient COBOL program. The saving of one byte in a given data definition can cause a significant increase in the number of instructions generated to perform arithmetic operations on the given data item. Conversely, a meaningful addition of one byte in the data declaration may result in fewer generated instructions.

Although the compiler resolves all of the allowable mixed data usages encountered, it is best to minimize mixed usages of related data items in procedure statements. A set of guidelines of numeric data declaration is as follows:

INDEX NUMBER
06-8.0

(1) Avoid mixed data formats.

(2) Match decimal places in related fields (Decimal-point alignment).

(3) Match integer places in related fields (Unequal-length fields).

(4) Include an S (sign) in all numeric fields (Sign control).

When specifying floating-point data in a COBOL program, minimize its usage with other data formats in compare and arithmetic statements. Non-floating-point data items used in conjunction with a floating-point data item are converted to floating-point. Conversions to floating-point require object time subroutines, thereby increasing execution time. Floating-point data is useful when positive or negative values greater than 18 digits in size are utilized or possibly generated. The floating-point mode of arithmetic operation is required whenever a fractional exponentiation is specified. Where possible, avoid the use of fractional exponentiation.

b. Cost of Decimal-Point Non-Alignment

(1) Intermediate instructions to align data.

(2) Intermediate instructions for sign movement.

EXAMPLE 1:

```
77 A Picture S999V99          COMPUTATIONAL-3
77 B Picture S999V9           COMPUTATIONAL-3
```

EXAMPLE 2:

```
ADD B TO A
Not efficient
```

EXAMPLE 2:

```
ADD 1 to A.  NOT EFFICIENT
It is more efficient as
ADD 1.00 to A.
```

ILLUSTRATION FOR 'b'

Procedure Division operations are most efficient when the decimal positions of the data item involved are aligned. If they are not, the compiler generates instruction to align the decimal positions before any

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

operation involving the data items can be executed.

Referring to Example 1, let us assume values for A and B of 100.45 and 100.2, respectively. Internally in packed decimal format, they will appear as 10045 + 01002 +, respectively. If you wanted to add B to A with the result being A, a simple add of the items as they are would give you a result of 11047 +. This is incorrect since the value should be 100.65 or 10065 + internally. To achieve the proper results, B must be aligned to match A by padding B with a low order zero. Three intermediate instructions totaling 18 bytes would be required to perform the padding and sign movement before the add instruction could be executed. Example 2 indicates how intermediate alignment instructions may be eliminated by specifying literals with proper alignment. The literal 1 has an implied picture of S9 while 1.00 has an implied picture of S9V99. Internally, in packed decimal format, 1 and 1.00 would appear as 1 + 100 +, respectively. The additonal byte required for the 1.00 literal is insignificant to the savings of 17 bytes of instructions plus the execution time.

c. Cost of Unequal-Length Fields

EXAMPLE 1:

```
A Picture S999
B Picture S99999
Move A to B
Cost in 10 additonal bytes of instructions for
zeroing high-order positions
```

Efficient definition of A would be:

```
A Picture S99999
```

ILLUSTRATION FOR 'c'

When moving an external decimal item (DISPLAY) to another external decimal item of larger length, the compiler generates instructions to insert zeros in the higher-order positions of the receiving field.

The generation of extra instructions can be avoided if the sending field is described with a length equal to or greater than the receiving field. Here again an increase in a data field size is more than compensated for by savings in object code and time.

In a similar manner this guideline applies to movement of alphabetic or alphanumeric items, except that spaces are inserted in the low order positions except when the JUSTIFIED RIGHT clause has been specified for the larger receiving field.

d. Sign Control and:

(1) Cost of unsigned numeric fields.

(2) Input data not checked for valid sign.

(3) Printing of unsigned vs. signed numeric fields.

EXAMPLE 1:

```
Picture 999
Picture S999
Picture S999

Move B to A
Move B to C
```

ILLUSTRATION FOR 'd'

When specifying pictures for numeric data items, it is a good rule to include a sign (S) in the picture unless absolute values are required. An S must be specified in the PICTURE clause of a binary item, since a binary item is always treated as a signed item.

For internal or external decimal items, the sign specification, is optional. Sign control is therefore a subject of concern for decimal items. For decimal items used as input, it is the programmer's responsibility to insure that the configuration in the sign position is valid. An invalid sign configuration in a decimal item involved in an arithmetic operation will cause abnormal termination of the program.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

e. Move Verb for:

(1) Numeric Data Minimize:

- Non-alignment of decimal points
- Mixed usage
- Unequal length fields
- Unsigned fields

(2) Careful usage of group moves containing numeric data

ILLUSTRATION FOR 'e'

Moves involving numeric data are governed by the same guidelines as are described above.

In addition, when specifying a group move containing numeric data, care should be taken to see that the sign position of a subordinate numeric item is not destroyed.

f. Arithmetic Verbs to:

(1) Minimize:

- Non-alignment of decimal points
- Mixed usages
- Unequal length fields
- Unsigned fields

(2) Initialize arithmetic fields

(3) Avoid exponentiation to a fractional power where possible

(4) Not duplicate sub-expressions in arithmetic expression

EXAMPLE:

```
COMPUTE X = A* (B/D) *C - B/D
```

ILLUSTRATION FOR 'f'

Example 1 is a summary of earlier recommendations concerning numeric data which applies to the use of the arithmetic verbs.

A COBOL programmer should make no assumptions with regard to initialization of arithmetic fields that he will use in computations. It is the programmer's resonsibility to initialize the arithmetic fields to the desired value. Failure to do so may yield invalid results or possibly result in abnormal termination of the program.

Whenever possible, avoid exponentiation to a fractional power since this would require the use of floating point instructions and may produce less accuracy than desired. When specifying an arithmetic expression do not use duplicate subexpression within the arithmetic expression. In doing this, the divide operation is performed only once, saving object code and execution time.

g. IF Statement

(1) Same general rules for arithmetic verbs apply

(2) Minimize use of arithmetic expressions in relational statements

EXAMPLE 1:

```
IF A * B - C = X - Y * Z ...

        instead use

Compute K = A * B - C
Compute J = X - Y * Z
IF K = J ...
```

(3) When specifying an IF Statement, the same guidelines apply as discussed earlier on numeric data. That is:

- Avoid mixed data formats
- Match decimal places in related fields
- Match integer places in related fields
- Include an S (sign) in all numeric pictures

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

ILLUSTRATION FOR 'g'

Minimize the use of arithmetic expressions in relational statements by performing computations prior to the IF statement, as indicated in the example in this figure. The IF statement then becomes a simple relational statement. This is recommended practice whenever the arithmetic expression may result in an intermediate value being truncated, thereby producing an invalid comparison. Minimize the use of compound IF statements, if for no other reason than that they are difficult to debug. Aside from the debugging difficulty, compound IF statements generally result in more internal labels being generated by the compiler in order to perform the necessary branches than would be required from a series of simple IF statements.

h. Subscripting

(1) Constant vs. variable subscripts

(2) Binary vs. other data format subscripts

(3) Frequently referenced subscripted fields

EXAMPLE 1:

```
INEFFICIENT   ADD K TO J (A,B,C).
              IF J (A,B,C) = M TO TO ERR.
              MOVE J (A,B,C) TO F.

EFFICIENT     MOVE J (A,B,C) TO WORK-FIELD
              ADD K TO WORK-FIELD.
              IF WORK-FIELD = M GO TO ERR.
              MOVE WORK-FIELD TO F.
```

ILLUSTRATION FOR 'h'

Subscripting provides the facility for referring to data items in a table or list that have not been assigned individual data-names. The subscript may be represented by a constant or a data name (variable subscript). All constant subscripts are resolved by the compiler. Variable subscripts are resolved at

INDEX NUMBER
06-8.0

object execution time through generated subroutines. Therefore, it is recommended that variable subscripts used in procedural statements be minimized, as object code and execution time would be reduced. Generally, when subscripted fields are referenced more than twice within a program it would be efficient to move the referenced subscripted field to a Working Storage field and refer to it by its non-subscripted name in Working Storage. The above example shows the problem and a solution.

i. Using the "REDEFINES" Clause

If an initial "VALUE" is to be given to an item and that item is to be "REDEFINED", the "VALUE" clause should be used in the description of the term to be "REDEFINED" and should not be used in the description of the item containing the "redefines" clause.

j. Perform and Alter

COBOL has a tendency to lull the programmer into becoming lazy. There are too many ways to save a little pencil lead by using programming "tricks". For example, there is a tendency to "PERFORM" portions of a program which are comprised of only a few lines of coding. This of course produces larger less efficient programs and increases the chances of coming up with a very messy debugging job. Caution must also be exercised in the over-use of the "ALTER" statement and of nested "PERFORM" statements. These programmer abuses of the COBOL language can result in a monumental debugging problem.

k. Conditional Branching

GO TO Procedure-name-1...DEPENDING ON data-name
vs.
A series of IF data-name = literal GO TO...statements.
The GO TO...statement is more efficient when data-name:

(1) Is an integral numeric item

(2) May have values of 1 through n

(3) Values may be greater than 2

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

ILLUSTRATION FOR 'k'

When using a multiple-way (more than two) branch switch in COBOL program, careful consideration should be given to the data format definition of the switch. If it is feasible to define the switch within the requirements of data-name for the GO TO...DEPENDING ON data-name statement, the conditional branch executed by the GO TO... will generally be faster than that from executing a series of IF statements. The requirements of data-name for GO TO... are:

- Must be an integral numeric item, DISPLAY, COMPUTATIONAL OR COMPUTATIONAL-3, not exceeding 4 digits in length.
- May have values ranging from 1 to n, where n equals the number of procedure-names specified in the GO TO... statement.

l. ACCEPT vs. READ

(1) READ file-name (INTO data-name) AT END...
File description required
Work with data in buffer
Automatic end-of-data set transfer

(2) ACCEPT data-name
Working-storage description required
An internal move is executed
Program recognition of end-of-data set
Object subrouting required

ILLUSTRATION FOR 'l'

In general, the READ statement is more efficient than using the ACCEPT statement for similar applications. A file description and its associated record description is required when using READ, use of ACCEPT just requires a working storage description of data-name, which can be less than the logical record size. In this case the use of ACCEPT is preferred as it requires less data definition coding. With READ you have the option to work in the buffer or in working storage (INTO option). ACCEPT

forces the programmer to work in working storage all the time. The issue here is an optional vs. constant internal move. READ is preferred in this instance. The AT END clause is required with READ, giving the programmer an automatic end-of-data set transfer while ACCEPT does not. Recognition of end-of-data set is a programmer's responsibility within the COBOL program through counter controls. If not provided for, the program will abnormally terminate when ACCEPT attempts to access a record and there are no more records to be retrieved. Consequently, READ is preferred under these conditions.

Both READ and ACCEPT utilize data management routines but ACCEPT, in addition, requires a special object time routine to interface between data management and the COBOL object program. The additional core and execution time required for the subroutine gives another case for preference of READ over ACCEPT. ACCEPT should generally be used to access data from the console.

m. WRITE vs. DISPLAY

(1) Write record-name (FROM data-name-1) AFTER
(data-name-2)
ADVANCING LINES
(integer)

Record specification required
Initialization of record area
Dynamic control character facility

(2) DISPLAY (data-name)
(literal) ...UPON (CONSOLE)
(SYSPUNCH)
(mnemomic-name)

Limited control character facility
Required object time subroutine

ILLUSTRATION FOR 'm'

Which is more efficient, the WRITE...AFTER ADVANCING or the DISPLAY statement? The advantages and disadvantages of both must be weighed for a given application in order

DATA PROCESSING
MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS
AND AIDS

DATE ISSUED | DATE REVISED

to make a fair judgment. The DISPLAY statement should be used whenever it is desired to occasionally print on the console. Any extensive on-line printing of an intermittent nature should be directed to the printer via the DISPLAY statement. When data is destined for the printer or punch in a continuous and homogeneous fashion, the WRITE...AFTER ADVANCING statements should be used. These are overall recommendations. A programmer's likes and dislikes will play some part in what statement is finally used.

WRITE requires a file description with an associated record description while DISPLAY does not. DISPLAY requires just those data-name definitions that would be displayed plus any literals specified in the DISPLAY statement. WRITE requires that the programmer initialize the record area before each write command while DISPLAY does not. DISPLAY requires an object time subroutine which initializes the buffer area automatically at each display command. WRITE has the ability to dynamically specify control characters for spacing, page, skipping, and pocket selection while DISPLAY does not. This is a definite advantage in using WRITE. In order to space using DISPLAY, you must DISPLAY SPACE for each blank line designed. Both WRITE and DISPLAY utilize Data Management routines but DISPLAY, in addition, requires a special object time subroutine to interface between Data Management and the COBOL object program. Additional core and execution time is required for the subroutine. Weighing these advantages and disadvantages, the programmer should be able to make an efficient judgment with regard to his application.

n. COBOL Subprograms

(1) Utilized to:

Obtain efficient application programs quickly.
Take advantage of overlay features of the operating system.

(2) Linkage section considerations:

Group vs. elementary definitions.
Storage allocation (none).

INDEX NUMBER
06-8.0

(3) Multiple entry points and program flow

(4) Argument list considerations

Addresses
Correspondence

ILLUSTRATION FOR 'n'

The ability to write COBOL subprograms has increased the utility and flexibility of the language. Where previously a large application program (5000 or more cards) was usually written by one programmer, now with the subprogram facility the writing may be distributed to several programmers. Efficient application programs should be generated in a shorter time as a result. The subprogram facility also gives the programmer the opportunity to take advantage of the overlay features of the operating system. This provides for efficient use of core for a program that would not fit as a single main program. A COBOL subprogram must contain a linkage section which describes data being passed from another program. Whenever possible, define the data being passed as a group instead of as an elementary item for each item being passed. This reduces the argument list and provides a larger amount of data that can be passed. Remember, there is no storage allocation for the linkage section in the subprogram. It is just a dummy working storage area. A COBOL subprogram may have multiple entry points but it is the responsibility of the programmer to see that program flow within the subprogram does not flow to an entry point. When a subprogram is called, it has passed an argument list. This list contains the addresses of the data-names specified in the argument list. No values or literals may be passed directly. The data-names used in the argument list of the CALL statement need not be the same as those used in the ENTRY statement. However, there is one for one correspondence assumed and therefore the argument lists should be specified accordingly.

DATA PROCESSING
MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS
AND AIDS

DATE ISSUED | DATE REVISED

9.0 PROGRAM PREPARATION

9.1 Source Program Preparation

a. After the coding has been reviewed and desk checked in its entirety, the program is ready for input preparation.

(1) A source program deck of cards will be keypunched and verified from the coding sheets.

(2) The source deck of punch cards should be created on an interpreting keypunch, if available. If not, then the deck shall be interpreted as a separate operation.

(3) An "80/80" listing or computer printout of the source deck should be made to further aid in desk checking the program coding.

9.2 Desk Checking

a. After a program or a module has been coded and keypunched and a source program listing obtained, but before testing begins, the activity of desk checking should take place. The purpose of this activity is to minimize both the computer time and elapsed time needed to debug the program by checking the module's logic before testing. This logic test can be performed in many ways; however, experience indicates the following procedures should be followed and are most practical.

(1) Using the program module specifications flow charts, and other documentation as a guide, sample data should be prepared. Usually these data can be constructed on a regular lined paper or flow chart form. This data need not be complex in order to achieve its purpose.

(2) Manually, the sample data that was created is then sequentially "run through" all steps of the module listing from the starting point. This procedure of programmer simulation of the computer processing will test the program logic and highlight many coding errors before the first compilation.

(3) When possible, another programmer should be asked to assist in the desk checking process to provide a "sounding board" for the programmer and to help validate logic.

b. While the prime purpose of desk checking is to validate logic, a secondary benefit is manual screening of the listing for keypunch errors or coding translation errors, which can be corrected before the first compilation. Desk checking can continue to be of value even after the initial compilation. Desk checking after the initial compilation should make use of all aids provided by the compiler diagnostics for correcting the program. Some of these are:

(1) Identified coding errors (undefined labels,wrong punctuation, illegal language, etc.).

(2) Warning given of truncation, irregular justification or zone stripping data movement.

(3) Optimization messages provided when compiler required to produce coding for inefficient operation.

c. Desk checking is concluded when the programmer receives the first error-free compilation.

9.3 Program Assembling or Compiling

a. In this stage of developing the program, the programmer must use the computer and the associated operating system to complete development. Computer systems are costly, so careful planning and desk checking before each submission for compiling is vital.

b. Before submitting a program for compilation, take these steps:

(1) Determine that the source deck has been set up with appropriate operating system control cards.

(2) Determine that any required compiler output options have been properly selected.

(3) Prepare a Computer Operations Run Sheet.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

(4) Prepare a Console Message Sheet.

(5) Prepare a Job Request Form.

(6) Submit the Compile/Test Request Form, Console Message Sheet, and Console Run Sheet together with the source program deck of cards to the Operation's personnel responsible for preparing and scheduling Compile/Test computer runs.

c. When the output is obtained from a compile, it should be desk checked as the Standard prescribed before any further compilations are made. Repeat the above desk checking procedures on each compiler run until an errorless compiled program listing is developed. Be sure to make any alterations or changes on the detail flow chart as well as on the listing. When a clean program listing has been developed, the program is ready for testing.

INDEX NUMBER
06-9.0

10.0 PROGRAM TESTING

Program testing is performed to assure that each program module and the entire program correctly executes the functions defined by the specifications.

10.1 Test Data

a. General Principles

(1) Test data cases should be comprehensive but simple. Each of the many test cases usually required to check out a program should be designed to test one particular feature or path. Beware of setting up test situations where many different operations are tested simultaniously making analysis impossible when something goes wrong. Each machine run must be planned to check specific processes, and the results analyzed to verify that these conditions are being handled properly.

(2) Test data falls into two categories:

- Data necessary to check out each block or module of the program.
- Data necessary to validate the manner in which the system requirements are met. Such data is used to test the module to module logic.

b. Responsibility

(1) Each programmer will be responsible for generating comprehensive test data consisting of one or more of the following types:

- File Maintenance (transactions for file loading together with adds, changes and deletes for three processing cycles).
- Update Runs (one for each transaction type, grouped for three processing cycles).
- Edit Runs (all transactions from file maintenance and update, with at least one additional transaction processed by the system).

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

- Processing Runs (for all master and interface files required by the run, created by their respective source program using the edited transactions mentioned above).

(2) The programmer should create test data needed to test any unusual condition in the program which is not covered by normal testing. All test data must fairly represent actual working conditions; transaction with all zeros are virtually useless for testing.

c. Live vs. Test Data

(1) The relative merits of "live" versus "test" data for program checkout should be considered. Live data is actual business transactions used in recording day-to-day activities. Test data is simulated business transactions prepared especially to check out a computer program. Live data can be used as test data, but live data must first be analyzed and altered by the programmer to meet test case requirements. Altering the live data usually requires more work than developing representative test data.

(2) Live data has these characteristics:

- May not be available when a completely new system is being prepared.
- May contain errors or conditions never specified and not desirable in early tests.
- Probably does not contain data to test all programmed conditions.
- Large volumes may be necessary to cover the variety of transactions necessary for a comprehensive test.
- Has output which is difficult to analyze accurately. (For instance, what should the correct total of 500 entries be?).
- Complete content of files is not fully known by the programmer.

INDEX NUMBER
06-10.0

- Because of high volume, run time is high, thus delaying the programmer's receipt of the test result.

- Final results may be impractical or impossible to be prepared in advance for validating the test run.

(3) Test data overcomes all disadvantages of live data except that the initial creation of effective test data may be highly time-consuming and tedious. With test data, complexity of tests may be progressively increased, and results can always be precomputed with pencil and paper. Therefore, live data should not be used during the initial testing of a program. When the user offers to furnish test data, the programmer is responsible for verifying that the data meets test requirements.

10.2 Test Case Design

a. Test cases require careful preparation or programs will fail repeatedly on production runs becuase errors were not detected and cleared in the check out phase. Weak test data causes costly production delays.

b. The surest way to waste programmer and computer time is to rush into testing and debugging, hoping to straighten out a poorly written program. Before testing, the programmer must know that:

(1) The detail logic, illustrated in the block diagram, solves the problem for all conditions.

(2) The coding correctly interprets and matches the logic in the block diagram.

c. The following procedure will produce satisfactory results regarding a test:

(1) For the initial test, submit only enough data to verify that the program has gone to end-of-job.

(2) Test the most likely case on first test. Test less likely paths and error condition(s) on subsequent tests.

(3) For modular programs, test each module separately if possible. Then test mainline module with whatever other module(s) (if any) are needed to have the program go to the End of Job. Continue adding modules to program test until entire program has been tested.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

(4) Determine in advance what test outputs will look like, based on familiarity with the program, test data, and the master files.

(5) Take a tape dump of all output files.

(6) Compare actual outputs to predetermined outputs, and mark all differences.

(7) To find the cause of an error, analyze the console messages print output and output tape dump(s) together with the program listing and test data used. It is useless to try and fix an error when the cause is unknown.

(8) If the error is in the logic, correct the logic chart first, and then correct the program.

(9) If the program does not reach end-of-job, get a core dump at the point that the program fails and use the core dump together with the items mentioned in point 7 to find the cause of failure.

d. The following factors should be considered in the design of test data:

(1) Test data (vs. live data) should normally be used for program testing.

(2) Operating system utilities should be used for creating test files where appropriate.

(3) Test cases should be arranged in sets of increasing complexity, with the first test set as simple as possible, and the final set a comprehensive program test case.

(4) Every piece of coding should be tested.

(5) End-of-file conditions must be thoroughly checked.

(6) Each piece of test data must be designed to check a particular feature of the program.

(7) Good test data has low volume and simple content, which speeds up the checking process.

INDEX NUMBER
06-10.0

(8) Test data should be carefully desk-checked against the coding prior to computer testing, so expected results can be predetermined.

(9) Computer files created from test data must be dumped and checked after use. Particular attention should be given to internal label specifications.

10.3 Procedure for Submitting Programs for Testing

a. When the program and test data are ready for testing, a test package should be developed. Since for quality control and internal control purposes, the Programmer should not have direct access to the computer, he must make test requests as clear and simple as possible. The following steps should be taken in preparing the test package:

(1) Prepare a Computer Operations Run Sheet.

(2) Prepare a Compile/Test Request.

(3) Prepare a Console Message Sheet.

(4) Prepare a Program Run Flowchart

(5) Mark test data and program clearly.

(6) If applicable, create External File Lables, to put on test files. The first four digits of the File Identification Number should be the letters "TEST". The programmer's name or employee number should also be put on the external file label. In this way, the File Library Listing will indicate which files are test files, and who is responsible for them. This will also allow periodic purging of unused test files from the file library thus freeing up the volumes for productive use.

(7) Indicate outputs to be saved and outputs to be scratched.

(8) Remember that every checkout is always a first-time run for the computer operator.

b. The program test package will then consist of:

(1) A Compile/Test Request.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

(2) Computer Operations Run Sheet.

(3) Console Message Sheet.

(4) Program Run Chart.

(5) Job Control Cards.

(6) Program deck.

(7) Test data.

(8) External File Labels.

c. When the package is completed, it should be submitted to the Operations Scheduler for setting up compilations and tests.

10.4 Testing Techniques

a. The key to program or module testing is to carefully plan the test session in advance. The following list of techniques and standards should aid in the testing process:

(1) Only one post list and linkage edit map should exist for any program at any given time.

(2) Never display debugging information upon the console.

(3) For large complex programs a useful approach is to include debugging print-outs when the program is first coded. These can be easily removed later or bypassed with GO TO or BRANCH statements when testing is complete.

(4) Any special-coded programs written to facilitate testing should include operations documentation.

(5) A modified version of the Master File Create and/or Maintenance program can be used to create a "Test" Master File.

(6) Where possible software test aids should be used. The use of file generators and selective trace routines can reduce the programmer's time involvement.

10.5 Release for Systems Test

Before releasing a program from program test to systems test, the programmer must review the test output with the responsible systems analyst and get approval for the release.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION
PROGRAMMING STANDARDS AND AIDS

DATE ISSUED | DATE REVISED

11.0 SYSTEM/PROGRAM MAINTENANCE

11.1 Purpose

Operating systems and programs like the agencies and governments they serve are dynamic. Thus, procedures are required that provide needed changes which are properly designed and implemented, and associated documentation upgraded.

11.2 System Maintenance

System maintenance refers to a change to an existing system and also represents a change in the System Documentation. This means that programs and/or files are being added or deleted to/from the system or that the file and/or record layouts are being altered. If implementing new programs, all the necessary steps and program documentation are required as the standards dictate. If modifying existing programs through the addition/deletion or modification of records and/or fields, then Program Maintenance below must be exercised.

11.3 Program Maintenance

a. Program maintenance is implementation of a change to an existing program and changes to the existing program documentation. It might also affect changes to the Systems and/or Operational documentation. The program specification package, in essence, contains the same items as spelled out in the Specifications Standards. However, the Program Narrative and any supportive tables, lists, etc., need only refer to the changes to be made and not to the entire program. It is suggested that a Program Revision History sheet (sample attached) to be filled out and sent along with the revised program specification package to programming.

b. The new Program Narrative, together with supporting tables, etc., is also added to the Final Program Docementation. When the program has been accepted for implementation, the new Program Revision History sheet can replace the old one in the Program Documentation Package.

PROGRAM REVISION HISTORY SHEET
(Example Form)

PROGRAM NO. _____ STATUS: UNDER DEVELOPMENT (start date) FINAL (Acceptance date)

PROGRAM TITLE _____ PAGE _____ OF _____

AGENCY _____ DEPARTMENT OR UNIT _____

DATE	ANALYST	PROGRAMMER	REASON
(Date Operations Released For Production)	(Person who made Revisions in Specifications and Related Specification Documentation)	(Person who made the Program Changes)	(Brief Description of Why Program needed Modifying)

TABLE OF CONTENTS

07 OPERATIONS STANDARDS

1.0 General Information

Standards within the Operation section are among the most important ones to the entire Data Processing Department. The world of Electronic Data Processing by its very nature, relies on standardization of data handling and communications. Adherence to these standards enables the department to realize use of the equipment to its fullest potential. It is very logical, very necessary to not only standardize systems but take one step further and standardize the handling of Jobs and procedures within the Operations function.

The standards for operations fall into three logical major catagories:

a. Standards on Jobs submitted to operations for processing.

b. Standard procedures on methods used within the operations function.

c. Standards of communications to help improve operations in all functions.

2.0 Standards on Submitting Jobs

2.1 Submitting Background or F2 Programs for Production

a. When testing of program is completed and ready for production, and have been catalogued in the Core Image Library or an object deck punched, the following items must be prepared. For each program submit the following to the Operations Scheduler at least five working days prior to the date of the first production run.

(1) Job control cards or job control cards and object decks.

(2) Computer Operations Run Sheet.

(3) Carriage tape if other than standard.

(4) Information sheet for entering program in Program List and reports in Report Distribution Log.

(5) Estimated run time.

b. The following procedure will apply when the Operations Scheduler cannot be previously handed the package before the first production run.

(1) Prepare and submit the following items to the Operations Scheduler.

- A written request to run the program(s) stating the estimated run time, the date when the program(s) should run and any assigned priority.
- Job control cards, source decks and/or object decks set up to run in the desired sequence.
- Computer Operations Run Sheet(s).
- Carriage tape(s) if other than standard.
- Input data if it is not in the computer room.
- Date cards and/or control cards required, unless they are to be prepared by computer operations.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

OPERATIONS STANDARDS

DATE ISSUED | DATE REVISED

(2) Prepare and submit the following for Data Control if any report(s) will be produced.

- A written notice to expect the report(s), stating the titles of report(s) to expect, the distribution of each and the date they should arrive in Data Control.

2.2 F1 Links - Control and Responsibility

a. Section 1 and Section 2 Analyst

(1) Sign approval of each link request and coordinate the effort.

(2) Establish central location for filing all operational link decks, listings, maps, catalog listings, and link requests.

(3) Set up and maintain a binder of all printer listing of catalog jobs and link maps. Keep in a central location for easy accessibility.

b. Programmer/Analyst

(1) A programmer who has F1 module to catalog will note on a link request his name and that the module has been given to Operations to catalog. When the catalog listing is returned, the appropriate blank will be signed to indicate that the module is in the relocatable library. If the deck and listing are not returned during working hours, the 'verified as catalogued' blank will be left blank, indicating the programmer does not know that the module is in the library.

(2) If the module is needed in the on-line system the next day, check to be sure of a good link.

(3) Return all catalog listings and decks to the programmer if he is present. If he is not present, place the deck and listing on the link table.

(4) The analyst or programmer doing the link will check the link request form to see if all modules that were supposed to be catalogued were catalogued. The programmer

INDEX NUMBER
07-2.0

or analyst doing the link must check the link table to see if the catalog listing has been returned from Operations on any modules not already marked as catalogued on the link request. If a module has not been catalogued, it will be his responsibility to see that it is catalogued before the link is started. He will sign the verified column when the catalog is verified.

(5) If any modules are marked as necessary to be included in a good link, then a good link must be done before the end of the test period.

(6) Any modules catalogued to be included in a good link must be thoroughly tested before the system is brought up in an operational status. Either the programmer must come in early and conduct the test or sufficient instructions and test data must be left for the programmer doing the link to conduct a reasonably thorough test.

(7) When there are no supervisory personnel available in person or by phone, the programmer, if conditions indicate the necessity, may fill out the link request, and indicate on it that no one was available to sign it, and proceed with the link.

c. Operations

(1) One link request will suffice for a test period.

(2) Computer operators are not to allow a link without a properly authorized request. If both section analysts are unavailable, a unit supervisor or the manager will sign the request. If one analyst is unavailable, the analyst who is present will check with programmers in both sections before signing the link. He will note on the request that the other analyst was not available.

d. Unit Supervisor

The Unit Supervisor will review all link requests on the morning following the link.

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
OPERATIONS STANDARDS
DATE ISSUED
DATE REVISED

2.3 Control Card Procedure -Program Control Cards

a. This procedure does not apply to programs obtained from other agencies, or to programs that do not require control cards.

b. Card Columns 7Ø-8Ø are to contain the following:

C/C 7Ø-74 Program ID (same as on the //Job card).
C/C 75-77 The letters CTL.
C/C 78-8Ø Sequence number (must be numbered consecutively starting with ØØ1).

c. Card columns 1-69 (except on date cards) should be fully utilized to keep the number of control cards to the minimum. Date cards should only contain information that changes from run to run of the program.

d. If a date card is used it must be sequence number ØØ1.

e. Control Cards supplied by Data Control and Control Cards prepared by Computer Operations must have lower sequence numbers than programmer-supplied Control Cards.

f. The total number of Control Cards required for a program and the Control Card source must be entered in the SPECIAL INSTRUCTIONS section of the Computer Operations Run Sheet as well as the format of any Control Cards which are to be prepared by Computer Operations.

g. Computer Operators are responsible for correct placement of Control Cards they put in the Job Control Deck (Control Cards supplied by Data Control or prepared by Computer Operations).

h. If the Control Cards are not in sequence or if the required Control Cards are not read in, the program should display an appropriate message on the console and terminate. Computer Operator action after program termination for these conditions is:

(1) If the message indicates out-of-sequence condition, put the Control Cards in sequence and be sure they are all present before restarting the program.

INDEX NUMBER
07-2.0

(2) If the message indicates the required Control Cards were not read in, locate the missing Control Cards (if possible) and put them in sequence before re-starting the program. If the missing Control Cards cannot be located notify the person responsible for the program.

2.4 Date Card Procedures

a. Programs which require a Date card can be classified as either; Type 1 (has Data cards as input), or Type 2 (does not have Data cards as input).

(1) Date cards for Type 1 programs will be submitted to Operations with the Data cards. Keypunch and Data Control procedures will contain instructions for preparing the Date card and making sure the Date card is placed properly in the deck of Data cards.

(2) Date cards for Type 2 programs will be prepared by the Computer Operations Unit. The SPECIAL INSTRUCTIONS section of the Computer Operations Run Sheet will contain instructions for preparing the Date card and its placement in the Job Control deck.

2.5 Preparation of DOS Job card

DOS Job cards are to be prepared as follows:

C/C 1-2	//
C/C 3	blank
C/C 4-6	JOB
C/C 7	blank
C/C 8-12	program name
C/C 13	Code for benefiting agency (Refer to Section 6 paragraph 3.7.1 for agency codes)
C/C 14	Code for system concerned (Refer to Section 6 paragraph 3.7.2 for system code)
C/C 15	Code for run category (Refer to Section 6 paragraph 3.7.3 for run category codes)
C/C 16	blank
C/C 17-72	Comments. Comments should not contain any remarks concerning job set-up or run time

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

OPERATIONS STANDARDS

DATE ISSUED | DATE REVISED

2.6 14Ø3, Printer paper and forms

Unless otherwise specified on the program run sheet, all printed output will be on single-ply standard printer continuous form paper.

2.7 Carriage control tapes

Unless otherwise specified on the program run sheet, the Division standard carriage control tape will be used on the 14Ø3 printer to control form spacing.

2.8 Date and time entries

a. The "Date" format for internal computer storage and operator entry is as follows:

MM/DD/YY M = month

D = day

Y = year

b. The "Time" format for internal computer storage and operator entry is as follows:

HH/MM/SS H = hours

M = minutes

S = seconds

3.0 Standard Operating Procedures

3.1 Five Functions of Operations

a. The Control Function

(1) The Control Function is normally assigned to a Data Control Section. Its objective is to maintain the accuracy of the work done by the Computer Center. It maintains established schedules and sees to it that the data arrives from the User as agreed - the Center establishs schedules, obtaining the User's acceptance of the schedule, and making sure the User adheres to the schedule.

(2) Data Control maintains and logs the status of work-in progress in the Center. The work process of a Data Center is not unlike a production operation. Synonymous with a production control system, the Data Control Section must know the status of all work in the Center and be prepared to handle unplanned delays in processing the work. Included in this effort is the detail work necessary to keep the User's data in balance. It is the Data Control Section's responsibility to obtain the requi. d controls from the User, verify these controls with the results of processing and resolve any differences in these results with the User when out-of-balance.

(3) Data Control also prepares the results of processing and distributes the results to the User. This includes printing, binding, and recording the fact the work was released to the User - all within the prescribed time limit.

b. The Scheduling Function

(1) The Scheduling Function is one of the major functions in the Computer Center's effort. Scheduling is a never-ending project. This function's cycle begins when the Systems Department completes its design. At this time, Systems knows the program job stream or job streams required to produce the desired results. The Scheduling Section must take the work flow designed by Systems, obtain estimated volumes of data to be processed and formulize an estimated time frame in which to produce the desired results.

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
OPERATIONS STANDARDS
DATE ISSUED
DATE REVISED

(2) The proposed work schedule must then be integrated with overall Computer Center work schedule, to insure enough time available to perform the work. This could result in a re-evaluation of the schedule. Once the schedule is established, affected sections within the Center would be notified by this section of the impending additional work load.

(3) Once the new system has been implemented, the Scheduling Section must maintain statistics about the system and how accurate the work estimates were. If the variances become significant the section should make adjustments accordingly. These adjustments could affect the User. Therefore, the communication between the Center and the User must include this possibility.

(4) The end result of this section's analysis is a daily Computer Center Schedule. This schedule will indicate workload by Computer Center section, both repeat work and "one-time" job work, and expected completion times for each section. In addition, projected schedules for the rest of the week, month, quarter, six-month period, annual period, and special periods, as adjusted, must also be developed.

c. The Data Acquisition Function

(1) The Data Acquisition Function encompasses those activities necessary to convert data in the form familiar to the User into the form used in data processing. This effort includes the key punch, key tape, and off-line data collection activities. This function is most important, in that data must flow through this activity prior to being massaged in the Computer Section.

d. The Processing Function

(1) The Computer Section is the reason for the entire EDP organization's existence. Its capabilities and performance can determine the life or death of the entire EDP organization. The Computer Section's objectives are threefold:

- To manipulate the data received from the User,
- To maintain the User's data files, and

- To produce the results desired by the User.

The objectives of the Processing Function are met by: The design of specific methods of processing that the computer operators follow; by effectively "marrying" the hardware and software into one cohesive, working unit; and even by the design of the computer room and the work space it contains. Note that the actual processing of the User's data on the computer is only one step in the entire process. Yet it is the most important step.

e. The Library Function

(1) The Library Section is a support part to the Computer Section. Its objectives are:

- To keep or maintain User data for future processing by the Computer Section.
- To maintain the history files for the User for possible need or reference.
- To provide a safe environment for the storage of all this data.

The development of procedures for handling and storing the various forms of storage; i.e., paper cards and tape, magnetic tape and disc is a function of this section.

f. The Standards Section

(1) The Standards Section is second in importance only to the Computer Section. For it is this section that establishes, and maintains, the means by which all functional areas converse with each other. It will also establish and maintain the means by which each functional area operates within its own boundaries. The section is delegated the responsibility for developing and maintaining procedures describing the flow of work through the EDP Center.

(2) The major work flow procedures will describe each section's function. They will also outline what each Section can expect to receive from any or all

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

OPERATIONS STANDARDS

DATE ISSUED | DATE REVISED

other Sections in the work flow pattern. The Standards Section will also develop and maintain those procedures necessary to effect a smooth flow of work through the various functional areas of the Center.

3.2 Initializing Disks and Tapes

a. Each disk unit volume being initialized will contain the following information in the volume label:

XX = Applicable File Number

	"VOL10000XX	blank	KCOMPD	blank "
cd col	1 10	11 41	42 47	48 80

b. Each tape unit volume being initialized will contain the following information in the volume label:

	"VOL10000XX	blank	KCMOPD	blank "
cd col	1 10	11 41	42 47	48 80

3.3 Tape File Protection

a. ALL mounted INPUT tape volumes will have the write ring removed for protection.

b. ALL mounted OUTPUT tape volumes will have the write ring inserted to enable writing operations.

3.4 Returning Computer Output Products

a. Mark all outputs according to any special instruction contained on the run sheet.

b. Return source decks, object decks, listings, and job control cards to sender or as specified on the run sheet.

3.5 Labels

a. Label cards for file processing in job control stream will specify retention period rather than specific dates. See formats for TLBL and DLBL C24-5036.

b. External labels for tapes and disks will use date format MM/DD/YY for creation and expiration dates. Requesters will specify retention periods on the run sheets in days, weeks, or months rather than actual days until expiration date.

3.6 Time Verfication

a. At the beginning of each shift, the computer operator will verify the wall clock time by phoning 'time of day'. This will then be compared to internal computer time and appropriate corrections made as necessary.

3.7 VTOC STANDARDS

a. All 2311 Disk Packs (1316) will have their VTOC labels formated on cylinder 199.

b. All 2314 Disk Packs (2316) will have their VTOC labels formated on the standard location cylinder 199 (excludes ALERT Files).

3.8 Library Log Records

a. Each time a new tape or disk file is created, entries will be made on the stick-on label as contained on the operations run sheet.

b. In addition, an entry will be made in the Library log book usually kept on the tape cart.

3.9 SYSRES BACKUP

a. Each Sunday, the day shift will back up SYSRES on tape, using the copy and restore disk utility program.

b. Each Sunday, the day shift will take a DSERV, and file in the book on the console.

3.10 File History on Disk Packs

a. Each time a disk pack is installed on the spindle or an address plug is relocated an entry will be made on the history card mounted on each module.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

OPERATIONS STANDARDS

DATE ISSUED | DATE REVISED

3.11 Running a Job and the Run Sheet

a. General Information

(1) Since all Volume Serial Numbers are established as 111111 for Disks and 000000 for tapes, Vol, DLAB, XTENT, TPLAB or TLBL and DLBL cards are standardized insofar as volume serial number is concerned. This allows any reel or disk to be used as an output.

(2) Standard Labels will be specified in all programs written for this installation including utility applications.

(3) A Reel or Disk Library number will be assigned as an external control identification for each physical reel or disk and they will be filed in the computer library according to library number. The external Library number is the means of identifying and recording on run sheets the appropriate inputs to a specific run. All output reel and disk numbers are entered on the run sheet by the operator as they are drawn from the repository of new or expired library volumes. At the time of use, appropriate labels are placed on the volume. Volumes will be filed and retained until expiration date at which time they are returned to scratch status, available for immediate use.

b. Computer Operations Run Sheet Use - See example showing the set up of a typical update application consisting of 3 programs run in series:

(1) Run 1 -- card to tape. Output is on X"181". Reel # 1Ø3Ø was selected at random from the library scratch repository. Library Reel number was entered by the operator. Retention period on this reel is Ø days and is indicated in Label Retention Column (LBL RET). The Library reel number is also entered on the External tape label as well as today's date, the expiration date or period, Job ID, physical unit used and File ID.

(2) Run 2 -- second program run in the series and is a sort application. Note the input reel #1Ø3Ø which was created in the previous run is already mounted on 181 therefore no STOP card is needed between RUN 1 and

INDEX NUMBER
07-3.0

RUN 2, provided units 191 and 182 were set up prior to executing RUN 1. A scratch pack work area is used on 191 and a new tape is mounted on 182 selected from the scratch library. Again the operator enters the Library reel number used and affixes the external label. A STOP card is needed between RUN 2 and 3 in order to STOP BG to allow the operator to perform necessary set-up changes.

(3) RUN 3 -- update run. Reel 1Ø35 which was created in the previous run is already mounted on 182. Reel 1Ø3Ø must be dismounted from unit 181 and reel 1Ø2Ø as specified by the run sheet mounted on this unit. Reel 1040 output is selected from the scratch repository by the operator. In addition to the tape output created by this run a deck of cards and a 2-part listing will be created. The operator will insure that both units are prepared for processing (i.e., blank cards in the punch hopper and 2-part paper on the printer) before the START BG command is issued. Since this is the last run in the series a STOP card is the final card in the Job Stream required to STOP BG. The output card deck and listing will be marked and dated as indicated on the run sheet.

Note: In the above example, FG1 has not been disturbed except for the brief time necessary to get attention on the keyboard to START BG.

(4) Upon completion of the series of runs the operator signs the run sheet, tears off the original copy of the console typewriter sheet, attachs it to the run sheet and batches it with all output listings and card decks to be distributed as indicated. All tapes and disks will remain in the computer center.

(5) The originator of the run sheet is responsible for filing and controling run sheets when completed. In many instances such as RUN 3 in the example, the reel number indicated for output on 183 becomes the reel to be used in the next run or as next month's input on 181. Other run sheets may contain reel numbers which may be used in various runs at another time.

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
OPERATIONS STANDARDS
DATE ISSUED
DATE REVISED

c. External Labels - See example of the file label prepared for the output indicated on 181, RUN 1.

INDEX NUMBER
07-3.0

K.C.P.D. COMPUTER OPERATIONS RUN SHEET

JOB	TITLE	PROGRAMMER	ALTERNATE PROG.
CFØ18	CIVIL INDEX LISTING	HARMON	BAILEY

TYPE OF RUN	FREQUENCY	SEQUENCE	ORIGINAL DATE	REVISED DATE
⊗PROD. ○TEST	MONTHLY	2 OF 2	2-8-72	

CARRIAGE TAPE ⊗STANDARD ○OTHER (see spec. instr.) — 2 PART PAPER

I/O	360 UNIT ADDR.	REEL OR DISK NUM.	FILE-ID – COMMENTS	LABEL RETEN.	DISPOSITION	370 UNIT ADDR.
I	28Ø	FROM CFØ11	LATEST 'CFØ11T1 CIVIL INDEX'		LIB	48Ø
O	ØØE		CFØ18L1		DATA CONTROL	Ø1E
W	133		SORT		SCRATCH	155

SPECIAL INSTRUCTIONS:

ONE DATE CARD REQUIRED. FORMAT: MONTH AND YEAR OF REPORT STARTING IN COLUMN 1.

(EXAMPLE: JANUARY, 1972). PUT 'CFØ18CTLØØ1' IN COLUMNS 7Ø-8Ø.

Form 190 (Rev. 1-72)

DATA PROCESSING

MANUAL OF STANDARDS

SECTION: OPERATIONS STANDARDS

DATE ISSUED: January 16, 1973

DATE REVISED:

FILE LABEL EXAMPLE

KCMOPD

LOC. # 1047 JOB-ID DITTO SYS. # X'481'

FILE-ID OS PTFTAPE REL/21.6

CREATION DATE 11/2/72 EXPIRATION DATE ---

SAVE UNTIL DATE INDEF

INDEX NUMBER
07-3.0

3.12 Time of Day for Assemblies, Compiles and Test

a. There have been two periods of time (Ø93Ø-11ØØ and 143Ø-16ØØ) set aside each day (Monday through Friday) for running assemblies, compiles, tests and catalogs that have been submitted prior to the starting time designated for the two periods. If the period is not filled by submitted jobs, operations may schedule those jobs submitted after the designated starting time or resume other work if the above two conditions are not met due to lack of submitted assemblies, compiles, etc.

b. Only jobs (Assemblies, Compiles, Tests and Catalogs) that take fifteen minutes or less will be scheduled and run in usual manner.

c. Production jobs and priority jobs are not be run during these time periods. Jobs which take fifteen minutes or less but did not get run during a time period will be entered on the schedule and run in the usual manner until the next available time period.

d. Jobs that run over the estimated test period duration will be given another three minutes to allow completion. If at this time the job has not completed, the job will be canceled. The listing and control deck are to be returned to the person's respective supervisor for disposition.

e. The Operations Scheduler will be responsible for the alignment of jobs within the time periods involved.

3.13 Assembling and Cataloging from Tape Operations

a. To increase through-put on relocatable assemblies, a procedure has been developed to utilize a tape drive for punching object decks and the ability to catalog these decks from tape. Existing procedures will be used to submit the job to operations.

b. All relocatable source decks to be assembled using a tape will be accompanied with a standard run sheet. The operator will determine which of five tapes are available for the assembly. Tapes to be used: 1493, 1494

c. To determine available tapes which change on a daily basis, an inventory log has been established for these tapes. It will be the responsibility of the operator running the relocatable job to:

(1) Reference log for available tapes.

(2) After a relocatable tape has been created, log the current information in the inventory log.

(3) Post the tape number on the run sheet of the tape used during the assembly.

d. Releasing of relocatable object deck tape

(1) The standard tape release form used for scratching tape will be submitted to Operations who will immediately make the entry in the log showing release of the tape for further use.

(2) If object tape is correct, the programmer will submit run sheet and control cards to catalog the tape into the relo-library. At completion of catalog the tape will be released for further use. All tapes not used after one day will also be automatically scratched.

e. Tape Labels

(1) Since the labels would have to be changed at each assembly, it has been determined that each label will be permanent; therefore, the inventory log will reflect current status of the five tapes.

Note: Do not alter or scratch label when creating or releasing a relocatable tape.

3.14 Multi-Volume Tape Files and Tape File Labeling

a. When multi-volume tape files have been created due to file growth and expansion the external label reflects the proper volume by the use of "VOL. 1-9" at the end of the file - I.D. the proper procedure.

b. In order to comply with this procedure all operations personnel are instructed to comply in all instances with this standard by identifying these files with "VOL 1-x".

3.15 Maintaining Iventory Control of the Magnetic File Library

a. Due to the large volume of tapes and disks that are used daily in conjunction with computer operations; an efficient means of supply and return is necessary to facilitate proper inventory control of the tape and disk library.

b. Operations Scheduler

(1) The Operations Scheduler will supply on a daily basis, all input tape and disk volumes used in conjunction with daily computer operations.

(2) Control "scratch" tape and disk volumes required for output files.

c. Duty Operator

(1) Each shift will be responsible for the prompt return of all tape and disk volumes used for either input or output volumes on the previous operations shift.

(2) Verify all output volumes created on the previous computer operations shift (includes tapes and disks).

(3) Verify all daily T/P logs for correct logging regardless of creation date.

- Use the Library Log form as visible proof of verification to output volumes. This will be facilitated by placing a check mark in the right hand column of the Library Log next to the verified entry. (Be sure not to place the verifying check mark in the left hand column because this indicates the entry has been keypunched).

d. Operations Scheduler

(1) Reconcile and dispose of all discrepancies discovered in the tape and disk verifying procedure above.

3.16 Tape Cleaning and Tape Testing Responsibilities

a. Due to the importance of the ALERT On-Line criminal files, this tape cleaning procedure is used to help monitor and preserve the integrity of the On-Line files. Tape Management Controls on the tape cleaning device are to be set on, "CLEAN FULL-CYCLE ONLY". Only with direct permission, from his supervisor, may an operator use the "Test-Cycle". This regulation should insure against possible accidents.

As a means of inventory control a log will be furnished to indicate daily progress in the Tape Management System. This log will be used to indicate the last volume cleaned or tested on the previous day.

This information will supply the designated third shift personnel with a methodical point of reference to indicate exactly which volume will be cleaned on the next day's cycle. This log will be reviewed on a daily basis by the Operations Scheduler.

b. Third Shift Operations Personnel

(1) Initiate on a sequential basis the methodical cleaning of all tape volumes in the KCMOPD Tape and Disk Library.

(2) Set the controls on the Kybe TMS-70 for this phase of the Tape Maintenance at 'Clean-Full Cycle' only.

(3) After completion of the Full-Cycle clean of each volume, place on each tape volume, a Tape Maintenance Program external label. Indicate in the "Initial use" field the current date of cleaning.

(4) Return the tape volume to the tape library for future use.

(5) It will be the responsibility of the Computer Supervisor to insure that the tape cleaning device is set and locked into position on a daily basis.

c. All Operations Personnel

(1) Check the appropriate usage box after each tape is used. This information will flag each respective tape for re-cleaning after five uses.

(2) Place all tapes used for the fifth time on the work table adjacent to the tape cleaner. These tapes will then be cleaned again and returned to the usage/cleaning cycle.

(3) The usage/cleaning cycle will continue until the tape volume is used for the twenty-fifth time and the data on the tape then copied to another volume or released.

(4) The volume will be tested first without cleaning. This will give an indication of errors before cleaning. If the tape has no physical damage, it will be cleaned and again tested for errors.

Note: The cleaning cycle will not erase data on the tape as opposed to the test cycle which will erase all data except for the header and volume one labels.

(5) After completion of the Test Cycle, indicate clearly in the fields provided on the Tape Management System label:

- The footage of the indicated tape volume.
- The number of write-check errors if less than or equal to five.
- Date of testing.

(6) If any physical damage is indicated do not release the damaged volume to the usage cycle regardless of any other conditions.

(7) If there are more than five write-check errors, route defective tape to the Operations Scheduler.

3.17 Time of day for IPL and F1 Log Tape Changes

a. A special internal Date/Time routine allows automatic changing of the date and time in the communications region of all three partitions at midnight, removing the need to IPL at midnight for a date change.

b. The F1 Log Tape will be changed at the end of job RB513, (General Index Reorganize) each morning between 0115 and 0130 hours, Tuesday through Friday. If RB513 is not scheduled, remove the F1 Log Tape at 0300 hours. This procedure will provide more uniform distribution of data on the F1 Log Tapes by having data from approximately 0115 one day to approximately 0115 the next day.

3.18 Log Recovery after Abnormal Termination

a. A BPS and a DOS program now exist to recover log records that are not written onto the log tape following an abnormal F1 termination.

(1) The BPS stand-alone program was developed to run following a BPS dump of a F1 termination or a CPU hardware malfunction (disable or hardstop).

- BPS - to run this program (RB567), insert the deck in the card reader and ready the reader. Dial in the reader address on the CPU and disable the interval timer. Press the load button on the CPU and the first few cards

of the deck will be read in causing the system to enter a wait state. Then press the interrupt button on the CPU and the program will continue.

- Do not run the BPS program directly after the IPL from SYSRES. In this case, if an abnormal F1 termination preceded the IPL, write a single tapemark on the log file and resume processing
- If the on-line log tape (X'483') is used during test periods, the DOS or BPS log recovery program should be run following any abnormal F1 termination during that test period.
- The BPS program deck is located in the card cabinet and a run sheet is listed in the run book filed under job name 'RB567'.

(2) The DOS program was created to run following an abnormal F1 termination after which no IPL procedure is initiated.

- DOS - after cancelling F1 and the pause statement 'EOB TO RESTART F1' prints out on the console, stop F1. Press the request button on the console. Once into the attention routine, enter 'LOGIT' on the console.
- Do not run the DOS program while F1 is still running. Results are unpredicatable.

b. Except for the run procedures and the system under which they run, both DOS and BPS programs run identical to each other.

c. The following messages may be printed on the console during a run of either program, or if operating on system 370 the message will print on the 1403 printer:

(1) 'NO DTF WAS FOUND, WRITE LOG FILE TAPEMARK'. If this message is received, the console operator must write a single tapemark on the log file prior to initiating F1 again.

(2) 'DTF FOUND CLOSED, NO TAPEMARK REQUIRED'. No action necessary after receiving this message.

(3) 'DTF FOUND OPEN, TAPEMARK WRITTEN'. No action necessary after receiving this message.

(4) 'LOG RECOVERY EOJ'. Resume normal processing after receiving this message.

d. In any instance when X'483' is down, due to malfunction, use tape drive X'482' as the log file drive. There is a second copy of RB567 (BPS) marked for X'482' in the card cabinet. Do not use the DOS program if X'483' is not the log file drive. The DOS program will only work properly if the log file is on X'483'.

TABLE OF CONTENTS

08 PROJECT MANAGEMENT STANDARDS

1.0 General Introduction

1.1 Discussion of the Need for Standard Project Management

The typical Criminal Justice Data Processing Department exists in a complex environment that requires a degree of order and communication in conjunction with a specific methodology of managing the implementation and development of the computer systems. The complexity of the telecommunication systems that are being employed in the criminal justice area, the nature of the typical user, and the usual unscheduled demand for system service necessitates a standard method of communication such that priorities can be re-established, projects can be completed on time, and if priorities are shifted, projects can be rescheduled with some degree of accuracy such that the user demands can be satisfied.

1.2 Definition of the Typical Systems Development Process

One of the major prerequisites for effective management of a systems project is a well defined, consistent systems development framework such that any system project can be defined in a standard methodology. The flowchart of the systems development at the end of this section describes the usual systems development process. A general understanding of each of these individual process phases is mandatory before discussion of a method of managing and projecting each of these phases.

The systems development process begins with an approval that some problem or project has been defined, and a group of resources has been allocated, to improve the definition of the problem or improvement. It ends with an installed and documented system within the scope of the related initial definition. The process, as described in the following flowchart, is divided into nine phases. Critical review points are not shown but it is expected that at the end of each major phase communication with appropriate management and user personnel should be accomplished such that user satisfaction and communication is maximized. A brief description of each of the phases follows:

a. Document Present System - The systems analyst, in order to design a new system which will meet the needs of the user, must have an understanding of how the present system operates, and why. The objective of this phase is to gain this understanding. In addition, the analyst should collect facts about the present system which will be basic to the design of the new system. Volumes and levels of effort, for example, should be established at this time so that they can be relied upon, both in establishing benefits and in designing processes, during the subsequent phases.

During this phase, the analyst often concentrates on documents and files, and what is done to them. By this concentration, he finds quickly the understanding he needs about how the system now works, what decision rules are critical, and where the new system benefits will come from. Activities of the phase are generally interviews of personnel involved in the present process and collecting copies of documents and records used or created by the process.

The level of detail at which the analyst works in this initial phase must, at the minimum, give him a thorough understanding of both the what and why of the present system. The scope of this understanding should include all functions which may be affected in any way by the new system. Beyond this minimum level needed to understand, the analyst should collect detailed data on functions, files, and documents which are subject to change by the new system.

The phase ends when the analyst has gained the necessary level of understanding and documented the processes which are to be changed by the proposed system.

b. Define New System - The principal objective of the New System Requirements is to define the new system from the point of view of the user organizations, and to make these organizations thoroughly aware of how they will be affected by the new system. Whatever the potential benefits of the system, it is the user who must finally cause all actual benefits to be realized. Only when he has a complete understanding of what he must do and what the system will do for him can the user be expected to attest that he will achieve the benefits.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

The definition of the system from the user's point of view is largely the identification of what the user will do and receive when the system is in operation. Major time requirements of the phase usually involve defining the inputs which need to be provided and outputs which are to be produced. The controls which the user will provide and manual processes which he must perform are also identified.

During this phase, the analyst works at a level of great detail with respect to user activities. The specific content of each input document and output report should be defined. Each user organization function in the new system must be identified in sufficient detail to allow knowledgeable estimates of time requirements. The level of greatest detail work in this phase is in the complete specification of all data elements in the system. This includes not only input elements, but also elements generated within the computer system together with the rules for their generation. It is, of course, the constant requirements of the analyst that he define the system within the capability of the equipment and personnel, which must make it work. This requirement can usually be met in this phase with only general definitions of the necessary computer processes.

The completion of this phase is a detailed documented user-oriented system specifications.

c. New System Design - The objectives of the system design phase are:

(1) To complete the system specifications, detailing what the computer system must do and when it must be done, and

(2) To define the specific way in which any data processing equipment will be used.

The activities of the phase follow the above sequence of objectives of the phase. First, the analyst identifies the required processing sequences and the logical files of the system, usually with process charts of the various processing cycles. Second, with perhaps the assistance of a computer specialist, he defines the actual files (content and organization) and computer runs (required functions).

INDEX NUMBER
08-1.0

The level of detail at which work is done in the System Design phase varies from the "big picture" of the functional computer process charts to the minute detail of input and report layouts. Recall that the previous phase has identified the inputs and outputs but only in a fairly general way. These should be made specific at this time to insure that requirements are communicated to the Program Specifications phase. Less detailed is the computer process definition, by intention. The analyst is encouraged to examine a number of computer processes before selecting the best approach.

d. Programming And Program Test - The objectives of the programming phase are to write and test computer programs for the system. The basic logic for the programs was developed in the preceding Program Specifications phase. Many of the techniques used in writing the programs are determined by the programming standards of the installation.

One of the first activities of this phase is the definition of the detailed computer logic. Flowcharts and/or decision tables are used to develop how the program is to operate. Following the definition of the logic, the program is coded for the computer. Once the program has been coded, tests are performed on the program to insure the requirements, as defined in the program specifications, have been met. At this point, typically, the testing orientation is toward the testing of individual programs as compared to the testing of all of the programs as a single system.

Following program testing, the program test data and test results are documented for possible future use in the System Test phase and in program maintenance.

e. User Training - The objective of this phase is to equip the using organizations for the operation of the new system. Included in the phase is the preparation of a detailed plan for training the users, together with identification of responsibilities for the various tasks of training and procedure writing. While many of these tasks will be performed by other persons, the analyst has the objective of scheduling the activities and seeing that they are successfully completed.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

It is highly desirable for the user departments to perform as many of the preparation tasks as possible. By means of this involvement, the user may begin to accept the system and obtain a thorough working knowledge of the details of the system.

f. Write Program Specifications - The objective of the Program Specifications phase is to document the requirements of each computer program. These specifications should illustrate completely what the program must accomplish. They are not concerned, however, with how the program is to be written, nor are they expected to repeat the programming standards of the installation. These latter items are considered within the scope of the Programming phase. For example, sequence checking of an input file would not be a specified program function if an installation standard called for all input files to be sequence-checked. Logic would be included in the specifications only as required to communicate processing requirements to the programmer.

One of the first activities of this phase is the final definition of files. All master and inter-run files are detailed, and all coding schemes are written down if this has not already been completed. Then the functions of each program are identified, and supporting documents are prepared to complete the definition of these functions. By the end of the phase the analyst has prepared a "package" for each program in the system. Included in these "packages" are many end products of earlier phases.

The level of detail of work in this phase is the level necessary to communicate exact requirements to a programmer. The specifications should be sufficient to allow a programmer to proceed after a brief introduction by the analyst. The specifications should be understandable to a programmer who wants to understand what is to be done. If the specifier is to be the programmer, less detailed specifications may be needed than at other times. In this case, the specifications are still required, however, if only as a basis for programming supervision or for review by the project leader.

The phase ends when each program specification has been accepted by the programming organization.

INDEX NUMBER
08-1.0

g. Conversion Planning - The objectives of this phase is to develop a detailed plan for converting the present system to the new or revised system. Included in the phase are plans and schedules for the following elements of conversion:

(1) Files to be created or modified.

(2) Forms necessary during the conversion period.

(3) Parallel processing of the old and new systems.

(4) Sequence in which organizations are to be converted.

(5) Manpower and equipment requirements during conversion.

Scheduling the activities necessary to install a new computer or other major data processing equipment is not covered in this phase. A separate project should be established for managing an equipment conversion.

h. System Test - The objective of this phase is to insure, under the direction of the Project Leader, that all of the components of the system will work together to meet the defined system requirements. Included in the test are not only the computer programs, but also the user activities and all data processing operations. The activities identified as belonging to the Systems Test phase assume that the computer programs have been individually tested with test data, and live data if available. It is further assumed that programs will operate from run-to-run on an error-free basis.

The System Test may consist of running of the old and new system in parallel, or of running historical data through the new system, or of processing a carefully constructed sample for comparison with predetermined results. Regardless of the specific data used for the test, it is the analyst's responsibility to see that each component of the system does what it should do. Each processing cycle should be tested.

To test the system, the analyst should try to make it fail. The programmer tends to choose test data which he knows will make the program run. The analyst must include test data which should not run. He should concentrate upon the obscure, infrequent operations of the system -- the main line has already been given much attention.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

It is often particularly effective to involve the eventual users of the system in checking the test case results. This helps to insure, before the system is "turned on," that the true requirements have been met. Furthermore, this approach involves the users in the system to the extent that they become comfortable in the use of the system, confident of what it does. When a system test is run with much user involvement, it can become the start of actual implementation of the system.

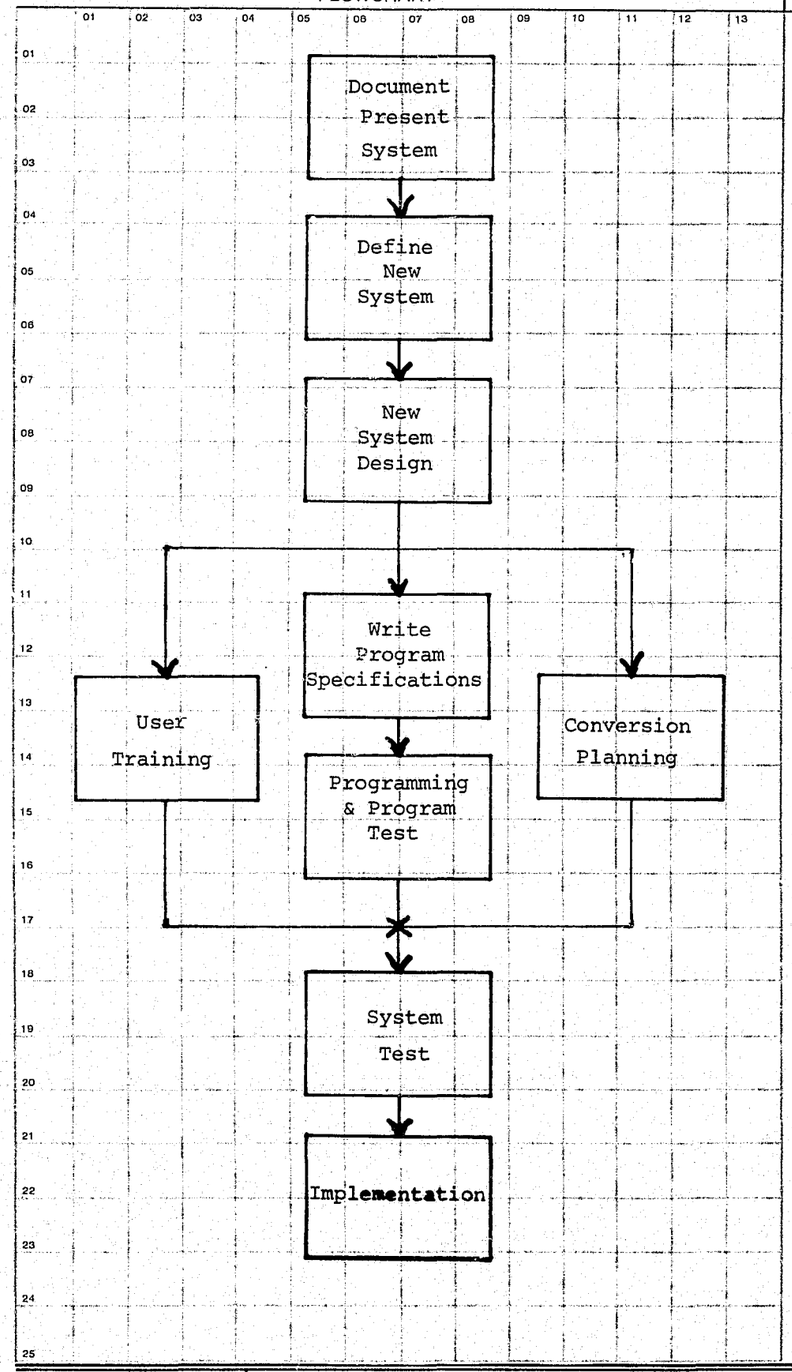
i. Implementation - The objective of this phase is to complete an orderly installation of the new or revised system. Recall that the Conversion Planning phase established schedules and responsibilities for the various activities of implementation. Certain of these, file conversion for example, may have been done before this time because they were required for system test. The remaining conversion tasks plus any other undone tasks which the system requires are performed during this implementation phase.

It is always difficult to tell just when a system is implemented. Nevertheless, the analyst should strive for a clear-cut end of the project. User organizations tend to depend upon the analyst for any and all questions. Some analysts, by responding to all questions, become permanent parts of the operation of the system. For these reasons it is essential that the users accept the system as their own.

SYSTEM FLOW CHART

Page 1 of 1

FLOWCHART



DESCRIPTION

System No.	System Title: Systems Development		
Date Prepared:	Prepared By:	Revision Date:	Revised By:
Date Approved:	Approved By:		

INDEX NUMBER 08-1.0

DATA PROCESSING

MANUAL OF STANDARDS

SECTION: PROJECT MANAGEMENT STANDARDS

DATE ISSUED

DATE REVISED

2.0 Description of the Specific Areas of Project Management

2.1 Introduction

In the above sub-section, the Systems Development Process was defined in order to establish a framework from which project management could be standardized. Project management is performed in all phases of Systems Development. In the initial phases of Systems Development, the attention is given to establishing project definition, scope and high-level planning of resources. With each ensuing project development phase, these plans and projections are updated and improved depending upon the new set of circumstances and knowledge gained in this Systems Development process. The project management process has been segregated into five specific areas. These areas are:

a. Establishing project definition and project scope.

b. Establishing project phases.

c. Work plan preparation, project resource projection and the determination of the critical path.

d. Establishing manning and equipment schedules for resources.

e. Project evaluation, progress reporting and management communication.

Each of these areas, as well as typical forms used in the standard process of project management, are included in the following sections.

2.2 Establishing the Project Scope

The establishment of a project scope is one of the most critical phases of any systems development effort. Miscalculation of effective scope can ruin a systems implementation and cause the user to be totally dissatisfied with the end product. To get a better understanding of the project scope, the initial phase of systems development should be undertaken, i.e., the documentation of the present system. All problems that are incurred with the

INDEX NUMBER 08-2.0

present system are important to the new systems development effort such that the new system must at least attempt to try to eliminate the present problems. The problems expressed by the user may simply be symptoms of an underlying problem and it is well advised for the systems analyst to not accept the surface definition of problems from the user but instead should spend enough time analyzing the present system such that the real causes of the problems are detected. During this initial systems development phase, any time constraints of processing, controls, whatever, must be documented since it will be important in the establishment of the project scope. Some systems can be batch-oriented systems if time constraints are not important but in the criminal justice area, if it is determined that information must be readily available within seven to ten seconds, it will require on-line communication support. Again, time constraints are a major important factor in defining the project scope. The conclusion of establishing the project scope is actually the definition of the service level that the Computer Processing Department is going to provide to the user agency. This service level, depending upon time constraints, funds available, equipment available, will dictate the end project scope. The definition of this service level and project scope should be defined, written in a management letter, and discussed with the user agency before any additional work is performed.

2.3 Establishing Project Phases

Most of the projects in the criminal justice area are very complex. The more complex a new system is, the less chance of success the department has in implementing the system. To reduce the amount of complexity, and allow the system to be installed as effectively as possible, all complex projects and systems should be broken down into smaller "byte size" phases. These phases will allow different groups of people to work on specific phases and/or allow the project to be implemented in a logical process. Many times it is impossible to implement a complex system in one huge phase and therefore even if it requires going back and making modification to earlier phases, it is still recommended that a phased approach be used for any complex system. It is important to define these phases before any other project management procedures are performed such that the phased approach can be used in the project management documentation. Once the phases have been defined, they become subprojects to the original project scope. A Gantt Chart, as depicted in the following example, is the standard method of scheduling and projecting the relationship between the various stages. A detailed description of a standard Gantt Chart and the procedure for completing the chart is detailed below:

a. Purpose of the form:

(1) To illustrate start and end dates for all activities and projects.

b. Procedure for completing the form:

(1) Complete the heading information.

- Identify the user organization.
- Indicate the title of the work unit: project or activity.

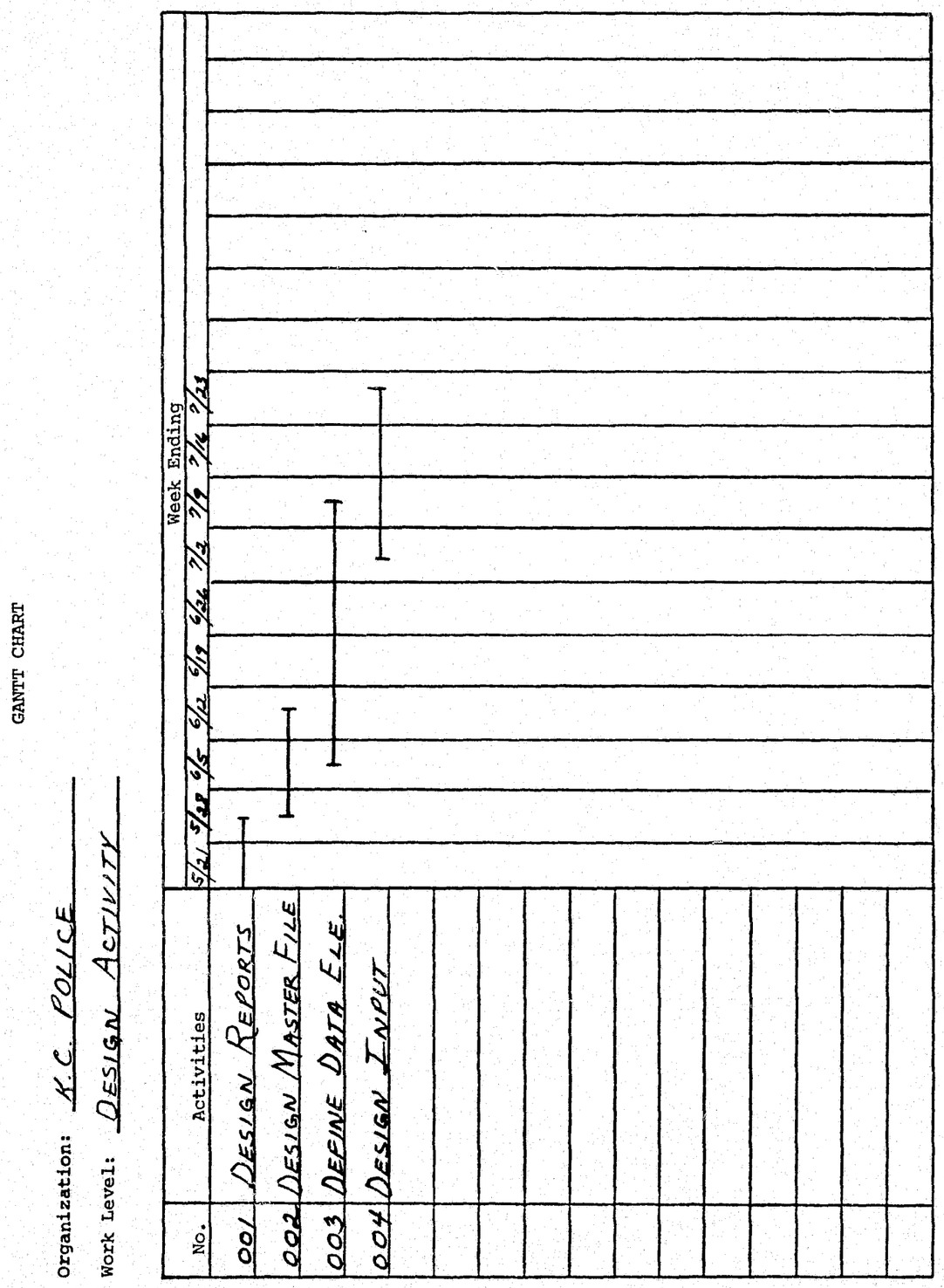
(2) Copy project and activity lists from Work Outline Forms.

(3) Develop and record time increments at the top of the chart.

(4) Draw lines from the projected start date to the projected end date for each activity and project.

(5) Use standard control chart symbols for work units: start, end, slide, etc.

INDEX NUMBER
08-2.0



2.4 Work Plan Preparation, Project Resource Projection and the Determination of the Critical Path

a. Developing the Work Plan

After defining the purposes and limits of each project, the systems group begins the work of planning by identifying the actual work to be accomplished. It is at this point that the job is broken down into its basic components. The systems group then prepares a list of activities required to complete each of these activities.

The activities then are broken down into smaller work units or tasks. Once the planner has identified these tasks, he can define the type skills required and estimate the amount of time each task will take to complete.

More specifically, the objective of this element of Project Planning is to identify the measurable work units that make up the project, the level of skill necessary to do the work, and the number of man-hours necessary to complete the project. The following outlines some of the steps involved in this element of planning:

Identify Work: The identification of the work involved is a process of stratification: beginning at the upper-most level (the project), the basic activities to be accomplished are identified and then are broken down, level by level, into the smallest work units (tasks) making up each project.

Activities: Small units of work assignable to an individual. At this level, the required personnel types or skill levels start to become apparent.

Sub-tasks: When a defined task is particularly complicated or it is difficult to predict the time required, or when the defined task's projected duration is in excess of two weeks, the task should be divided into smaller work units called sub-tasks.

During the project planning stages, it is not always possible or desirable to define all the activities of the project at the final level of detail. Those activities that remain to be

defined should be identified in the work plan along with the dates when more detail planning will be possible. However, the project plan should contain the dates for completing the initial activities and the dates when the later activities will be defined in more detail.

Estimate Time Requirements: In estimating the man-hours required to complete a project, the process begins at the lowest level, task or sub-task. Estimated time is assigned to each work unit. These units are then summed to obtain estimated times for activities which in turn are summed to obtain estimated times for projects, and finally, project estimates are summarized to obtain estimates for the entire job.

The estimates of time required to complete individual tasks are difficult to establish because of the many unknown conditions that can affect those estimates. Peculiarities of the job itself can affect how fast it is done and variations in experience or actual skill can affect the speed of analysts with the same job classification.

The forms indicated for use in this section can be used for all types of jobs. They can, for example, be used in planning new clerical systems as well as planning the implementation of EDP Systems.

The following is a description of a work plan document and how to complete it.

(1) Purpose of the form:

To prepare work lists of projects, activities, tasks and sub-tasks; to define skill levels; and to estimate the manpower required to complete the work.

(2) Procedure for completing the form:

- Complete heading information. Identify the organization and work level (project, activity, or task.)
- Identify the work units.
- Estimate manpower required by skill level or job title.
- Summarize by activity, project, and job.
- Set estimated start and target dates.

WORK PLAN

Date 2/10/23 Page 1 of 1

Organization: K.C. POLICE DEPT.

Description: NEW OFFENSE SYSTEM

No.	Work Element	Personnel Type	Man Days Required	Start Date	Target Date
001	REVIEW CURRENT SYSTEM	ANALYST	15	5/1/23	5/16/23
002	INTERVIEW USERS	ANALYST	10	5/10/23	6/5/23
003					

b. Critical Path Method of Scheduling and Controlling Projects

Data Processing Projects frequently require many concurrent as well as sequentially definable tasks to be performed in order to successfully implement the project. The Critical Path Method is a technique that can help the data processing manager schedule and control such projects. The advantages of CPM are that it provides for:

(1) Disciplined planning of a project by detailed definition of activities.

(2) Specification of the logical inter-relationships (dependencies, independencies, and inter-dependencies) of activities and manpower within and among application groups.

(3) Pinpointing of responsibilities.

(4) Assignment of time for the performance of individual activities can be accummulated to give the total project duration, as well as the scheduled start and finish basis.

(5) Calculation of the amount of leeway available for each activity and on an overall project basis.

(6) Target dates can be realistically assigned as goals of meeting milestones.

(7) Immediate determination of the effect of slippage in the performance of an application group on all other groups.

(8) Project updating to keep management abreast of progress.

(9) A means of evaluating alternative approaches, strategies, and objectives.

(10) Establishing better communications between personnel involved with the project.

The above list contains a few of the advantages of using CPM. In order to implement the technique, there must be understanding of the technique. It cannot be imposed and cannot be used for the purpose of evaluating individual performance. It is used primarily as a tool for project monitoring, to pinpoint critical areas, and to forecast effects of slippages so that early remedial actions may be taken.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

c. Input Requirements

To use Critical Path Method, the overall objectives must first be defined. Each application group (if there is more than one group) then defines its own sub-objective and the activities necessary to reach that sub-objective. The listing of activities should be done by the project supervisor.

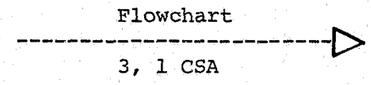
With the activity list, each project leader/supervisor is responsible for the construction of a network to show the relationship among activities. Then, an estimated duration is assigned to each activity. Estimated resource utilization should also be attached to each activity. If there is more than one project group the sub-networks of individual groups are integrated into a master network by the coordinators so that sub-objectives and important milestones can be phased into the large network.

d. Project Development

(1) Network Construction

The initial preparation involves the definition of activities, the estimation of durations, the assignment of numbers to events and the drawing of a network. The following conventions shall be adopted in constructing networks:

- An arrow represents an activity. Each activity, except for dummy activities, will have a unique alpha-numeric description or identifier above the arrow and the duration in units of days and manpower requirements below the arrow. The example below,



shows that the function "Flowchart" is estimated to take 3 days and require one computer systems analyst full time. The length of each arrow is arbitrary and should be drawn for the best clarity; no relationship exists between activity duration and the length of the arrow.

INDEX NUMBER
08-2.0

- A node, or event, represented by a circle, serves as a connector between activities. A node signifies the instant when one or more activities begin or end; it does not occupy time, utilizes no resources, and costs no money. A unique number and an optional description are placed near the circle. For example:

The above node shows Event 25 as the milestone when testing is started. An activity may be identified by a unique set of nodes preceding and succeeding it. For example, Activity (4,5) is the unique activity connecting nodes 4 and 5.

e. Input Sample

To demonstrate the use of CPM in coordination of system development, a sample network is shown in Figure 05.16-1. The activities for each application group for system development is grouped into six phases.

(1) Document Present System.

(2) Define New System.

(3) New System Design and Writing Program Specification.

(4) Programming and Programming Test, User Training and Conversion Planning.

(5) System Test and Implementation.

(6) Operation.

The network in Figure 05.16-1 depicts the activities of the first five phases for one application group. The sub-objective of the effort is to complete conversion and parallel operation as scheduled; the sixth phase, operation, begins with the successful completion of phase five and continues indefinitely.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

The network contains a representative set of activities in each phase, but it is by no means necessarily comprehensive. The first three activities comprise the first phase. The activities between nodes 12 and 42 are designated as the second phase. The third phase is represented by the activities between nodes 42 and 48, 50 and 56. The fourth phase consists of the activities from 48 to 90, 50 to 90 and 56 to 90. The fifth Phase is represented by the activities between 90 and 96.

The list of activities in the network shows little parallelism, or few concurrent activities, at the beginning of the project. The programming, testing, and procedural development phase is shown with much parallelism. In that phase, three different runs, each composed of more than one program module, are broken down into more detailed activities. If a network cannot be completely defined at the beginning, only the initial phases are to be developed in detail. An activity should then be included in the early stage to refine activities for later phases.

There are activities or constraint dates that each application group must be aware of, but has little control over. For example, training may be part of every group's sequence of activities, but the training session will only be held at given times, to which each group must adjust. Therefore, the training coordinators and the liaison, when the sub-networks are integrated. Another example is constraint on the availability of software.

Time estimates shown on the network are typical times expected in a project, but they are not necessarily valid for any project group. It is to be emphasized that each project supervisor is to construct his own network embodying the logic and time estimates required for his effort. Figure 05.16-1 serves merely as a conceptural guide.

f. Usage

The CPM diagram can be translated into a series of Gantt charts representing the absolute dates to start and finish the identified tasks and the associated personnel to be assigned to these tasks. The CPM diagram and Gantt charts can now be used by the manager to:

(1) Plan and schedule the start of various tasks.

(2) Determine resource assignments.

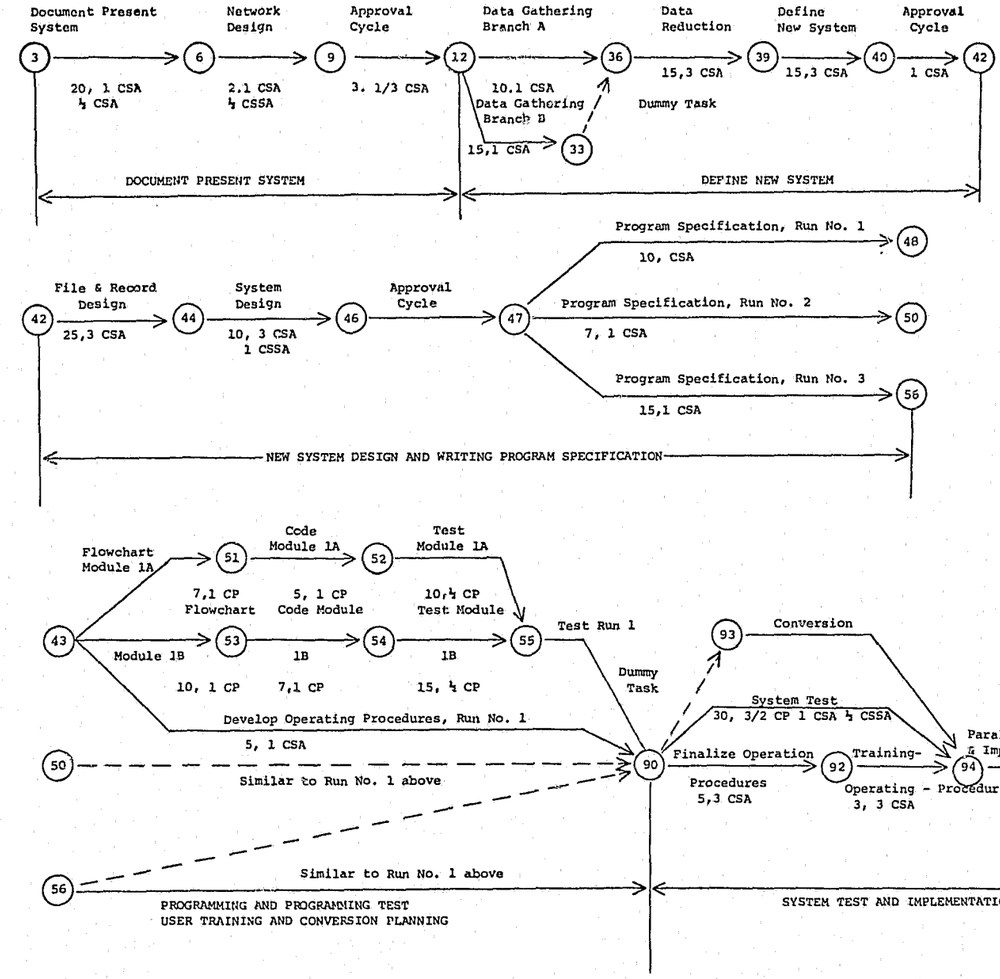
(3) Analyze the effect of changes in the plan.

INDEX NUMBER
08-2.0

Periodically, say every two weeks, the status of where the project is can be reviewed and appropriate steps taken. In addition, as the project develops, specific tasks to be performed in the latter portions of the project will be identified and the diagram and charts updated, thus keeping management aware of the true status of the project.

Computer programs are available to process these diagrams together with update information. In most cases, the size of the project will not be great enough to warrant the effort involved in procedurally utilizing the program. However, for very large projects, the use of the program may be justified.

CPM NETWORK



Legend
Code No. - X
CSA - Computer System Analyst
CP - Computer Programmer
CSSA- Computer Software System Analyst

FIGURE 05.16-1

2.5 Establishing Manning and Equipment Schedules

Scheduling human and machine resources for complex projects is difficult for a single project let alone a data processing department that may have four or five major projects being worked on at one time. It is typical for a department to have a fixed number of staff and therefore the projected day of when projects can be completed is dependent upon the number of individual staff members and the priorities of each project and the loading or staff requirements per project. The detail work plan and the critical path is very important in trying to assign staff members to individual projects. Once the critical path has been determined for each project the staffing of the critical path is the most important section for staffing. The non-critical paths can be staffed as is needed and therefore reduces the amount of confusion of attempting to staff projects. Establishing the manning of each individual project is a two-phase problem. The first phase is determining what resources are needed for each individual work element and the second is insuring that each individual staff member has a full work load but not an impossible work load. It is important to note that each individual is different and that by reducing confusion by establishing good project plans and predicting effectively the work that each individual is responsible for will reduce the amount of tension and confusion within the department.

a. Personal Planning for Each Project

A personnel planning form is to be completed for each major project. The planning form breaks down each individual activity and relates the individuals that will be working on each activity within the work plan. Again, the personnel planning form is completely dependent upon the effective definition of a work plan and the formalization of defining the elements within each work plan. The personnel planning form gives a picture of the resources that are to be used on the project but it does not give information as to the work loading per individual in total nor the loading of each project per individual. The details for completing this form and a sample of the standard form is enclosed.

(1) Purpose of the form:

To provide a means of identifying specific personnel by activity. This form relates specific personnel to an activity within a time frame.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

(2) Procedure for completing the form:

- Complete the heading information:

  Identify the organization.
  Identify the project.
  Check the work units in days or hours.

- Copy the activity lists from Work Plan form.

- List the names of personnel across top of page.

- Record the time related to the activity and the personnel in the rectangle.

  Write the start and the end dates in the upper two rectangles.

  Record the man-hours/days in the lower half of the rectangle.

- Total all columns and record the total man-hours/days for each person.

- Total across and record the total effort to be applied against each activity.

INDEX NUMBER
08-2.0

PERSONNEL PLANNING FORM

Date: 2/15/22

Page: 1 of 1

Organization: K.C. POLICE DEPT.

Project Name: ARREST SYSTEM

Activity Description / Names	SMITH	JONES	RAY										Total Hours
DESIGN SYSTEM	5/1 5/12	5/1 5/19	5/1 5/15										22
	10	12	10										
	6/5 6/20	6/1 6/10	6/1 6/15										7
	7	5	10										
		6/5 6/25											26
		9											
													2
Sub-Totals	17	26	10										

Prepared by: JOE DOKES

b. Individual Work Loading

The individual work loading must be performed by the group leader that supervises each individual. Each individual must be reviewed separately and therefore a separate loading form is prepared for each individual for the next projected six months. It is important to remember that an individual should not be overloaded with work or responsibilities and to eliminate conflicts between projects, close coordination with the individual as to how they are progressing with each individual project and activity list is mandatory. The individual work loading form is designed such that each project that individuals are responsible or working on are listed down the left-hand side of the page and the months projected for the next six months are listed at the top of the page from left to right. The number of hours that have been scheduled from the Personnel Planning Form is posted to the correct month that the individual is to work on each project. Totals are then totaled per month and it is expected that if the total hours for an individual exceeds the expected hourly figure per month, the work activities will be rescheduled or transferred to a different individual. The directions for completing the form and a sample of the standard form is enclosed.

(1) Purpose of the form:

To provide a work sheet for each individual the number of hours that is expected for each project and to insure that an individual is not being under-loaded or over-loaded.

(2) Frequence prepared by the group leader and is performed once a month or as needed.

(3) Procedure for completing the form:

- Complete the individual line by writing the individual's name.
- List the projects down the left-hand side of the form under the project's heading.
- List the next six months one month per column at the top of the form.

- Post the hours expected for each individual from the Personnel Planning Form to each individual's activity by month square.
- Total the number of hours per month for the individual insuring that they do not exceed 176 hours.

INDEX NUMBER
08-2.0

INDIVIDUAL WORK LOADING FOR SIX MONTHS

Date: 7/15/23

Page: 1 of 1

Individual: JOE SMITH

Projects \ Months	JUNE	JULY									Total Hours
ARREST	10	20									30
OFFENSE	30	10									40
ASAP	10	30									40
TRAFFIC	20	20									40
Sub-totals	70	80									150

Prepared by: JACK DAY

2.6 Project Evaluation, Progress Reporting and Management Communication

The management of the department and of the user agencies are continually bombarded with questions as to when the project is to be completed, when or if the project is on schedule, and if the project is behind schedule how much is the project delayed? It is typical for data processing individuals to talk about percentage completion but it is well understood that percentage completion of a project is normally and usually way overestimated. There are many projects that are approximately 95 percent completed but in fact never do become completed. To eliminate the amount of guesswork and to state as factually as is possible the status of a project and enabling management as well as individuals within the project the evaluation of the project and for maximizing management communication and eliminating inconsistencies it is recommended that a procedure of reporting earned hours versus projected hours be used as a standard method of communication. This principle warrants a definition of the term earned man-hours.

> In the preceding discussion the definition and the use of a specific work plan form was established as a standard. The summary of a work plan is the basis of reporting the status of projected man-hours versus earned man-hours. Once an individual element of a work plan has been completed the number of projected man-hours in the work plan now becomes earned man-hours. If it was decided that a specific activity was going to take 100 man-hours and the individual spent 200 man-hours, at the completion of that activity 100 man-hours were actually earned.

By reporting earned man-hours versus calendar days or weeks it is possible to determine whether the project is behind schedule or ahead of schedule according to the original work plan. If the original work plan is grossly in error the project will be either behind schedule or ahead of schedule depending upon the correctness of the original work plan projection. The directions on completing the project schedule report and a sample of the standard project schedule is included.

a. Purpose of the form:

> To graphically illustrate to management the current status of a project. This document is used to communicate the status of a project to both technical and non-technical members of management.

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

PROJECT MANAGEMENT STANDARDS

DATE ISSUED | DATE REVISED

b. Frequency:

> Initiated by the project manager when the work plan has been completed and is used for a weekly report or monthly report as is warranted.

c. Procedure for completing the form:

> (1) Project name
>
> Write the project name in the appropriate space.
>
> (2) Project leader
>
> Enter the name of the individual who is responsible for the project.
>
> (3) Project start date
>
> Enter the planned start date for the project.
>
> (4) Project completion date
>
> Enter the planned completion date.

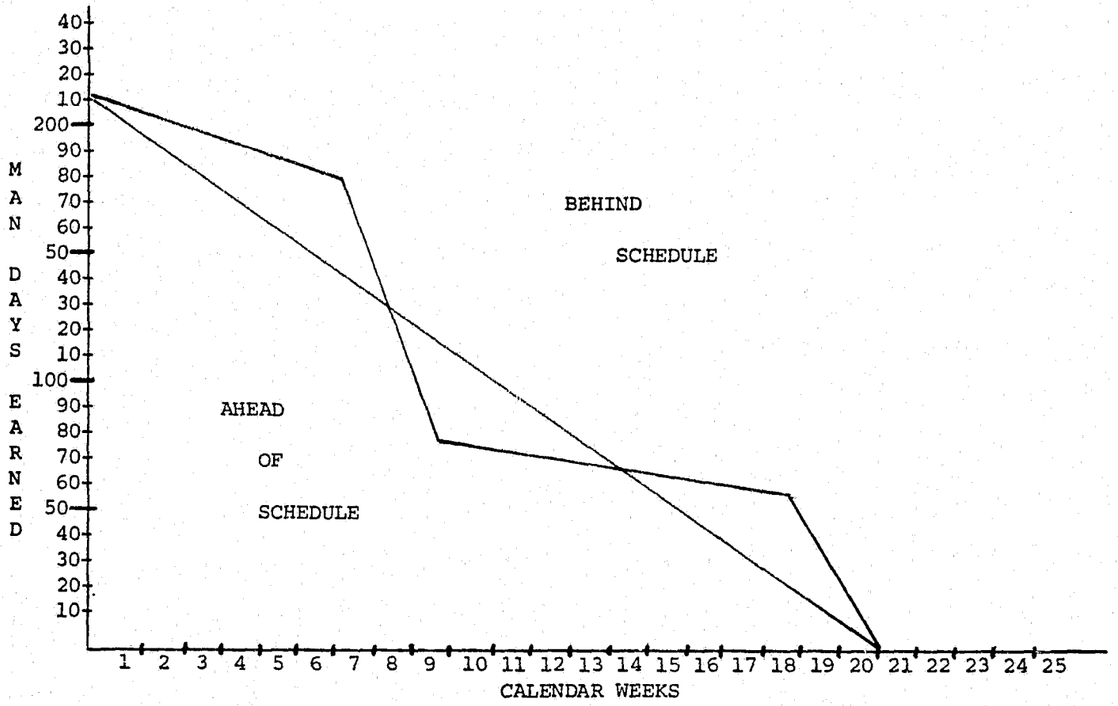
Draw a line from the number of man-days or man-hours on the vertical access on the left side of the schedule to the point on the horizontal schedule of calendar weeks when the project is to be completed. This line represents the progression of earned hours in conjunction with calendar and depicts a project that is perfectly on schedule. Complete the Summary Work Plan by listing the summary schedule of the work plan and the projected man-days or man-hours for each individual work element. The earned man-days will be posted to each individual work element as the man-days are earned in the reporting process. When using the form for reporting purposes the earned man-hours or man-days are posted to the summary work plan and totaled and the total earned man-hours or day is graphed on the project schedule and a dotted line drawn from the date the project was started to that point. If the dotted line is in the Behind Schedule area of the chart this represents the magnitude and severity of the lateness of the project.

INDEX NUMBER
08-2.0

PROJECT SCHEDULE REPORT

Project Name: K.C. POLICE DEPT Project Start: 12/1/72

Project Leader: JOE DAVIS Project Completion: 7/1/73



SUMMARY WORK PLAN

No.	Description	Projected Man Days	Earned Man Days
001	REVIEW CURRENT SYSTEM	45	45
002	DESIGN NEW SYSTEM	100	0
003			

d. Earned Hours Work Sheet

In order to complete the project schedule report it is mandatory that earned hours be accummulated by individual per project such that when time comes for a project report it is possible to accummulate the earned hours per the summary work plan and complete the project schedule report. The description of completing the earned hours work sheet and a sample of the form is included.

(1) Purpose of the form:

To provide a work sheet for consolidating the earned hours for a project. This information will be used to update the project schedule report.

(2) Frequency:

Prepared weekly by group leader.

(3) Procedure for completing the form:

Information	Action
Project Name	Indicate name of project.
Start Date	Enter date the work sheet is first used on the project.

For each task checked completed on the Time Report the following information is entered:

Individual	Enter name of individual assigned to the job.
Current Week Earned Hours	Record planned/earned hours for completed work using the individuals Time Report.
Cumulative Earned Hours	Update to reflect earned hours reported for the week.

EARNED HOURS WORKSHEET

Project Name: ARREST

Worksheet Start Date: 7/1/73   Page 1 of 1

Individual	Current Week Earned Hours	Cumulative Earned Hours
SMITH	25	100
JONES	10	65
TOTAL	35	165

TABLE OF CONTENTS

09 PROJECT CODE CATALOG

09 PROJECT CODE CATALOG

1.0 Agency Codes

A = KCMOPD
B = City
C = Jackson County
D = Johnson County
E = Municipal Court
F = Other (Commercial)
G = Other (Governmental)
H = Overland Park
I = Independence
J = Juvenile Court
K = Joplin
L = Lee's Summit
M = Roeland Park
N = OS Conversion
O = SOP Task Force
P
Q
R = Regional
S = State (Missouri)
T = Topeka
U = St. Joseph
V = Kansas City, Kansas
W = Carroll County

2.0 System Codes

A = Teleprocessing System
B = Real Time System
C = Payroll
D = Traffic
E = Patrol Workload
F = Offense
G = Personnel
H = DOS/SYS Maint
I - LEMRAS
J = Accident
K = U C R
L = Detective
M = Outside Agency
N = Court Docket
O = Modus Operandi
P = Property Inventory
Q = NCIC Operational
R = Teleprocessing Operation
S = Scheduled Hardware Maint
T = Unscheduled Hardware Maint
U = Arrest
V = Computer Systems Division
W = Regional Laboratory
X = Warrant System
Y = Fleet Maintenance
Z = Planning Model Simulation System
Ø = ASAP (Alcohol Safety Action Program)
1 = Sheriff's On-Line System
2 = Circuit Court Reporting System
3 = Prosecuting Attorney's Update System
4 = Parole and Probation Reporting System
5 = Mobile Terminal
6 = Caseload Analysis
7 = Corrections System

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

DEPARTMENT CODE CATALOG

DATE ISSUED | DATE REVISED

3.0 Run Category Codes (Note - Codes F, G, H, I and J are for use by Computer Operations personnel)

A = Operational Run
B = Assembly
C = Test
D = Training
E = Catalog
F = Scheduled Maint
G = Unscheduled Maint
H = On-Idle
I = F1 Down Time
J = DOS System Error Recovery

4.0 Data Element Code List

Please see following pages for above code list.

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
DEPARTMENT CODE CATALOG
DATE ISSUED
DATE REVISED

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Accident Class	1	Accident			
Accident Involved	1	Accident			
Actions	6	Asap			
Activist	1	Real Time			
Actual Apartment Number	4	Civil Index			
Actual St. Address Number	5	Arrests			
Actual Street Direction	2	Arrest			
Actual Street Name	10	Arrest			
Actual Street Suffix Code	2	Arrest			
Add or subtract Money Stolen	1	Offense			
Add or subtract Value of Auto Stolen	1	Offense			
Add or Sub. Value of Auto Recovered	1	Offense			
Add or Sub. Value of Clothing Stolen	1	Offense			
Add or Sub. Value of Furs rec.	1	Offense			
Add or Sub. Value of Furns Stolen	1	Offense			
Add or Sub. Value of Jewels Rec.	1	Offense			
Add or Sub. Value of Jewelry Stolen	1	Offense			
Add or Sub. Value of Misc. Rec.	1	Offense			
Add or Sub. Value of Misc. Stolen	1	Offense			
Add or Sub. Value of Money Rec.	1	Offense			
Address Type	1	Dispatch			
Age	2	Offense			
Age	2	Tickets			
Agency Criminal Jacket/File No.	8	Arrest			
Agency Case Number	8	Real Time			
Age of Arrestee	2	Arrest			
Age of Offender	2	Offense			
Age of Person	2	Accident			
Alert Number	7	Real Time	X	X	
Alias Flag	1	Real Time			
Alias Known As	1	Real Time			
Amount of Fine	4	Arrest	X		
Amount of Forfeiture Collected	7				X
Amount of For. Reduced to Judgement	7				X
Amount of For. Set Aside	7				X
Apartment Number	4	Offense			
Apartment Number	4	Tickets	X		
Apartment Number	7	Real Time			
Appeal	1				X
Area-Geographical	1			X	
Arraignment Date	6			X	X
Arraignment Court	1				X
Arraignment Court Division	2				X
Arraignment Judge	2				X
Arraignment Plea	1				X
Arraignment Prosecutor	2				X
Arraignment Time	4			X	

INDEX NUMBER
09-4.0

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Arrest Count	1	Arrest			
Arresting Officers Unit	2	Arrest			
Arrest Number	8	ASAP			X
Arrest Number 2	8				X
Arrest Number 3	8				X
Arrest Number 4	8				X
Arrest Number 5	8				X
Arrest Number 6	8				X
Arrest Number 7	8				X
Arrest Number 8	8				X
Arrest Number 9	8				X
Arrest Report Number	8	Arrest	X		X
Arrest Type	1	Arrest	X		
Assault Police	1	Real Time			
Assisting Car 1	4	Dispatch			
Assisting Car 2	4	Dispatch			
Assisting Car 3	4	Dispatch			
Assisting Car 4	4	Dispatch			
Associated Alert or Referral No.	7			X	
Associated Arrest Number	8	Arrest			
Associated ID	8	Real Time			
Attitude	13	ASAP			
Attorney	1			X	
Auto License	8	Tickets	X		
Auto License Number	8	Real Time			
Auto License State	2	Real Time			
Auto License State	2	Tickets	X		
Auto License Type	2	Real Time			
Auto License Type	2	Tickets	X		
Auto License Year	1	Real Time			
Auto License Year	1	Tickets	X		
Autos Stolen	3	Offense			
Auto Theft	1	Real Time			
Balance	7	ASAP			
Beat Number	4	Real Time			
Beat of Occurrence	4	Tickets			
Beat of Occurrence	4	Dispatch			
Beat of Occurrence	4	Arrest			
Beat of Occurrence	4	Offense			
Beat of Occurrence	4	Ticket			
Bond Amount	7				X
Bond Amount	4	Real Time			
Bond Amount Forfeited	7				X
Bond Court	1				X
Bond Number	6				X
Bondsman	10				X
Bond type	1				X
Booking Number	6			X	

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
DEPARTMENT CODE CATALOG
DATE ISSUED
DATE REVISED

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Breath	4	ASAP			
Breath Test	1	ASAP			
Build	4	Modus Operandi			
Burglary	1	Real Time			
Business Name	37	Dispatch			
Business Name	43	Real Time	X		
Call Classification	4	Dispatch			
Call Classification Reclassified	4	Dispatch			
Cancel Code if Record to be Can.	6				X
Car Radio Number	4	Dispatch			
Case Number	8	Real Time			
Case Report Number	8	Real Time			
Case Report No. 2 (Jacket No.)	8	Real Time			
Case Report No. 3 (Jacket No.)	8	Real Time			
Case Report No. 4 (Jacket No.)	8	Real Time			
Case Report No. 5 (Jacket No.)	8	Real Time			
Census Block	4	Real Time			
Census Block	4	Arrest			
Census Block	4	Offense			
Census Block	4	Accident			
Census Block	4	Ticket			
Census Track	4	Real Time			
Census Track	4	Real Time			
Census Track	4	Real Time			
Census Track	4	Arrest			
Census Track	4	Offense			
Census Track	4	Accident			
Census Track	4	Ticket			
Change of Venne	2				X
Charge Number	1				X
Charge Number 1	5				X
Charge Number 2	5				X
Charge Number 3	5				X
Charge Number 4	5				X
Charge Number 5	5				X
Charge Number 6	5				X
Charge Number 7	5				X
Charge Number 8	5				X
Charge Number 9	5				X
Charge Type	1	Real Time			
Charges Declined	1				X
Child Lives With	7			X	
City	10	Tickets			
City	15				X
City Name	10	Real Time	X		
City of Occurrence	10	Real Time			
City Occurred	10	Accident			
City of Occurrence	10	Offense			
City of Occurrence	10	Tickets			

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
City of Occurrence	10	Arrest			
Clearance Add Sub Code	1	Offense			
Clothes Stolen	3	Offense			
Clothing	3	ASAP			
Complaint Date	6				X
Concurrent Sentence	1				X
Continuance Court	1				X
Continuance Requested by agreement	2				X
Continuance Requested by Court	2				X
Continuance Requested by Deft.	2				X
Continuance Requested by State	2				X
Continuity	3	ASAP			
Contributing Circumstances	2	Accident			
Conviction Type	1	Arrest	X		
Court Jurisdiction	1	Arrest	X		
County of Occurrence	1	Accident			
Court Case Number	6				X
Court Date	6	Tickets	X		
Court Room	1	Tickets	X		
Court Status	2			X	
Court Time	4	Tickets	X		
Criminal Status	1	Real Time			
Current Court Date	10				X
Current Judge	2				X
Date	6				X
Date	6	Real Time			
Date Cleared	6	Offense			
Date Continuance Requested	6				X
Date in Detention	6			X	
Date Issued to Officer	6	Ticket			
Date Occurred	6	Offense			
Date Occurred	6	Ticket			
Date Occurred	6	Accident			
Date of Birth	6	Real Time			
Date of Disposition	6				X
Date of Disposition	6	Arrest	X		
Date of First Setting	10				X
Date of Occurrence	6	Arrest			
Date of Occurrence	6	Dispatch			
Date of Occurrence	6	ASAP			
Date of Theft	4	Real Time			
Date of Test	6	ASAP			
Date of Want	6	Real Time			
Date of Warrant	6	Real Time			
Date of Probation Revolked					X
Date Reported	6	Accident			
Date Reported	6	Offense			
Date Returned	6				X
Date Sentenced	6				X
Day of Week	1	Accident			
Day of Week	1	Dispatch			
Day of Week	1	Offense			
Day of Week	1	Ticket			
Defendant in Custody	1				X
Delete Code	1	Real Time			
Description	16				X
Detention	2			X	
Directional Analysis	2	Accident			
Disposition Appealed	1				X
Disposition Charge	10			X	
Disposition Code	1	ASAP	X		
Disposition Code	2	ASAP	X		
Disposition Code	2	Ticket	X		
Disposition Court	1				X
Disposition Court Division	2				X
Disposition Date	6			X	
Disposition Date	6	Ticket	X		
Disposition Judge	2				X
Disposition of Call	1	Dispatch			
Disposition Type	2				X
Disposition Type				X	
District Number	4	Real Time			
Dock Refind	5				X
Driving Ability	1	ASAP			
Drivers Condition	1	Accident			
Driver Drinking	1	Accident			
Drivers Education	1	Accident			
Drivers License Number	16	Real Time			
Drivers License State	2	Real Time			
Drivers License Type	1	Accident			
Drivers License Year	1	Real Time			
East/West Street	10	Offense			
East/West Street	10	Accident			
East/West Street	10	Arrest			
East/West Street	10	Tickets			
East/West Street	10	Dispatch			
East/West Street Suffix	2	Accident			
East/West Street Suffix	2	Arrest			
East/West Street Suffix	2	Ticket			
East/West Street Suffix	2	Offense			
East/West Street Suffix	2	Ticket			
Effects	1	ASAP			
Ejection	1	Accident			
Entry Type	1	Real Time			
Expiration Date	6	Real Time			
Extension	1	Ticket			X

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
DEPARTMENT CODE CATALOG
DATE ISSUED
DATE REVISED
INDEX NUMBER
09-4.0

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Extra Point	1	Ticket	X		
Eyes	5	ASAP			
Eyes	2	Real Time			
FBI Number	9	Real Time			
Felony Arrest	3	Real Time			
Felony Conviction	3	Real Time			
Final Charge	4	Arrest	X		
Fine	7				X
Fine in Dollars	3	Ticket	X		
Fingerprint Classification	20	Arrest			
Fingerprint Classification	20	Real Time			
Final Disposition	1				X
First Car Switch	1	Dispatch			
First Case Report No. (Jacket No.)	8	Real Time			
First Defense, Attorneys First Name	7				X
First Defense, Attorneys Middle Initial	1				X
First Name	11	Ticket	X		
First Name	11	Real Time			
First Name	12				X
Fixed Type Code	1	Real Time			
Furs Recovered	3	Offense			
Furs Stolen	3	Offense			
Hair	2	Real Time			
Hearing Continuance	2			X	
Hearing Date	6			X	
Hearing Docket	1			X	
Hearing Time	4			X	
Height	3	Real Time			
Held for Prosecution	1	Arrest			
Hour of Occurrence	2	Arrest			
Hour Occurred	2	Offense			
Hour Reported	4	Offense			
How Attacked	3	Offense			
How Call Initiated	1	Dispatch			
How Cleared	1	Offense			
Illness	1	Arrest			
Injury Class	1	Accident			
Injury Victim	1	Offense			
Intersection Address Direct	2	Arrest			
Institution	2				X
Issuing Unit	2	Real Time			
Jail Credit	1				X
Jacket Number	8	Real Time			
Jewelry Recovered	3	Offense			
Jewelry Stolen	3	Offense			
Jr/Sr	1	Real Time			
Judge Assessing Disposition	2		X		
Juvenile Court	1	Arrest			

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

DEPARTMENT CODE CATALOG

DATE ISSUED | DATE REVISED

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Juvenile Parole or Probation	1	Arrest			
Juvenile Previous Offense	1	Arrest			
Last Name	17				X
Last Name	17	Real Time			
Last Name	17	Ticket	X		
Length of Parole Probation	3	Ticket	X		
Length of Probation Parole	3	Arrest	X		
Length of Sentence in Days	2	Arrest	X		
Length of Sentence in Months	3	Arrest	X		
License Attacked	1	Ticket			
License Number	8	Real Time			
License State	2	Ticket			
License State	2	Real Time			
License Type	2	Real Time			
License Type	2	Ticket			
License Year	1	Real Time			
License Year	1	Ti-ket			
Life Number	6			X	
Light Condition	1	Accident			
Local Intell.	1	Real Time			
Log Point	3	Accident			
Mask	3	ASAP			
Mental	1	Real Time			
Middle Initial	1	Real Time			
Middle Initial	1	Tickets	X		
Militant	1	Real Time			
Miscellaneous Stolen	3	Offense			
Misdem. Arrest	3	Real Time			
Misdem. Conviction	3	Real Time			
Molestation	1	Real Time			
Money Recovered	3	Offense			
Money Stolen	3	Offense			
Motive	2	Offense			
Name Type	1	Real Time			
Narcotics - Deals in	1	Real Time			
Narcotic - Known to Possess	1	Real Time			
Narcotics Involved	1	Arrest			
National Intell.	1	Real Time			
Nature	2	Offense			
Nature of Change	2	Arrest			
NCIC Number	10	Real Time			
NCIC Request	1	Real Time			
New Apartment Number	4	Real Time			
New Arrest Number	8	Arrest			
New Arrest Type	1	Arrest			
New Beat of Occurrence	4	Accident			
New Beat of Occurrence	4	Offense			
New Case Report Number	8	Accident			

INDEX NUMBER
09-4.0

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
New Case Report Number	8	Arrest			
New City Name	10	Real Time			
New Date Occurred	6	Offense			
New Date of Occurrence	6	Accident			
New First Name	11	Real Time			
New Hour Occurred	2	Offense			
New Last Name	17	Real Time			
New License Number	8	Real Time			
New License State	2	Real Time			
New License Year	1	Real Time			
New Middle Initial	1	Real Time			
New Nature Code	2	Offense			
New North/South Street	10	Offense			
New Offense Code	4	Offense			
New ORI	8	Accident			
New ORI	8	Arrest			
New Report Data	6	Accident			
New Report Data	6	Offense			
New State	2	Real Time			
New Street Address Number	5	Real Time			
New Street Direction	2	Real Time			
New Street Suffix Code	2	Real Time			
New Time of Occurrence	4	Accident			
New Trial Date	10			X	
New VIN	15	Real Time			
New Zip Code	5	Real Time			
North/South Street	10	Offense			
North/South Street	10	Accident			
North/South Street	10	Arrest			
North/South Street	10	Tickets			
North/South Street Direction	2	Accidents			
North/South Street Direction	2	Dispatch			
North/South Street Direction	2	Tickets			
North/South Street Name	10	Dispatch			
North/South Street Suffix	2	Accident			
North/South Street Suffix	2	Arrest			
North/South Street Suffix	2	Dispatch			
North/South Street Suffix	2	Tickets			
North/South Suffix	2	Offense			
Number of Reference	2			X	
Offense Code	4	Accident			
Offense Code	3	Arrest			
Offense Code	4	Real Time			
Officer Serial Number 1	4				X
Officer Serial Number 2	4				X
Officer Serial Number 3	4				X
Officer Serial Number 4	4				X
Officer Serial Number 5	4				X
Officer Serial Number 6	4				X
Officer Serial Number 7	4				X
Officer Serial Number 8	4				X
Officer Serial Number 9	4				X
Officer Serial Number 10	4				X
Officer Serial Number 11	4				X
Officer Serial Number 12	4				X
Officer Serial Number 13	4				X
Officer Serial Number 14	4				X
Officer Serial Number 15	4				X
Officer Serial Number 16	4				X
Officer Serial Number 17	4				X
Officer Serial Number 18	4				X
Officers Serial Number	4	Arrest			
Officers Serial Number	4	ASAP			
Officers Serial Number	4	Real Time			
Officers Serial Number	4	Ticket			
Officers Unit	2	Tickets			
Old Beat of Occurrence	4	Offense			
Old Date of Occurrence	6	Offense			
Old Date Reported	6	Offense			
Old Nature	4	Offense			
Old Offense Code	4	Offense			
Operators License Number	16	Real Time			
Operators License State	2	Real Time			
Operator License Year	1	Real Time			
Ordinance Arrest	3	Real Time			
Ordinance Code	6	Ticket			
Ordinance Conviction	3	Real Time			
Ordinance Number	6	Accident			
Original Court Date	10				X
Originating Case Agency Number	8	Real Time	X		
Parent Marital Status	5			X	
Pedestrian Action	2	Accident			
Pedestrian Condition	1	Accident			
Pedestrian Drinking	1	Accident			
Person Arrested	1	Accident			
Persons Location	1	Accident			
Petition Number	5			X	
Phone	7				X
Phone Extension	4			X	X
Picking Up Coins	5	ASAP			
Place of Birth	2	Real Time			
Plea	1				X
Pre-Sentence Investigation	1				X
Previous Void	4	Ticket	X		
Probation Days	3				X

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

DEPARTMENT CODE CATALOG

DATE ISSUED | DATE REVISED

INDEX NUMBER 09-4.0

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Probation Months	3				X
Probation/Parole	1	Arrest	X		
Probation/Parole	1	Ticket	X		
Probation Revoked	1				X
Probation Years	3				X
Property Attacked	3	Offense			
Prosecutors Office Number	7				X
Pupils	4	ASAP			
Race	1	Offense			
Race	1	Real Time			
Race	1	Tickets			
Race of Arrestee	1	Arrest			
Race of Offender	1	Offense			
Race of Person	1	Accident			
Reason in Detention	4			X	
Reason for Continuance	1			X	
Reason Released	1			X	
Record Indent.	1			X	
Recovered Add/Sub	1	Offense			
Recovered Autos	3	Offense			
Recovered Auto Theft	1	Offense			
Recovered Clothes	3	Offense			
Recovered Miscellaneous	3	Offense			
Reduced Charge	5			X	
Reduced Charge Code	1			X	
Referal Agency	1			X	
Referal Charge	10			X	
Referal Date	6			X	
Referal Number	7			X	
Referal Type	1			X	
Related Case Number One	7				X
Related Case Number Two	7				X
Related Case Number Three	7				X
Related Case Number Four	7				X
Related Office Number 1	7				X
Related Office Number 2	7				X
Related Office Number 3	7				X
Related Office Number 4	7				X
Released on Own Recognizance	1				X
Remark	40				X
Remark Type	1				X
Report Car	4	Dispatch			
Report Date	6	Ticket			
Reporting Beat	4	Ticket			
Reporting Beat	4	Arrest			
Reporting Beat	4	Offense			
Reporting Beat	4	Ticket			
Reporting Date	6	Accident			

DATA PROCESSING

MANUAL OF STANDARDS

SECTION

DEPARTMENT CODE CATALOG

DATE ISSUED | DATE REVISED

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Reporting Date	6	Arrest			
Reporting Date	6	Ticket			
Request Code	2	Real Time			
Requested By	1				X
Residency Code	1	Ticket			
Residency Code	1	Accident			
Results	2	ASAP			
Road Condition	1	Accident			
Road Surface	1	Accident			
Road Type	1	Accident			
Robbery	1	Real Time			
R O R	1				X
School Name	10			X	
Seat Belts	1	Accident			
Second Defense Attorney's First Name	7			X	
Second Defense Attorney's Last Name	12			X	
Second Attorney's Middle Initial	1				X
Second Officers Serial Number	4	Arrest			
Sector of City	1	Dispatch			
Sentence and/or Fine Code	1	Arrest	X		
Sentence and/or Fine Code	1	Ticket	X		
Sentence Days	3				X
Sentence in Days	2	Tickets	X		
Sentence in Months	2	Ticket	X		
Sentence Months	3				X
Sentence Suspended	2				X
Sentence Type	1				X
Sentence Years	3				X
Sequence No.	3				X
Sex	1	Offense			
Sex	1	Real Time			
Sex	1	Ticket			
Sex of Arrestee	1	Arrest			
Sex of Offender	1	Offense			
Sex of Person	1	Accident			
Shoplifting	1	Real Time'			
Social Security Number	9	Real Time			
Soiled by	5	ASAP			
Speech	7	ASAP			
State	2				X
State	2	Ticket	X		
State Code	2	Real Time			
State ID	10	Real Time			
State of Address	2	Real Time			

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
State Occured	2	Accident			
State of Occurrence	2	Arrest			
State of Occurrence	2	Offense			
State of Occurrence	2	Tickets			
State or County of Birth	2	Real Time			
Stolen add/sub	1	Offense			
Street Address	5	Tickets			
Street Address	5	Accident			
Street Address	5	Real Time			
Street Address Number	5	Offense			
Street Address Number	5	Real Time			
Street Address Number	5	Tickets	X		
Street Class	1	Accident			
Street Direction	2	Accident			
Street Direction	2	Dispatch			
Street Direction	2	Offense			
Street Direction	2	Real Time			
Street Direction	2	Tickets			
Street Direction Code	2	Tickets	X		
Street Name	10	Accident			
Street Name	10	Dispatch			
Street Name	10	Offense			
Street Name	10	Real Time			
Street Name	10	Tickets	X		
Street Suffix	2	Dispatch			
Street Suffix	2	Offense			
Street Suffix	2	Real Time			
Street Suffix	2	Tickets			
Street Suffix Code	2	Accident			
Street Suffix Code	2	Real Time			
Street Suffix Code	2	Tickets	X		
Supplemental	1	Offense			
Text	40	Real Time		X	
Ticket Number	6	ASAP			
Ticket Number	6 (7)	Ticket	X		
Ticket Type	2	Ticket			
Time Call Recieved	4	Accident			
Time Call Received	4	Dispatch			
Time Car Clears	4	Dispatch			
Time Car Sent	4	Dispatch			
Time First Car Arrives	4	Dispatch			
Time Occured	4	Ticket			
Time Occured	4	Accident			
Time of Test	4	Arrest			
Time Type	1	Accident			
To Delete a Record	6			X	

DATA PROCESSING
MANUAL OF STANDARDS
SECTION
DEPARTMENT CODE CATALOG
DATE ISSUED
DATE REVISED

Data Element	Data Size	Law Enforcement	Mun. Crt.	Juv. Crt.	Pros.
Traffic Arrest	3	Real Time			
Traffic Control	1	Accident			
Traffic Conviction	3	Real Time			
Traffic School Days	1	Ticket	X		
Traffic Ticket Type	2	Tickets			
Trial Division	2				X
Trial Prosecutor	2				X
Turning	7	ASAP			
Type Area	1	Accident			
Unit	2	ASAP			
Unit Clearing	2	Offense			
Unit Code	1	Accident			
Value of Autos Recovered	5	Offense			
Value of Autos Stolen	5	Offense			
Value of Clothes Recovered	5	Offense			
Value of Clothes Stolen	5	Offense			
Value of Furs Recovered	5	Offense			
Value of Furs Stolen	5	Offense			
Value of Jewelry Recovered	5	Offense			
Value of Jewelry Stolen	5	Offense			
Value of Misc. Recovered	5	Offense			
Value of Misc. Stolen	5	Offense			
Value of Money Recovered	5	Offense			
Value of Money Stolen	5	Offense			
Variable Type	1	Real Time			
Vehicle Action	2	Accident			
Vehicle Color	6	Tickets			
Vehicle Color	6	Real Time			
Vehicle Damage	2	Accident			
Vehicle ID no.	15	Ticket			
Vehicle ID Number	15	Real Time			
Vehicle Make	4	Real Time			
Vehicle Make	4	Ticket			
Vehicle Model	3	Real Time			
Vehicle Model	3	Tickets			
Vehicle Style	2	Real Time			
Vehicle Style	2	Ticket			
Vehicle Type	1	Accident			
Vehicle Year	2	Real Time			
Vehicle Year	2	Ticket			
Vehicle Year of Manufacture	2	Accident			
Video Tape	2	Asap			
Vision Obscured	2	Accident			
Void Conviction Type	1	Arrest	X		
Void Pervious Disposition	7	Arrest	X		
Walking	7	Asap			
Warrant Status	1	Real Time			

INDEX NUMBER
09-4.0

END