# MODIFICATION OF A COMPUTER WAR GAME FOR COMBINATION COMPUTER/MANUAL PLAY

ZEANIOUS L. NEWCOMB

MODIFICATION OF A COMPUTER WAR GAME

FOR COMBINATION COMPUTER/MANUAL PLAY

* * * * *

Zeanious L. Newcomb

# MODIFICATION OF A COMPUTER WAR GAME
# FOR COMBINATION COMPUTER/MANUAL PLAY

by

Zeanious L. Newcomb

Lieutenant, United States Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE
IN
OPERATIONS RESEARCH

United States Naval Postgraduate School
Monterey, California

1 9 6 4

MODIFICATION OF A COMPUTER WAR GAME

FOR COMBINATION COMPUTER/MANUAL PLAY

by

Zeanious L. Newcomb

This work is accepted as fulfilling

the thesis requirements for the degree of

MASTER OF SCIENCE

IN

OPERATIONS RESEARCH

from the

United States Naval Postgraduate School

# ABSTRACT

This document points out the rigidity of certain processes commonly used in computerized war games to perform decision making functions. Such procedures are believed to have detrimental effects upon the value of these simulations for many analytical applications. As a partial solution to this problem, a combination computer/manual game is proposed which provides for a human decisionmaker and adds much in the way of flexibility to the game. The ease with which a large computerized war game can be modified for combination computer/manual play is demonstrated by describing such a modification to an existing model and displaying flow charts of the logic for the modification.

In addition to improving the analytical capabilities of the simulation, a combination computer/manual game provides educational and training capabilities not present in the completely computerized game.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

# I

# INTRODUCTION

## 1.1 Background

The simulation of military orders of battle and the interactions of the opposing forces on large digital computers, or war gaming as it is called, has become one of the military operations analyst's most important tools. The capacity of modern computers for handling large quantities of data, the speed with which computations and operations are performed, and the ability to repeat a simulation over and over with the same parameters or with one or more parameters changed have contributed to the usefulness of this analytical tool in attacking military problems. These characteristics are especially important to analysis of present day military problems where massive numbers of forces may be introduced into a conflict situation and the action accelerated at a very rapid rate. The pace of research and development is such as to make today's wonder tomorrow's antique; thus, the need for considering improvements in equipment capabilities, etc., which frequently are represented by parameters in a computerized war game, is great. The capability of changing these parameters and replaying the game is a definite asset.

The many benefits derived from computerizing the war game have not been acquired free or without accepting some handicap. The decision making processes suffer from a lack of flexibility because they of necessity involve selection according to pre-selected rules. The rules are, of course, established by the builder who has attempted to anticipate the decisions which could confront the decisionmaker being simulated and has also decided what criteria the decisionmaker should

use to resolve the issue. There are many persons who believe that modern methods of waging war demand this sort of predetermined decision - one for every conceivable situation that might arise. For those of us who believe that there are elements of every conflict situation which are never recognized or perceived until the situation is being played out its one and only time on the world stage and that the ability of the human mind to crank these elements into the decision making process is the thing that makes humans indispensable, these predetermined decisions are a source of concern.

## 1.2 Purpose

As a partial solution to this dilemma, this thesis proposes a combination computer/manual game. The concept to be set forth here is not that involved in the NEWS at the Naval War College in Newport, Rhode Island,which really involves a computer assisted manual type game but rather would involve a game that is essentially computerized but affords a human the opportunity to participate in the decision making. If one envisions the manual war game and the computer war game as graphically located at either ends of a line segment which represents the differences between them, the NEWS would be located on the end nearest the manual game while the concept to be demonstrated here would be located near the computer.

The idea is to have the computer programmed so that when major decisions are to be made it halts its simulation operations and displays on some external equipment - such as a printer, a cathode ray tube, a fluorescent screen, some photographic device, or one of the other types of display devices which are under development and

rapidly becoming available - the information available to the decisionmaker of the side involved. A human player simulating the decisionmaker in a more realistic fashion would view this data, integrate it with information given him before the start of the game and at other decision points by the computer and indicate his decision to the computer. This might be done through any of the usual facilities for inputting data to the computer, or, if some of the more refined display equipment were used, it could be done through the keyboard capabilities of the equipment.

A good example of equipment which is particularly well suited to this technique is that being used in the war game research project at the U.S. Naval Postgraduate School. Selected portions of a program running on the large CDC 1604 may, at certain pre-programmed spots, be read out of that machine and into the memory of the smaller CDC 160. An executive program for that computer permits an operator at the control panel of a dd65 Data Display Unit to selectively view segments of the program and to make changes in the segment being viewed through the keyboard of the dd65. This technique has direct applications to the player participation in war games proposed herein.

The introduction of a human decisionmaker does not, of course, solve the problem of not being able to anticipate all elements of a conflict situation before it occurs. It merely provides an intellect that is conditioned to react to the unexpected and has a capacity for original thought.

This leads to another use to which this type war game may be put - that of providing experience to prospective decisionmakers by having them occupy the player position and make the decisions required as the

game is played. This is conceptually similar to the NEWS, but as hopefully will be shown in this thesis is much easier to acquire, less elaborate, and less expensive; thus, might be within the capabilities of more activities.

A third possible use for such a player particpation game might be for the teaching of war game techniques and concepts. The player in such a game would definitely gain a great deal of insight into the structure of the model and an understanding of many war gaming procedures.

1.3 War Game to be Modified

A one sided computer war game built by the Applied Physics Laboratory for the Office of the Chief of Naval Operations will be utilized as a vehicle for demonstrating the combination computer/manual game for two reasons:

(1) It will keep the task within the time limitations on this thesis.

(2) It is hoped that demonstrating how little effort is involved in modifying a computer game to permit manual participation will encourage others to experiment with the idea.

It should be pointed out that the method is not restricted to one sided games but on the contrary lends itself quite readily to player competition if duplicate display facilities are available.

The game chosen for a vehicle is the "Naval Air Strike Model - Mod Zero" which is documented in the reports of the Planning Analysis Group of the Applied Physics Laboratory listed in the bibliography. This game is one-sided in the sense that all decisions of the offense

are made before the game is played and are programmed into it. Only the defense has alternatives which require decisions as the game is being played; consequently, this thesis will be concerned with modifying the logic of the defensive side of the game to permit decision making by a player at some type display unit.

A general description of "The Naval Air Strike Model - Mod Zero" is given in Section II of this thesis. Section III discusses the modifications necessary to convert the game, and Section IV offers a comparison of the two type games and points up the advantages which the author feels are present in the modified game. Section V discusses different type display concepts, and Section VI lists the author's conclusions and recommendations.

## II

## BRIEF DESCRIPTION OF THE NAVAL AIR STRIKE MODEL - MOD ZERO[1]

### 2.1 Conflict Situation

"The Naval Air Strike Model - Mod Zero" simulates a strike by manned bombers, cruise missiles, or ballistic missiles against a region of "enemy" territory which may be defended by anti-aircraft guns, manned interceptors, and/or surface-to-air missiles. The flight plans of the attacking forces are pre-planned and introduced into the game as input parameters and variables; thus, the actions simulated in the game involve the attacking forces attempting to execute their prescribed flight plans and to deliver their payloads against the efforts of the defensive forces to destroy the attackers before weapon delivery.

### 2.2 Game Geography

All action simulated within the computer takes place within the Early Warning Radar circle of the defense. This circle may be as large as the game user desires so long as its center and some segment of its circumference is contained within a 2000 by 2000 mile square, the boundaries of which are described in the game input. Game action is then restricted to the area intersection of the EWR circle and this square. This area is referred to as enemy territory in the game literature. Objects of interest are located in the game by giving them coordinates in the reference plane formed by using this square as the

[1] No discussion of this model's intended use or special applications will be included here. The reader is referred to the documentation listed in the bibliography for such information.

first quadrant of an XY-coordinate system.

The size of the playing area is limited by the choice of 1/16 mile as the basic unit of distance and by the computer word size of the IBM 7090 for which the Mod Zero was programmed. This means that two objects separated by less than 1/16 mile are considered to be located at the same spot by the computer. It is possible to expand the playing area by letting the 1/16 mile unit represent a larger distance, say 10 miles; however, in this case, objects 10 miles apart are considered to be at the same location. This is usually not required as the playing area is large enough to permit the consideration of large integrated air defense complexes.

## 2.3 Game Events

The Naval Air Strike Model is an event store[1] computer simulation. A list of the game events will be a useful reference in the discussion that follows and some insight into this type game may be achieved by considering the list which is given below. In this list the name represents the action to be simulated, the designation is an abbreviation for the name, and the type is the numerical representation of this action in the computer program.

| TYPE | DESIGNATION | NAME |
|---|---|---|
| 1 | EET | Raid Enters Enemy Territory |
| 1A | RDE | Raid Detected by Early Warning Radar |
| 2 | IAS | Interceptor Assignment |
| 4 | ITO | Interceptor Take-off |
| 5 | EGZ | Raid Enters GCI Zone |

[1] See Appendix I for a discussion of event store logic.

| | | |
|---|---|---|
| 5A | RDG | Raid Detected by GCI Radar |
| 5B | CTA | Crossing Target Check at GCI Zone |
| 6 | WSI | White Scope at Intercept |
| 7 | SSI | Strobed Scope at Intercept |
| 8 | BSI | Blobbed Scope at Intercept |
| 9 | CCI | Close Control at Intercept |
| 10 | MAN | Raid Maneuver |
| 11 | ILA | Interceptor Landing |
| 12 | IOR | Interceptor Loiters on Station |
| 13 | EMA | Raid Enters Missile Area |
| 13A | RDM | Raid Detected by SAM area |
| 13B | TAL | Target List Entry |
| 13C | ICA | Interceptors Cancelled from Raid |
| 14 | PML | Possible Missile Launch |
| 14A | MII | Missile Intercept |
| 16 | WRE | Weapon Release |
| 17 | WDD | Weapon Detonation |
| 18 | RNR | Reset Number of Radars Free |
| 19 | RLL | Reset Launcher Loaded Bit |
| 20 | DBD | Damaged Bomber Dies |
| 21 | LGZ | Raid Leaves a GCI Zone |
| 22 | LMA | Raid Leaves a Missile Area |
| 23 | LET | Raid Leaves Enemy Territory |
| 25 | IHH | Interceptor Heads for Home |
| 26 | PPC | Periodic Position Check |

NOTE: There are no events numbered 3, 15, and 24.

The above events may be grouped into three categories which are

helpful toward understanding the structure of the game:

(1) Actions initiated by the offense - Event Types 1, 5, 10, 13, 16, 21, 22, 23.

(2) Actions initiated by the defense - Event Types 1A, 2, 4, 5A, 5B, 6, 7, 8, 9, 11, 12, 13A, 13B, 13C, 14, 14A, 18, 19, 25.

(3) Bookkeeping functions - Event Types 17, 20, 26.

The primary functions and basic logic of each event will be described in detail in Section III in order to explain how the proposed modification will affect the event or to show why it does not need to be modified.

2.4 Offensive Forces

The offensive forces may consist of manned bombers, cruise missiles, or ballistic missiles. Ballistic missiles can only be used when the defense has no counter weapon against them. This is an assumption in the model. The flight paths and targets of offensive forces are pre-programmed and are executed in a straight forward manner during the play of the game. An ECM environment may be generated by the offense which has several degrees of effectiveness against the defense's radars. These are: white scope, multiple strobes, single strobes, and false blips.

2.5 Defensive Forces

The defensive forces consist of anti-aircraft gunnery, manned interceptors, and surface-to-air missiles. Each is discussed separately in the paragraphs below with a few comments about how they are handled in the model.

Anti-Aircraft Gunnery - Model doctrine assumes that anti-aircraft

gunnery units are generally mobile and that their positions would be impossible to predict; consequently, it is assumed that their coverage is uniformly distributed over enemy territory. As each bomber enters enemy territory, computations to determine if it will be shot down by anti-aircraft fire are made using Monte Carlo techniques. If a kill is predicted, the position of fall is computed and a Type 20 Event (Damaged Bomber Dies) is schedule for the appropriate time. This capability may be eliminated from the game by inputting a zero probability of kill by anti-aircraft fire. Specific anti-aircraft sites may be simulated also by inputting these as surface-to-air missile sites.

Manned Interceptors - The defensive detection system consists of the Early Warning (EW) and Ground Controlled Intercept (GCI) radars. It is assumed that airfields are located at each GCI site. Interceptors from these airfields are assigned to raids upon detection provided that interceptors are available and needed. The model allows for four types of intercept, represented by Events 6, 7, 8, and 9. The events differ mainly in the size of the interceptor's probability of kill. The type of intercept simulated depends upon whether detection has been made by EW radar only or by GCI radar also, and upon the ECM level currently existing at the GCI radar site.

Surface-to-Air Missiles - Missile areas are formed by radar maximum range circles around missile sites. When a raid enters one of these areas, the time and position of its detection by the site are computed according to probability distributions based on the characteristics of the equipment attributed to the defensive forces. The time interval after detection required for the missile directors

to acquire the target and the probability of a hit are dependent upon the ECM level at the missile sites.

## 2.6 Significant Details and Limitations of the Model

There are certain significant details and limitations associated with any simulation model or computer program which are imposed by various sources - the size of the computer, the logic, the doctrine, self-imposed restrictions by the builder to reduce the size of the effort required, etc. These details and limitations can be of particular interest to a potential user trying to evaluate the suitability of the game for his needs. Some such details for "The Naval Air Strike Model - Mod Zero" have been mentioned already. In addition to those, the following are considered significant and worthy of note:

(a) The unit of time in the game is four seconds; thus, events occurring less than four seconds apart occur at the same time in the game.

(b) The maximum amount of game time that can be simulated is 18 hours, 12 minutes and 12 seconds. A running time shorter than this can be an input to the game. The game will stop automatically when there are no more raids in enemy territory.

(c) All arithmetic computations for the movement of aircraft, missiles, etc., are in two dimensions in the XY-plane. Altitude is considered only as a check to see if intercepts and detections are permissible.

(d) The offense may have up to fifteen different type bombers. A raid (defined as a group of aircraft of the same type always occupying the same position) may have up to fifteen

bombers but the weapons carried by all of them must have the same yield. The maximum number of raids allowed in the game is 127 and the maximum number of bombers is 480. There may be as many different weapon yields are there are raids.

(e) The defense is allowed a maximum of 31 GCI sites coincident with 31 airfields.

(f) Fifteen different types of interceptors may be simulated but all interceptors at a particular airfield must be of the same type.

(g) Up to 31 SAM sites may be simulated and each site may track as many as fifteen targets at one time. Each site may have a maximum of 255 salvos.

(h) Communications between defensive units are implicit. Actually, what is really implicit is a central command and control center with a single area commander who has knowledge of all contacts and directs all defensive activities. This commander has perfect communications with all GCI sites, SAM sites, and airfields. As a result, no two GCI units ever unintentionally direct an attack against the same target; the proper number of interceptors are always assigned if they are available; there are no garbled communications or lost messages; etc.

(i) A Fortran pre-processing program has been written for "The Naval Air Strike Model - Mod Zero" which transfers geographical coordinates into game coordinates, computes time of entry into enemy territory, time of weapon release, etc., so that these are all in proper format for input to the game.

(j) "The Naval Air Strike Model - Mod Zero" uses up approximately 28,000 computer storage locations when used with its present capacity and force levels. This leaves only about 4000 locations for monitoring devices and the re-programming for the modifications which will be proposed. This could become restrictive and require some reduction in the sizes of the forces used.

## 2.7 Illustration of Game Procedures

The reader's attention is directed to Figure 1 which shows a sector of a hypothetical defense complex - a segment of an early warning radar line, one GCI area and its airfield, one missile site, and a target. Also shown is the flight path of an equally hypothetical raid with points labelled $E_i$ (i=1,2,.....,16) to indicate the position of the raid at the occurrence of some of the events that might be executed by "The Naval Air Strike Model - Mod Zero" if this flight were simulated. A discussion of the manner in which these events (and a few others which are not indicated in the figure) would be stored and executed should given the reader a better understanding of the model.

First, consider how the game is initiated. Events $E_1$, $E_{11}$, $E_{12}$, and $E_{14}$ would be pre-stored in an initial event store table. This table for the very simple game being considered here is shown in Figure 2. $E_1$ represents an EET Event (Raid Enters Enemy Territory) and begins the game since this is the first raid to enter. If there were other raids, their time of entering enemy territory would be part of the game input. One function performed by the EET Event is to search this input to determine which is the next raid to enter and to

store an EET Event for that raid. A second function of the EET Event is to compute, by Monte Carlo methods, how far the raid will penetrate into enemy territory before it is detected and to set up a RDE Event for the raid at the time of detection. This event occurs at point $E_2$ in the figure. Another function of the EET Event for the first raid to enter enemy territory is to store the first PPC Event. This event projects all raids in the game along their respective flight paths for a specified interval of time; stores any EGZ, LGZ, EMA, LMA, or LET Events that might occur when the raid flies the projected track; and restores itself to recur at the end of that interval.

The RDE Event marks the detection of the raid by EW radar and alerts the defensive forces by scheduling an IAS Event to occur after a short interval representing reaction time for the defense. Point $E_3$ indicates the raid's position when the IAS Event takes place. The assignment of interceptors involves the scheduling of take-offs which is done by storing ITO Events for each interceptor. The position of the raid at the execution of these take-offs is not shown; however, $E_6$ represents the position of the first intercept which will be assumed to be a CCI Event.

$E_4$ is the position of the raid at the execution of an EGZ Event for the raid. The EGZ Event was stored by the last regular PPC Event to occur. It will determine how far the raid penetrates into the GCI area before detection and store an RDG Event for the time of detection - represented by point $E_5$.

Any bombers killed by the interceptors may result in WDD Events being stored if a Monte Carlo computation shows that the pilot set a "dead-man" switch before the kill and that the switch was operable.

These events are omitted from the figure for simplicity.

Assuming that some live bombers are left in the raid, a LGZ Event will be stored to occur at $E_7$ by the PPC Event which projects the raid's track out of the GCI area. A later PPC Event will project the track into the missile area $M_1$ and store an EMA Event for point $E_8$. The EMA Event will determine the distance of penetration into the missile area by the raid before detection and store a RDM Event to occur at the time the raid is at point $E_9$. The RDM Event stores a TAL, ICA, and/or PML Event as needed. None of the last three events are represented in the figure; however, any PML events which are executed may result in a MII, i e., a missile intercept, one of which is represented by point $E_{10}$. Several of these events may occur while the raid is in the missile area, but the one will suffice to illustrate the point.

As mentioned earlier $E_{11}$ and $E_{12}$ represent pre-stored events. $E_{11}$ is a maneuver event (MAN) that might cause a change of altitude as well as heading. $E_{12}$ is a WRE Event. It stores a WDD Event for the weapons in addition to indicating their release. The position of the raid at the execution of the WDD is omitted from the figure.

Assuming at least one bomber remains alive, $E_{13}$ is the position for a LMA Event that is stored by the PCC Event which projects the raid out of the missile area. $E_{14}$ is a pre-stored MAN Event and again causes a change in both altitude and heading. $E_{15}$ is the position for a DBD Event which might have been placed in store as the result of either an interceptor or a missile damaging a bomber.

The LET Event at $E_{16}$ would be stored by the last PPC Event and would end this very simple game. In a larger game, all raids would have been handled simultaneously and similar events stored and executed

for each. Several Event Types for interceptors were not mentioned but will be treated in Section III. The purpose here was not to explain each event but to indicate the procedures by which the play is carried forth. Figure 3 lists the sixteen labelled points of Figure 1 together with the designation for the type event occurring there.

FIGURE 1

Illustration of Game Procedures

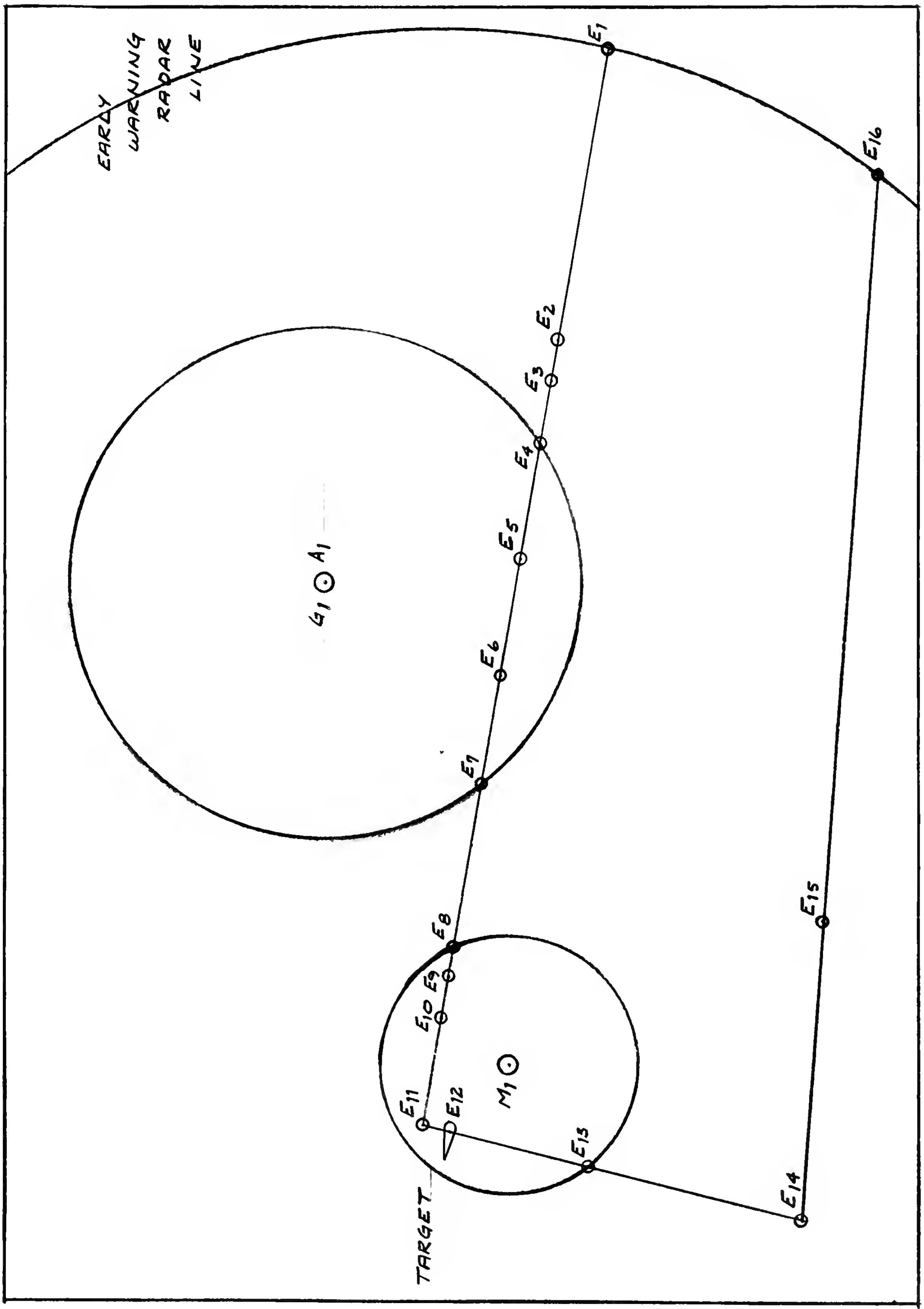


FIGURE 2

Initial Event Store Table[1]

| Game Time (Min.) | Event Type | Event Name |
|---|---|---|
| 00.0 | 1 | Raid Enters Enemy Territory |
| 42.6 | 10 | Raid Maneuver (Alter Heading and Altitude) |
| 43.4 | 16 | Weapon Release |
| 57.8 | 10 | Raid Maneuver (Alter Heading and Altitude) |

[1] This table indicates the information stored in the event store table within the computer but not the form in which it is stored.

FIGURE 3

Event Designations Corresponding to Points in Figure 1

| Point | Event Designation |
|---|---|
| $E_1$ | EET |
| $E_2$ | RDE |
| $E_3$ | IAS |
| $E_4$ | EGZ |
| $E_5$ | RDG |
| $E_6$ | CCI |
| $E_7$ | LGZ |
| $E_8$ | EMA |
| $E_9$ | RDM |
| $E_{10}$ | MII |
| $E_{11}$ | MAN |
| $E_{12}$ | WRE |
| $E_{13}$ | LMA |
| $E_{14}$ | MAN |
| $E_{15}$ | DBD |
| $E_{16}$ | LET |

# III

# MODIFICATION OF THE NAVAL AIR STRIKE MODEL - MOD ZERO

The purpose here is to indicate the modifications that need to be made in the logic of "The Naval Air Strike Model - Mod Zero" to provide the capability for human decision making during the play of the game; however, it should be emphasized how little modification is required. To accomplish these ends, the following procedure will be used. This section will be divided into three sub-sections. The first will discuss those events for which no modification is involved. A brief description of what each event does will be given which will illustrate why the event does not need any changes.

The second sub-section will deal with the events to be modified and will discuss the modification. Detailed flow charts of the modifications are given in Appendix II and the flow charts of the original events (marked to indicate where changes occur) are reproduced in Appendix III.

The third sub-section describes a new subroutine and a new event which must be introduced into the model. The new subroutine will form the connection between the computer and whatever display equipment is utilized. It will be called at any point in the program at which information or an opportunity to make a decision is to be given to the player. The new event will be a bookkeeping device for the convenience of the player. It will periodically check the raids which have been detected and inform him of the number of live targets remaining and the number of interceptors committed to that raid which are still alive.

## 3.1 Discussion of Events Which Are Not to be Modified

Event Type 1 (Raid Entering Enemy Territory) - The time at which each raid is to cross the early warning radar line, i.e., enters enemy territory, must be pre-computed and entered into the game as input. A Type 1 Event must be stored in the initial event store table (as described in sub-section 2.7) for the first raid to enter. The functions of this event are: indicate that this raid is now in enemy territory, determine in accordance with inputted probability distributions how far this raid will penetrate into enemy territory before it is detected, store a Type 1A Event for this raid for the time of detection, and determine when the next raid is to enter and store a Type 1 Event for the next raid.

Event Type 1A (Raid Detected by Early Warning Radar) - This event indicates the raid has now been detected by EW radar, computes the delay that must occur before the defense can begin assigning interceptors to the raid, and schedules a Type 2 Event to occur at that time. The delay is a function of the reaction time attributed to the defense.

Event Type 5 (Raid Enters GCI Zone) - Monte Carlo methods are used here to determine how far a raid entering a GCI area will penetrate before detection by the GCI site occurs. Consideration is given to altitude, radar cross-section, and whether the site has been alerted by a previous detection by early warning radar. A Type 5A Event is stored for the time of detection and it is in the execution of that event that the player will receive information appropriate to this detection.

Event Type 10 (Raid Maneuver) - This event alters the raid's velocity vectors and/or altitude to conform to pre-stored data. Any Type 10 Events required are pre-stored for all raids in the initial

event store table. A check is made to see if the change in heading or altitude will cause a change in the detection status of the raid, if so, the change is noted. If interceptors have been assigned, checks are made to cancel intercepts no longer possible, adjust times of take-offs, and set up new take-off times as necessary. One might at first think that the player should be immediately informed of the new raid parameters; however, further consideration indicates that this is unrealistic. It seems more in line with true radar information if he is informed later by the new raid check event discussed in sub-section 3.3 or by the occurrence of one of the other events, discussed in sub-section 3.2, for this raid.

Event Type 12 (Interceptor Loiters on Station) - This event occurs as a result of an intercept loitering on station waiting for further instructions. The action taken by the execution of this event is to assign the aircraft to an intercept if possible, and if not, determine if the aircraft should continue to loiter or should land. If the aircraft is to land and the home airfield is dead, a new airfield is located and he is vectored there immediately, i.e., a Type 11 Event is stored.

An explanation of the circumstances under which an interceptor is required to loiter is in order here. Interceptors may be committed to either long or short range intercepts on the basis of EW radar contact and the raid's projected track; however, GCI radar contact is necessary for execution of the short range commitment.[1] The interceptor may actually take off before the GCI contact is established and

[1] See the discussion of a Type 2 Event in sub-section 3.2 for additional information of the two types of commitments.

this results in his loitering on station awaiting GCI detection and a Type 12 Event being stored.

Game doctrine does not permit an interceptor to be controlled by another airfield's GCI radar. It would be desirable to allow the player in the computer/manual game to consider assigning the interceptor to another raid; however, this runs counter to basic game doctrine which does not permit airborne interceptors to be re-assigned to other raids. To change this would go beyond the modification intended here.

Event Type 13 (Raid Enters Missile Area) - This event occurs when a raid reaches the maximum radar range of a missile site. The time and position of detection are determined similar to a Type 5 Event and a Type 13A Event is stored for the raid at the time of detection.

Event Type 13B (Target List Entry) - This event occurs as a result of the Type 13A Event. The target is placed (according to prescribed criteria) in the proper position on a priority list for target selection by the missile site involved. Changes will be made in the Type 13A Event to permit the player to withhold the target from the missile site's target list if he so desires.

Event Type 14 (Possible Missile Launch) - This event selects targets from the target list of a missile site and, if a launcher and a radar are available, indicates that a missile salvo has been launched against this target. Target selection for a missile battery is commonly performed by pre-determined criteria and frequently by computers; consequently, this function is left to the computer here.

Event Type 16 (Weapon Release) - Weapon release events for all raids are pre-stored in the initial event store table. This event

determines the number of weapons released, and stores a Type 17 Event for the computed time of detonation.

Event Types 18 and 19 (Reset Number of Radars Free and Reset Launcher Loaded Bit) - These events perform the functions indicated by their names. The first indicates the guidance radar is available for a new target; the latter indicates that a previously used launcher is loaded and ready for use.

Event Type 21 (Raid Leaves a GCI Zone) - The functions of this event are primarily of a bookkeeping nature. The actions taken are to: indicate that the raid has left the GCI area and is no longer detected by its radar, remove the GCI radar from the effect of the raid's ECM efforts, look for another radar towards which these efforts might be directed, and delete take-off events for aircraft assigned to the raid from this site's airfield but not yet airborne.

Event Type 22 (Raid Leaves Missile Area) - This event performs the same functions for a missile site as Type 21 above does for a GCI site. The comments made there are pertinent here. It should be noted that this event stores a Type 2 Event for the raid; therefore, it is not necessary to modify this event to provide the player with an opportunity to assign interceptors.

Event Type 23 (Raid Leaves Enemy Territory) - This event indicates that the raid is no longer in enemy territory. All events in store for this raid are cancelled since leaving enemy territory is synonomous with leaving the game.

Event Type 25 (Interceptor Heads for Home) - The failure of an interceptor to detect the raid due to poor ECM conditions or the cancellation of the interceptor by a Type 13C Event will result in

this event being stored for the interceptor. This event removes the commitment of the interceptor, vectors him home if his airfield is alive, finds him a new one if it is not, and stores a Type 11 Event for the time of landing. The comments concerning an opportunity for the player to re-assign the interceptor made in the discussions on Event Type 12 apply here also.

Event Type 26 (Periodic Position Check) - The purpose of this event is to store events pertaining to all raids entering and leaving enemy defensive areas. It utilizes a technique which is very beneficial in an event store type game. To keep the event store table from getting long and burdensome, raids are only projected along their tracks for a short time interval (the length of the interval is an input parameter) and the events which will occur solely as a result of the raid travelling that projected track are placed in store. There include the events designated by EGZ, LGZ, EMA, LMA, and LET. All other events pertaining to defense actions occur as a result of these events.

The Periodic Position Check Event is self storing and comes up at the end of the time interval and projects all raids forward for one more interval. The advantage of this procedure is that if all of the events that occur over the raid's entire flight path were stored in the initial event store table for all raids, one can easily imagine that the table would become unmanageable in a large game.

3.2 Events Requiring Modification

In the general flow charts for the modifications of the events to follow, the following symbols will be used:

 - indicates action taken by the computer

 - indicates decisions by the player

 - indicates computer check to determine player's decision

Event Type 2 (Interceptor Assignment) - The action taken by this event is to locate available interceptors and assign them to the raid. The modification to this event constitutes the major logical difference between the completely computerized model and the combination computer/manual game proposed in this thesis. When it is ascertained that there are both live bombers in the raid and live airfields, a commitment ratio is determined. The commitment ratio may have one of two values, one for initial commitment and the other for recommitment. Both values are input and may be the same if the game user desires. The product of the ratio used and the number of live bombers in the raid gives the number of interceptors required for the raid. The computer then attempts to assign interceptors to bring the total number of live interceptors committed to the raid up to the required level.

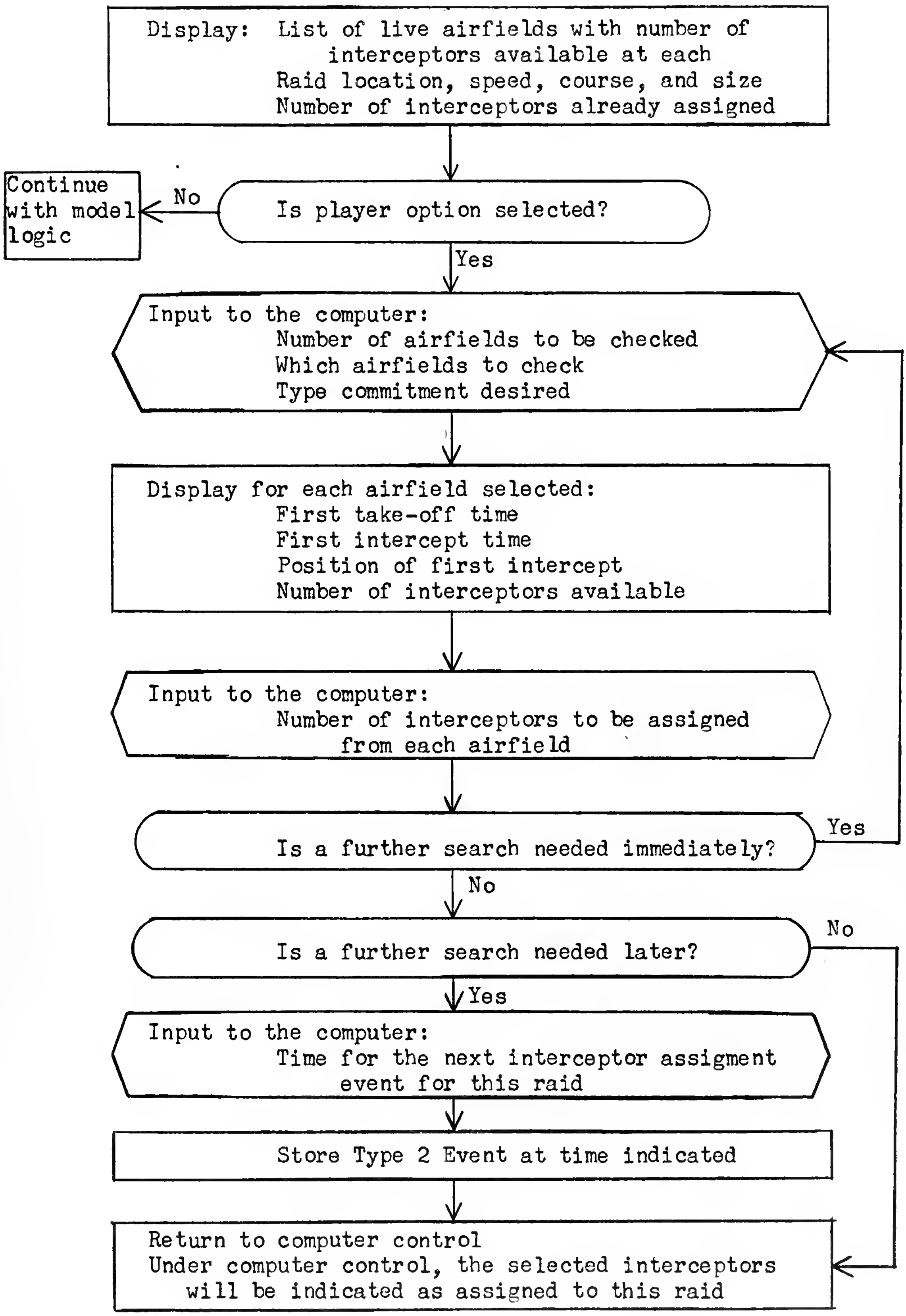
In making interceptor assignments the model considers two types of commitments - short range and long range. The short range commitment is for a raid passing through a GCI area at an altitude to place it above the horizon and permit detection. The long range commitment is for the raid that is either outside of the GCI area or too low to be above the horizon. Short range intercepts are preferred in the model and the computer checks this possibility first, assuming the raid continues on the same course and altitude. As many interceptors as are available are assigned up to the required number. When not enough

interceptors are available for short range intercept, an attempt is made to cover the deficiency with long range commitments. When several airfields exist that can provide the same type of intercept, interceptors are selected first from the airfield closest to the raid's current position. A Type 4 Event (Interceptor Take-off) is stored for every interceptor assigned for the time of take-off which has been computed to provide the earliest possible intercept but not have the interceptor loiter. Should the required number of interceptors not be located after considering both short and long range possibilities, another Type 2 Event is stored for the raid to occur after a prescribed interval of time when another search will be made. This time interval is an input value.

The modification proposed for this event is that when an interceptor assignment is to be made the player should be given the option to determine the number of interceptors needed and from what airfields they will be assigned to the raid. To accomplish this, the computer should be programmed to display to the player the number of available aircraft at each airfield, the current position of the raid, its course and ground speed, its size, and the number of live interceptors already assigned, if any. The player could then allow the computer to assign interceptors as described in the preceding paragraph or make his own assignments. To do the latter he should be allowed to have the computer's assistance. Appendix II provides a suggested list of the information to be displayed and the information needed by the computer to continue. When selecting his option the player will indicate the type commitment desired and the numbers of airfields to be checked.

FIGURE 4

General Logic for Modification of Type 2 Event



Upon receiving this information, the computer will make the necessary computations and provide a second display showing which of the airfields selected can make the type commitment desired. Included in this display should be the time of the first take-off, the associated time and position of the first intercept, and the number of interceptors available.

With this information and a knowledge of the current situation gained from a briefing prior to starting the game and from his participation to this point, the player can assign interceptors by whatever criterion he deems appropriate. This can be done by inputting to the computer at this time the number of interceptors to be assigned from each of the desired airfields. The player should also have the capability of asking the computer to check other airfields or to check for a different type commitment in order to provide the proper number of interceptors. He might also like to store another Type 2 Event to occur at some later time if he is playing a "waiting game" or has knowledge that aircraft will be available later from a more desirable airfield, etc.

The general logic[1] for the modified event is shown in Figure 4.

Event Type 4 (Interceptor Take-off) - This event occurs at the scheduled time of take-off by an interceptor. In the execution of this event it must be determined that the airfield is alive, that there are live bombers remaining in the raid to which the interceptor is committed, and that the raid is still in the airfield's area. Given

[1] The original flow charts for this and other modified events are reproduced in Appendix III. Detailed flow charts for the modification and suggested displays are given in Appendix II.

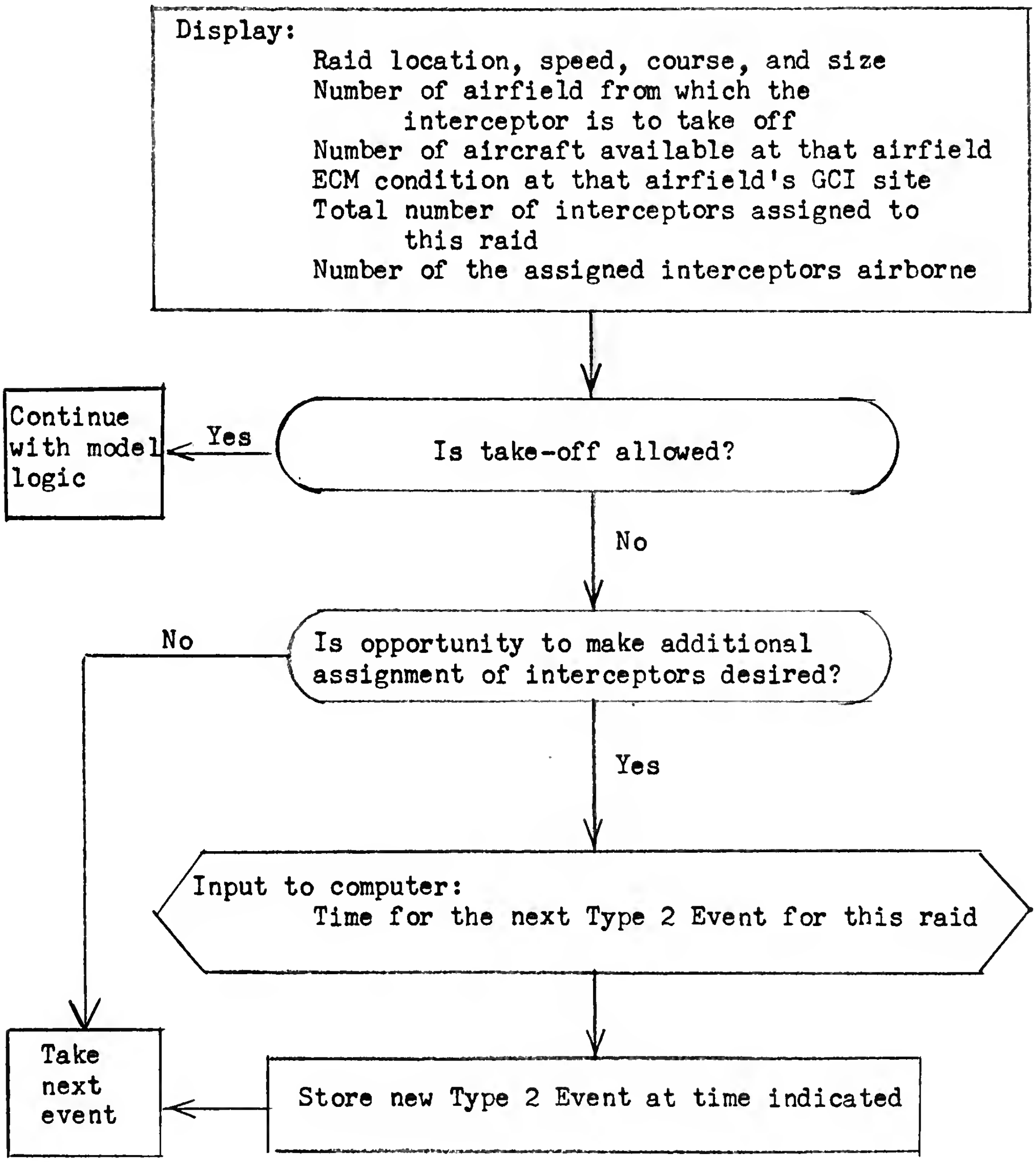
positive results, a probabilistic computation is made to see if the interceptor aborts on take-off. If not, it becomes airborne and the computer proceeds to ascertain which of the four type intercepts will occur and stores the associated event. The type intercept that does occur is a function of the radar conditions existing at the GCI site.

It might seem unusual to wait until the aircraft is airborne and then consider the radar conditions before the intercept type is determined; however, it should be noted that there is no capability for jamming early warning radars in the model. Therefore, an intercept under early warning radar contact only is possible even when the raid is not detected at the GCI site or when white scope conditions exist at the GCI site. This is another internal scheme of the model wherein a very intelligent Area Commander is assumed to be directing the defensive forces.

The modification proposed for this event will allow the player to take a quick look at the situation and make the decision to cancel the take-off and assigned intercept, or to permit the interceptor to continue. The modification involves a deviation from the original model logic after it has been determined that the raid is still in the area. To aid the player in this decision the radar information at the GCI site should be displayed along with the airfield identification; number of unassigned aircraft at the field; position, course, and ground speed of the raid to which the interceptor under consideration is assigned; the number of interceptors assigned to the raid; and the number of these interceptors that are airborne.

The player would then only be required to input to the computer an indicator to either allow the take-off to proceed or to cancel it.

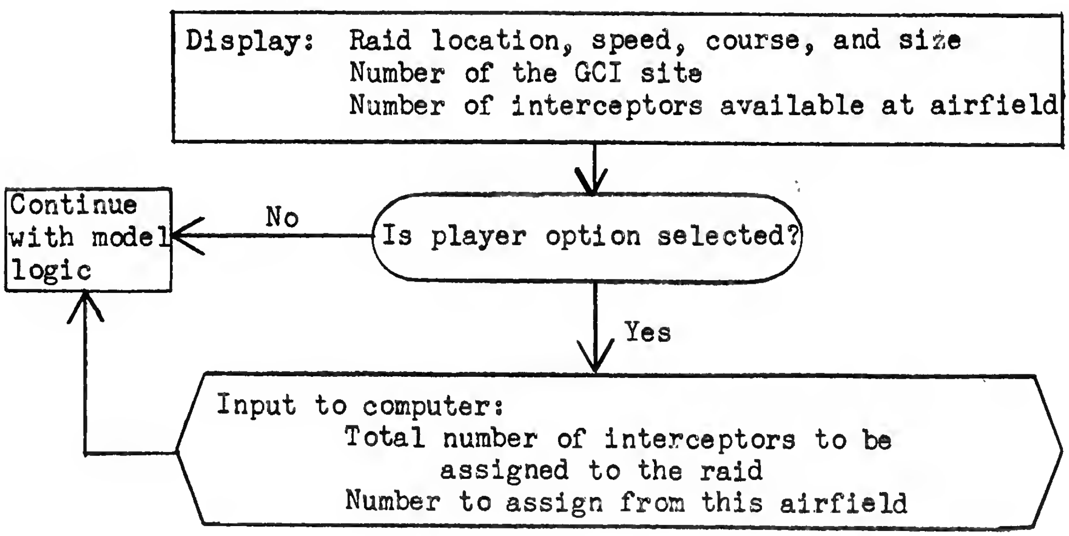
If this take-off event is cancelled the player should be allowed to schedule another Type 2 Event for the raid. In this way the raid may be reconsidered for interceptor assignment. The general logic for the modification is shown below:




<u>Event Type 5A</u> (Raid Detected by GCI Radar) - An indication that the raid has been detected by the GCI site is made when execution of this event begins. After the ECM environment caused by this raid at

the site is determined, a check is made to determine if any aircraft from this airfield previously assigned to the raid are airborne. If this is the case the type intercept will be altered as a function of ECM environment and type commitment. If this is the initial detection of the raid, one of two operations may occur: (1) a white scope, multiple strobe or single strobe ECM condition existing at this airfield causes a Type 5B Event to be placed in store after a time interval described by an input parameter, (2) any other ECM condition existing at this airfield establishes an EW detection and a subroutine (PISUB) is called which assigns interceptors from this airfield if there are enough available to meet the demands of the commitment ratio. A Type 2 Event is stored for the raid when additional interceptors are required.

If interceptors are to be assigned to this raid from this airfield as a result of this detection, the selection of these interceptors should become the option of the player. What must be modified in this event then is the subroutine PISUB in order to have the player control the assignment of interceptors. The following diagram reflects the logic of the modified PISUB routine:



Event Type 5B (Crossing Target Check at GCI Zone) - This event occurs as a result of poor ECM conditions and no detection by early warning radar as described in Event Type 5A above. When this event does occur a check is made to see if EW radar detection has occurred. If it has not, another Type 5B Event is stored for execution after a time interval fixed by input. If there has been an EW detection, the player must be given the option to assign interceptors to the raid from this airfield. The modification to be made again applies only to the PISUB subroutine and is the same as that described for Event Type 5A.

Event Types 6, 7, 8, and 9 (Intercept Events) - These may all be discussed together since all of them perform essentially the same function - the evaluation of the success of an attempted intercept. They differ in the probabilities by which the evaluations are made and in the manner in which a missed intercept is handled. The white scope condition at intercept simulated in the Type 6 Event represents the poorest intercept condition and has the smallest probability of success. Under this condition if the aircraft fails to detect, it will fly on in a search mode until it reaches maximum range or until another interceptor detects the raid, whichever is sooner. The interceptor is then vectored home by a Type 25 Event. The strobe scope condition of Event Type 7 gives a better probability of success but the results of a missed intercept are the same. Event Type 8 represents chaff conditions and Event Type 9 represents a close controlled intercept. The probabilities of success for these events are increased appropriately. A missed intercept in these two events results in the interceptor being vectored home immediately. Model doctrine states that long range

commitments are always assigned Type 8 intercepts.

This event does not require a decision or option to be made by the player. Consequently, the only modification to be made is in the T6B subroutine which actually assesses the results of the attempted intercept. The object of the modification will be simply to have the subroutine inform the player if a bomber is damaged or killed. The change is straight forward and the interested reader may consult the flow chart in Appendix II for the details. It would be desirable to offer the player an opportunity to reassign interceptors which fail to convert; however, as was pointed out earlier, this involves a change in the basic game doctrine and goes beyond the scope of the effort here.

Event Type 11 (Interceptor Landing) - The first operation performed in this event is a check to see if the interceptor's home field is alive. If it is not, the remaining airborne time is computed and a search is made for another airfield. A new landing event is then stored for the time the interceptor will arrive at the new airfield. When the home field is alive, a Monte Carlo computation determines whether the interceptor aborts on landing. A delay in turn around time is computed to reflect the time required for refueling and rearming. It must also be ascertained whether this airfield has exceeded its normal complement of aircraft by receiving aircraft from airfields that have been destroyed. If this is the case, the field's reaction time is increased in proportion to the excess.

The only modification required for this event is to inform the player of the fate of this interceptor and the time it will be available for another assignment if it has survived. The logic of this

modification, outlined below, should be added at the end of the event:

Display: Number of airfield
Fate of interceptor
Time interceptor will be available

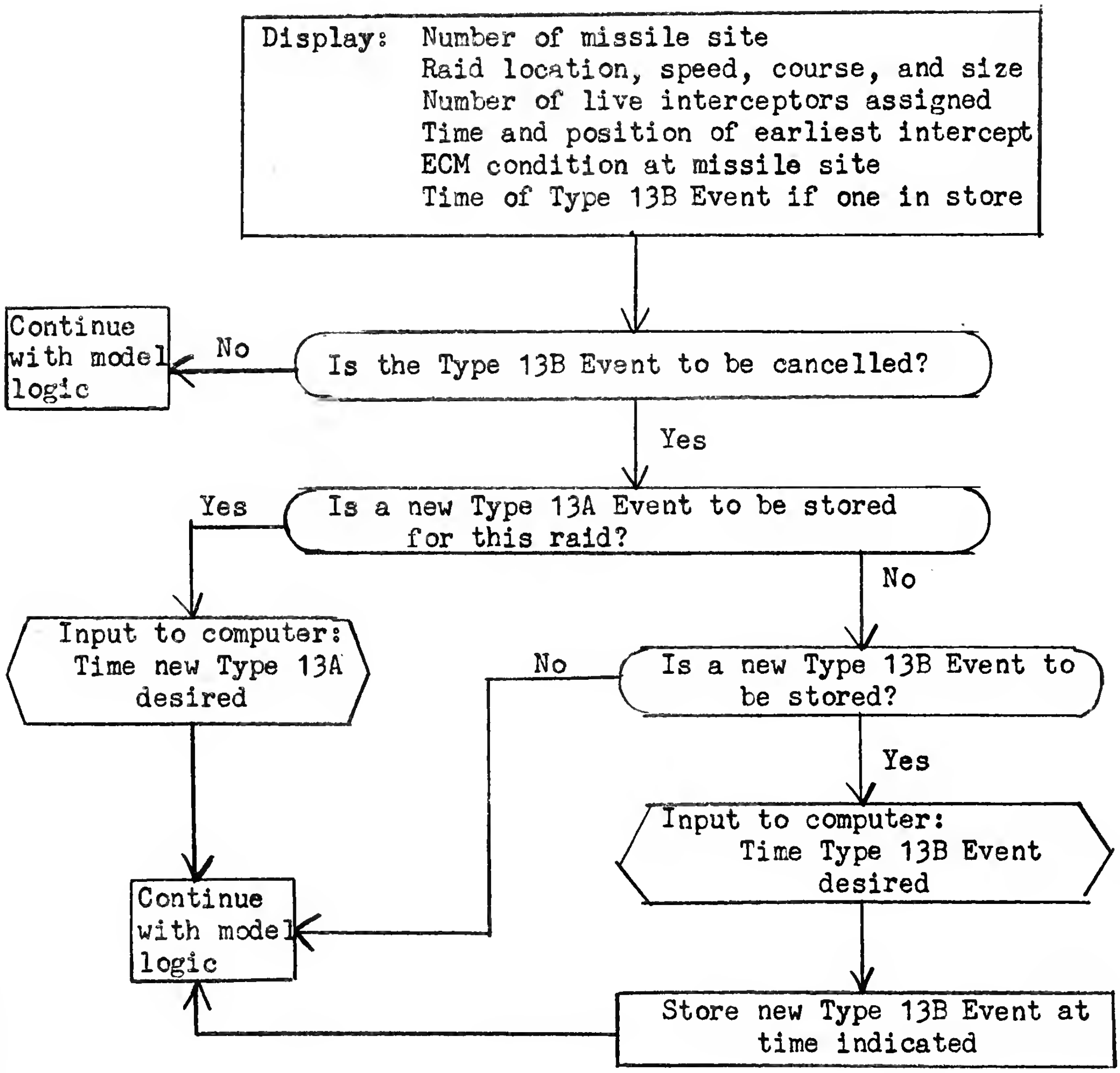
↓

Continue with model logic

<u>Event Type 13A</u> (Raid Detection by Missile Area) - The purpose of this event is to indicate that the raid is now detected by the missile site's search radar and to set up a missile firing routine. If it is determined that the missile area is alive and operable and that live bombers remain in the raid, the ECM conditions at the missile site are used to adjust parameters and events accordingly. If the raid has not been detected by EW radar and a white scope condition exists at the missile site, a check is made to see if the raid will still be within the area of radar coverage after a fixed input time interval. A negative result causes no further action; however, a positive one results in the storage of a new Type 13A Event to occur at the end of the interval. More favorable ECM conditions or an independent EW detection lead to the setting of appropriate acquisition delay times and the storage of Event Types 13B, 13C, and 14. If this is a "false missile site" (an anti-aircraft area) a Type 13C Event is not stored.

The player should be given the option here of determining whether or not he wants this raid placed upon the missile site's target list. The modification required here is that needed to give him control over the storage of a Type 13B Event which causes the raid to be entered on the missile site's target list. The player should also be provided the opportunity to determine the time at which a Type 13B Event is to

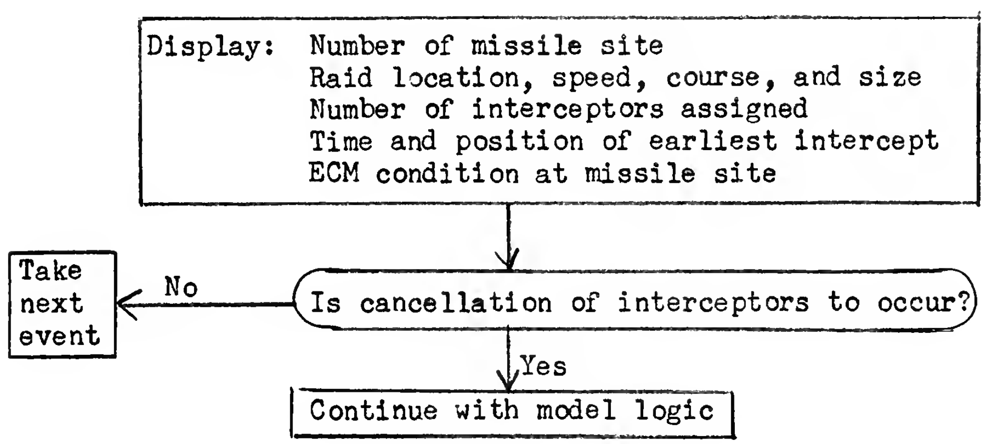
occur. This decision may be postponed by the player storing a new Type 13A Event for some later time. The logic is as follows after the event has performed all of the above operations:




<u>Event Type 13C</u> (Interceptors Cancelled from Raid) - This event cancels interceptors from a raid in a missile area after first checking to be sure the missile site is alive, is operable, and has detected the raid. If there are live bombers in the raid, this event also stores a new Type 2 Event to occur at the end of a specified time interval. This provides for the assignment of interceptors to clean up any targets

that are missed by the missile site.

The decision to prevent interceptors from entering missile areas should be made by the player. A modification to this event must be made, therefore, to permit the player to prevent cancellation of the interceptors if he so desires. An example of a case in which this might be the best course of action would be an interception about to occur on the fringes of a missile area while the site was busily engaged in another sector of its area. The following logic should be inserted into the event logic just prior to the cancellation of the interceptors.




<u>Event Type 14A</u> (Missile Intercept) - The function of this event is to evaluate the results of a missile launch. Monte Carlo computations decide whether the bomber is missed, damaged, or killed. For a miss, the computer checks to see if the bomber is passed the point of closest approach and, if not, sets up a re-engagement. If it is, a Type 14 Event is stored to be executed after a time interval representing an acquisition delay for a new target. For a damaged bomber, a Type 20 Event is stored at a time determined by a Monte Carlo computation. A check is made for a possible re-engagement as described

above for a miss. A kill causes a weapon detonation event to be stored at the time of fall, if applicable, and sets up a new Type 14 Event after the acquisition delay interval.

The results of this must be made known to the player. The modification of this event then is to simply have the intercept results displayed. The logic for this modification is straight forward and may easily be followed on the flow chart in Appendix II.

Event Type 17 (Weapon Detonation) - Damage assessment for weapons delivered by the offense or set to detonate when a damaged bomber dies is carried out by this event. Trajectory error on ballistic missiles is determined here also. Navigational and guidance errors for other type weapons are taken care of in the Weapon Release Event. If more than one weapon is detonated, model logic assumes they are all of the same type and yield. Heavy, medium, and light damage radii of the group are computed by multiplying the appropriate radius of a single weapon by the cube root of the number detonated. Damage is assessed to airfields and missile sites by determining which airfields and missile sites are in the damage zones. Heavy damage is synonomous with total destruction and the airfields and missile sites receiving heavy damage are removed from the game. Essentially, medium damage reduces the capability of the airfields and missile sites to 1/3 its former value and light damage reduces it to 2/3 that value.

Modification to this event will be for the purpose of informing the player of the damage inflicted upon his forces. The general logic needed is illustrated below:

Display: Numbers of airfields within damage range of detonation
Indication of type damage to each airfield
Numbers of missile sites within damage range of detonation
Indication of type damage to each missile site

↓

Continue with model logic

If interceptors have been cancelled from any raids which were in missile areas that are now dead, Type 2 Events, if needed, are stored for the raids.

Event Type 20 (Damaged Bomber Dies) - The functions performed by this event are essentially bookkeeping in nature: indicators are set to show that the bomber is dead and a check is made to see if the bomber has fallen in enemy territory. A positive result stores a Type 17 Event at this time provided any weapons aboard are operable and have a dead-man switch.

The modification required in this event is to inform the player that the bomber is now dead. The logic is as follows:

Display: Raid location, speed, course, and size
Indication that a bomber has died

↓

Continue with model logic

3.3 Discussion of Subroutine and Event to be Added

As mentioned in the introduction to this section, a new subroutine must be added to the model to form a connection between the computer and whatever external equipment is used as a display device. It is

also proposed that a new event be created for the convenience of the player which will periodically check all raids that have been detected and report their current status.

The logic for the subroutine is the same regardless of the device used for displaying the required information; the details, of course, must be tailored to the characteristics of the particular equipment to be utilized. In the general flow charts shown with the discussion of events in sub-section 3.2, the operations requiring display of data will lead to the calling of the subroutine proposed here. The subroutine would in turn take the information stored in the designated locations and transfer it to the display equipment.

In order to utilize the same section of computer memory for all types of display data, each event that displays data will set an indicator, ZE, to a specified value before calling the subroutine. The first operation in the subroutine will be to determine the value of ZE. This will tell it the nature of the data found in the locations reserved for display information. It will, in turn, inform the display equipment as required. The details of logic may be observed by reference to the flow charts for modifications to the events shown in Appendix II and for the subroutine shown in Figures 5A and 5B.

The flow chart for the new event - arbitrarily designated Type 27 (Summary on Detected Raids) is given in Figure 6. The time interval between occurrences of this event could be determined by an initial input to the computer or by a parameter to be adjusted by the player each time the event comes up. The logic here assumes that it will be done the second way. This event and the display of aircraft availability incorporated into the Type 2 Event are to help the player with

his bookkeeping. The sort of bookkeeping that might be required of him will be discussed in Section V. A suggested format for the display of the data to be provided by Event Type 27 is given in Figure 7.

FIGURE 5A

Flow Diagram for Subroutine DPSUB

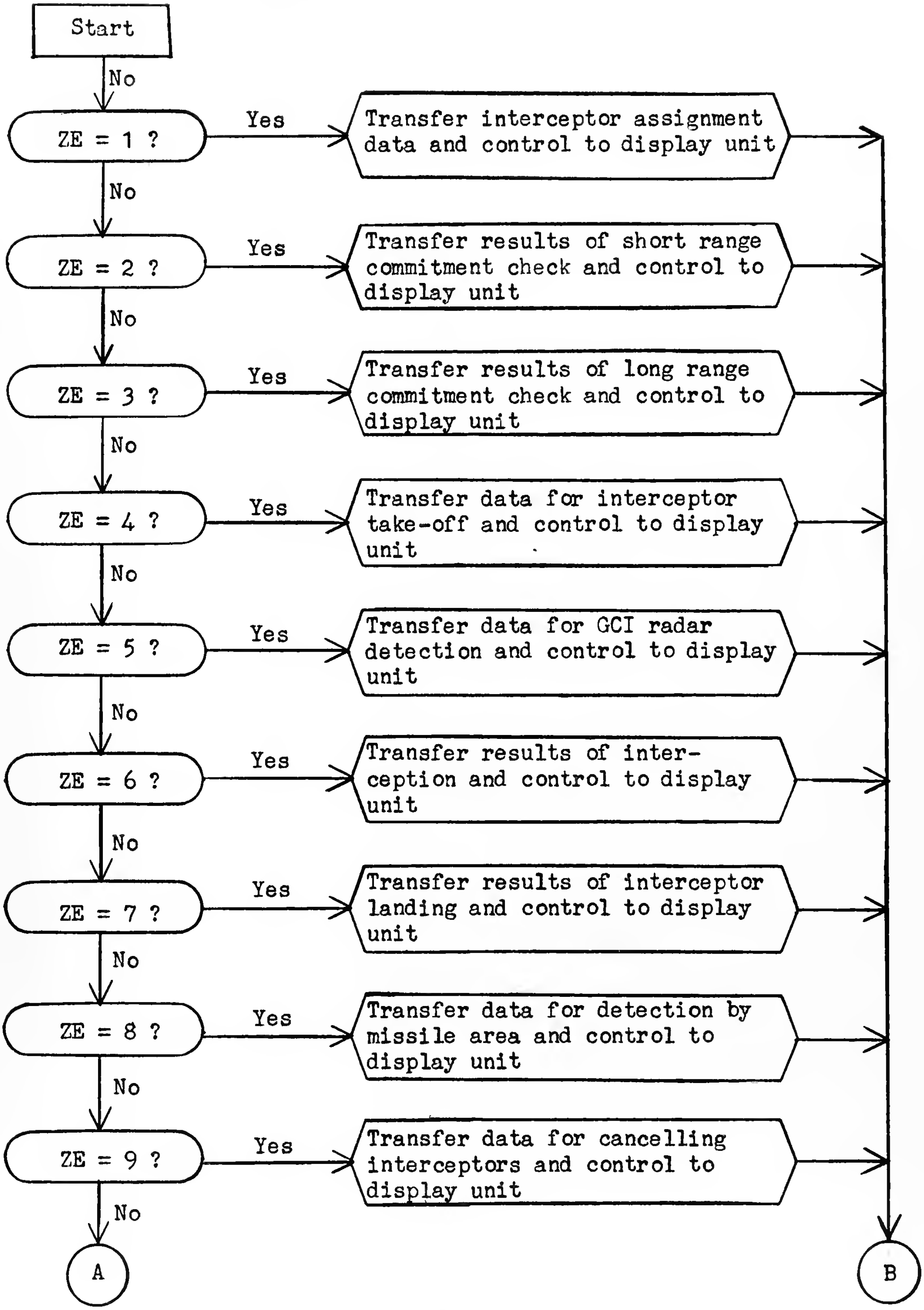


FIGURE 5B

Flow Diagram for Subroutine DPSUB (Continued)

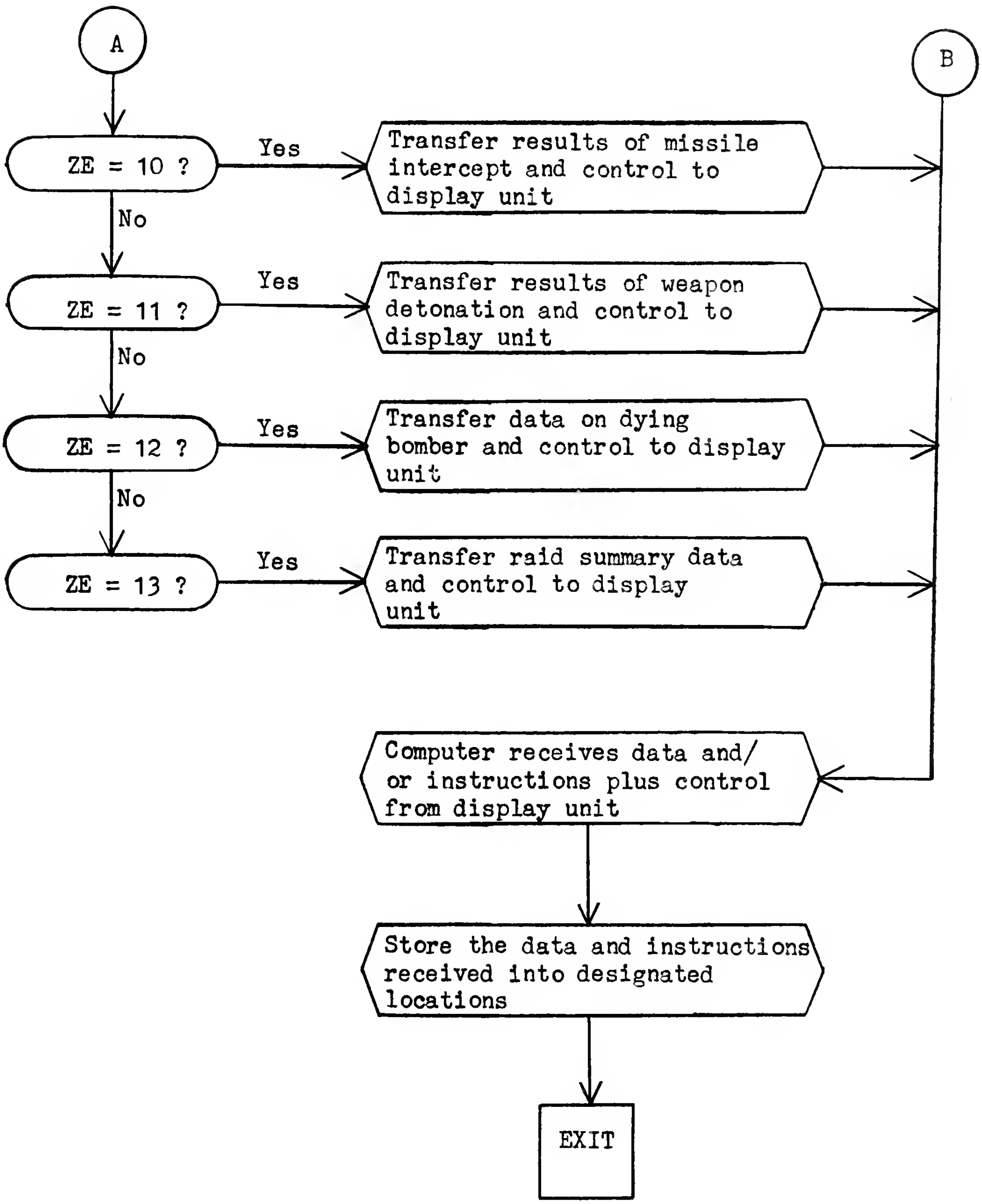


FIGURE 6

Flow Chart for Event Type 27 (Summary on Detected Raids)

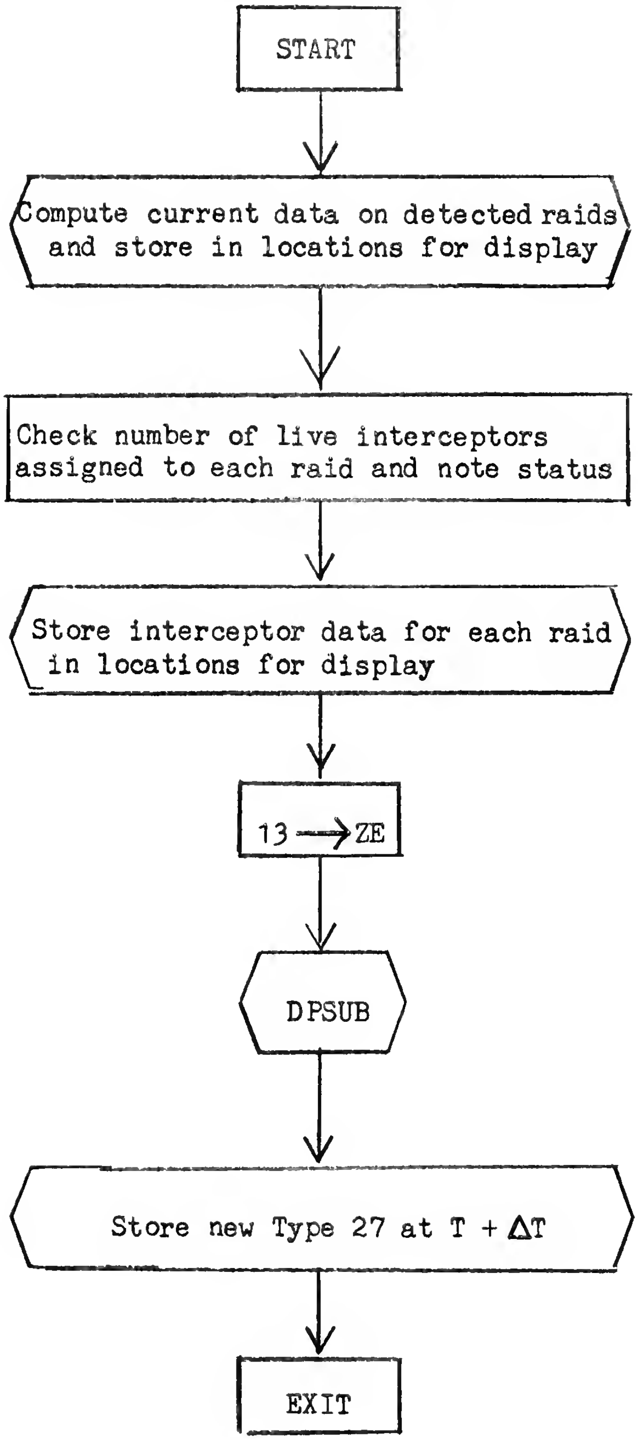


FIGURE 7

Suggested Display for Event Type 27

| | X Coordinate | Y Coordinate | A/F Number | Number Intercepts Assigned | Number Airborne |
|---|---|---|---|---|---|
| Raid Position: | ________ | ________ | ______ | ________ | ________ |
| | | | ______ | ________ | ________ |
| | | | ______ | ________ | ________ |

etc.

Raid Position:

Raid Position:

etc.

## IV

## COMPARISON OF THE COMPLETELY COMPUTERIZED GAME WITH THE PROPOSED COMPUTER/MANUAL GAME

It has been pointed out in the introduction to this thesis that in spite of the tremendous assets of the computerized war game - speed, capacity of handling large quantities of data, ability to play many repetitions of a game, etc. - it suffers from "narrow-mindedness", i. e., a lack of flexibility in its decision making processes. Further, it was observed that this type of decision making is inherent in the structure of the computer and not necessarily the fault of the builder of war games. What this thesis proposes is a partial solution to the problem in the form of a combination computer/manual game that permits a human to make the major decisions during the play of the game while the computer does all the work. This has until very recent times been the traditional relationship between man and machine and one which this author believes should be perpetuated.

This change in the decision making functions is the primary difference between a completely computerized game and the combination game being proposed. The introduction of the human decisionmaker will, of course, slow down the play of the game and increase the running time significantly. It could very well result in an intolerable monopolization of a large digital computer if the operation were not carefully planned and prepared for; however, the problems related to integrating the play of such a game into the busy schedules of most computers will be discussed in Section V. It is the purpose of this chapter to discuss the manner in which several particular decisions are handled and point up the advantages of the combination man/machine type game.

The discussion here will rely heavily upon the description of the original model and the modifications to it given in Section II and III, respectively.

## 4.1 Assignment of Interceptors in the Computer Model

First, consider the process by which the computerized model assigns interceptors. The decision making procedure involved is only a matter of following a set of criterion for bookkeeping procedures. If interceptors have previously been assigned to the raid concerned the recommitment ratio is used, if not, the initial commitment ratio is selected. This ratio is multiplied by the number of bombers in the raid and that number of interceptors is assigned if they are available. No consideration is given to whether the bombers are headed toward low or high priority targets relative to other raids that may be over the defensive territory, to whether they are inbound or outbound from target areas, or to whether one airfield is being utilized to full capacity while another only a few miles further from the action is completely idle. Essentially, the procedure involves assigning by a fixed ratio until the supply of available interceptors is exhausted and then making periodic checks to determine if additional interceptors have become available. In addition to these items interceptors are not held back even though a raid may enter a missile area before it becomes a significant threat. Interceptors are always assigned to the raid if available and are only cancelled from the raid when it is actually under the surveillance of the missile site's search radar and within an input "buffer zone" area.

When interceptors are selected for assignment it should be

remembered that interceptors are always assigned with a priority for short range commitment. This means that if the required number of interceptors can be assigned to a short range commitment, no long range commitments will be considered. Figure 8 depicts a possible undesirable situation that may result from following this procedure. The two circles around each airfield represent the short and long range commitment ranges, $R_1$ and $R_2$, of the two airfields. Note that airfield $A_2$ can make both types while $A_1$ could only make a long range commitment. The model logic dictates that interceptors would be assigned from $A_2$ if they were available even though $A_1$ could intercept much earlier and even though $A_2$ might profitably send more than the standard ratio to intercept the raid on the lower track.

A similar situation may arise if and when airfields sustain damage. According to game doctrine an undamaged airfield may be neglected in the search for interceptors and a damaged one selected because of its relative position to the raid at the time of interceptor assignment. This would result in using lower intercept kill probabilities than is necessary. Because of inflicted damage the probability of a successful take-off will have been significantly reduced and the probabilities for detection, conversion, and kill will have each been reduced also.

It should also be restated here that game doctrine in "The Naval Air Strike Model - Mod Zero" prohibits the re-assignment of airborne interceptors to other raids.

4.2 Target List Entry and Selection in the Computer Game

The criterion by which targets are entered on and selected from

the target lists of missile sites by the computer is also firmly fixed in the game doctrine. All raids that enter the missile area and are detected by the search radar are automatically placed upon the missile site's target list without any consideration of the relative threat of the raids or to the possibility that a raid may be better handled by interceptors. Once entered upon the target list, the raid becomes an actual target for a surface-to-air missile based, generally, upon the principle of "nearest and least engaged". As an example of a situation that might occur using this doctrine, the reader's attention is directed to Figure 9. Assume that the $M_2$ missile site has been very busy with earlier raids and has only a few missiles left and that it has sustained light damage. Model doctrine requires that Raid V and Raid VI be placed upon the $M_2$ missile site's target list and its remaining missiles would most likely be expended on them even though they will soon pass through other defensive areas, airfield $A_1$ and missile site $M_1$, before becoming a significant threat to the defense. This would leave missile site $M_2$ void of missiles with which to attack Raid VII which is coming in behind Raids V and VI and does not enter the other defensive areas.

4.3 Advantages of the Combination Game

The comments made here, although directed at the model under discussion, are not intended as criticism of this particular model but to illustrate characteristics of computer models in general. In fairness to the Naval Air Strike Model, it should be noted that it was not built to test interceptor assignment policies and some of the items that have been pointed out could have been improved within a computerized model had it been so intended; yet, these improvements could not have gone

beyond having the computer check a finite number of factors which would in combination describe a finite set of possible situations - those which the builder thought most likely to occur.

Also, the remarks here should not be construed to mean that the author does not recognize that the completely computerized war game is a valuable tool of analysis. The combination game as proposed is offered as a refinement to make it more valuable for certain applications, but not all. Every war game has limitations. Its capabilities are a function of the skill of both the builder and the user. The numbers accumulated during the play of any game are meaningless except when interpreted and analyzed within the framework of a complete knowledge of the model and the input. For the user that is aware of this, the completely computerized war game will continue to be more acceptable for many analytical purposes. In a game of this type, it is felt that interceptor and missile assignment doctrines are two parameters assigned to the defense that cannot be factored out of any final analysis of game results and treated as constants. The modifications proposed for the combination game described in this thesis will truly let assignment doctrines be treated as parameters and not necessarily constants.

Finally, the combination game has definite advantages for educational purposes. It emphasizes decision points in the game by presenting them to the player for resolution. The displays themselves are a running commentary on the progress of the simulation and the course it is following. The learning by doing concept comes in to play and involves the acquisition of knowledge of both gaming concepts and of elements of the situation being simulated.

FIGURE 8

Illustration of Interceptor Commitment Procedures

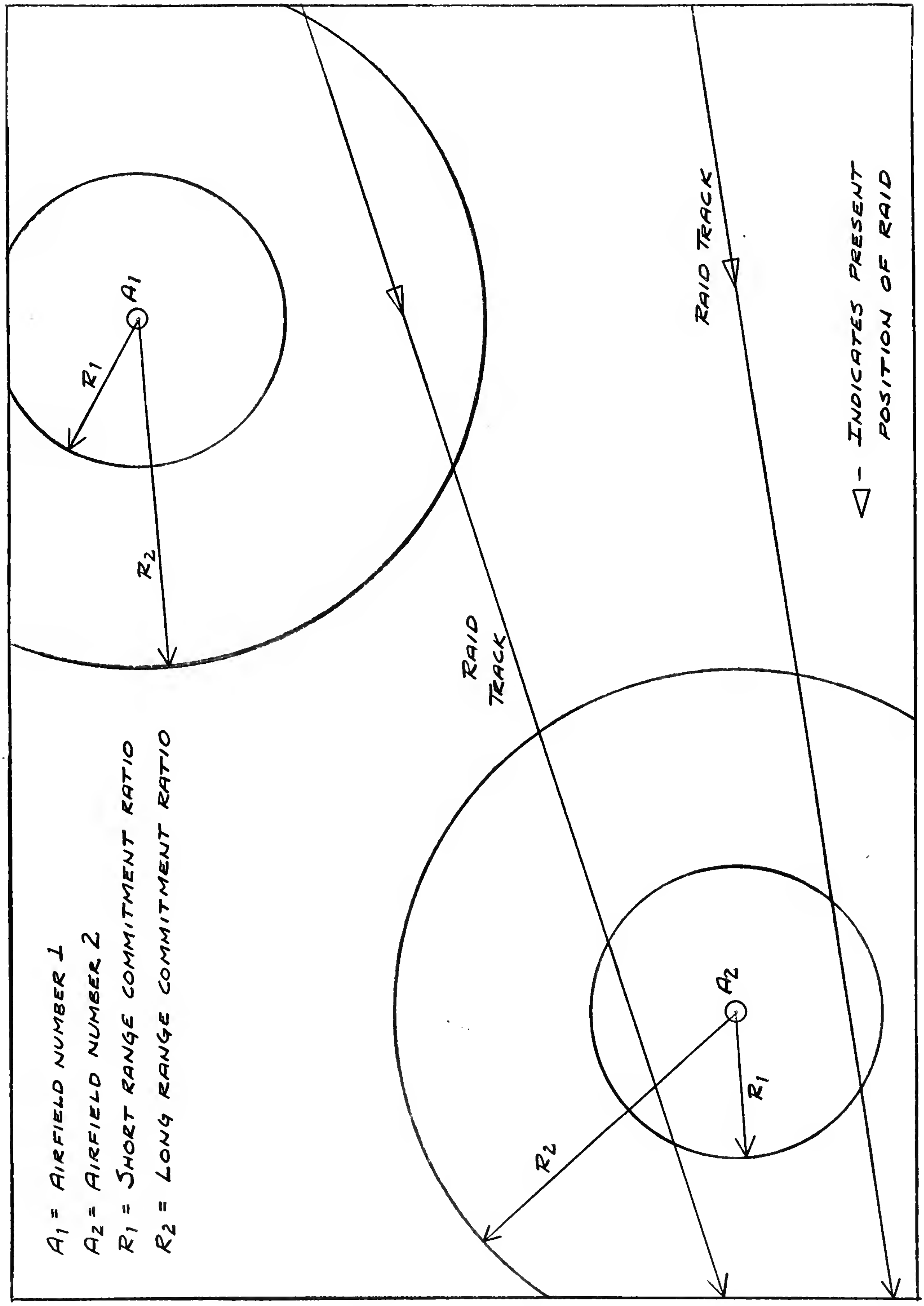
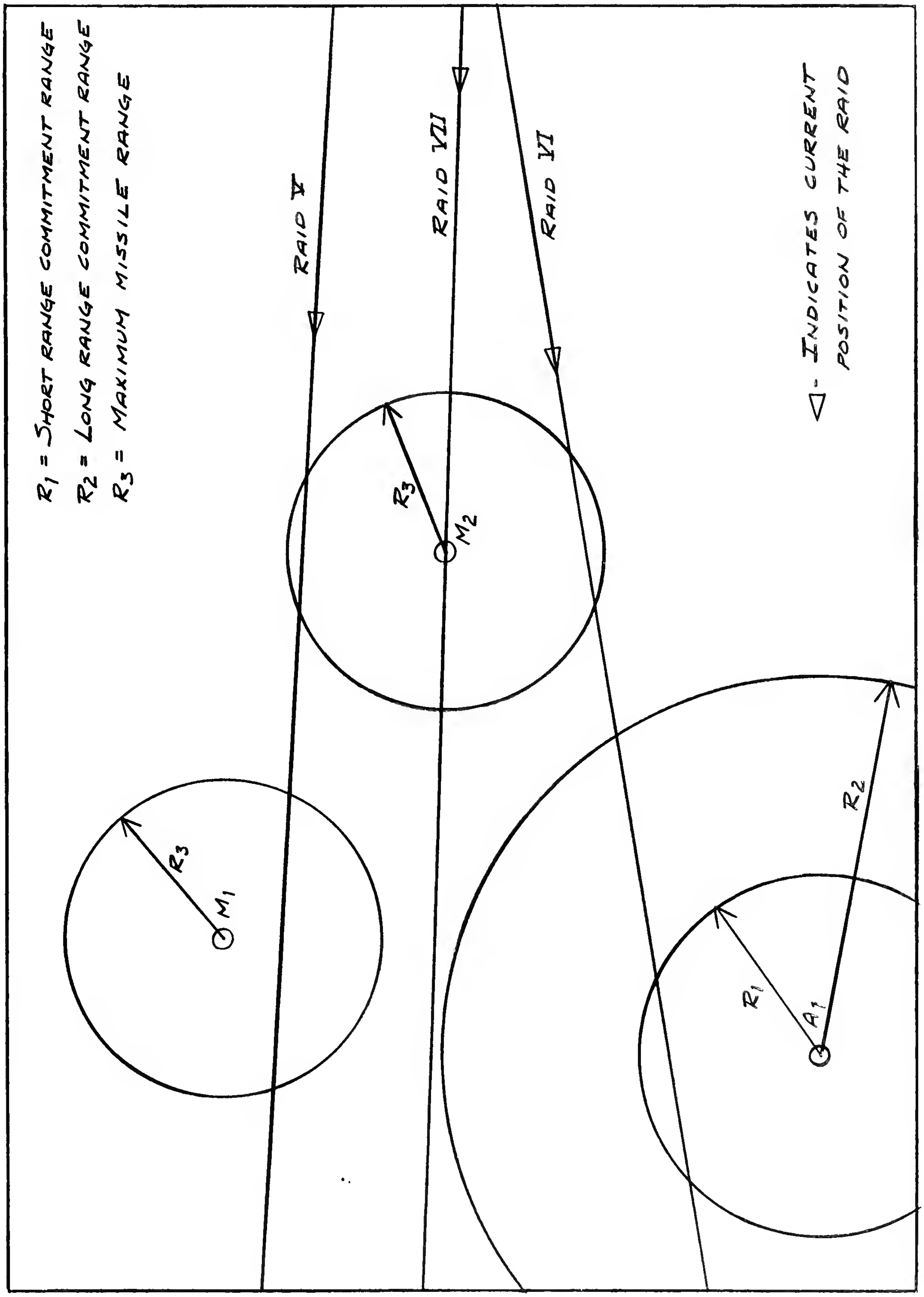


FIGURE 9

Illustration of Missile Target List Entry Procedures



V

# DISCUSSION OF DISPLAY TECHNIQUES

A discussion of display techniques and the related problems of putting the display device on line to the computer is in order at this point. The business of putting the display device on line is itself no longer a problem. It simply requires that the program for the model be written so that the computer is told when and what data is to be passed to the display unit, and what to do with the data it receives from that unit upon getting a signal indicating that the player has made a decision and wants to return control to the computer. That is the essence of the modifications that have been proposed in Section III. The related problem which will require solution is how to provide for utilization of the computer during the interval of time in which the player examines the data being displayed to him, takes any actions necessary to integrate the new information with that previously received, and reaches his decision. Since this problem is the same regardless of the type display used, further discussion will be postponed until methods of display have been considered.

There are two basic types - a static display and a dynamic display. The first is less expensive, easier to acquire, and very well suited to the concept proposed for the combination computer/manual game; consequently, it is recommended and will be discussed in greater detail. The dynamic display is more elaborate and if properly done has definite advantages over the first.

## 5.1 Dynamic Display

A dynamic display would require the reproduction of the entire

playing area (or at least as much of it as action was to take place in) upon the face of the display unit. All forces of which the side viewing the display had knowledge would be displayed in their proper positions and these positions continuously updated to maintain a current picture of the game situation. The nature of the forces would be identified by appropriate symbols and such intelligence as had been revealed to the player should be depicted. For example, the number of bombers in a raid could be revealed by a number beside the symbol representing the raid, its ground speed by a second number, and its track by a third number or by showing a series of its previous positions to form a track line behind the symbol. Both the advantages and disadvantages of this type display should be readily apparent. The player has a complete and current picture of the entire situation displayed before him at all times and has no plotting and very little record keeping to do. On the other hand, if very many forces were to be employed, the size of the display unit would have to be quite large and the design very sophisticated; thus, increasing the cost to unreasonable magnitude. Also, additional programming would be required and more computer time would be utilized during the play of the game. The high costs expected for such a setup might be justified for an activity conducting a program which would utilize the equipment full time, but not for one studying the feasibility or evaluating the concepts of such a game on a part time basis.

### 5.2 Static Display

A static display may take several forms ranging from periodic displays of the type discussed above through displays of isolated portions of the playing area (in static form) to display words and

numbers which convey the desired information to a viewer. The latter includes such very common techniques as on line printers and card punch equipment. Both the static and dynamic techniques must provide for return of information to the computer which amounts to instructions and/or data to permit the computer to continue its operation. The means of handling this function range from keying in information on the control console of the computer through pre-punched data cards to keyboard arrangements on the display equipment itself.

As was mentioned in the introduction, a display device in use at the U.S. Naval Postgraduate School is very well suited to the requirements of the combination war game and some consideration has been given to employing it in this manner. The device is the Data Display Model dd65 built by Data Display Incorporated of Saint Paul, Minnesota. Two cathode ray tubes (approximately ten inches in diameter) are used for display and the equipment has its own keyboard through which alterations and additions may be made in the display. The logic controlling the keyboard of the display unit must be part of the program of the computer itself. Such alterations and additions are made in corresponding locations in the machine's memory and are thereby introduced into the system. The present utilization at the Postgraduate School employs a CDC 160 to control the logic of the dd65 and as a connection between the dd65 and the CDC 1604. The use of the CDC 160's memory frees the CDC 1604 while data is being displayed and viewed on the dd65. The dd65 may actually be programmed for either a dynamic display, within limitations, or static display; however, the static is recommended in view of the small display scope and the complexity of a dynamic display.

There are a number of display devices available of both a static and dynamic type which are suitable to a system such as that proposed here, but would, of course, require special adaptations. All things considered, a static type display is the most likely to be employed because it costs less and requires less special adapting; therefore, the suggested display formats presented in Appendix II are in keeping with this mode of operation. A static display would require the player to maintain a plot of the playing area and a status board or sheet to keep himself informed. Preferably, he should have a plot similar to that maintained in a typical CIC Room and several assistants to post and plot information as he called it out while viewing his display. If the player were a Naval Officer this would place him in somewhat familiar surroundings; thus, affording him maximum utilization of passed learning in his present situation and at the same time be conducive to carrying his learning from the game forth to his future assignments.

Figure 10 is representative of the type plot of his own forces with which a player might begin a game and Figure 11 illustrates how it may look one hour of game time later. The player's version should be much larger of course - on the order of 6 x 8 feet or 8 x 10 feet. Two numbers are given beside raid and interceptor flight positions in the figure. The one above the line gives game time in minutes; the one below, the number of targets in the raid or interceptors in the flight. By reference to such a plot, appropriate status boards for his own forces, and the data currently before him on the display, a player could make intelligent decisions under circumstances which realistically simulate a true military situation.

## 5.3 Time Sharing

On the problem of insuring continuous use of a large digital computer using a time sharing executive routine while playing a combination computer/manual game, only a few comments need to be made. It is possible to build sophisticated executive routines for digital computers which will pick up other small programs automatically and run them while monitoring the display unit for its signal that it is ready to return to operation of the war game. The executive routine can be designed to return immediately to the war game program or to finish the program it is currently processing and then return. From that level of sophistication, one may descend to a setup such as the one at the Naval Postgraduate School in which the entire war game program may be read out into satellite equipment while the CDC 160 is programmed to present material to the display unit. An operator at the control panel or a less sophisticated time sharing type monitoring routine could be signalled when return to the main computer was desired and would control that return according to established procedures. There are still less sophisticated techniques that could be employed but the significant fact to recognize is that it is quite feasible to do all of these things. The method will have to be chosen that is commensurate with the time and effort to be invested in this type war gaming.

FIGURE 10

Initial Plot of Playing Area

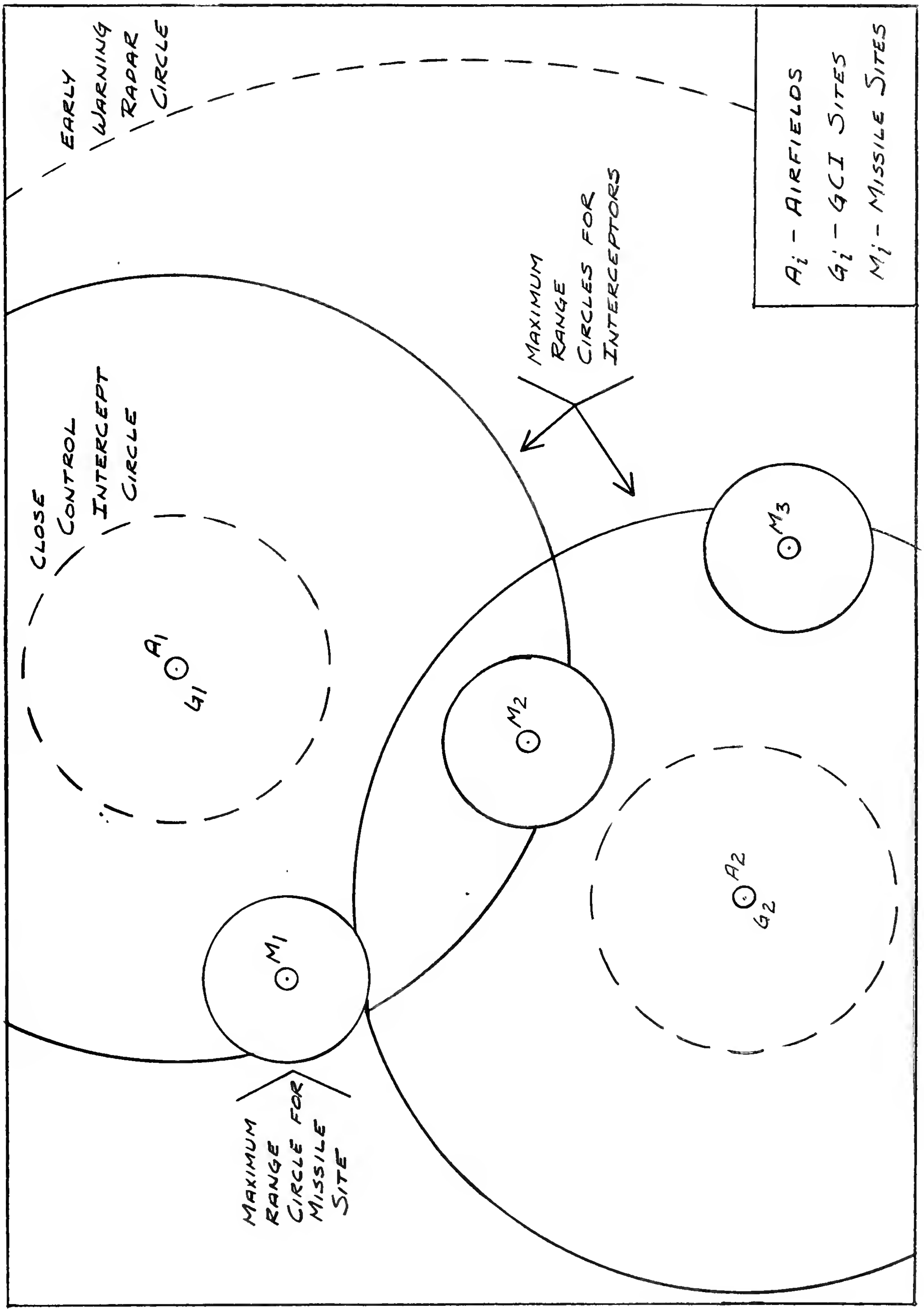
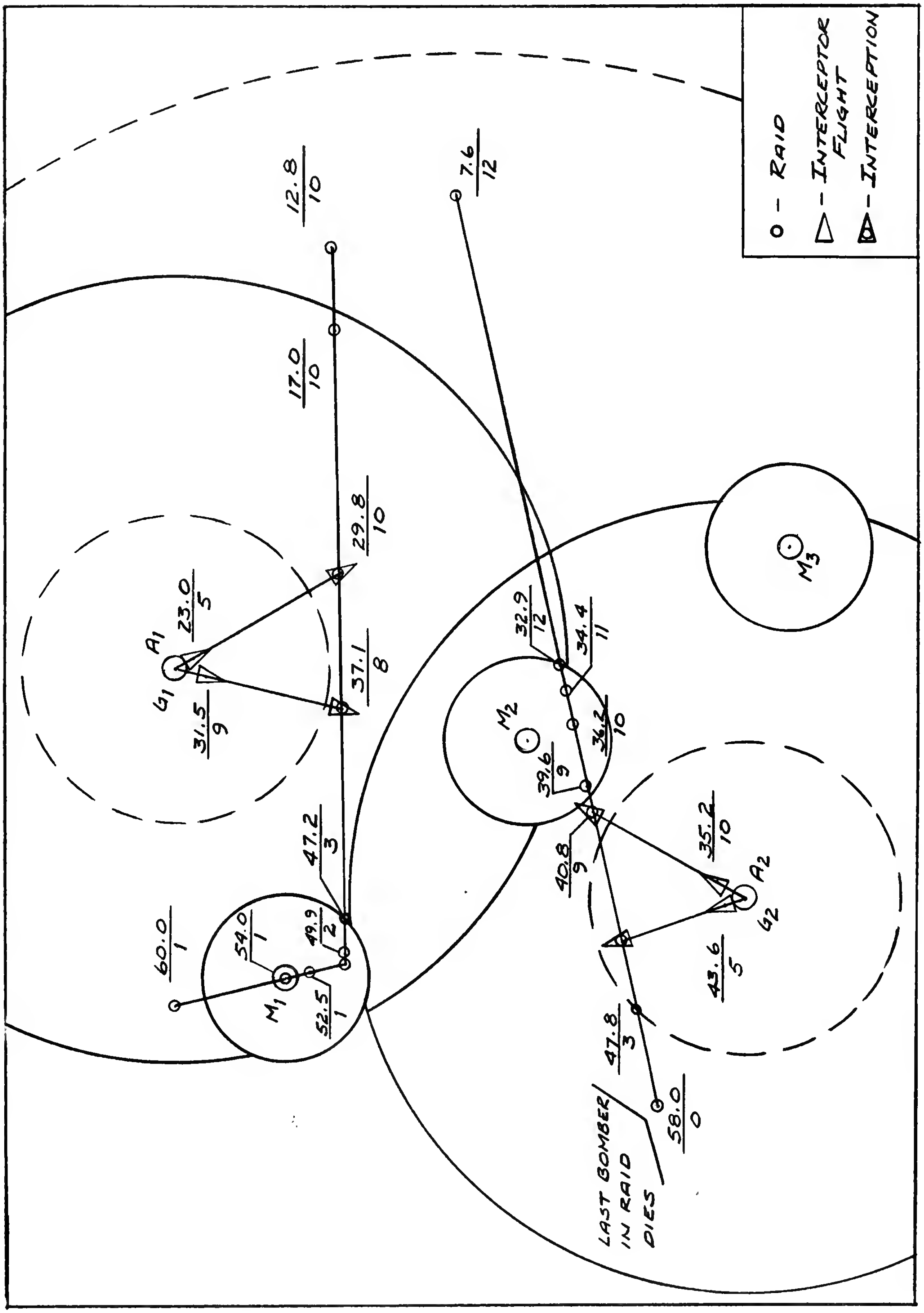


FIGURE 11

Plot of Playing Area After One Hour of Game Time



VI

# CONCLUSIONS AND RECOMMENDATIONS

## 6.1 Conclusions

Several conclusions may be drawn from the description in Section III of the modifications required to convert a game for player participation and from the comparison in Section IV of the completely computerized war game with the combination computer/manual model. Others follow from the discussion in Section V of problems associated with putting the combination game on the computer without wasting valuable computer time while data is being displayed. These are summarized here for the convenience of the reader.

(a) The combination computer/manual war game provides a feasible solution to the inflexible decision processes required by a computer.

(b) The addition of the human decisionmaker results in more realistic reaction by the defensive forces in the particular game discussed here and is likely to have a similar effect on any game.

(c) While a game built from scratch as a combination man/machine model could realize greater benefits from the concept, an existing computer model may be modified for this type play with very satisfying results and the amount of effort involved is far from prohibitive.

(d) All things considered, a static display of data would seem most appropriate to the requirements of this game - at least until an extensive evaluation has been made.

(e) The decision as to how best to utilize the computer during

periods of data display is one which must be made in terms of the needs and capabilities of the activity proposing to use the technique of a combination game.

## 6.2 Recommendations

In view of the discussion contained in this thesis and the conclusions stated above, it is recommended:

(a) That activities involved in the construction and utilization of war games consider the technique proposed here as an improvement in the decision making functions in large computerized war games.

(b) That activities concerned with the teaching of war gaming methods and techniques consider the understanding of these concepts that might be acquired from participating as the player in a game such as that described here.

(c) That activities concerned with the training of command and control center personnel and others who must make decisions quickly on the basis of information pieced together from a number of sources consider the value of experience gained as the player in a combination man/machine simulation of a typical situation in which they might be required to act.

(d) That those activities involved in the evaluation of tactics to be employed by a system (such as the defense system in the game discussed here) consider the merits of performing the evaluation by having a player utilize the tactics in a war game of this type.

(e) That any game built or modified for combination computer/

manual play have an indicator, the value of which is an initial input, which would inform the computer as to whether a play of the game was to be in the combination mode or the completely computerized one. The computer should be programmed to check this indicator at each point that data is to be displayed to a player. One value of the indicator would tell the computer to continue with the display; another would mean skip the display and proceed with a completely computerized play of the game.

## BIBLIOGRAPHY


1. Applied Physics Laboratory, Johns Hopkins University. Descriptions of Air Strike Model Input Sheets, by R. F. Meier. March, 1962. PAM-47.

2. Applied Physics Laboratory, Johns Hopkins University. Air Strike Model (Flow Charts), by A. F. Andrus and R. F. Meier. May, 1962 PAM-54.

3. Applied Physics Laboratory, Johns Hopkins University. Input Description and Output Format for Air Strike Model FORTRAN Processor, by A. F. Andrus and R. F. Meier. June, 1962. PAM-55.

4. Applied Physics Laboratory, Johns Hopkins University. Output Format of the Air Strike Model, by A. F. Andrus and R. F. Meier. June, 1962. PAM-56.

5. United States Naval War College. Fundamentals of War Gaming, by Francis J. McHugh. Second edition, November, 1961.

## APPENDIX I

## SIMULATION LOGIC

There are two basic types of logic commonly used in large computerized war games by which the computer carries out the play of the game. These are the time step logic and the event store logic.

The time step is the older of the two since it is easier and more natural in manual games. The size of the step becomes the unit of time in the game. The computer proceeds at the beginning of each step to compute what may happen during the time interval based upon information about both sides stored in its memory. This stored information is then adjusted (or updated) to reflect the interactions occurring in the interval and the computer steps forward to the next interval. The time intervals frequently are as long as five or more minutes. One need not think very long or hard to realize that treating everything that happens in a five minute interval as if it all occurred at the same instant could lead to gross errors. Couple this with the fact that the computer must have a fixed order for considering the various actions that may occur and must consider one player first and then the other, and it will be obvious that a bias is apt to be introduced. The usual method of countering this is to decrease the size of the time steps; however, decreasing them enough to eliminate the chance of bias frequently results in increasing the running time for the game beyond tolerable limits. Another counter is the event store logic.

In the event store type game there are a finite number of possible events which may occur during the play of the game. Certain of these events may generate certain of the others. The computer maintains an

event store table which is a chronological list of events scheduled to occur. The game usually starts with a list of pre-planned events. As the computer proceeds to execute these, others are generating and their time of occurrence computed. These are inserted into the table in the proper chronological order and the game is played out by the computer processing each event in the list in its proper chronological sequence.

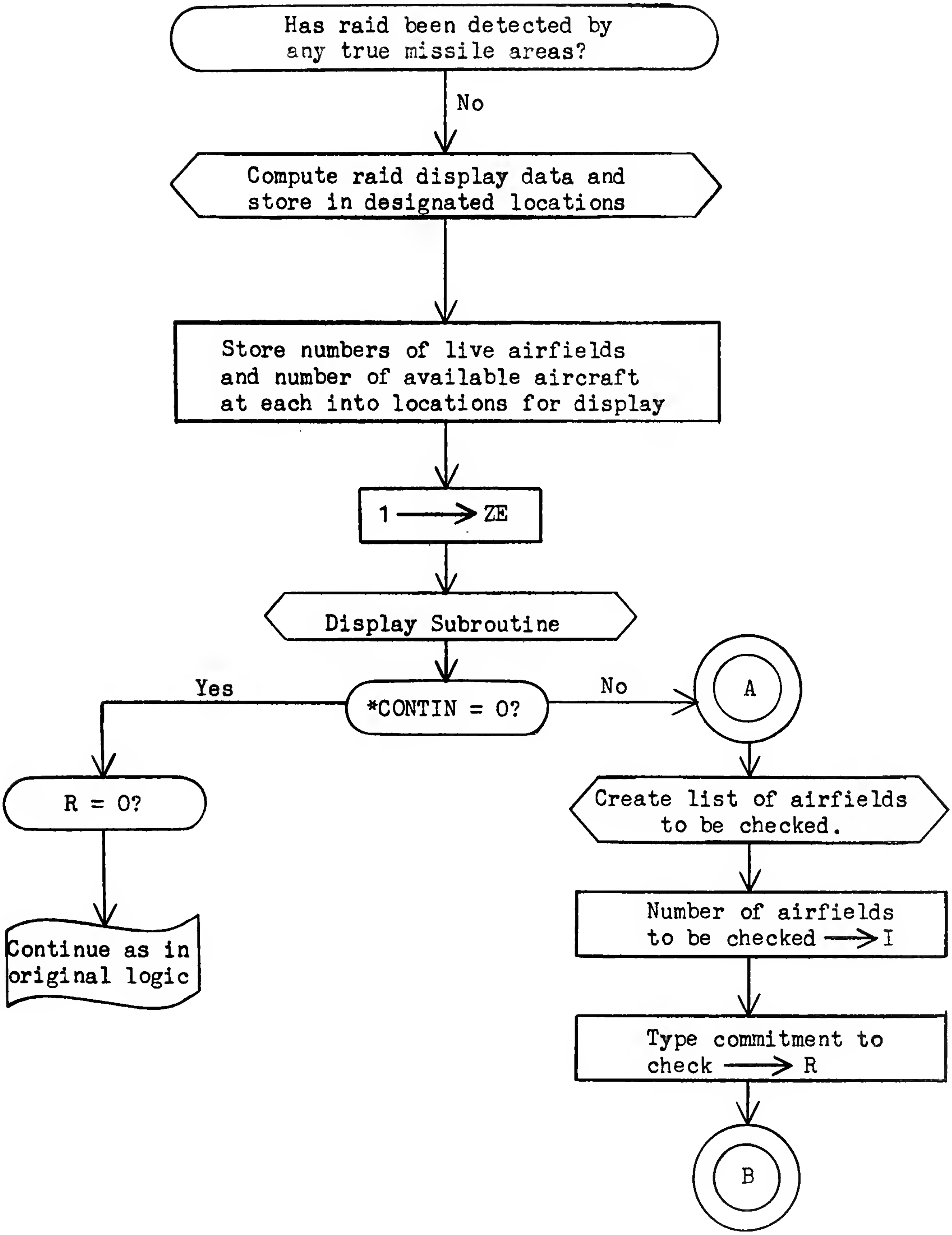
# APPENDIX II

## FLOW CHARTS FOR MODIFICATIONS TO EVENTS

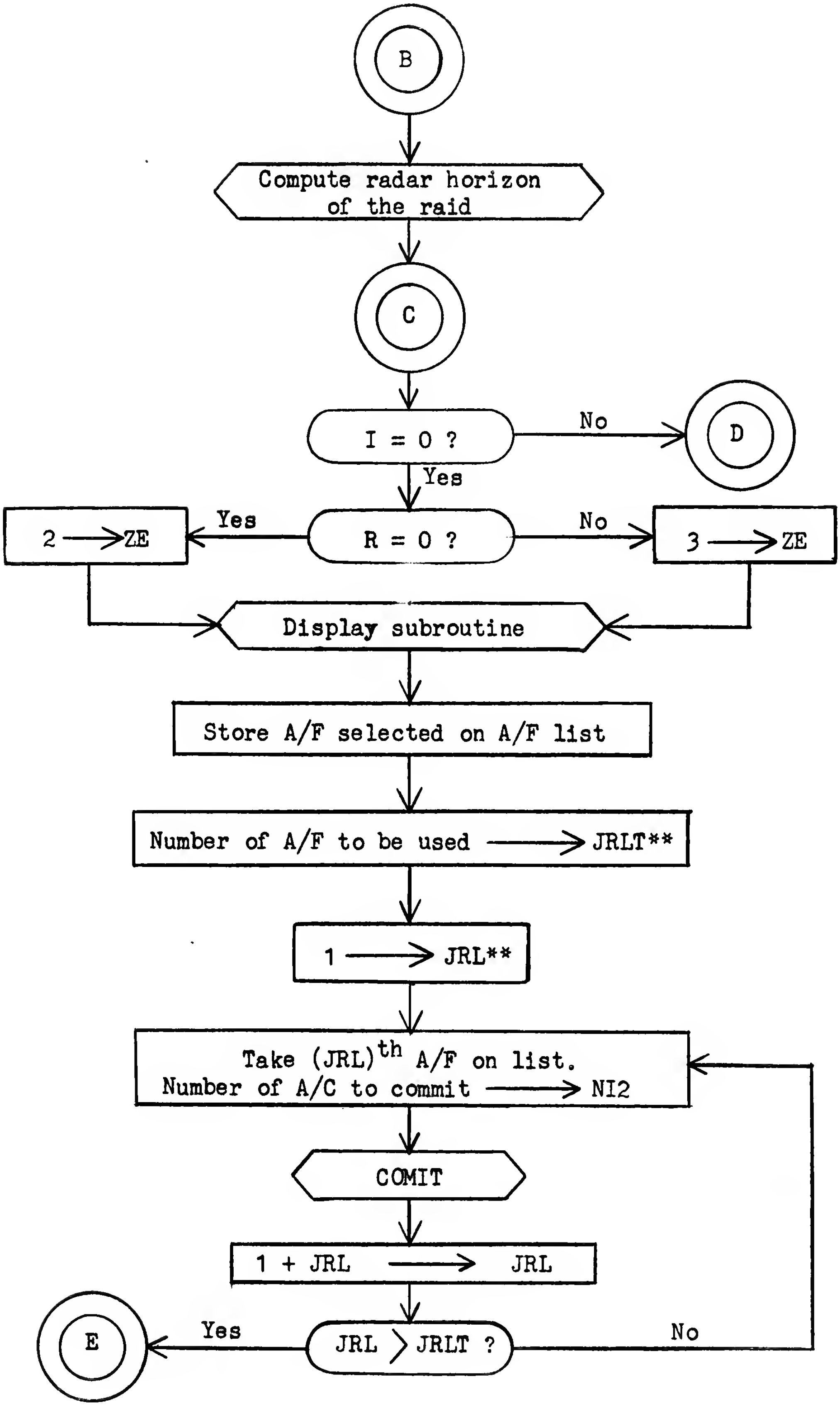
This appendix contains all the detailed flow charts for the necessary modifications to the events of "The Naval Air Strike Model - Mod Zero" to convert it for combination computer/manual type play. Detailed flow charts for the original events that are being modified are reproduced in Appendix III and labelled to indicate the point at which a deviation would be made to follow the logic shown here. A second point is marked on the original flow charts to show where one would return to that logic upon reaching the end of the logic for the modification. To further assist the reader in fitting the proposed modifications into the original logic, the last element of the original flow charts prior to shifting to the modification and the first element to be entered upon returning to the original are shown as the first and last element respectively in the flow chart of the modification.

The data that should be displayed to the player for each of the modified events is listed after the logic for the modification and is presented in a format that might be useful for some types of static displays.

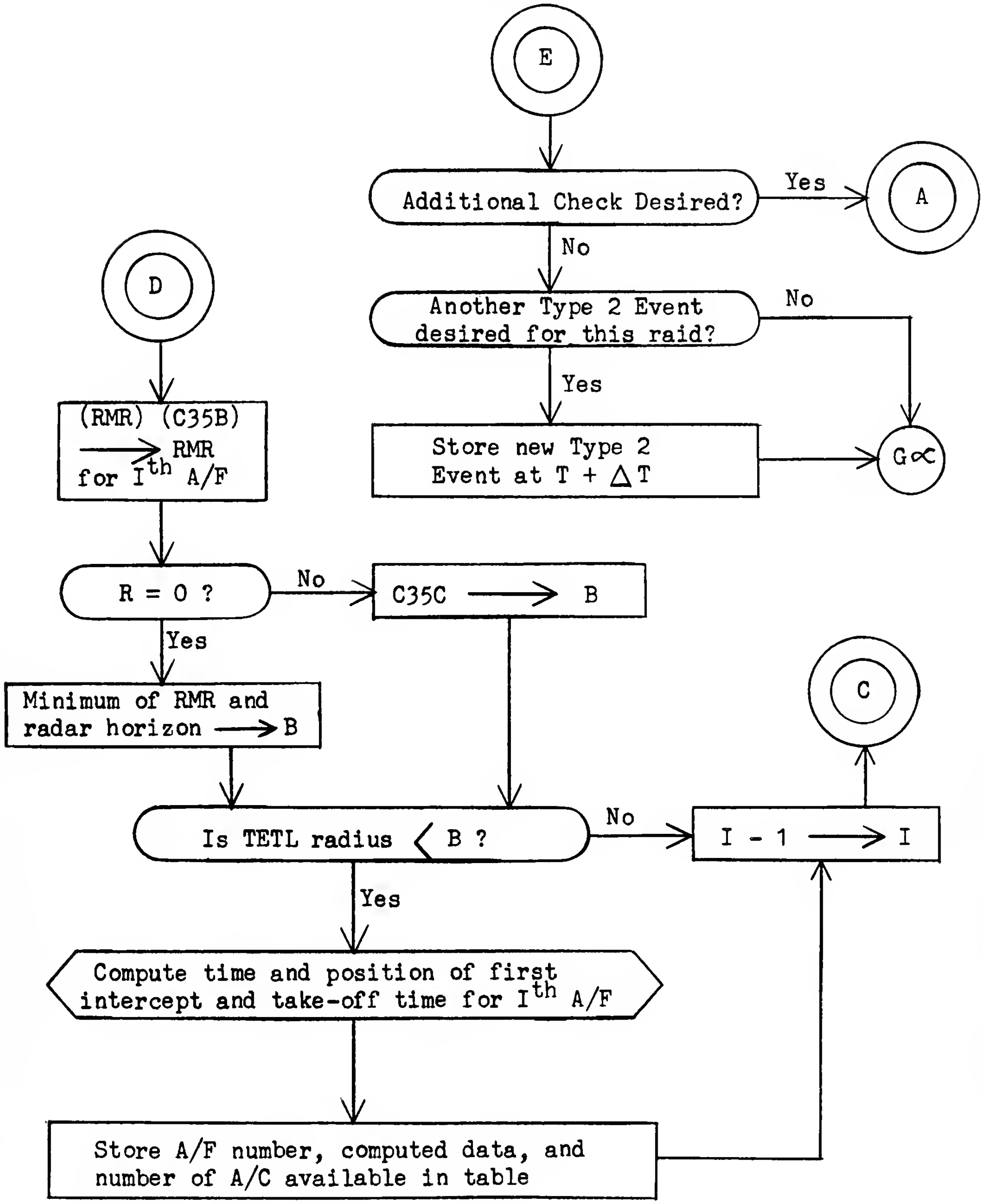
MODIFICATION TO EVENT TYPE 2
(Interceptor Assignment Event)



MODIFICATION TO EVENT TYPE 2
(continued)



MODIFICATION TO EVENT TYPE 2
(continued)




*CONTIN is an indicator: CONTIN = 0 means for the computer to complete the event in normal manner
CONTIN = 1 means manual control will be exercised

**This variable is defined by its use here.

DATA TO BE DISPLAYED FOR A TYPE 2 EVENT
(Initial Display)

Table showing all live airfields and the number of available aircraft at each:

| AIRFIELD NUMBER | NUMBER OF UNASSIGNED A/C | NUMBER ASSIGNED BUT NOT AIRBORNE |
|---|---|---|
| ______ | ______ | ______ |
| ______ | ______ | ______ |

(etc., for all live airfields in the game)

| | | | |
|---|---|---|---|
| RAID LOCATION: | X-coordinate | (game coordinate) | ______ |
| | Y-coordinate | (game coordinate) | ______ |
| | Time | (game time) | ______ |
| RAID COURSE | | (degrees true) | ______ |
| RAID SPEED | | | ______ |
| RAID SIZE | | | ______ |
| NUMBER OF LIVE INTERCEPTORS ALREADY ASSIGNED TO THIS RAID | | | ______ |
| TYPE COMMITMENT TO CHECK FOR* (0 = short, 1 = long) | | | ______ |
| TOTAL NUMBER OF AIRFIELDS TO BE CHECKED* | | | ______ |
| NUMBERS OF THE AIRFIELDS TO BE CHECKED* | | | ______ |
| | | | ______ |
| | | | ______ |

(etc., listing all desired to be checked)

* Indicates an item to be inputted by the player

DATA TO BE DISPLAYED FOR A TYPE 2 EVENT
(Results of Airfield Check)

TYPE COMMITMENT CHECKED FOR ________

| A/F NUMBER | EARLIEST TAKE-OFF TIME | POSITION OF FIRST INTERCEPT X-COORD. | Y-COORD. | TIME OF FIRST INTERCEPT | NUMBER OF A/C AVAILABLE |
|---|---|---|---|---|---|
| _____ | _____ | _____ | _____ | _____ | _____ |
| _____ | _____ | _____ | _____ | _____ | _____ |
| _____ | _____ | _____ | _____ | _____ | _____ |

(etc., list in table form all airfields for which the check was positive)

TOTAL NUMBER OF AIRCRAFT TO BE ASSIGNED* _____

TOTAL NUMBER OF AIRFIELDS TO BE USED* _____

| A/F NUMBER* | NUMBER OF INTERCEPTORS* |
|---|---|
| _____ | _____ |
| _____ | _____ |
| _____ | _____ |

(etc., for each airfield from which interceptors are assigned)

ADDITIONAL CHECK DESIRED* (0 = yes, 1 = no) _____

TYPE COMMITMENT TO CHECK FOR* (0 = short, 1 = long) _____

TOTAL NUMBER OF AIRFIELDS TO BE CHECKED _____

NUMBERS OF THE AIRFIELDS TO BE CHECKED _____

_____

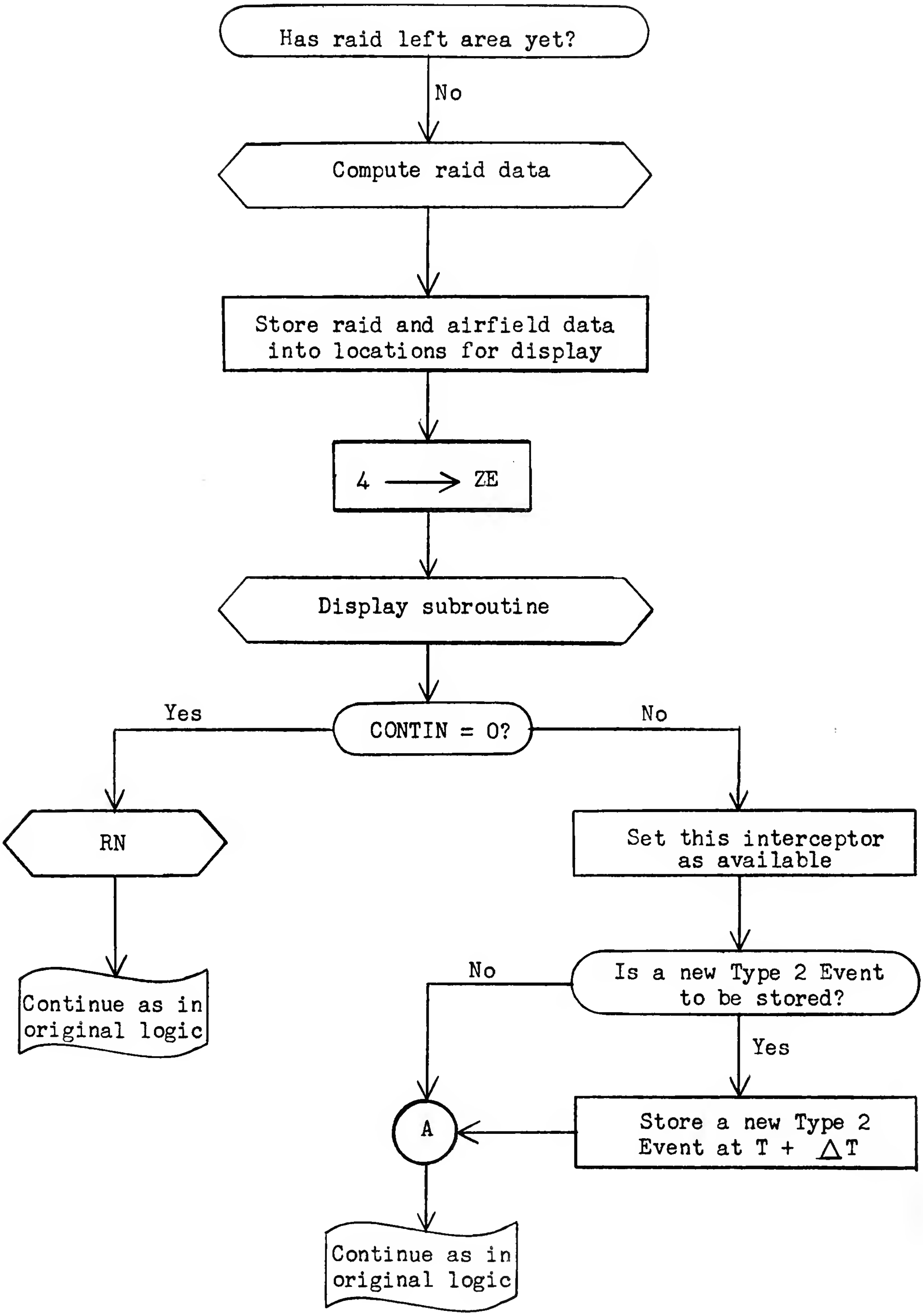
(etc., listing all desired to be checked) _____

IS ANOTHER TYPE 2 EVENT TO BE SCHEDULE?* (0 = yes, 1 = no) _____

TIME INTERVAL* (This interval to be added to present time to give time for next Type 2 Event) _____

* Indicates an item to be inputted by the player

MODIFICATION TO EVENT TYPE 4
(Interceptor Take-off)



DATA TO BE DISPLAYED FOR A TYPE 4 EVENT

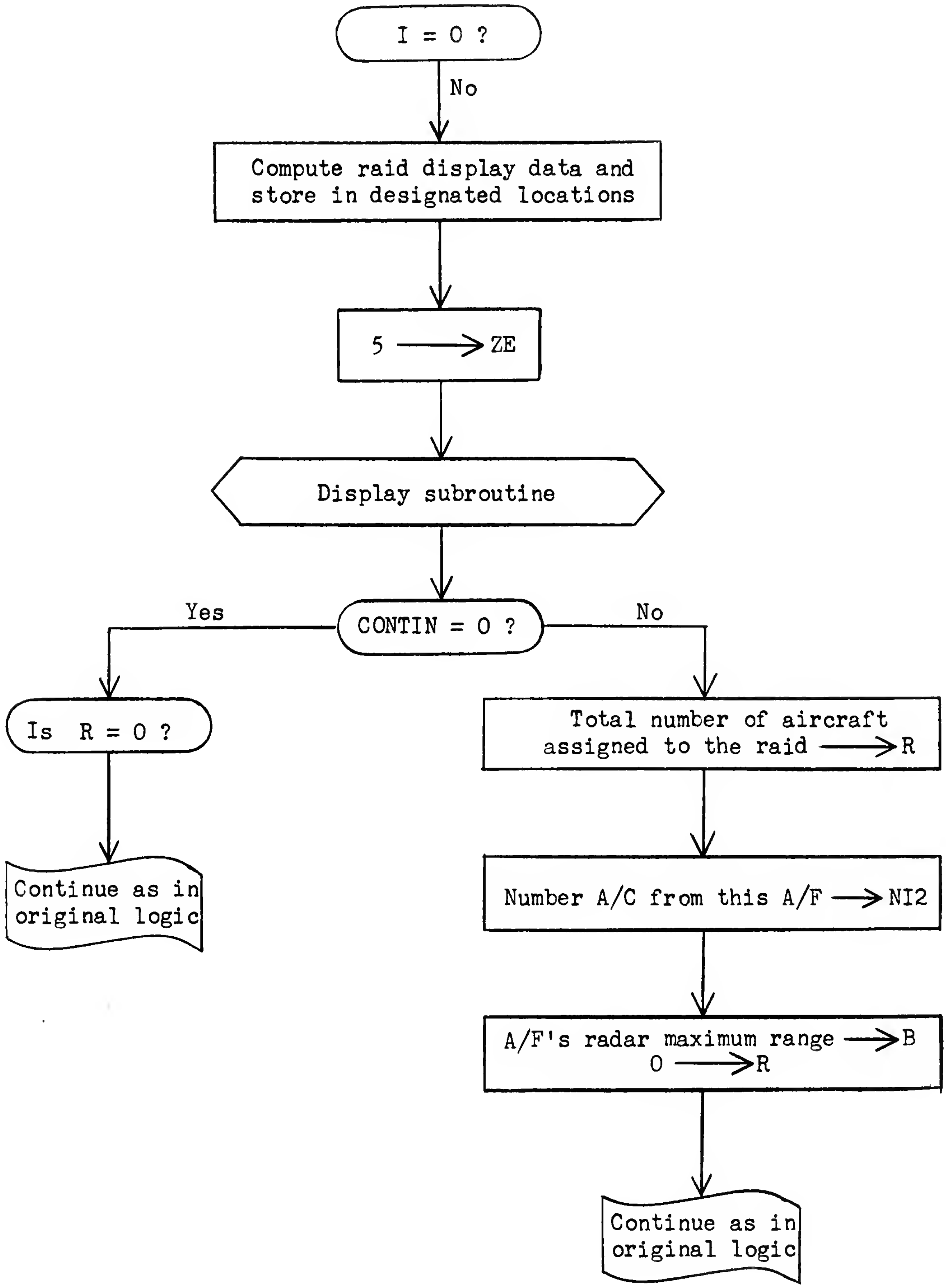
| | | | |
|---|---|---|---|
| AIRFIELD NUMBER | | | ______ |
| NUMBER OF UNASSIGNED AIRCRAFT AVAILABLE AT THIS FIELD | | | ______ |
| ECM CONDITION AT THIS GCI SITE** | | | ______ |
| FOR THE INTERCEPTOR ABOUT TO TAKE-OFF: | | | |
| RAID LOCATION: | X-coordinate | (game coordinates) | ______ |
| | Y-coordinate | (game coordinates) | ______ |
| | Time | (game time) | ______ |
| RAID COURSE | | | ______ |
| RAID SPEED | | | ______ |
| RAID SIZE | | | ______ |
| TOTAL NUMBER OF LIVE INTERCEPTORS ASSIGNED | | | ______ |
| NUMBER OF THESE INTERCEPTORS AIRBORNE | | | ______ |
| CONTIN* (See Type 2 Event diagrams for explanation) | | | ______ |
| IS NEW TYPE 2 EVENT TO BE STORED FOR THIS RAID?* (0 = Yes, 1 = No) | | | ______ |
| TIME INTERVAL* (This interval to be added to present time to give time of new Type 2 Event) | | | ______ |

* Indicates an item to be inputted by the player

** ECM conditions might be indicated by the following:

1 = White scope
2 = Multiple strobes
3 = Multiple blips
4 = Clear condition

MODIFICATION OF SUBROUTINE PISUB FOR EVENT TYPES 5A AND 5B
(Raid Detected by GCI Radar and Crossing Target Check at GCI Zone)

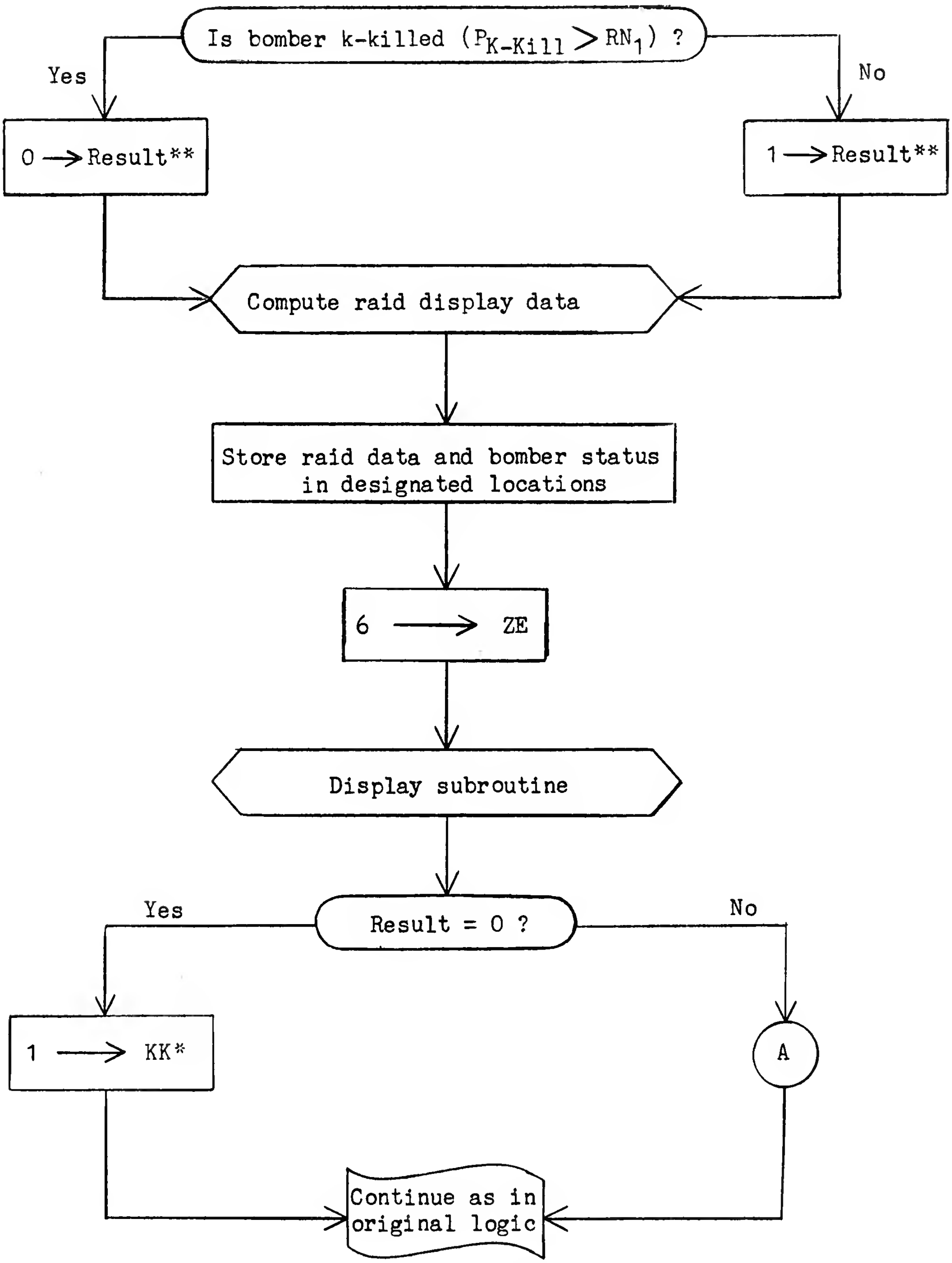


DATA TO BE DISPLAYED FOR EVENT TYPES 5A AND 5B

| | | |
|---|---|---|
| CONTIN* | | ______ |
| RAID LOCATION: | X-coordinate | ______ |
| | Y-coordinate | ______ |
| | Time | ______ |
| RAID COURSE | | ______ |
| RAID SPEED | | ______ |
| RAID SIZE | | ______ |
| NUMBER OF THIS GCI SITE | | ______ |
| NUMBER OF AVAILABLE AIRCRAFT AT THIS AIRFIELD | | ______ |
| TOTAL NUMBER OF AIRCRAFT TO BE ASSIGNED TO THIS RAID* | | ______ |
| NUMBER TO ASSIGN FROM THIS AIRFIELD* | | ______ |

*Indicates an item to be inputted by the player

MODIFICATION TO SUBROUTINE T6C FOR EVENT TYPES 6, 7, 8, AND 9
(Intercept Events)




** This variable defined by its use here

DATA TO BE DISPLAYED FOR EVENT TYPES 6, 7, 8, AND 9

RAID LOCATION: X-coordinate ______
Y-coordinate ______
Time ______

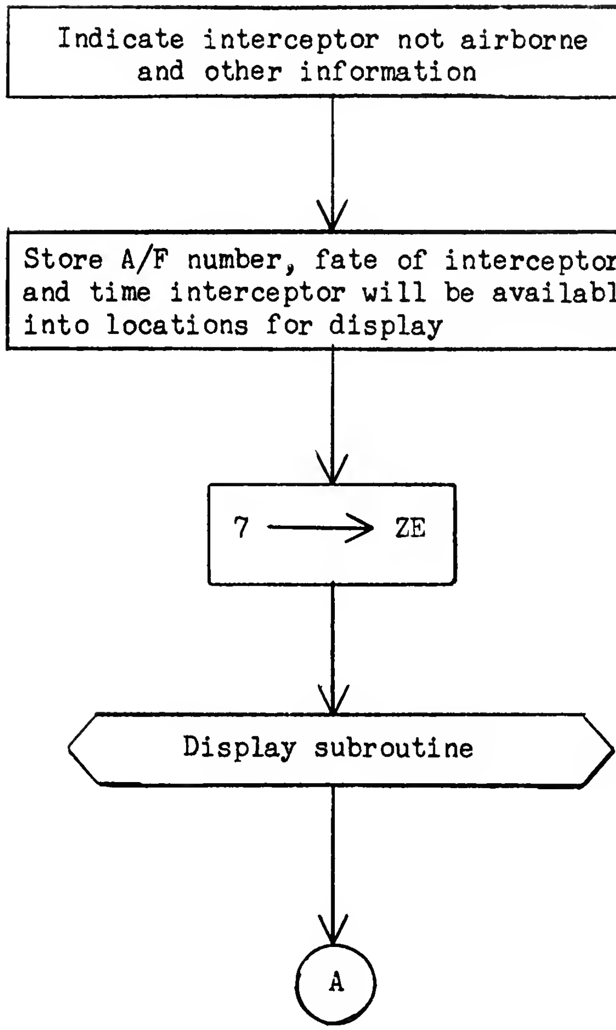
RAID COURSE ______

RAID SPEED ______

STATUS OF BOMBER (0 = killed, 1 = damaged) ______

RAID SIZE ______

MODIFICATION TO EVENT TYPE 11
(Interceptor Landing)



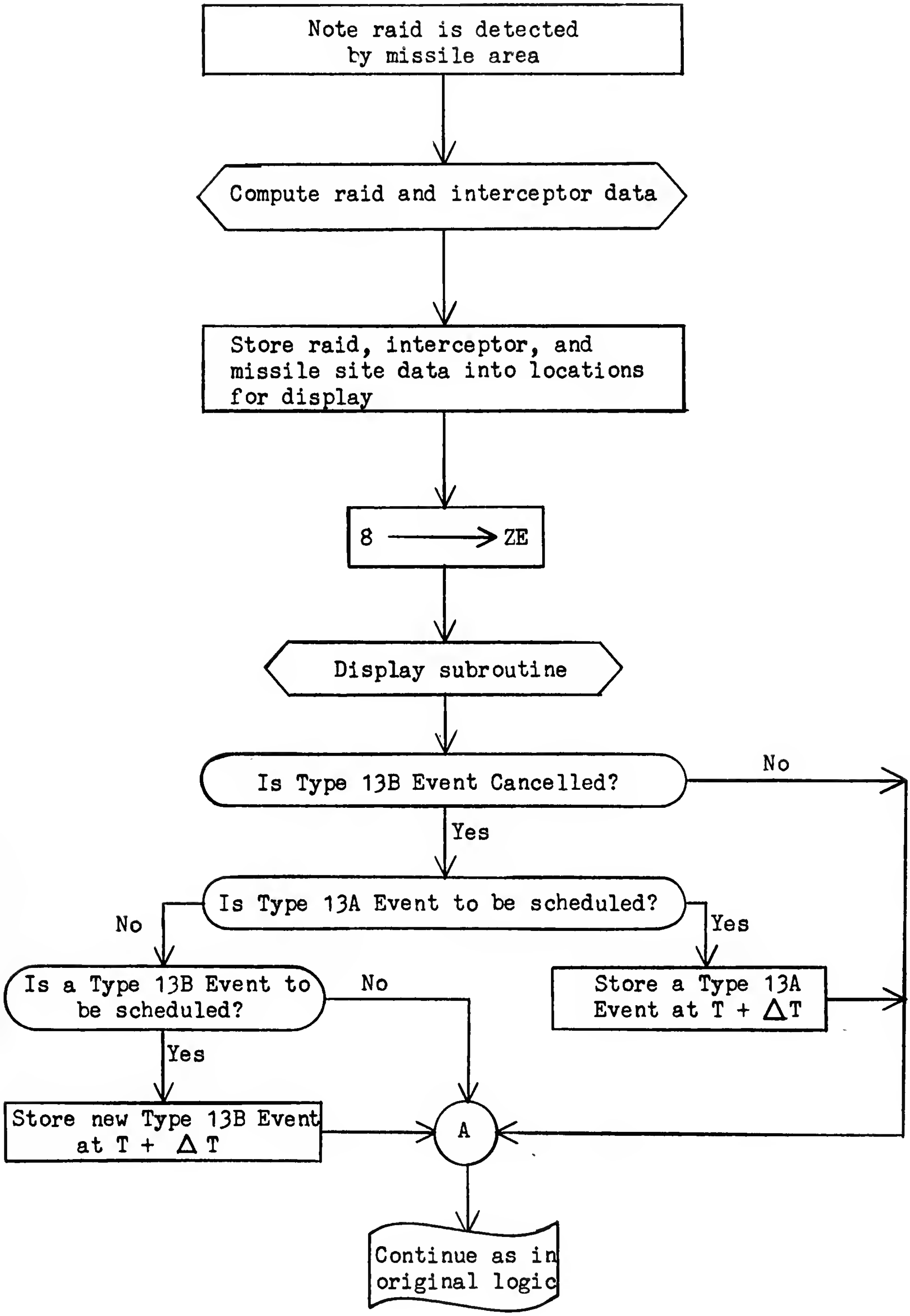
DATA TO BE DISPLAYED FOR A TYPE 11 EVENT

AIRFIELD NUMBER ______

FATE OF INTERCEPTOR (0 = dead, 1 = damaged, 2 = safe) ______

TIME INTERCEPTOR WILL BE AVAILABLE FOR FURTHER ASSIGNMENT (Game time) ______

MODIFICATION TO EVENT TYPE 13A
(Raid Detected by SAM Area)



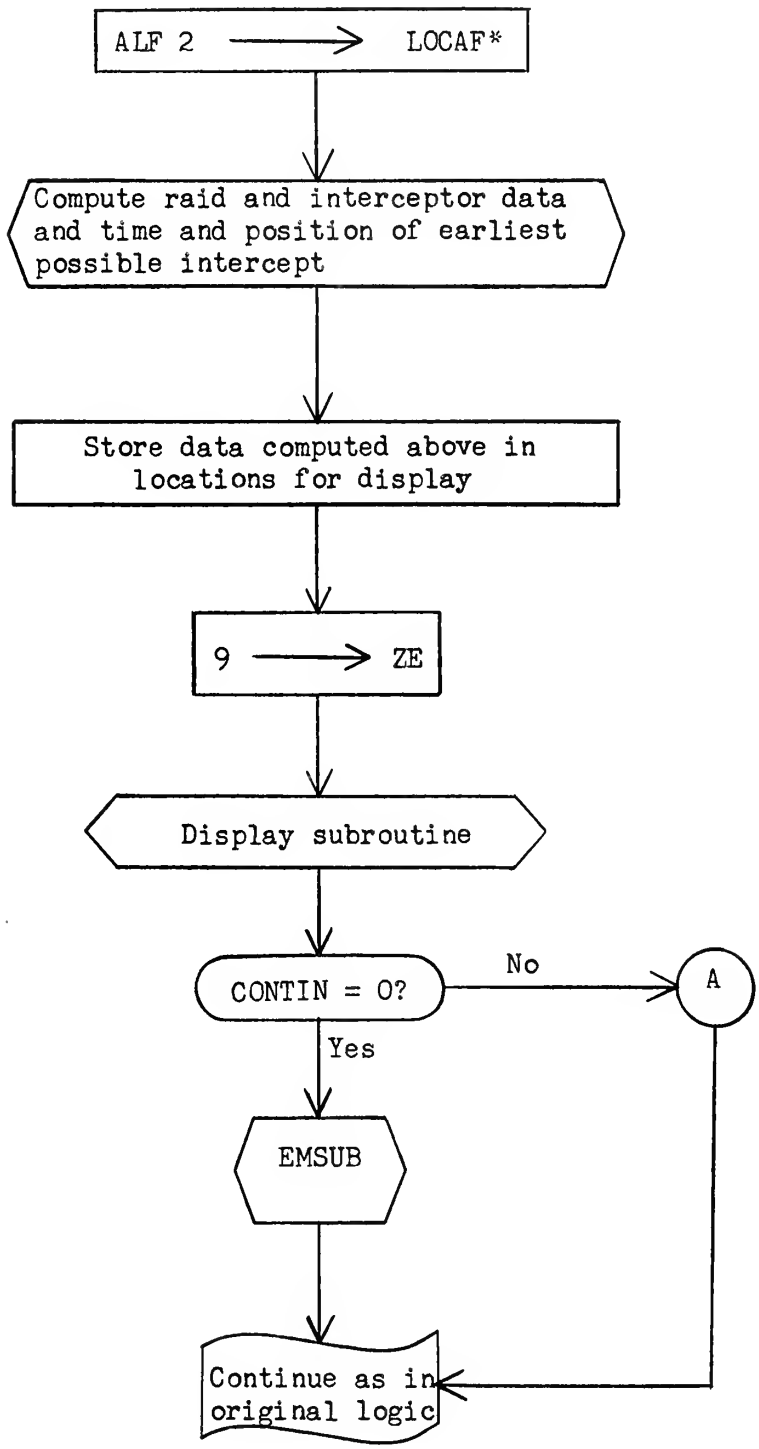
DATA TO BE DISPLAYED FOR A TYPE 13A EVENT

| | | |
|---|---|---|
| RAID LOCATION: X-coordinate (game coordinates) | | ______ |
| Y-coordinate (game coordinates) | | ______ |
| Time (game time) | | ______ |
| RAID COURSE | | ______ |
| RAID SPEED | | ______ |
| RAID SIZE | | ______ |
| NUMBER OF LIVE INTERCEPTORS ASSIGNED | | ______ |
| TIME OF FIRST POSSIBLE INTERCEPT (game time) | | ______ |
| POSITION OF FIRST POSSIBLE INTERCEPT: X-coordinate (game coordinates) | | ______ |
| Y-coordinate (game coordinates) | | ______ |
| NUMBER OF THIS MISSILE AREA | | ______ |
| NUMBER OF MISSILES AVAILABLE AT THIS SITE | | ______ |
| ECM CONDITION AT THIS MISSILE SITE** | | ______ |
| IS A TYPE 13B EVENT IN STORE? (0 = yes, 1 = no) | | ______ |
| TIME OF THAT TYPE 13B EVENT (game time) | | ______ |
| IS THAT TYPE 13B EVENT TO BE CANCELLED?*(0 = yes, 1 = no) | | ______ |
| IS A NEW TYPE 13A EVENT TO BE SCHEDULED?*(0 = yes, 1 = no) | | ______ |
| IS A NEW TYPE 13B EVENT TO BE SCHEDULED?*(0 = yes, 1 = no) | | ______ |
| TIME INTERVAL*(This interval added to present game time gives time of new Type 13A or 13B Events if either is to be stored) | | ______ |

* Indicates an item to be inputted by the player

** See flow chart for Event Type 4 for suggested code

MODIFICATION TO EVENT TYPE 13C
(Interceptors Cancelled from Raid)



DATA TO BE DISPLAYED FOR A TYPE 13C EVENT

GAME TIME ______

RAID LOCATION: X-coordinate (game coordinates) ______
Y-coordinate (game coordinates) ______

RAID COURSE ______

RAID SPEED ______

RAID SIZE ______

POSITION OF CLOSEST INTERCEPTOR:
X-coordinate (game coordinates) ______
Y-coordinate (game coordinates) ______

INTERCEPTOR COURSE ______

INTERCEPTOR SPEED ______

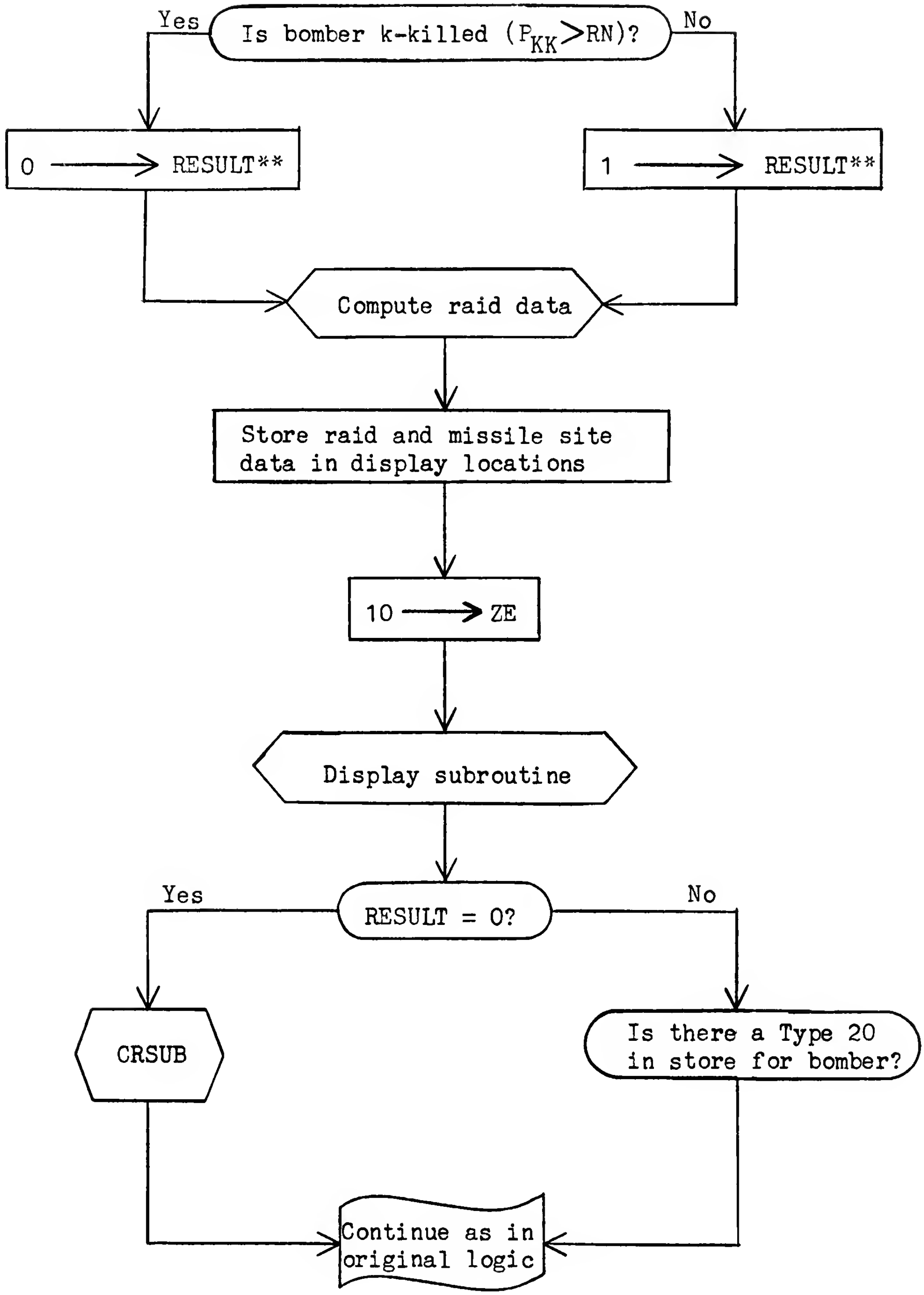
POSITION OF EARLIEST INTERCEPT:
X-coordinate (game coordinates) ______
Y-coordinate (game coordinates) ______

TIME OF EARLIEST INTERCEPT (game time) ______

CONTIN* ______

* Indicates an item to be inputted by the player

MODIFICATION TO EVENT TYPE 14A
(Missile Intercept)

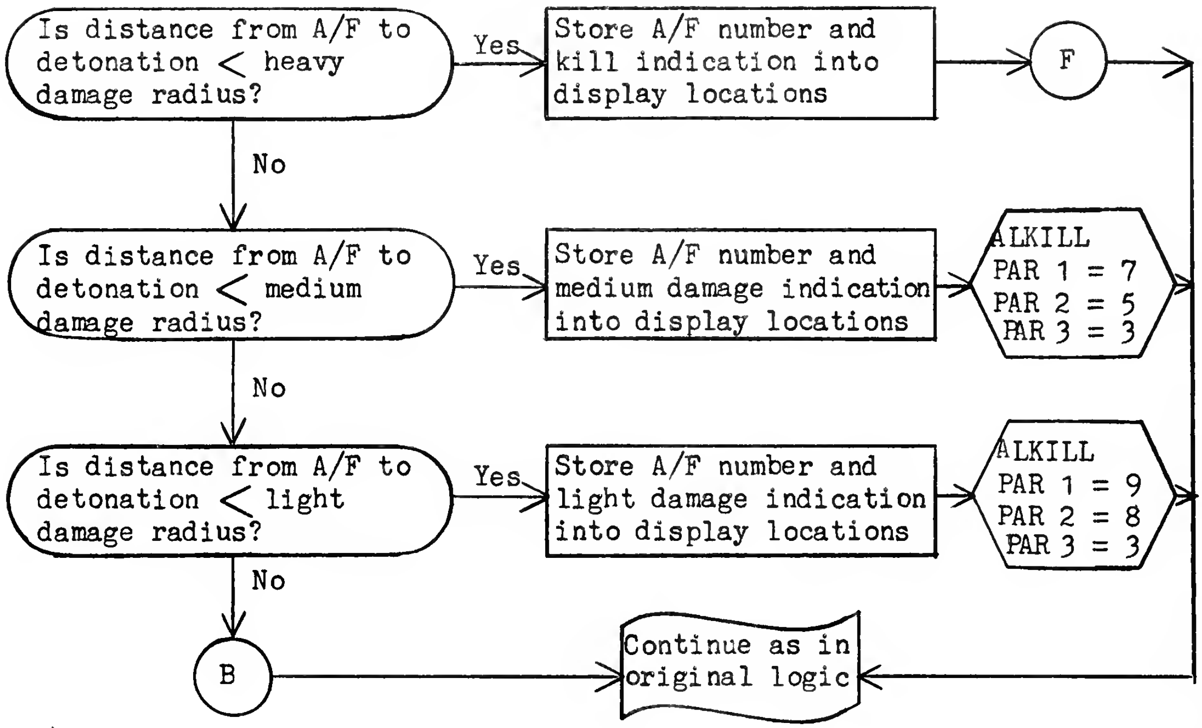


DATA TO BE DISPLAYED FOR A TYPE 14A EVENT

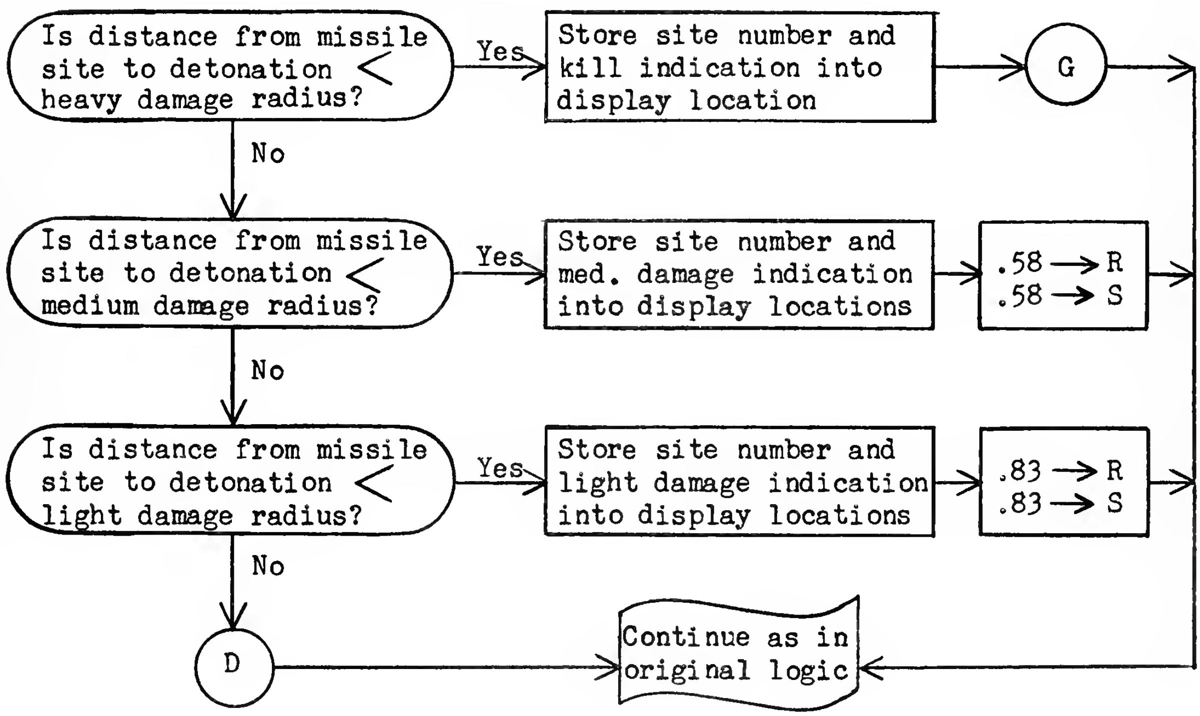
| | | |
|---|---|---|
| RAID LOCATION: X-coordinate (game coordinates) | | ______ |
| Y-coordinate (game coordinates) | | ______ |
| Time (game time) | | ______ |
| RAID COURSE | | ______ |
| RAID SPEED | | ______ |
| STATUS OF BOMBER (0 = killed, 1 = damaged) | | ______ |
| NUMBER OF TARGETS REMAINING IN THIS RAID | | ______ |
| NUMBER OF MISSILE SITE | | ______ |
| NUMBER OF MISSILES REMAINING AT THE SITE | | ______ |

MODIFICATION TO EVENT TYPE 17
(Weapon Detonation)

Airfield Section of Event

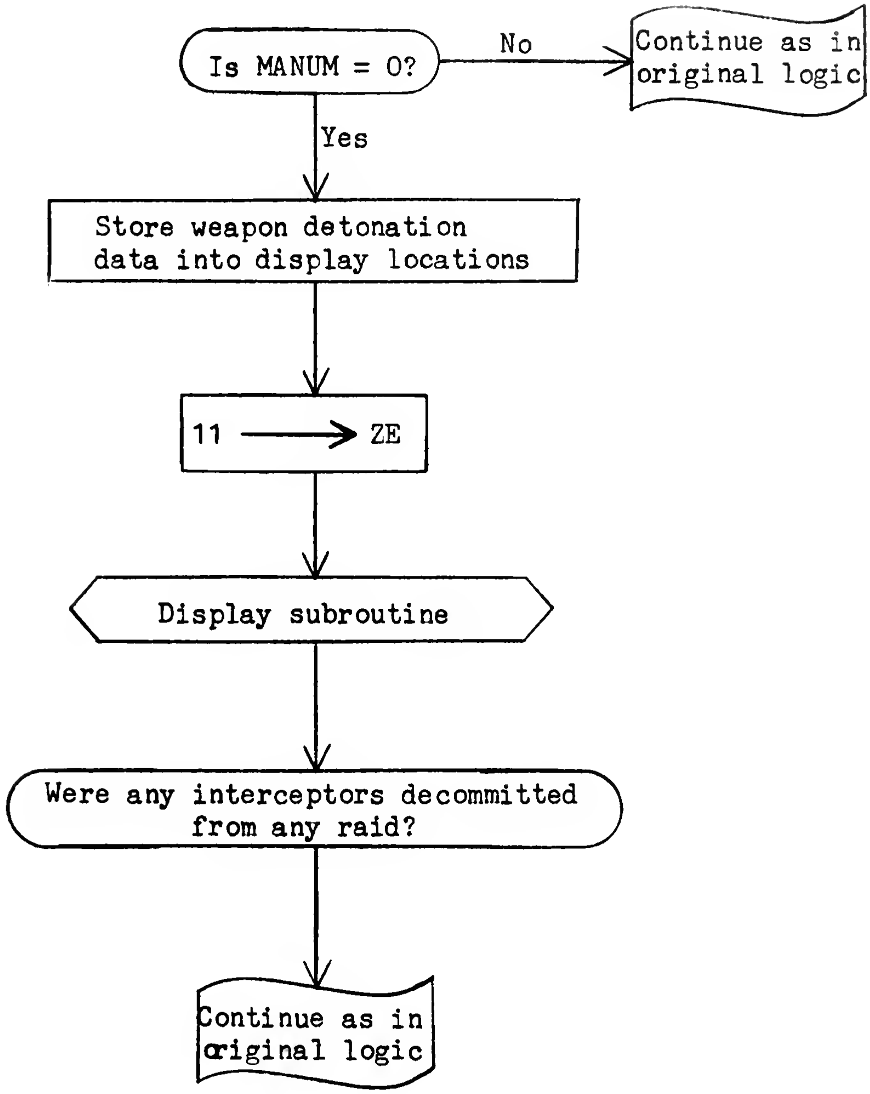



Missile Site Section of Event



MODIFICATION TO EVENT TYPE 17
(continued)

Final Section



DATA TO BE DISPLAYED FOR A TYPE 17 EVENT

LOCATION OF WEAPON DETONATION:

| | |
|---|---|
| X-coordinate | (game coordinates) |
| Y-coordinate | (game coordinates) |
| Time | (game time) |

RESULTS:

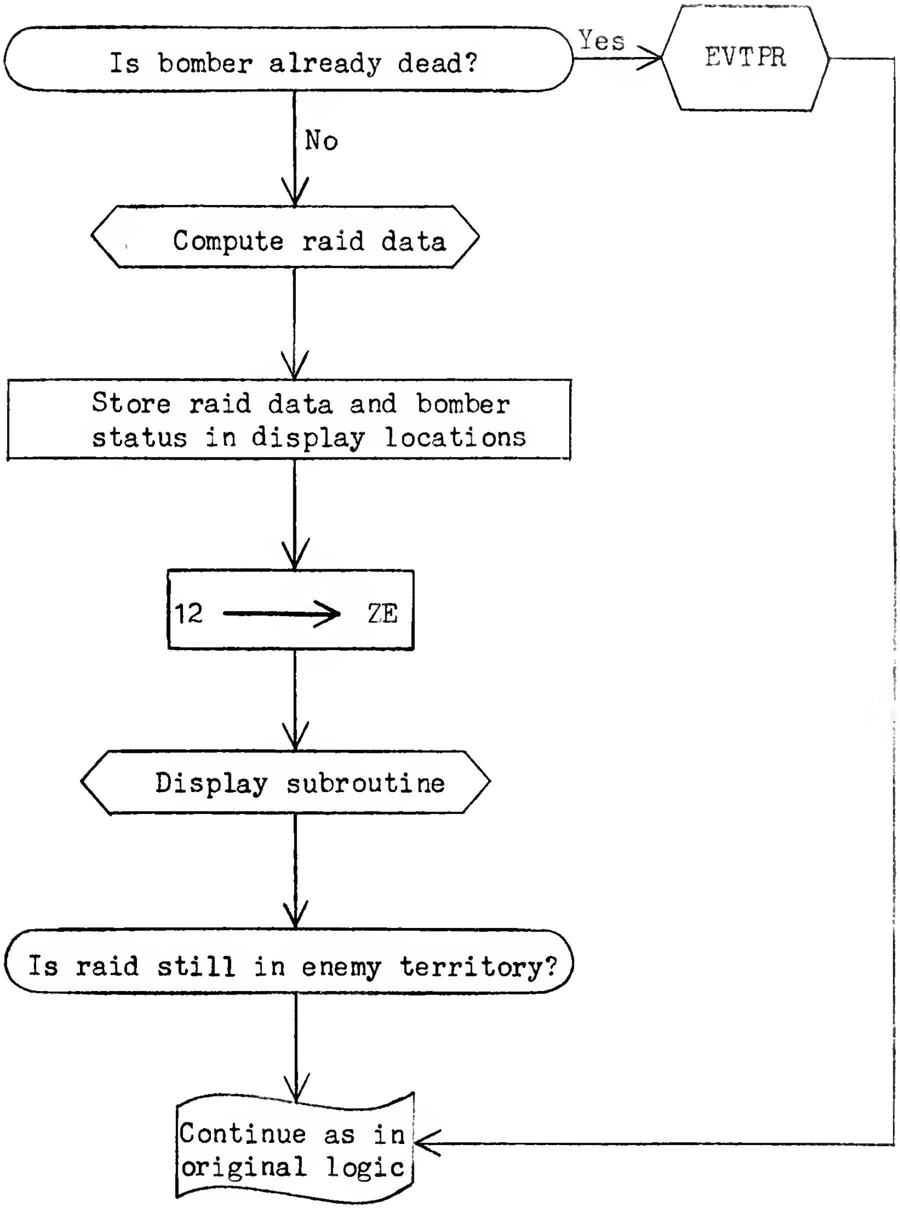
| A/F NUMBER | TYPE DAMAGE |
|---|---|
| ______ | ______ |
| ______ | ______ |
| ______ | ______ |

(etc., listing all damaged or destroyed airfields)

| MISSILE SITE NUMBER | TYPE DAMAGE |
|---|---|
| ______ | ______ |
| ______ | ______ |
| ______ | ______ |

(etc., listing all damaged or destroyed missile sites)

MODIFICATION TO EVENT TYPE 20
(Damaged Bomber Dies)



DATA TO BE DISPLAYED FOR A TYPE 20 EVENT

| | | | |
|---|---|---|---|
| RAID LOCATION: | X-coordinate | (game coordinates) | ______ |
| | Y-coordinate | (game coordinates) | ______ |
| | Time | (game time) | ______ |
| RAID COURSE | | | ______ |
| RAID SPEED | | | ______ |
| INDICATE THAT A DAMAGED BOMBER DIED | | | |
| RAID SIZE | | | ______ |

# APPENDIX III

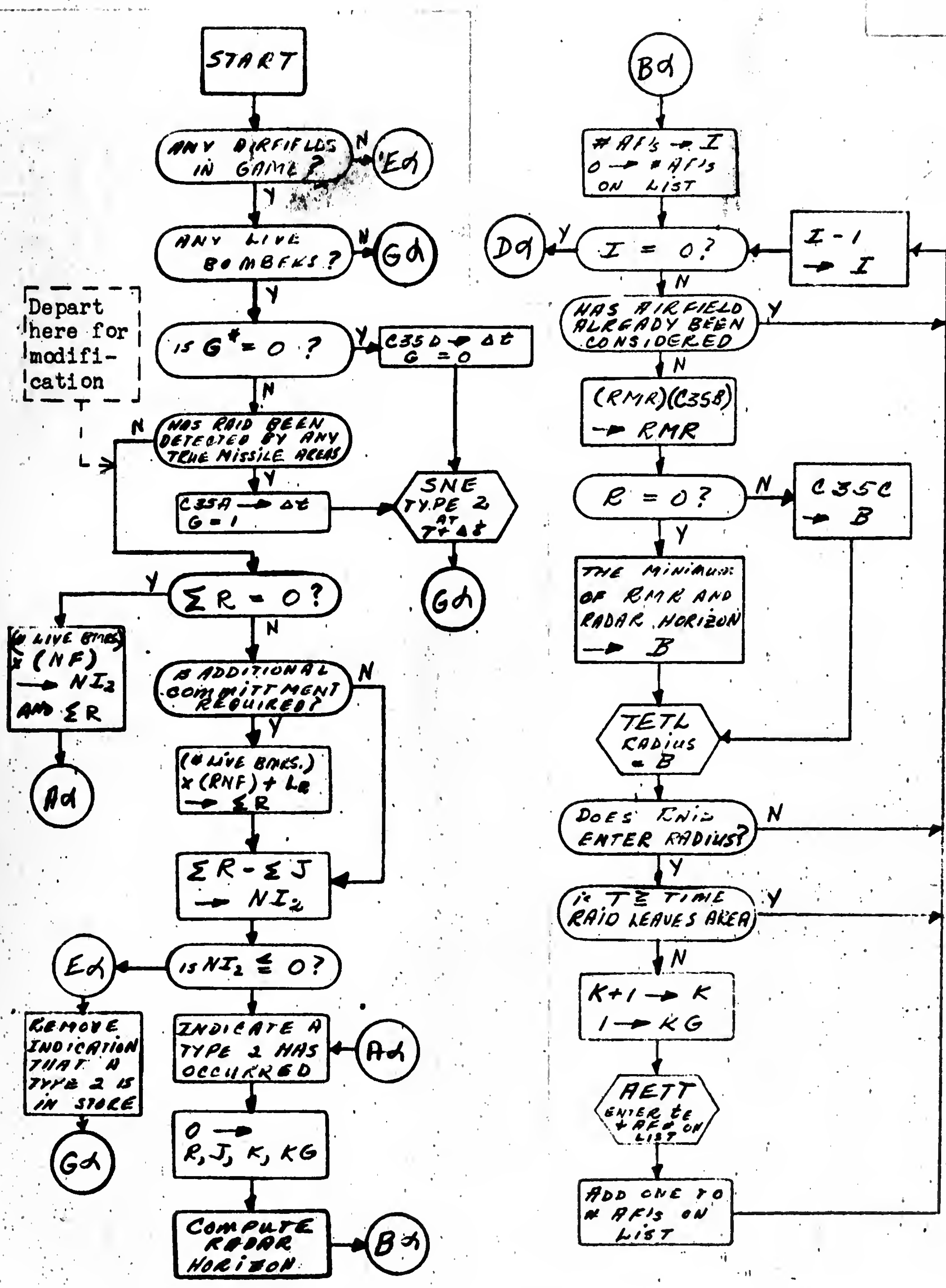
## ORIGINAL FLOW CHARTS OF MODIFIED EVENTS

This Appendix contains the original flow charts of the events or subroutines for which this thesis has proposed modifications. Arrows have been inserted in the original flow charts and labelled to show where the logic flow should deviate to follow the proposed modifications and where it would return to the original.

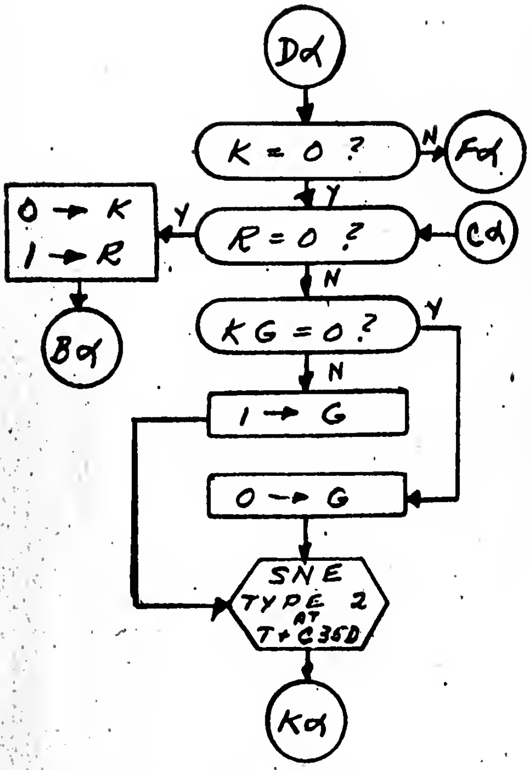
The flow charts provided here were reproduced from the documentation on "The Naval Air Strike Model - Mod Zero" which is listed in the bibliography. The reader is referred to this documentation for flow charts of other events and subroutines.

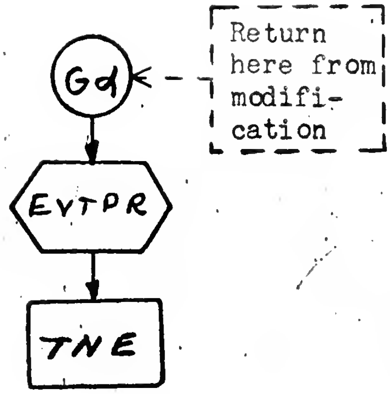
# TYPE 2 (IAS)

INTERCEPTORS ASSIGNED TO RAID



# TYPE 2
(CONT'D)




RMR - RADAR MAXIMUM RANGE

G - IS AN INDICATOR OF WHETHER OR NOT A RAID'S PATH IS IN COMMITTMENT RANGE OF ANY AIRFIELD. 0 = NO, 1 = YES

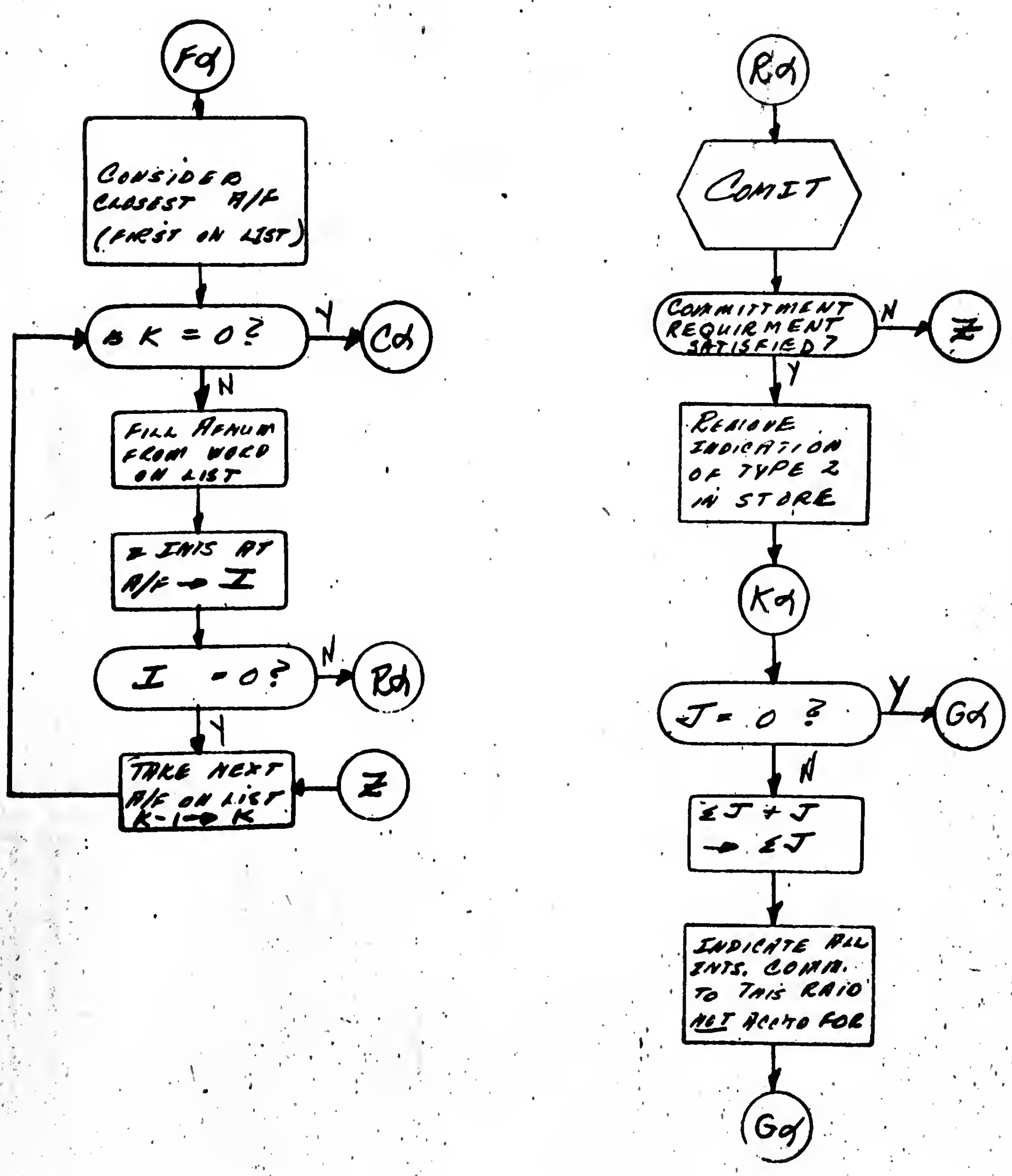
ΣR - IS THE CUMALATIVE NUMBER OF INTERCEPTORS THAT SHOULD BE COMMITTED TO THIS RAID THROUGHOUT THE GAME.

$L_R$ - IS THE NUMBER OF INTERCEPTORS WHICH HAVE BEEN BUT, ARE NO LONGER COMMITTED TO THIS RAID

ΣJ - IS THE CUMULATIVE NUMBER OF INTERCEPTORS COMMITTED TO THIS RAID, DURING THE GAME.

$NI_2$ - IS THE NUMBER THAT SHOULD BE COMMITTED BY THIS EVENT.

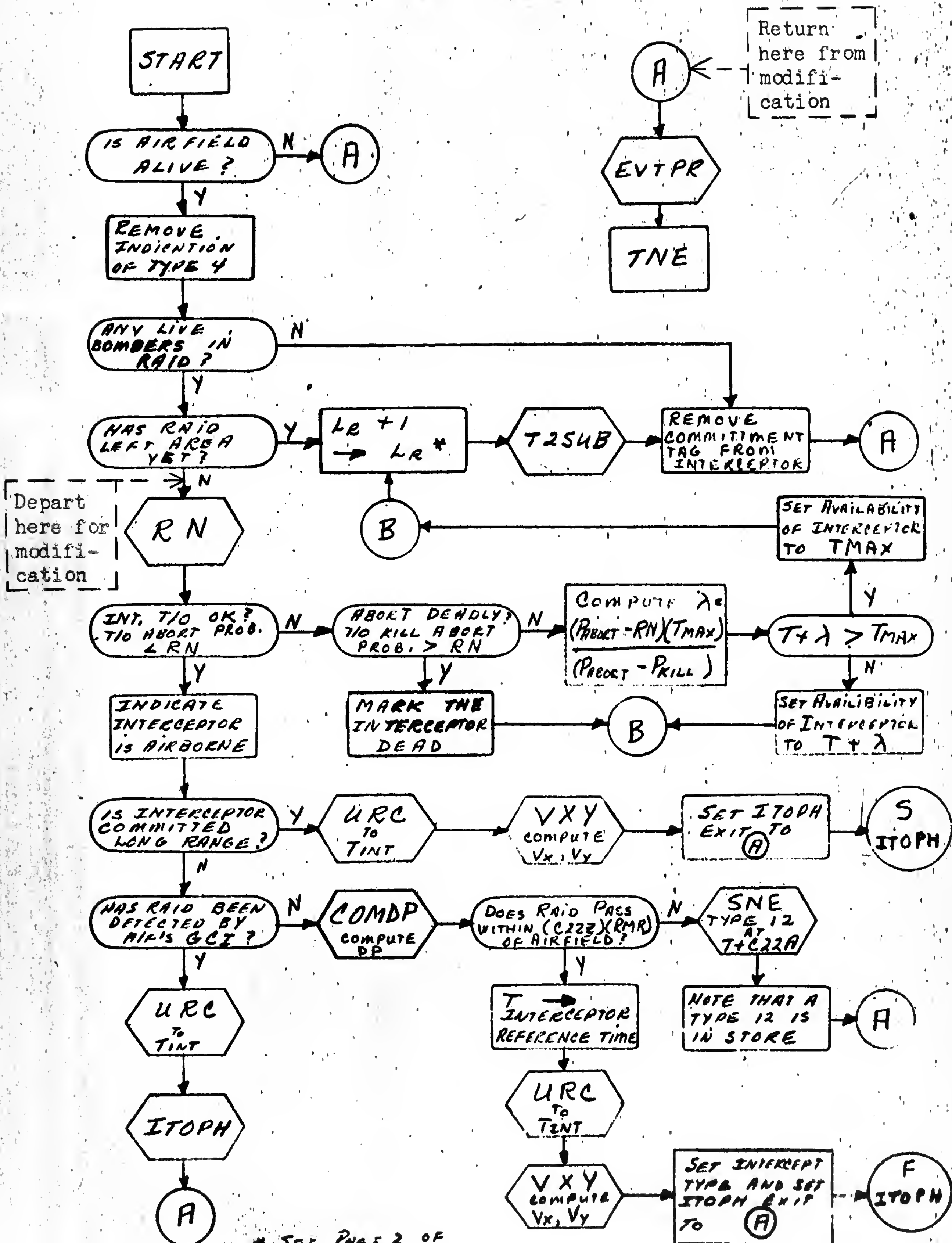
R - INDICATES LONG OR SHORT RANGE COMMITTMENT, 0= SHORT, 1= LONG
J - IS COUNTER OF ALL INTERCEPTORS COMMITTED BY THIS EVENT.
K - IS COUNTER OF ALL AIRFIELDS IN COMMITTMENT RANGE OF RAID PATH.
KG - IS CHECK IF K EVER > ZERO


Fα
CONSIDER CLOSEST A/F (FIRST ON LIST)
ΔK = 0?
Y
Cα
N
FILL AFNUM FROM WORD ON LIST
Σ INTS AT A/F → I
I = 0?
N
Rα
Y
TAKE NEXT A/F ON LIST K-1 → K
Z
Rα
COMIT
COMMITTMENT REQUIREMENT SATISFIED?
N
Z
Y
REMOVE INDICATION OF TYPE 2 IN STORE
Kα
J = 0 ?
Y
Gα
N
ΣJ + J → ΣJ
INDICATE ALL INTS. COMM. TO THIS RAID NOT ACC'TD FOR
Gα

7/13/61

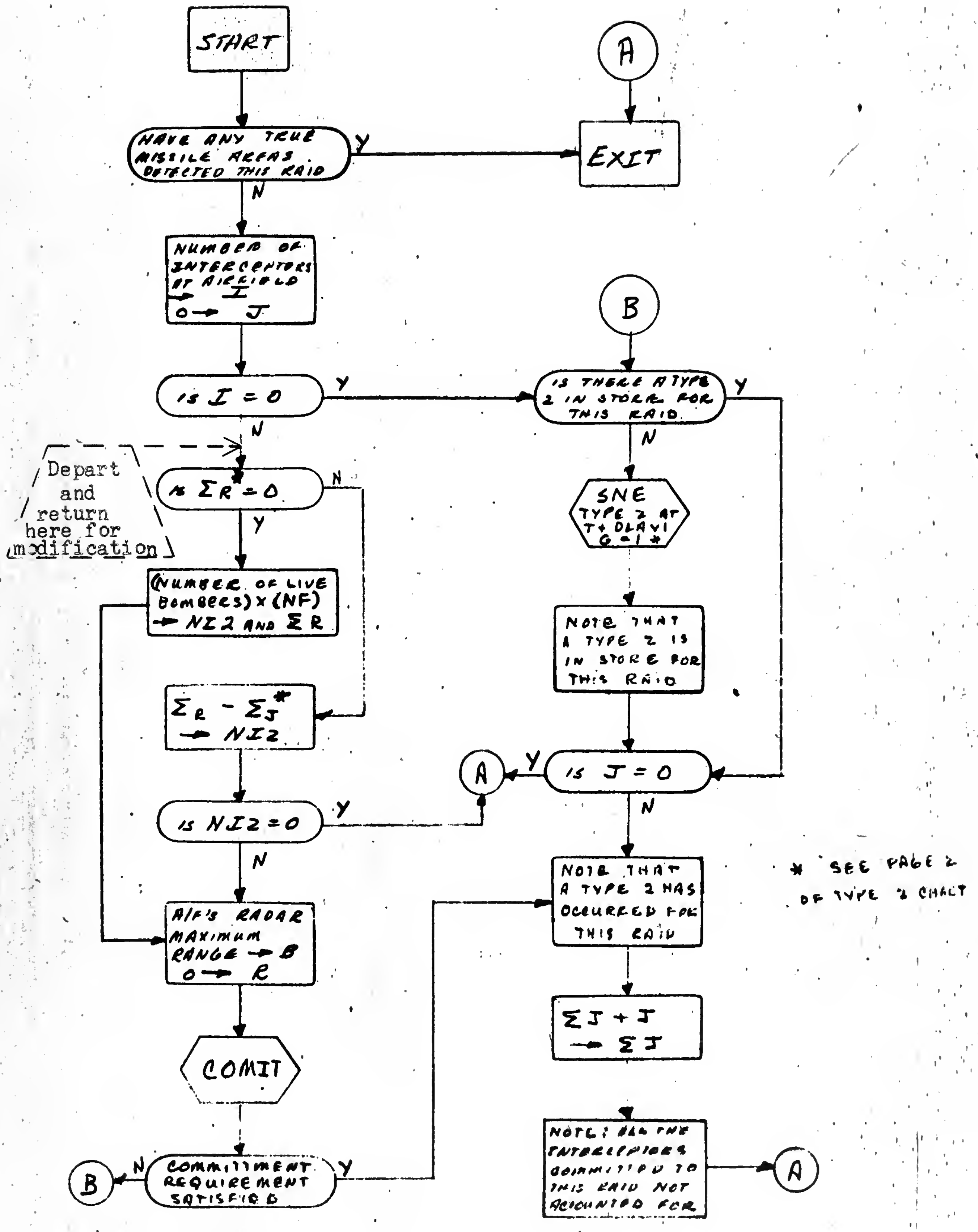
# TYPE 4 (ITO)

## INTERCEPTOR TAKE-OFF

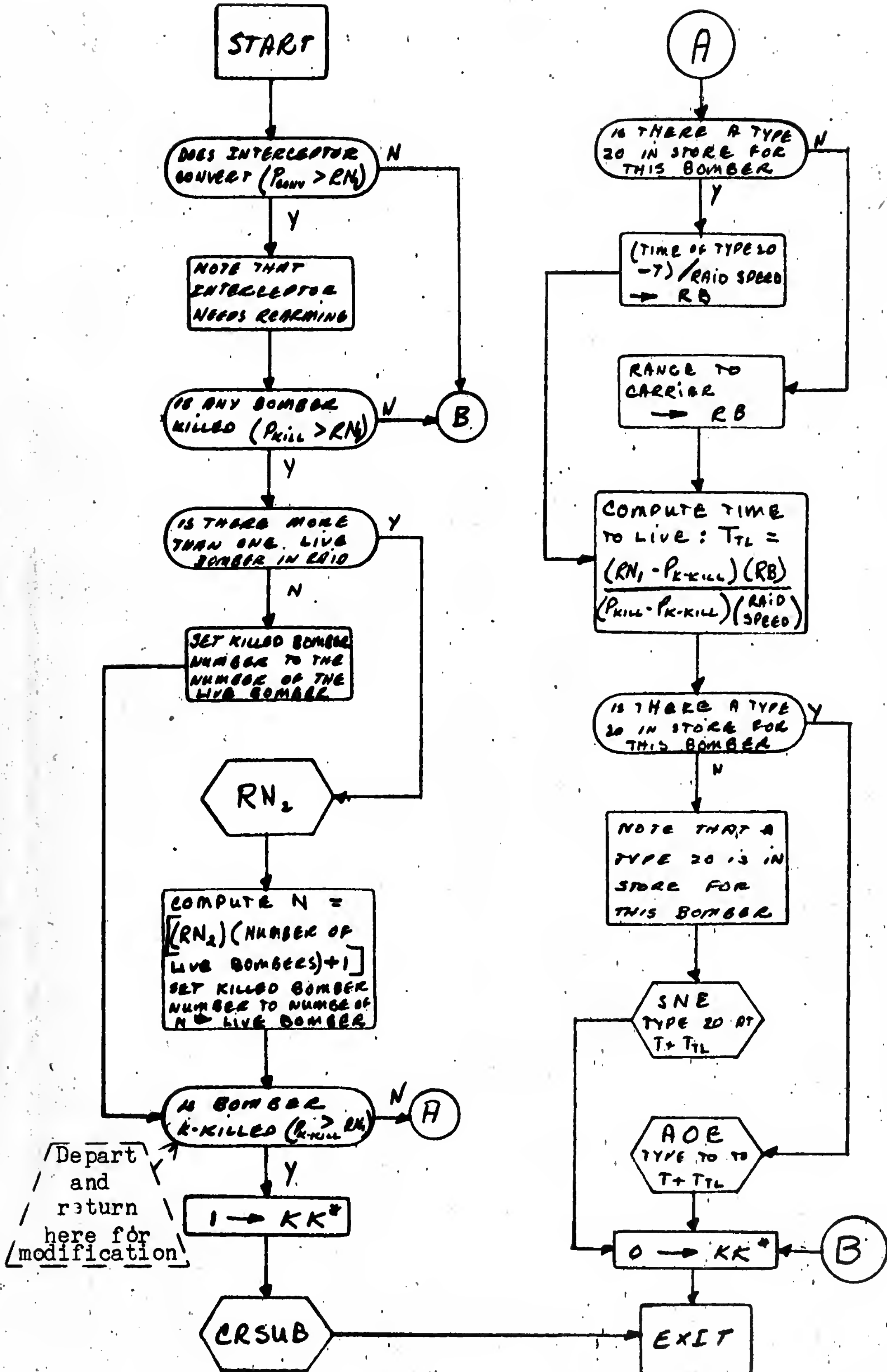



* SEE PAGE 2 OF TYPE 2 CHART FOR DEFINITION

# PISUB


START
HAVE ANY TRUE MISSILE AREAS DETECTED THIS RAID
Y
EXIT
A
N
NUMBER OF INTERCEPTORS AT AIRFIELD → I
0 → J
B
IS I = 0
Y
IS THERE A TYPE 2 IN STORE FOR THIS RAID
Y
N
Depart and return here for modification
IS ΣR* = 0
N
Y
SNE TYPE 2 AT T + DLAY1 G = 1 *
(NUMBER OF LIVE BOMBERS) x (NF) → NI2 AND ΣR
NOTE THAT A TYPE 2 IS IN STORE FOR THIS RAID
ΣR - ΣJ* → NI2
A
Y
IS J = 0
IS NI2 = 0
Y
N
N
NOTE THAT A TYPE 2 HAS OCCURRED FOR THIS RAID
* SEE PAGE 2 OF TYPE 2 CHART
A/F'S RADAR MAXIMUM RANGE → B
0 → R
ΣJ + J → ΣJ
COMIT
NOTE: ALL THE INTERCEPTORS COMMITTED TO THIS RAID NOT ACCOUNTED FOR
A
B
N
COMMITMENT REQUIREMENT SATISFIED
Y

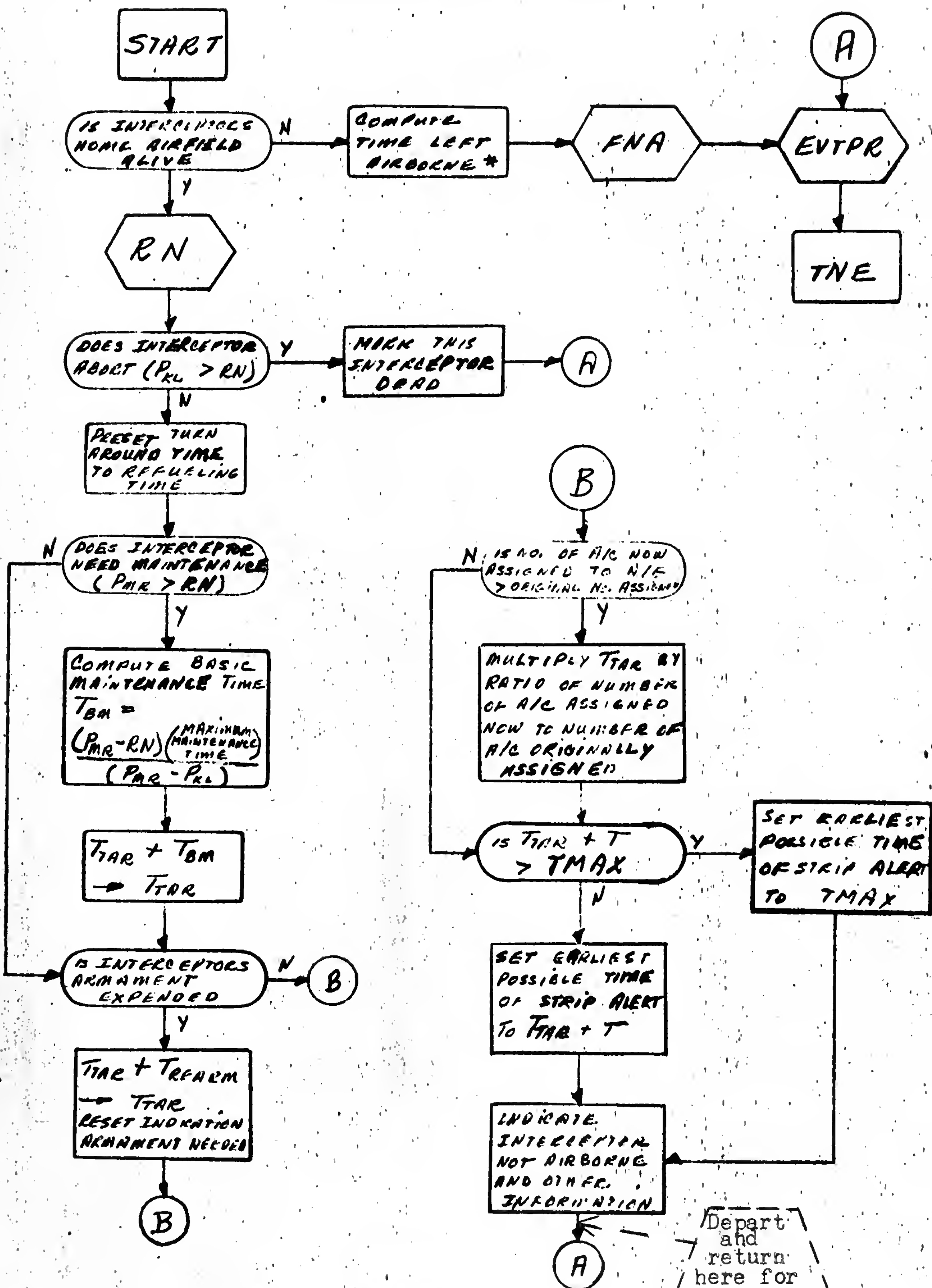
T6B




* SET FOR T6C SUBROUTINE

## TYPE II (ILA)

### INTERCEPTOR LANDING

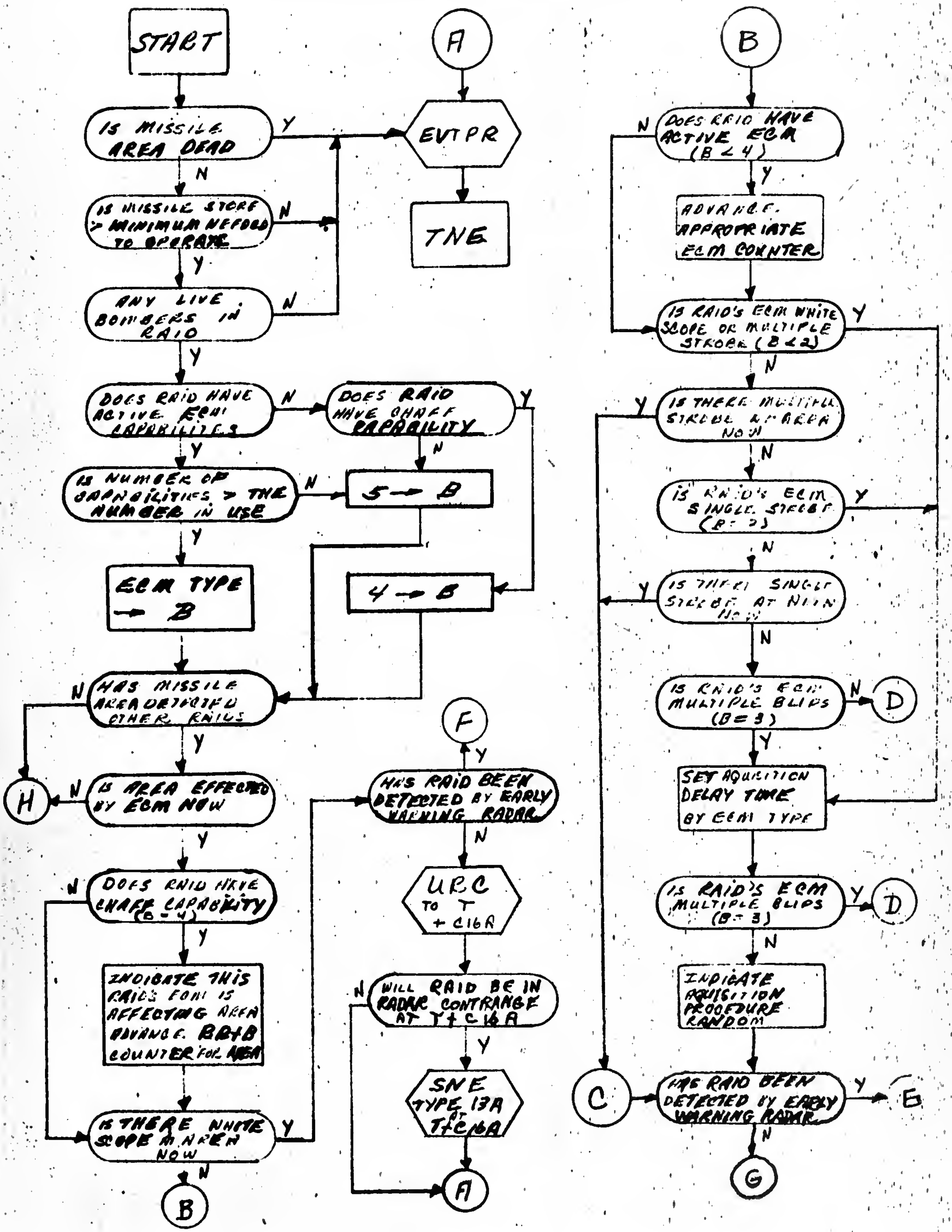


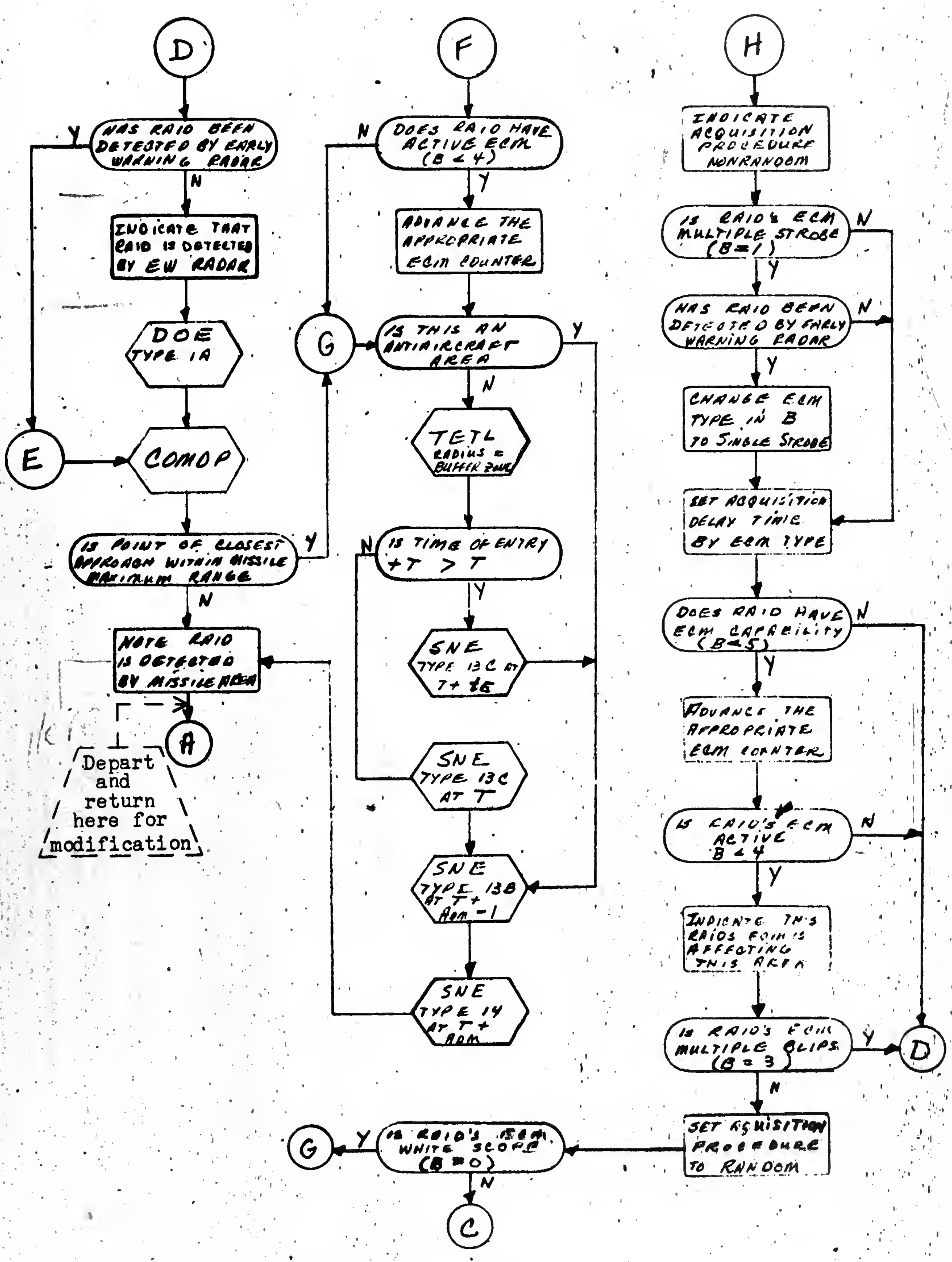

* TIME LEFT AIRBORNE

= MAX. TIME AIRBORNE + TAKE-OFF TIME − T

# TYPE 13A (RDM)

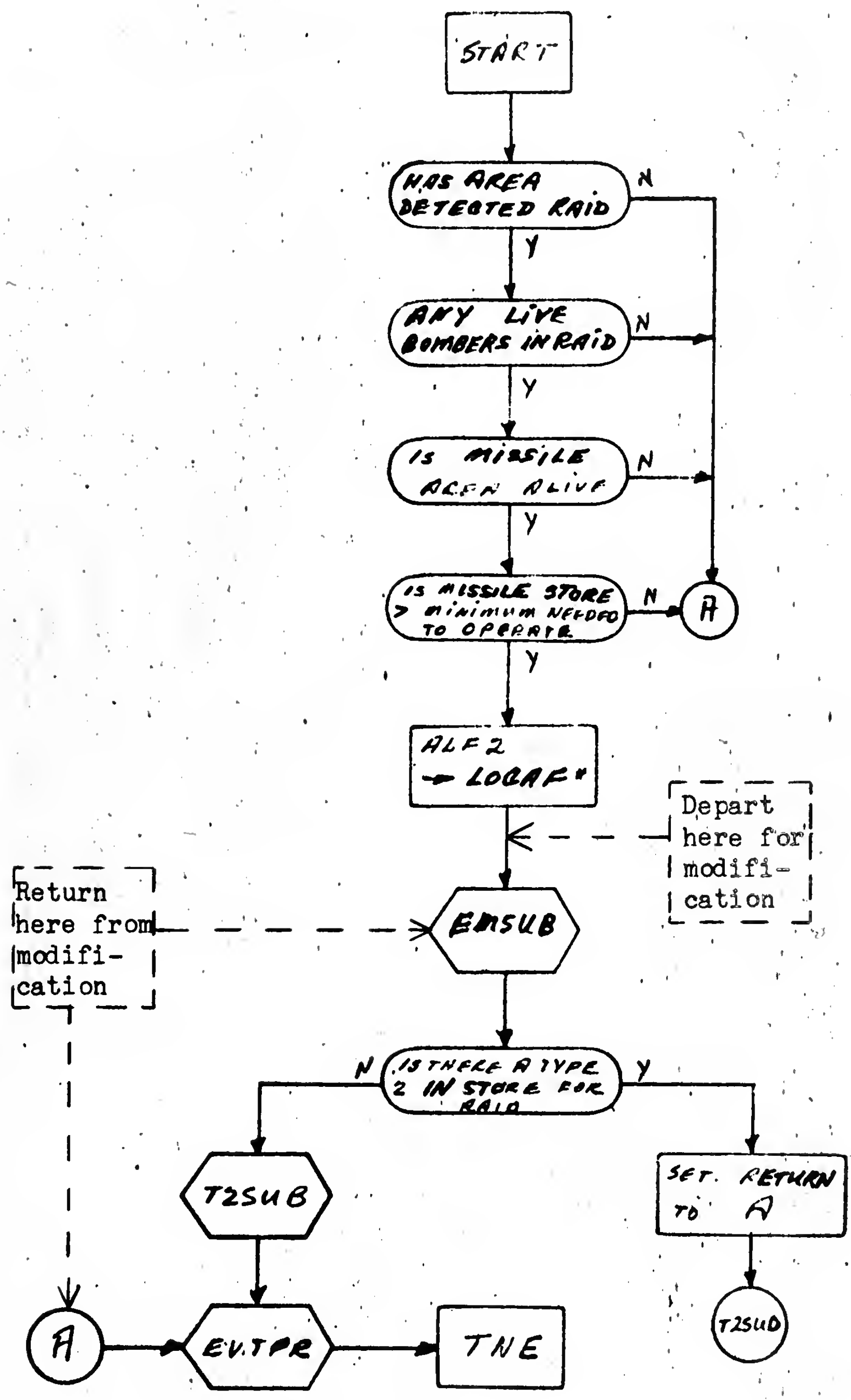
## RAID DETECTION BY MISSILE AREA




D
HAS RAID BEEN DETECTED BY EARLY WARNING RADAR
Y
N
INDICATE THAT RAID IS DETECTED BY EW RADAR
DOE TYPE 1A
E
COMOP
IS POINT OF CLOSEST APPROACH WITHIN MISSILE MAXIMUM RANGE
Y
N
NOTE RAID IS DETECTED BY MISSILE AREA
A
Depart and return here for modification
F
DOES RAID HAVE ACTIVE ECM (B < 4)
N
Y
ADVANCE THE APPROPRIATE ECM COUNTER
G
IS THIS AN ANTIAIRCRAFT AREA
Y
N
TETL RADIUS = BUFFER ZONE
IS TIME OF ENTRY +T > T
N
Y
SNE TYPE 13C AT T+ tE
SNE TYPE 13C AT T
SNE TYPE 13B AT T+ Aem -1
SNE TYPE 14 AT T+ Aem
H
INDICATE ACQUISITION PROCEDURE NONRANDOM
IS RAID'S ECM MULTIPLE STROBE (B=1)
N
Y
HAS RAID BEEN DETECTED BY EARLY WARNING RADAR
N
Y
CHANGE ECM TYPE IN B TO SINGLE STROBE
SET ACQUISITION DELAY TIME BY ECM TYPE
DOES RAID HAVE ECM CAPABILITY (B≤5)
N
Y
ADVANCE THE APPROPRIATE ECM COUNTER
IS RAID'S ECM ACTIVE B < 4
N
Y
INDICATE THIS RAIDS ECM IS AFFECTING THIS AREA
IS RAID'S ECM MULTIPLE BLIPS (B = 3)
Y
D
N
SET ACQUISITION PROCEDURE TO RANDOM
IS RAID'S ECM WHITE SCOPE (B = 0)
Y
G
N
C

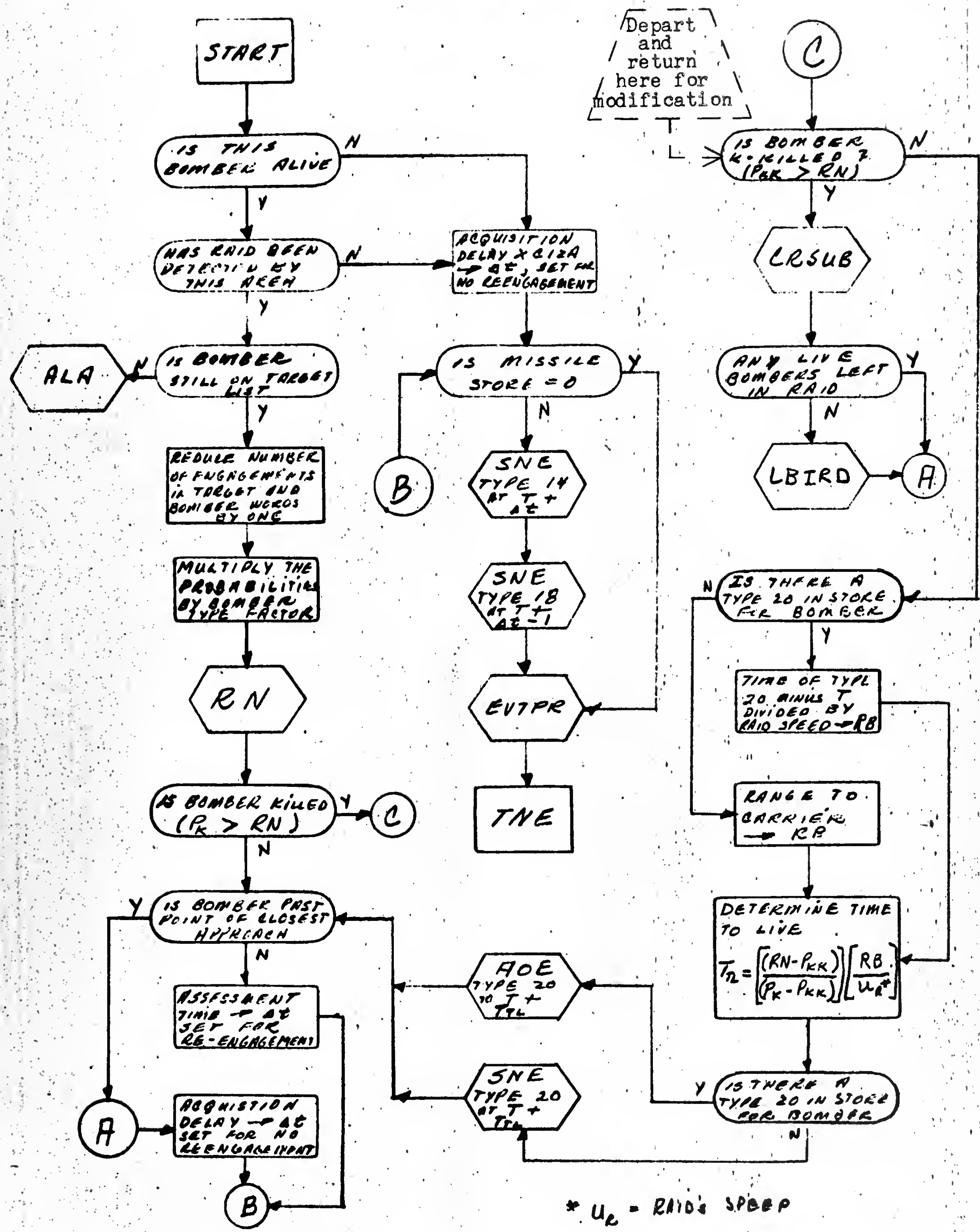
TYPE 13C (ICA)

INTERCEPTORS CANCELLED FROM RAID



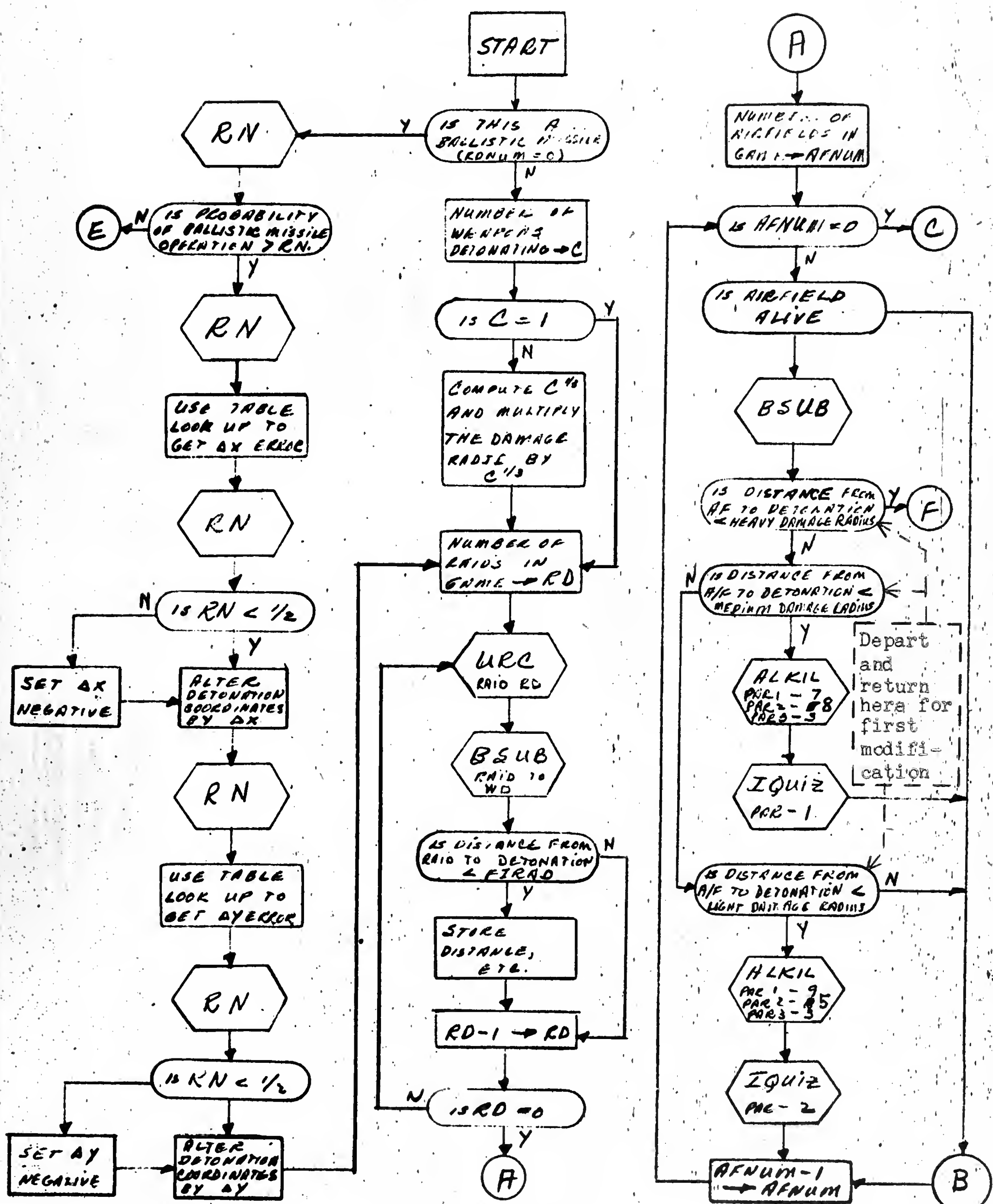

* TEMPORARY LOCATION USED IN EMSUB

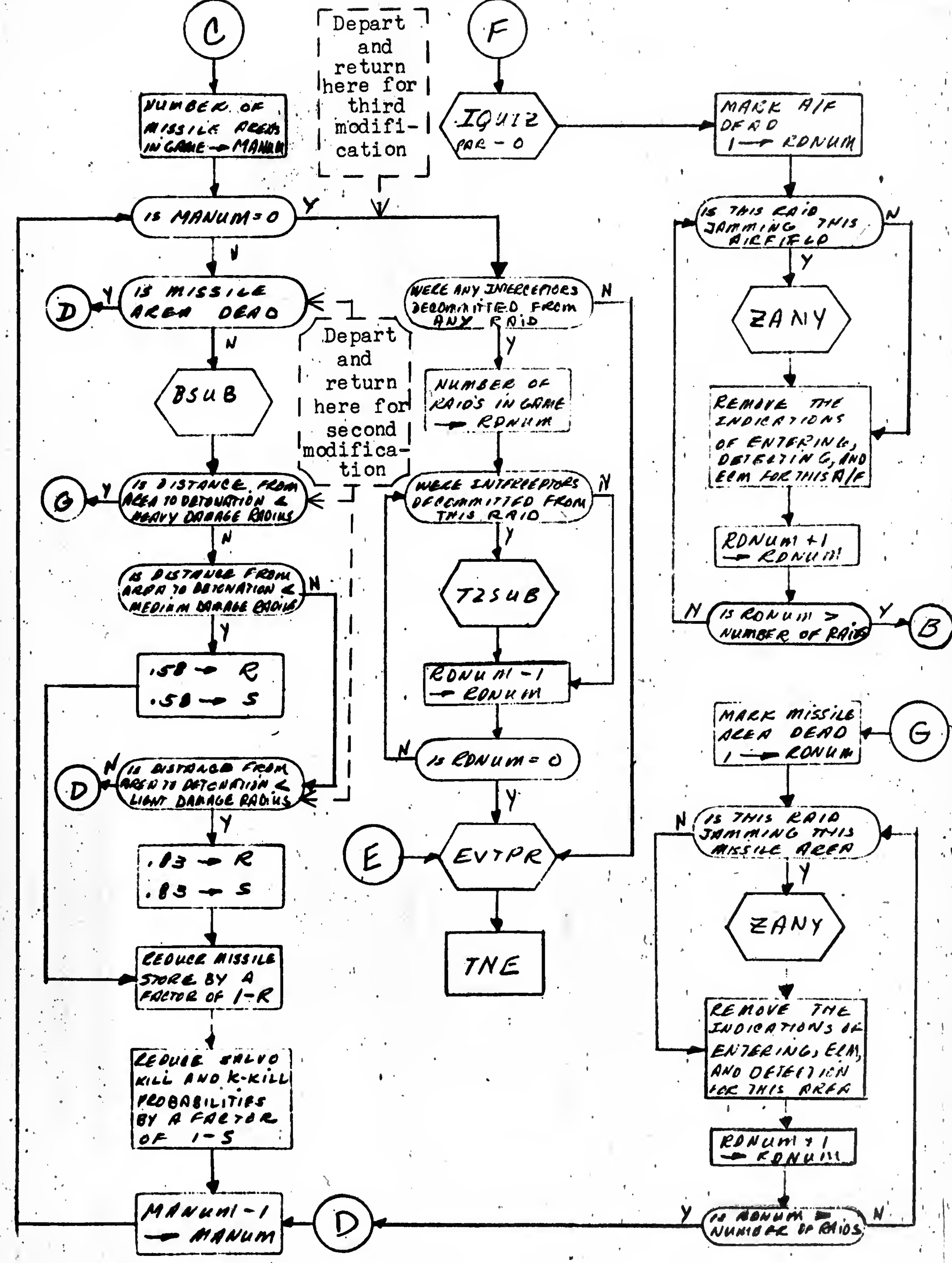
## TYPE 14A (MII)
## MISSILE INTERCEPT




* $u_R$ = RAID'S SPEED

TYPE 17 (WDD)
WEAPON DETONATION




C
NUMBER OF MISSILE AREAS IN GAME → MANUM
Depart and return here for third modification
F
IQUIZ PAR - 0
MARK A/F DEAD 1 → RDNUM
IS MANUM = 0
IS THIS RAID JAMMING THIS AIRFIELD
IS MISSILE AREA DEAD
WERE ANY INTERCEPTORS DECOMMITTED FROM ANY RAID
ZANY
D
Depart and return here for second modification
BSUB
NUMBER OF RAIDS IN GAME → RDNUM
REMOVE THE INDICATIONS OF ENTERING, DETECTING, AND ECM FOR THIS A/F
G
IS DISTANCE FROM AREA TO DETONATION < HEAVY DAMAGE RADIUS
WERE INTERCEPTORS DECOMMITTED FROM THIS RAID
RDNUM + 1 → RDNUM
IS DISTANCE FROM AREA TO DETONATION < MEDIUM DAMAGE RADIUS
T2SUB
IS RDNUM > NUMBER OF RAIDS
B
.58 → R
.58 → S
RDNUM - 1 → RDNUM
MARK MISSILE AREA DEAD 1 → RDNUM
IS DISTANCE FROM AREA TO DETONATION < LIGHT DAMAGE RADIUS
IS RDNUM = 0
IS THIS RAID JAMMING THIS MISSILE AREA
E
EVTPR
.83 → R
.83 → S
ZANY
REDUCE MISSILE STORE BY A FACTOR OF 1-R
TNE
REMOVE THE INDICATIONS OF ENTERING, ECM, AND DETECTION FOR THIS AREA
REDUCE SALVO KILL AND K-KILL PROBABILITIES BY A FACTOR OF 1-S
RDNUM + 1 → RDNUM
MANUM - 1 → MANUM
IS RDNUM = NUMBER OF RAIDS
Y
N

## TYPE 20 (DBD)

DAMAGED BOMBER DIES