GNC ASC 2102
Ascent Guidance, Navigation, and Flight Control Workbook

Advanced Training Series

For Training Purposes Only
Crew Training and Procedures Division
Flight Training Branch

May 15, 1979

NASA
National Aeronautics and
Space Administration

Lyndon B. Johnson Space Center
Houston, Texas

GNC

FOREWORD

DEVELOPER Jenifer Herrman — MAIL CODE DG6

BRANCH Flight Training Branch — TELEPHONE 483-4637

LOCATION BLDG 4 /ROOM 2081A

This training material is part of a series of lectures, handouts, and workbooks for instruction on Avionics Systems

This training package should be studied before attending any classroom session on this material or taking any lesson for which this is a prerequisite.

The Avionics Systemslesson sequence chart shows where this lesson fits in the series, the prerequisites for each lesson, and the optimum presentation order. The lesson sequence charts are located in the Shuttle Training Program catalog.

This material is for training purposes only; it should not be used as a source of operational data.

This workbook is part of the advanced training series with data for specific examples taken from the STS- 1 Flight. This should not detract from its training value for later STS Flights.

The last two pages of this workbook are two forms:

(1) The "Lesson Critique Sheet" is used to obtain your comments on this workbook. Please fill it out and mail to the instructor when you have completed this workbook. (It need not be placed in an envelope nor be signed.)

(2) The "Training Report" applies to FOD personnel ONLY. It provides the information to enter this training in your training record. To receive credit when you have completed this workbook, please complete the "Training Report" and mail to DG

HOW TO USE THIS WORKBOOK

This book is organized into sections. You should proceed through the sections in the order they are presented. Within the section, however, you control your own progress. You should begin each section by reading the overview and the learning material.

If you are already quite familiar with the material presented, skip right to the exercise items. For other sections with which you are somewhat familiar, you may want to review the material, then go on to the exercise.

Before going on, you should always make certain that you understand the correct answers to each of the exercise items. The correct answers are provided immediately following the exercise pages.

A content bibliography outline may be provided at the end of the lesson for those persons who may desire more detailed data on a specific topic.

This workbook is provided for your use. You may make notes or write comments anywhere you like. You may retain the text for later reference.

The vertical bars in the right-hand margin, as shown on the right, indicate changes since the last revision of this document. The subsections that have changed are noted in the Contents by an asterisk.

CONTENTS

SECTION 1
INTRODUCTION

This workbook is a-stand-alone, self-instructing session on Shuttle ascent navigation, guidance, and flight control software. Also included in the workbook are chapters discussing software/hardware sequences applicable to ascent and trajectory displays available on the CRT. This workbook covers the portion of ascent ranging from prelift-off functions to ET separation.

The workbook is designed to allow the reader to progress at a comfortable pace and concentrate on areas of interest or need.

Exercises are provided to aid the student in recognizing his/her extent of knowledge in a given area.

The book is intended to reflect the state of the art at the publication date.

SECTION 2
OVERVIEW

For a vehicle such as the Space Shuttle to be able to "fly" automatically, (or be "flown" manually) an abundance of information must be provided via the onboard computers (GPC's) and vehicle sensors (i.e., AA's, IMU's, RGA's, etc). In order to manage this abundance of information, the computer's applications software is conveniently divided into application programs. Examples of application programs are: guidance, navigation, flight control, redundancy management, and sequencing.

The navigation software provides to users (i.e., guidance, flight control, crew) the current vehicle state (i.e., position and velocity) along with related information such as altitude, angle of attack, range, bearing, etc. Guidance is the software that contains predetermined (I-loaded) targeting data. Each computation cycle, guidance determines if the vehicle is "on course" and computes course corrections commands if it is not.

Flight control is responsible for converting these course correction commands into vehicle effector (i.e. engine bells, elevons) movements so as to put the vehicle back "on course."

Figure 2-1 is an overview of Guidance Navigation and Flight Control and is meant to give the reader a "birdseye view" of GN&C before getting into the details of the software.

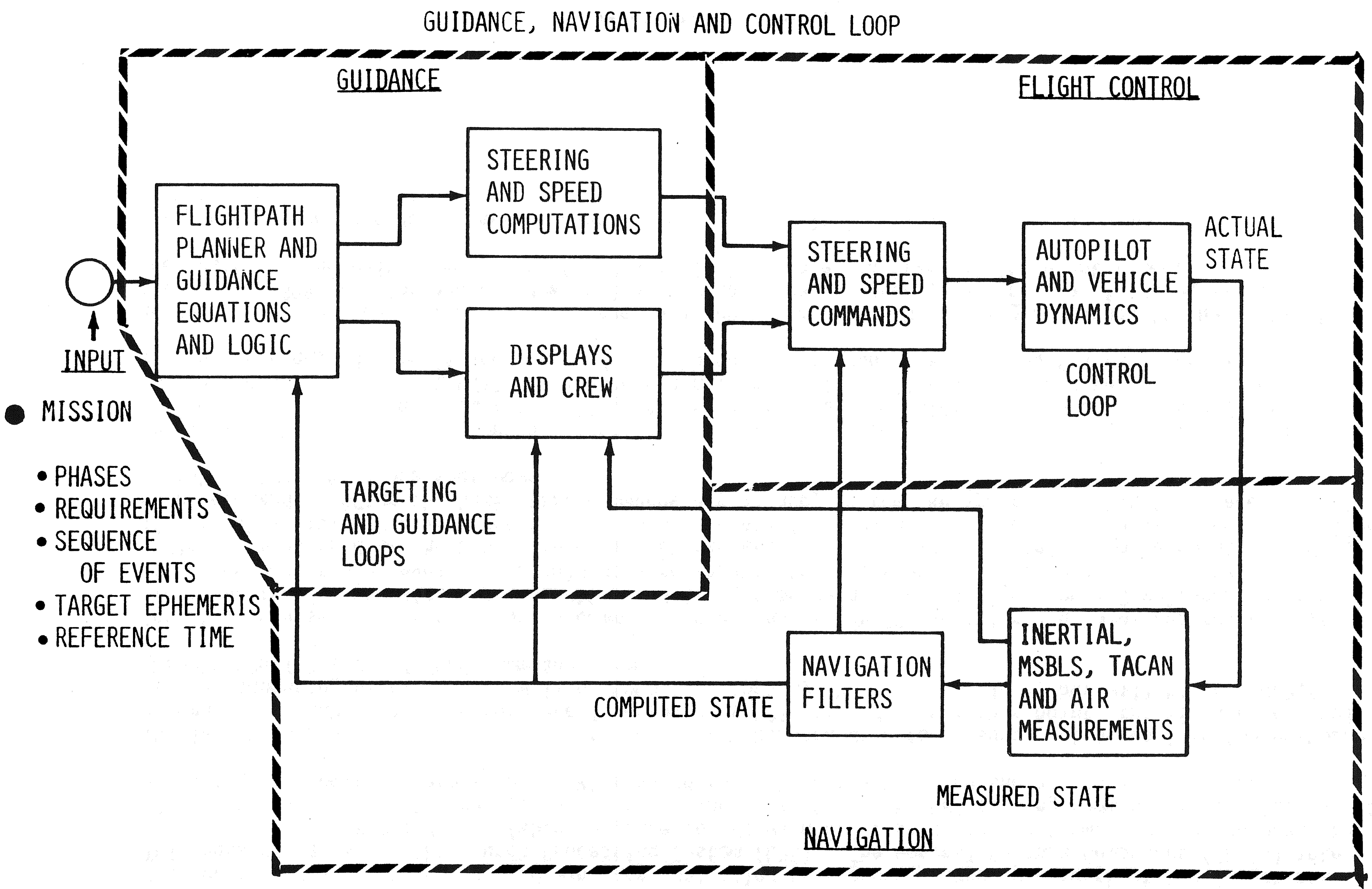


Figure 2-1.- Guidance, navigation, and control loop.

2.1 PRELAUNCH FUNCTIONS

Prelaunch functions, prior to T-20 min, are controlled by a ground computer network at the launch site. This network is called the Launch Processing System (LPS). The General Purpose Computers (GPC's) interact with the LPS from T-20 min (when they are loaded with OPS 1 software and formed into a Redundant Set (RS)) until lift-off. Basically, this interaction consists of the LPS sending commands to the RS, the RS executing the commands, and then the RS responding back to the LPS when the action is completed.

The launch countdown is controlled by the LPS until 25 seconds before launch, at which time the on-board automatic RS launch sequence software is enabled by LPS command. From this point, the RS takes control of the sequencing of events and will perform functions by the on-board clock, but will honor "HOLD," "RESUME COUNT," and "RECYCLE" commands from LPS.

The RS launch sequence sets flags to command the arming of the SRB ignition and hold down release system PIC's and the T-0 umbilical release PIC's. After a time delay, the SRB ignition PIC voltages are monitored for acceptable levels. The hold down release system PIC's and the T_0 umbilical release system PIC's are monitored by the LPS. The RS launch sequence logic provides for initiating a countdown "Hold" if the SRB ignition PIC voltages fall below an acceptable level at any time prior to issuance of the SSME start commands. After the SSME start commands are issued, if the SRB ignition PIC voltages are not acceptable, the SSME's are shut down.

The RS launch sequence also controls certain critical main propulsion system valves and monitors the engine ready indications from the SSME's. After the main engine "START" commands are issued, the sequence monitors the thrust buildup of each engine; and, unless all engines reach the required level (90%) within the required time (4 sec), an orderly shutdown is commanded and safing functions are initiated.

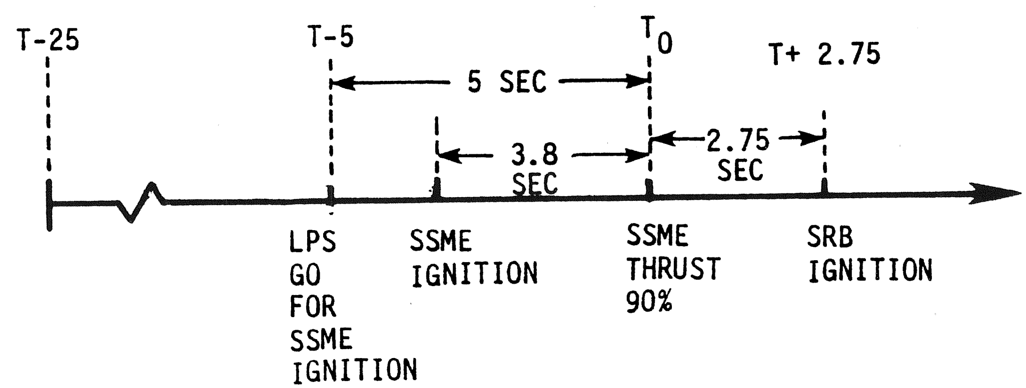
Normal thrust buildup to the required level will result in the SSME's being commanded to the lift-off position, the SRB ignition and holddown release commands being issued, termination of LPS polling, reset of the master timing unit, commanding of T-0 umbilical release and start of the event timer.

A typical launch countdown sequence chart is given in table 1, which outlines the major functions performed prior to lift-off.

There are no guidance functions performed prelaunch. Also, there are no active flight control functions (only the commands to gimbal the SSME's for test and launch positions and slew the surfaces and SRB nozzles).

However, navigation is initialized prelaunch. At T-8 seconds, the launch pad location (latitude, longitude, and altitude) of the nav base is transformed to an M50 position vector. An M50 velocity vector is computed from the earth's angular velocity vector. These values provide an initial state vector which will be propagated using the basic gravity model and sensed velocity changes from the IMU's. The IMU's are set inertial at T-20 min but are biased for earth rotation until T-12 sec when the bias is removed.

The following timeline depicts the countdown sequence near lift-off.



(It is still in question as to whether T-0 will be SSME THRUST 90 percent or SRB IGNITION. In this workbook, we'll assume it to be SSME THRUST 90 percent.)

The 2.75 sec delay time between SSME THRUST 90 percent and SRB IGNITION is to allow the vehicle to recover from the "twang" effect experienced when igniting the main engines. FIRE 1 and FIRE 2 commands are sent simultaneously to the SRB's 2.75 sec after all three SSME's have reached 90 per cent. FIRE 2 modes the redundant set to MM102 and releases the T-0 umbilical. All three engines will be shutdown and the SRB's inhibited from firing if any one of the SSME's do not reach 90 percent thrust 4.6 sec after SSME ignition.

EXERCISE

1. The ground computer network which aids in controlling prelaunch functions is the __________.

2. T/F. The LPS launches the Shuttle. ______

3. T/F. When the RS Launch Sequence begins performing functions by the on-board clock, it will honor a "HOLD" command from the LPS. _______

4. T/F. Navigation is initialized at lift-off.

5. The SRB's are fired _______ sec after the SSME's reach _______ percent thrust. This allows the vehicle to damp out rates generated by the "_______" effect at SSME ignition.

EXERCISE ANSWERS

1. The ground computer network which aids in controlling prelaunch functions is the LPS.
2. T/F. The LPS launches the Shuttle. False
3. T/F. When the RS Launch Sequence begins performing functions by the on-board clock, it will honor a "HOLD" command from the LPS. True
4. T/F. Navigation is initialized at lift-off. False
5. The SRB's are fired 2.75 sec after the SSME's reach 90 percent thrust. This allows the vehicle to damp out rates generated by the "twang" effect at SSME ignition.

SECTION 3
ASCENT FIRST STAGE

3.1 NAVIGATION

The navigation application program is responsible for providing accurate knowledge of the vehicle state vector through the use of IMU sensed velocity changes and a mathematical model of the earth's gravitational forces. For those of you who are familiar with the so-called Super-G navigation, it is the scheme which is used during ascent and is discussed under state propagation.

3.1.1 Gravity Model

The model used is a multi-term series expansion which describes the earth's gravitational field as a set of spherical harmonics. Each additional term adds accuracy to the model but it also adds computational complexity. Figure 3-1 gives a pictorial view of the various terms which comprise the gravity model. Only the central and J2 terms are used in the ascent model.

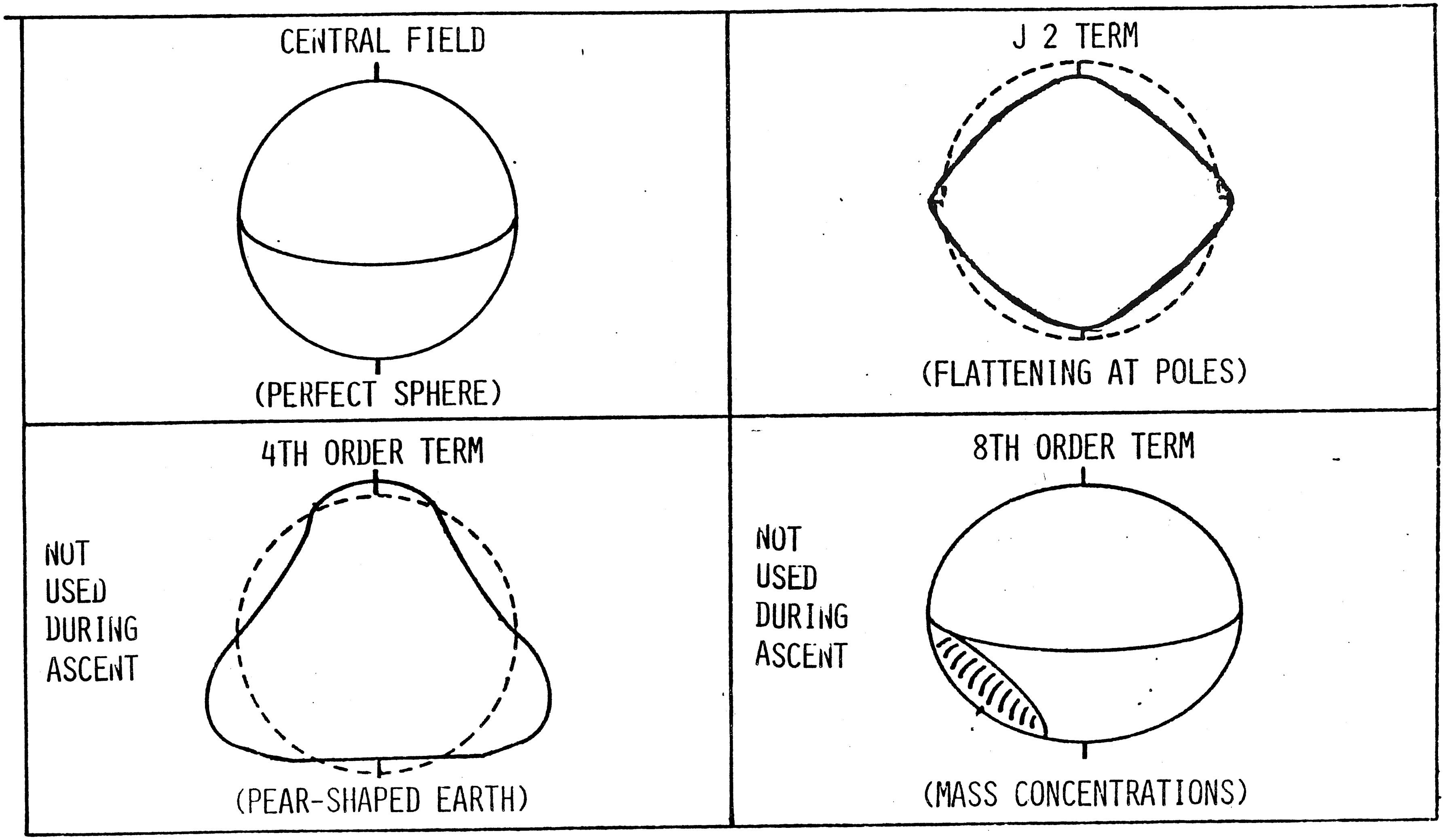


Figure 3-1.- Gravity models used by navigation.

3.1.2 State Propagation

This principal function provides the logic that integrates the past vehicle state to the current time. The integration scheme used is Super-G, which means the integrated state is corrected for the change in gravity over the integration time. The sequence of events in the Super-G scheme is as follows:

- At T_1, select gravity value $\overline{G}_1$ = function of $\overline{R}_1$, $\overline{V}_1$, gravity model.
- Now estimate current state at T_2.
- $\overline{R}_2 = \overline{R}_1 + \Delta T\ (\overline{V}_1 + 1/2\ (\text{IMU sensed } \Delta\overline{V} + \overline{G}_1\ \Delta T))$
- Select gravity value $\overline{G}_2$ = function of $\overline{R}_2$, $\overline{V}_1$, gravity model.
- Compute $\overline{V}_2$.
- $\overline{V}_2 = \overline{V}_1 + \Delta\overline{V} + 1/2\ \Delta T\ (\overline{G}_1 + \overline{G}_2)$
- Correct R_2 for gravity change over last step.
- $\overline{R}_2 = \overline{R}_2 + 1/6\ (\overline{G}_2 - \overline{G}_1)\ \Delta T^2$
- The current estimated state is now ($\overline{R}_2$, $\overline{V}_2$, T_2).

3.1.3 State Prediction Function

This function in navigation provides the capability for predicting the position and velocity of the Orbiter at some time in the future or past when an initial state and time are given. The function assumes coasting flight; i.e., only gravity is acting on the vehicle. A potential use of this function would be for aiding in the prediction of the burn-out state at MECO. Also, this function can be used to aid in driving the predictor and trailing bugs on the CRT displays.

3.1.4 User Parameter Processing (State Propagation)

Since navigation cycles at a slow rate (.25 Hz or .5 Hz), there is a need for a function to provide an updated state vector at a fast rate (12.5 Hz or 25 Hz). This fast rate state propagation is needed because of the demand, by other users, for an accurate state vector at a faster rate than navigation can provide. User Parameter Processing (UPP) integrates the equations of motion and provides the state as requested. When a new NAV state is available (i.e., each navigation cycle), UPP throws away its propagated state and replaces it with the more accurate NAV state (fig. 3-2).

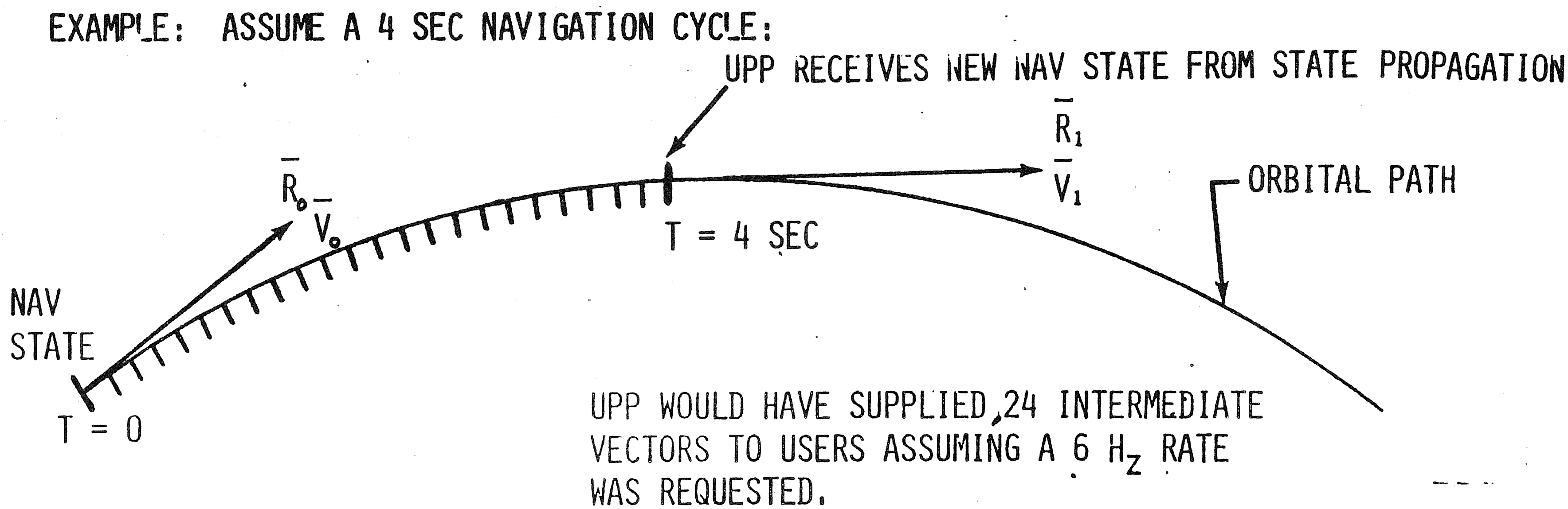


Figure 3-2.- UPP state.

EXERCISE

1. The navigation scheme, Super-G, gets its name because it ______________________ over the integration time.

2. The navigation estimated state consists of ________, ________, and ________.

3. The gravity model used during ascent models two terms in the series expansion. Name what the two terms describe.

    a. ________________

    b. ________________

4. T/F. UPP provides state info to users that need it at a faster rate than navigation can provide it.

    ____

5. T/F. Since UPP uses a simple integration scheme, the ground periodically has to uplink a new state in order to purify the UPP state vector. ________

EXERCISE ANSWERS

1. The navigation scheme, Super-G, gets its name because it corrects for the gravity change over the integration time.

2. The navigation estimated state consists of $\bar{R}$, $\bar{V}$, and T.

3. The gravity model used during ascent models two terms in the series expansion. Name what the two terms describe.

   a. central field

   b. flattening at the poles

4. T/F. UPP provides state info to users that need it at a faster rate than navigation can provide it.
   True

5. T/F. Since UPP uses a simple integration scheme, the ground periodically has to uplink a new state in order to purify the UPP state vector. False

3.2 GUIDANCE

The guidance scheme used in first stage is the simplest scheme used throughout the entire flight. The guidance scheme basically works open loop in that it selects an I-loaded roll, pitch, and yaw command based on Orbiter relative velocity. There is actually an I-loaded table that the software accesses in order to command the appropriate vehicle attitude needed to obtain the desired SRB separation altitude and orbit inclination. Guidance is also responsible for issuing throttling commands to the main engines. The throttle commands are also I-loaded, as are the attitude commands. There are four I-loaded slots for throttle settings and they are also referenced to relative velocity. A typical throttle command plot is given in Figure 3-3. (Note, the main engine controller limits the rate of change of throttle setting to 10 percent/sec.)

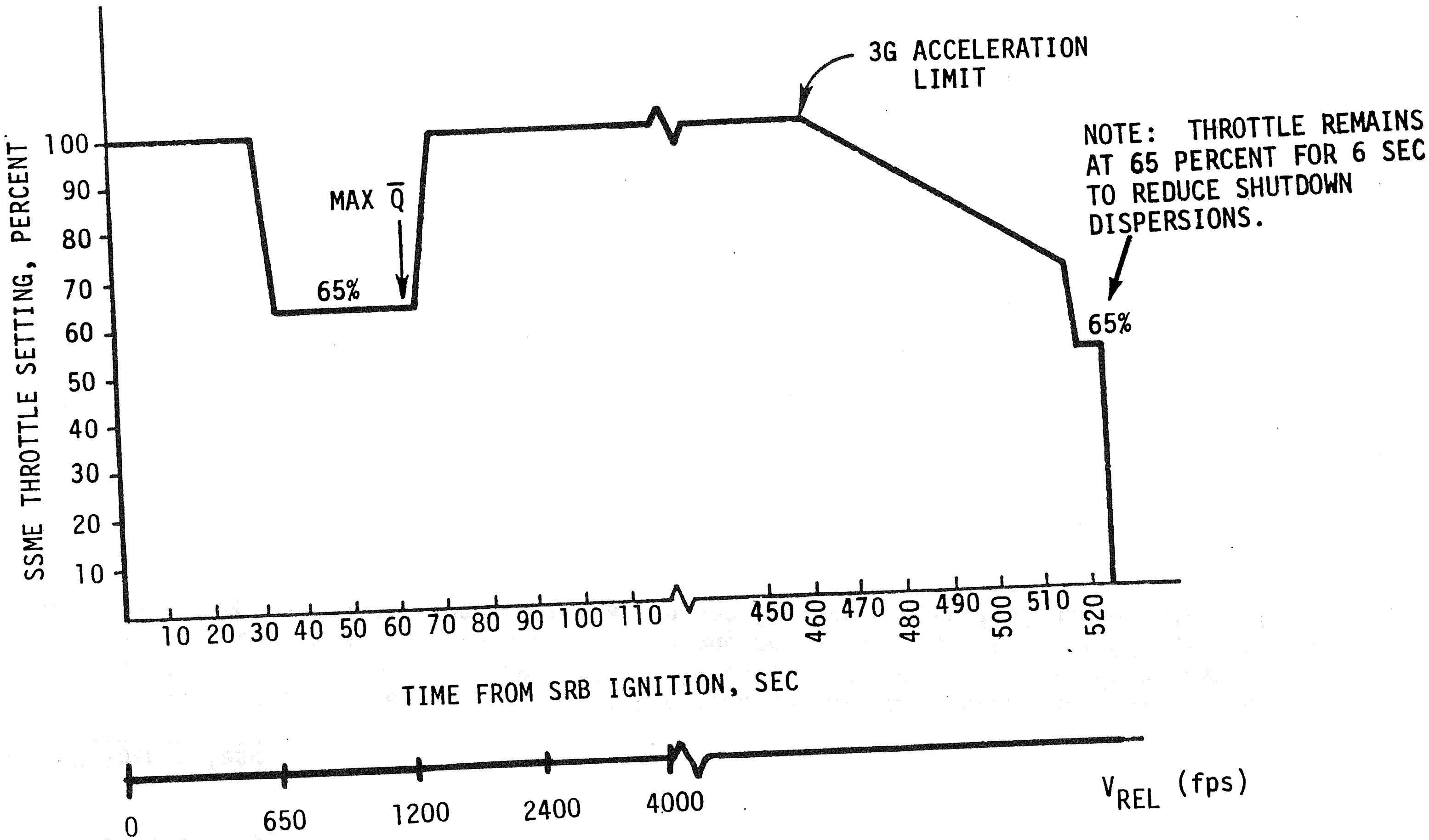


Figure 3-3.- SSME throttle history.

Guidance performs the preceding functions by utilizing a number of different software tasks. They are discussed in the following paragraphs.

3.2.1 Boost Guidance Task

This is the task responsible for performing the "table look-up" of the appropriate roll, pitch, and yaw command depending on relative velocity. The table is I-loaded and contains attitude ephemeris to loft the trajectory in the event of an SSME failure. If an engine fails prior to V = 1100 fps, guidance automatically modes to the appropriate I-loads for that engine failure. A nominal ascent altitude vs. time plot is shown in figure 3-4.

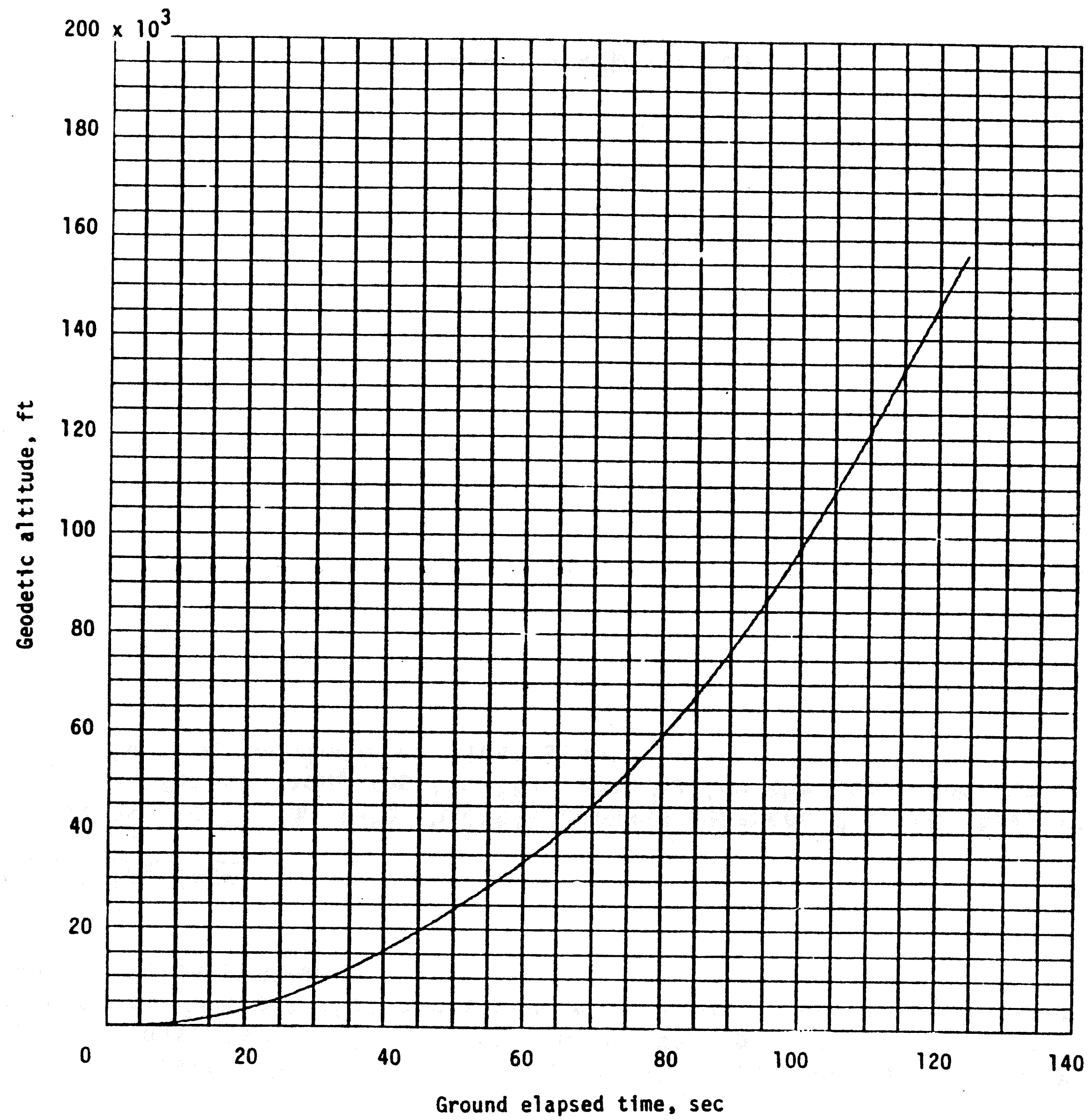


Figure 3-4.- First-stage geodetic altitude versus ground elapsed time.

Boost guidance commands an attitude hold for approximately the first 10 sec after lift-off so that the tower is cleared. Then, at about V_{REL} = 117.8 fps, the dynamic attitude profile begins. This profile is followed until SRB SEP CMD, at which time a 3-axis attitude hold is commanded for 4 sec. At the end of the 4 sec attitude hold, second stage guidance takes over control.

This software is also responsible for converting the attitude commands to a form usable by G/C STEER (a software module to be discussed in the Flight Control section), i.e., a quaternion. Recall from the Quaternion Supplemental Workbook that a quaternion is a convenient form of representing the transformation between two reference systems.

3.2.2 Boost Throttling Task

This software provides the appropriate throttle command referenced to relative velocity. If there is a main engine failure in first stage, the throttle command for the remaining two engines is set to 100 percent and held there for the remainder of first stage.

In the event of manual throttling, the SBTC throttle setting is output to all active engines. See Section 3.2.4 below for procedures on manual throttle takeover.

Referring back to Figure 3-3, the "bucket" that occurs in the plot is due to max-q control. The SSME's throttle down so as not to exceed dynamic pressure constraints on the vehicle. This throttle down is built into the throttle profile.

3.2.3 SSME Out Safing Task

This task is active in first and second stage and also the powered portion of RTLS. In the event of a main engine failure, this task notes the velocity at the time of the failure. This velocity will be used by abort guidance and targeting functions if an abort is subsequently selected. This task also informs other software that an engine has failed.

3.2.4 Manual Throttling (Via Pilot's SBTC)

The procedure for manual takeover is as follows:

- Depress takeover switch on SBTC. (Freezes auto command.)
- Rotate SBTC until "MAN" light on the SPD BK/THROT PBI is lit. (This means you have matched the frozen auto command.)
- Release takeover switch. Pilot now has control of the throttle.

This procedure will prevent undesirable transients in throttle commands during takeover.

EXERCISE

1. T/F. First stage guidance consists of an I-loaded attitude profile referenced to vehicle velocity. ________
2. T/F. Guidance begins to command a vehicle dynamic profile at lift-off. ________
3. T/F. Guidance commands 100% throttle through all of first stage. ________
4. T/F. There are no provisions made in guidance to mode for an SSME failure. ________
5. T/F. Large transients in manual throttle takeover are unlikely. ________

EXERCISE ANSWERS

1. T/F. First stage guidance consists of an I-loaded attitude profile referenced to vehicle velocity. True
2. T/F. Guidance begins to command a vehicle dynamic profile at lift-off. False
3. T/F. Guidance commands 100% throttle through all of first stage. False
4. T/F. There are no provisions made in guidance to mode for an SSME failure. False
5. T/F. Large transients in manual throttle takeover are unlikely. True

3.3 ASCENT FLIGHT CONTROL

Now it's time to take a look at what happens to the guidance commands once they've been computed and sent to flight control. First, a look at the hardware involved.

3.3.1 Hardware

It is important to realize that the Digital Autopilot (DAP) is just that, digital, and resides in the GPC's in the form of software. The hardware that is involved consists of sensors (RGA's and AA's) and thrusters (SSME's and SRB's). The thrusters, besides providing thrust, also provide Thrust Vector Control (TVC) by the movement of the nozzles. The IMU's are also used in the flight control software to provide vehicle attitude information. Figure 3-5 shows the location of the flight control related hardware of the Shuttle.

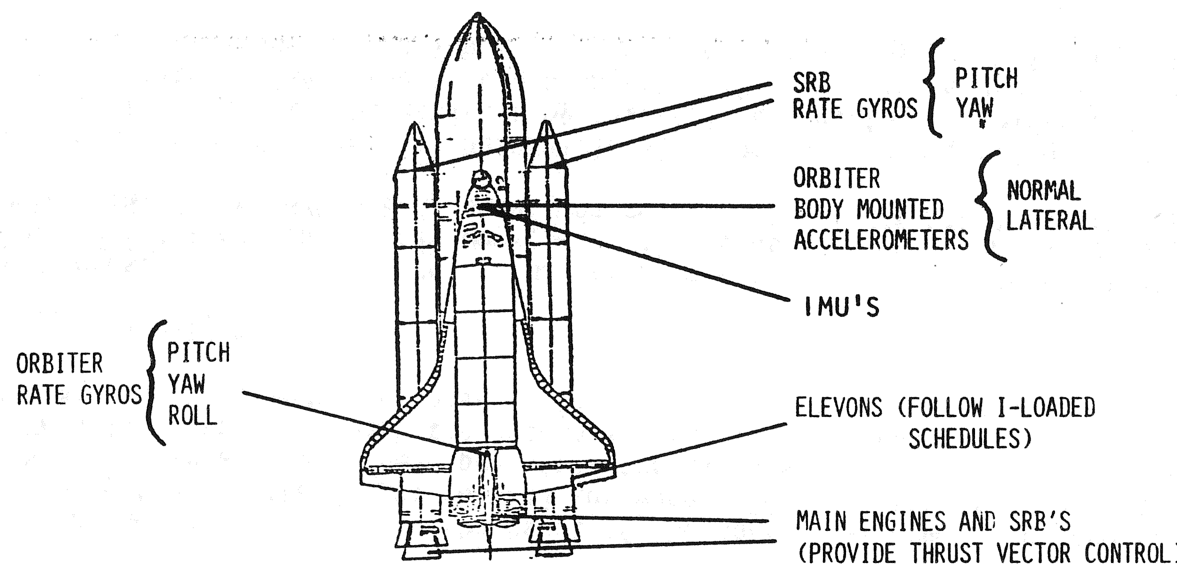


Figure 3-5.- Flight control related hardware for MM 102.

3.3.2 Software

Since the DAP is actually programmed into the computer, it is convenient to present the software in blocks or principal functions. This will be the approach taken here.

Although the DAP, a principal function, is the heart of the control system, there are two other principle functions that are essential to the flight control software. They are the Attitude Processor and Guidance/Control Steering Interface. These principle functions will be discussed in detail in the following paragraphs. We will first start with the automatic system and discuss the entire system. Then we'll take a look at the manual (CSS) mode, and discuss the differences from the automatic system.

3.3.3 Automatic Control System

3.3.3.1 Attitude Processor. - The function of the attitude processor is to derive attitude related data for ~~several user principal~~ functions. This workbook will discuss the two primary users which are GUIDANCE/CONTROL STEERING INTERFACE (G/C STEER, sometimes referred to as STEERING) and the ADI.

Attitude Processor is structured to derive the vehicle attitude quaternion, using a selected IMU, at a low rate, 1.04 Hz, and to propagate the attitude by integrating a quaternion differential equation, driven by selected, prefiltered RGA outputs at a high rate, 12.5 Hz. There are some users, namely flight control, that require the attitude quaternion at 12.5 Hz; so that is why the RGA's (SRB RGA's) are used to propagate the attitude quaternion.

A quaternion is just a four element representation of a nine element transformation matrix. Here, we are transforming from the Aries Mean of 1950 coordinate frame (inertial) to the body frame. For more information on quaternions (more than you'd ever want to know), read the Quaternions Supplemental Workbook.

The attitude processor also provides body attitude information which drives the attitude indicator (8-Ball).

Figure 3-6 is a functional view of the attitude processor for ascent.

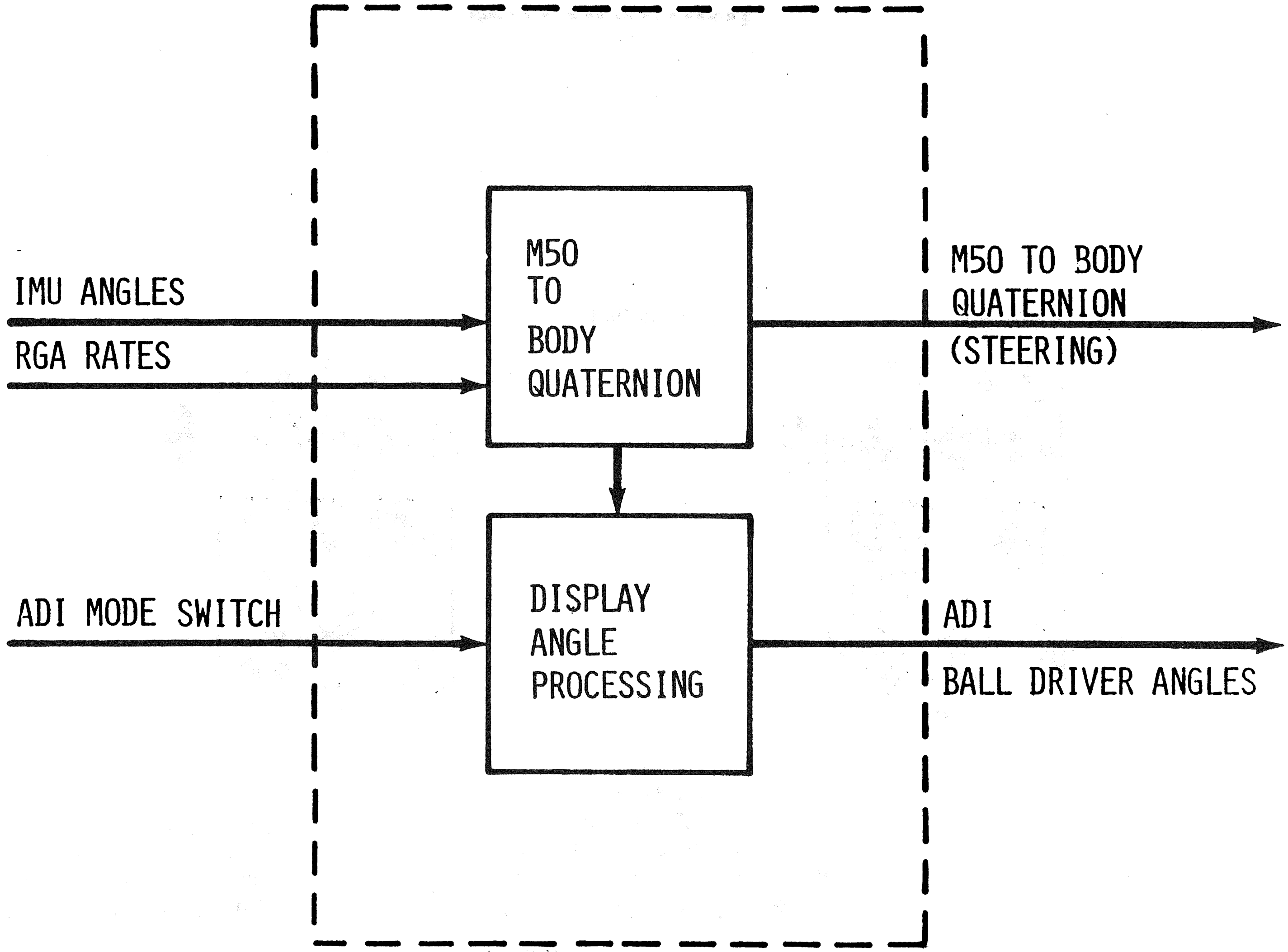


Figure 3-6.- Attitude processor (ascent).

The ADI attitude mode switch, located on panels F6 and F8, is shown below in figure 3-7. It allows the crew to select the reference frame of the corresponding attitude indicator (8-Ball).

As of the publishing date, the switch positions contain the following reference frame information:

REF	LVLH	INRTL
Pitch, yaw, roll sequence x_s along firing azimuth y_s completes triad z_s down to earth center	Yaw, pitch, roll sequence x_s along launch azimuth y_s completes triad z_s down to earth center (so-called LVIY)	OMS 2 TIG attitude

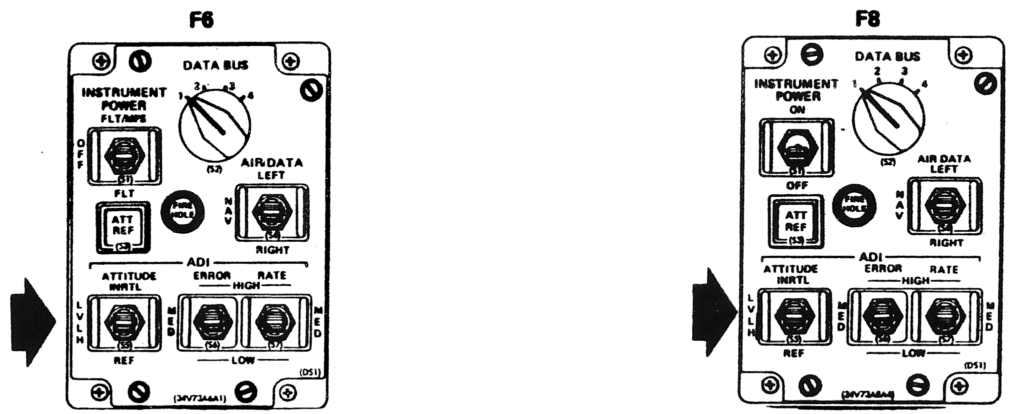


Figure 3-7

For more information on the ADI attitude mode switch, see GNC DISP 2102, section 2.2.1.

The important point to note about the attitude processor is that it provides G/C STEER with current attitude information; i.e., the attitude quaternion.

Now that we've formulated a base to work from, before going any further, let's take a look at what we've learned and what we're going to do with it.

Assuming you've read the Automatic Flight Control Supplemental Workbook or have the equivalent knowledge, you should know that the control system computes and flies out errors between the guidance commanded attitude and the vehicle actual attitude. So, now that we've seen how guidance computes a commanded attitude quaternion and the attitude processor computes the actual vehicle attitude quaternion, you might guess that the two quaternions would be compared to form an error and the DAP would try to fly out that error. Good guess, but it's not completely right. With vehicle structural limits in mind, the commanded attitude must first be smoothed or metered out so as not to exceed these limits during the maneuver. Therefore, there is a principal function named G/C STEER, which contains the logic to limit the command signal used in computing the attitude error which is sent to the DAP.

3.3.3.2 Guidance/control steering interface.- Along with smoothing or metering the attitude command signal, G/C STEER computes and sends the attitude error signal to the ADI processor which drives the attitude error needles on the ADI.

First, let's take a look at what G/C STEER does to the commanded attitude in order to smooth it. Figure 3-8 shows the commanded attitude quaternion coming in from guidance, passing through a second order filter (you need not understand the technicalities of the filter) and coming out as a smoothed "desired" command.

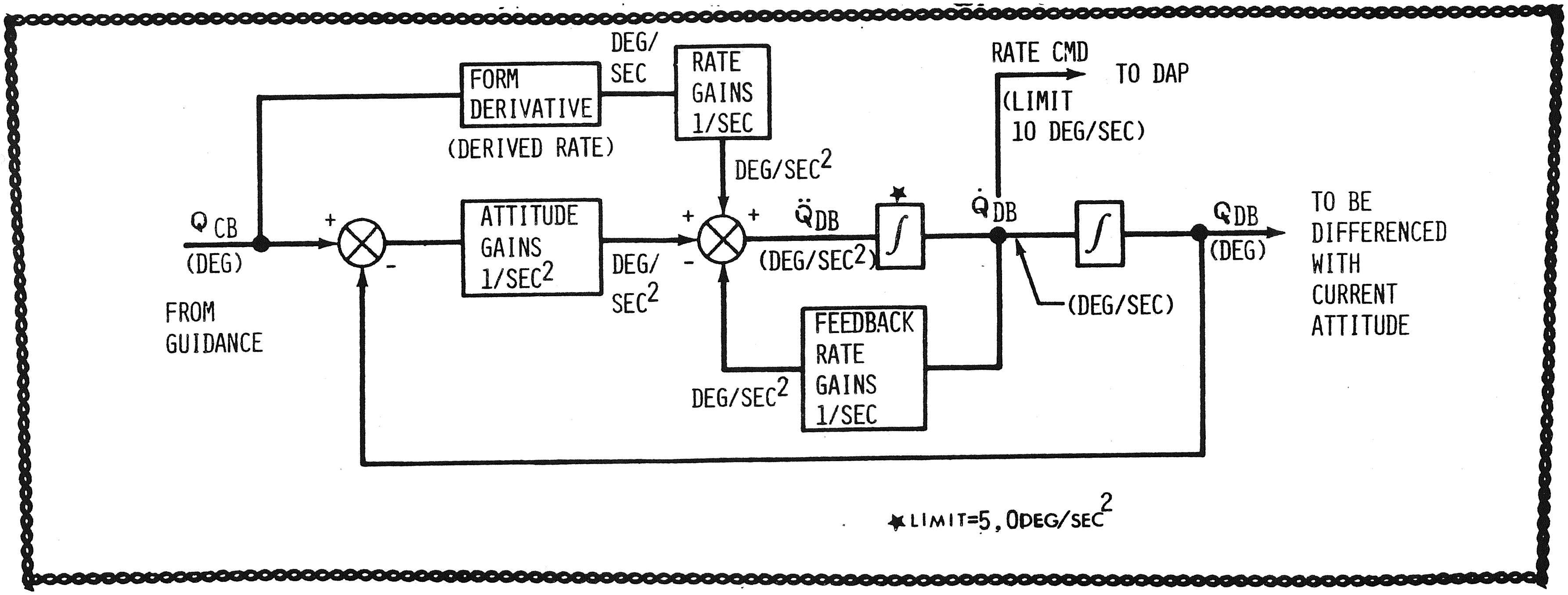


Figure 3-8. - G/C STEER second order filter

where Q_{CB} ∽ Commanded Body quaternion

Q_{DB} ∽ Desired Body quaternion

A convenient way to understand what the filter does is to think of a 25° attitude command going into the filter and the curve shown in Figure 3-9 being the output of the filter. It actually took 13 sec before the output command settled to 25°; therefore, smoothing the commanded attitude so as not to overstress the vehicle.

The filter logic is such that we can extract a rate command from the attitude command that is formed in the filter (fig. 3-8). This becomes the rate command. This is sent to the DAP to be differenced with the RGA rates to form the rate error. Note that there are software signal limits in the filter that limit both rate and acceleration.

We have now seen the two major signals that G/C STEER is responsible for generating, filtered attitude error and rate command (for roll, pitch, and yaw axes). The filtered error is also sent to the ADI processor for display by the attitude error needles on the ADI. As will be seen in a later section, G/C STEER also contains the logic for manual thrust vector control. For now though, we'll limit our discussion to automatic control.

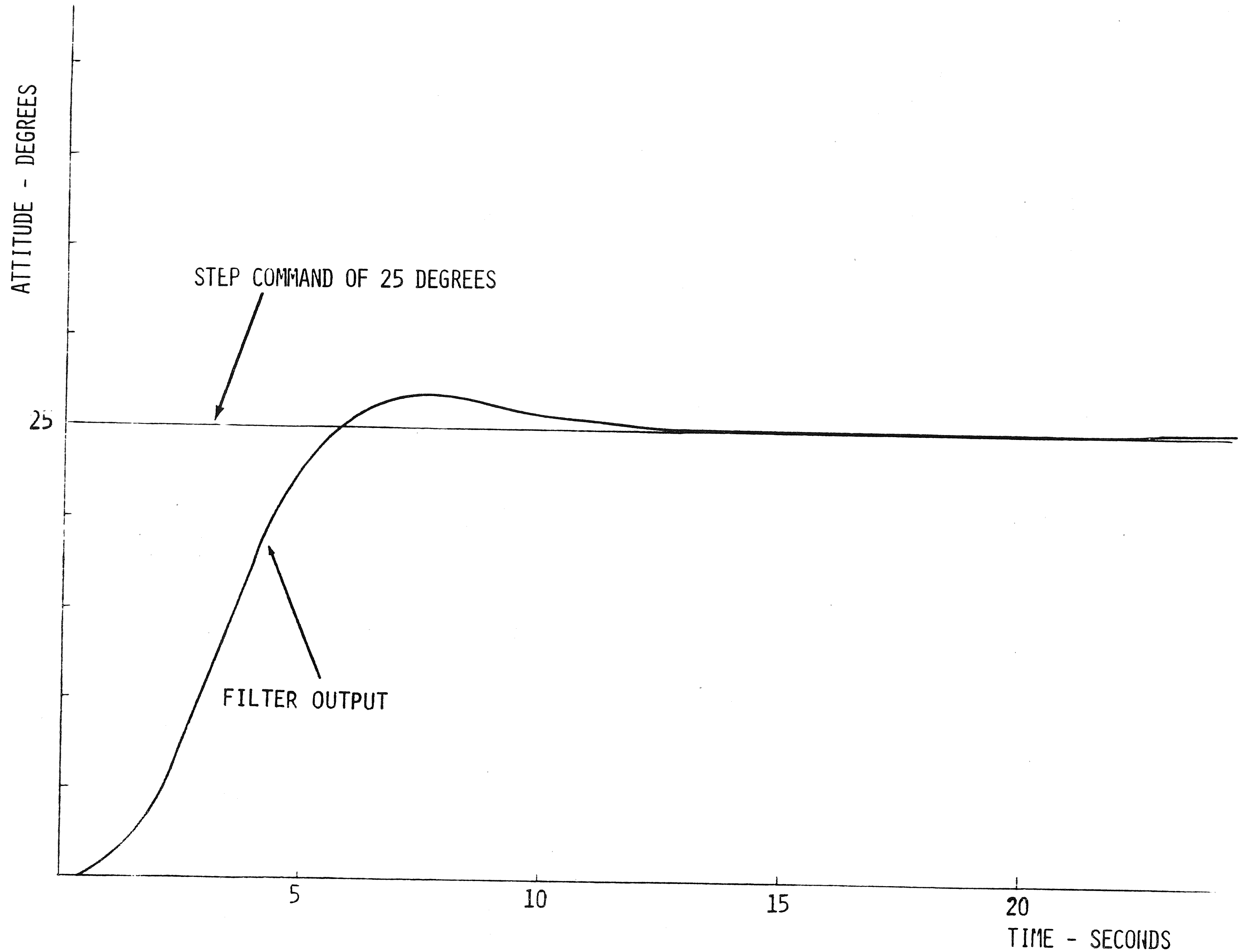


Figure 3-9.- G/C STEER filter output.

EXERCISE

1. T/F. G/C STEER generates current attitude information from the IMU's in the form of a quaternion. ________

2. The commanded attitude from guidance is __________ in G/C STEER.

3. The two major signals that G/C STEER is responsible for generating are:

   a. ____________________
   b. ____________________

4. T/F. The smoothed attitude error is used to drive the ADI error needles. ________

EXERCISE ANSWERS

1. T/F. G/C STEER generates current attitude information from the IMU's in the form of a quaternion. False

2. The commanded attitude from guidance is smoothed in G/C STEER.

3. The two major signals that G/C STEER is responsible for generating are:

   a. attitude error
   b. rate command

4. T/F. The smoothed attitude error is used to drive the ADI error needles. True

3.3.3.3 Digital autopilot.- The Digital Autopilot (DAP) is divided into seven separate software modules. There are two modules of which it would suffice to simply know that they exist. They are covered very briefly in the next two sections; reconfiguration logic and linear interpolations. The remaining modules are important and are covered in detail. Figure 3-10 shows a functional organization of the modules within the DAP.

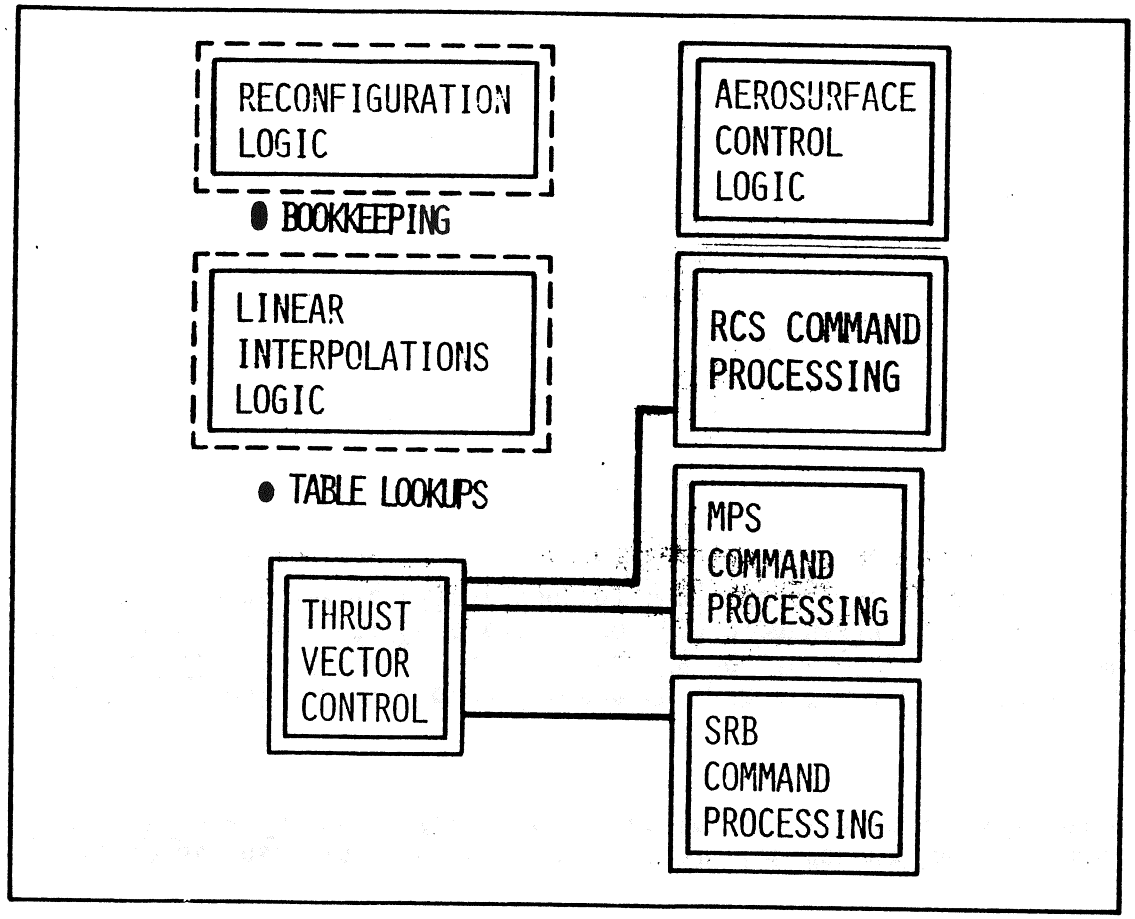


Figure 3-10.- Ascent Digital Autopilot.

3.3.3.3.1 Reconfiguration logic: This logic is basically the bookkeeper of the flight control software. In response to control mode changes, changes in failure status, and the occurrence of events, reconfiguration logic generates the indicators that are needed by both the DAP and G/C STEER for moding sequencing and initialization.

3.3.3.3.2 Linear interpolations: This logic is responsible for generating acceleration profiles, trims, elevon schedules, and scheduled gains as functions of earth relative velocity magnitude, mission elapsed time, and vehicle mass.

3.3.3.3.3 Thrust vector control: As you probably guessed, Thrust Vector Control (TVC) is the hub of the wheel of flight control. Its function is to close the acceleration and rate loops within the outer attitude loop to generate body axis attitude rate errors to eventually be flown out by the SSME's and SRB's (fig. 3-11 - taken from Automatic Flight Control Supplemental Workbook).

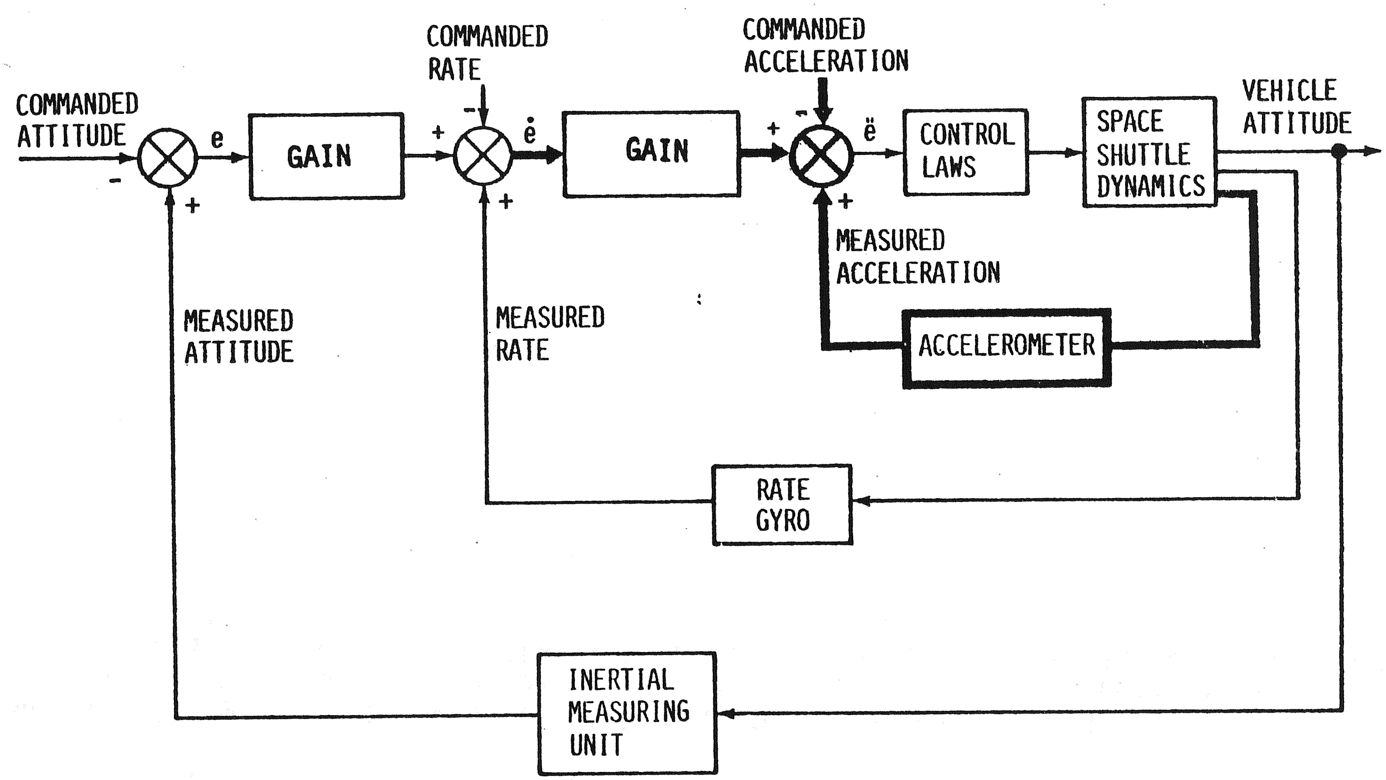


Figure 3-11.- Generic control loop with attitude feedback, rate feedback, and acceleration feedback.

Recalling that the inputs to TVC are attitude errors and rate commands, let's take a look at what happens to them.

The approach taken here is to "build" the control loop in pieces to facilitate comprehension. We'll start out with the pitch channel, then we'll build the yaw channel, and finally the roll channel.

First of all let's review for a moment, first stage flight.

For about the first 10 sec after lift-off the vehicle is in attitude hold. This is convenient in that it prevents recontact with the tower. After tower clear, the pitch profile (I-loaded) begins and also a roll to 180° is commanded (upside down so the horizon can be seen through the windows). By about T+20 sec the vehicle is at 180° roll and 78° pitch.

When a relative velocity of 547 fps (T-0 + 25 sec) is reached the automatic load relief logic kicks in. This logic will actually fly the vehicle away from the planned profile, if it is necessary, to relieve the loads on the vehicle in the high q region. Load relief is done by flying out acceleration error as opposed to attitude error. Bad winds and wind shears make load relief necessary. Because vehicle load relief is performed, hinge moments may build on the elevon hinges. Therefore, in addition, active load relief for the elevons is performed by driving them to the appropriate positions to relieve adverse hinge moments.

We'll save the talk on the elevons for the aerosurface control section.

Load relief is actually performed by utilizing a number of scheduled gains. Recall that a signal that is "gained" is just multiplied by that gain value. The gains are scheduled as a function of relative velocity (represents dynamic pressure). So what happens is, when the vehicle is in the high q region (i.e., $547 < V_{REL} < 2500$) the gains that multiply the acceleration error increase, and the gains that multiply the attitude error decrease. Hence, less attention is paid to the attitude error and more attention to acceleration error. If you're thoroughly confused, maybe the following charts will help (confuse you more).

Let's begin, as we said before, with the pitch channel. Recalling that one of the two inputs to TVC from G/C STEER (steering) is attitude error, let's start with that, figure 3-12.

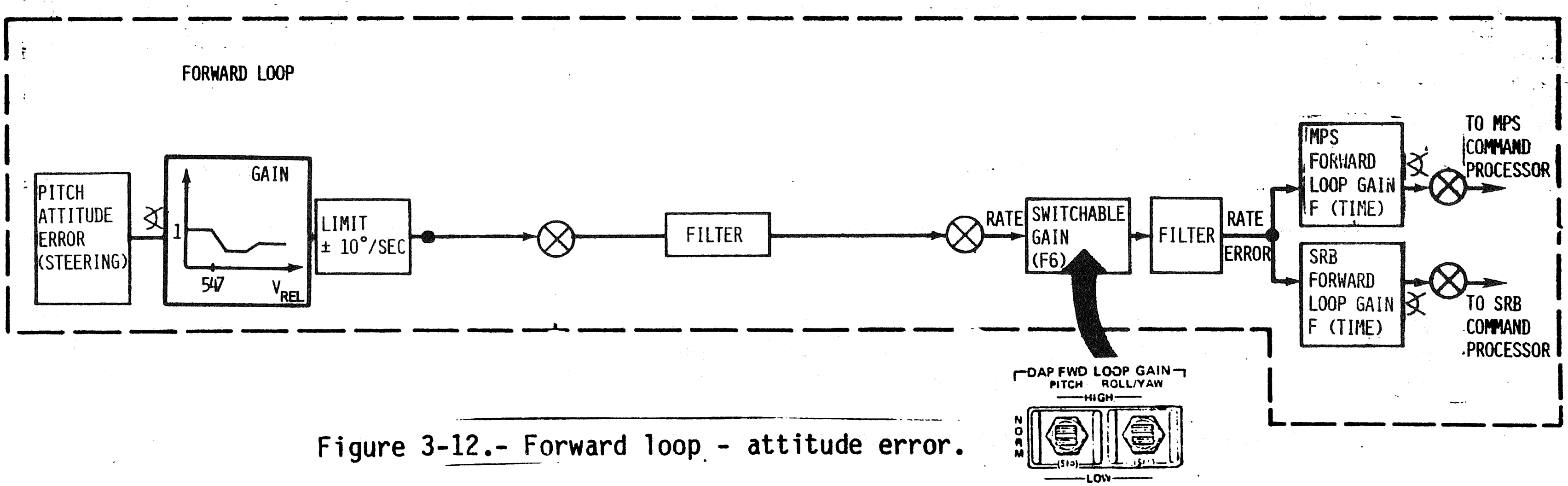


Figure 3-12.- Forward loop - attitude error.

Now the attitude error is multiplied by a gain that is variable with relative velocity. The reason for the values of gain will become more obvious as we complete the control loop. This gain also changes the units of the signal, so that now it becomes attitude rate error.

The signal is next limited to 10°/sec, which is a structural constraint on the vehicle.

So now we have an error signal that was generated basically by using the IMU's. We can now take a look at the RGA feedback signals from the SRB's and compare them with the commanded vehicle rate that was generated in G/C STEER, remember? That is done by just differencing the two signals to generate an error (fig. 3-13).

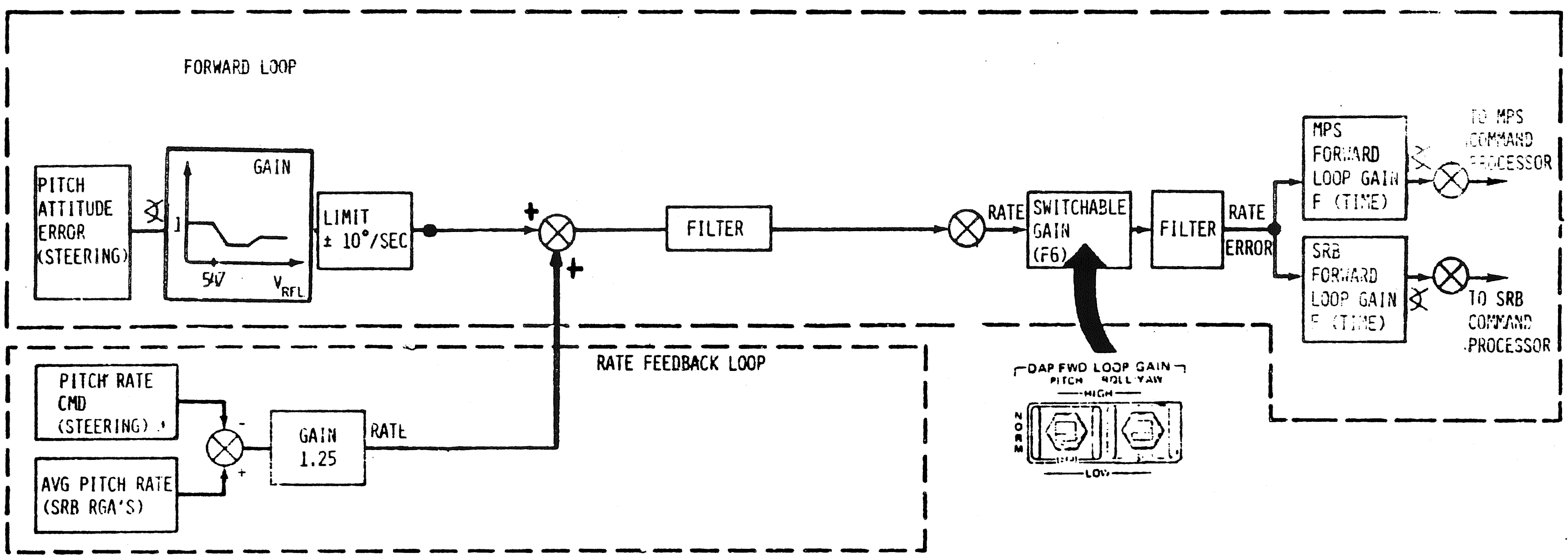


Figure 3-13.- Rate feedback loop - rate error.

Note that that the polarity is actual minus commanded. The rate error signal is then gained and added to the attitude rate error that was previously computed.

The final means of feedback, namely the accelerometer, is the next signal to bring in. It is the signal from the accelerometer mounted on the body Z axis and therefore measures normal acceleration, NZ. We next difference that with a commanded acceleration which is I-loaded in the form of a table (value depends on relative velocity). This signal is called the load factor error (fig. 3-14).

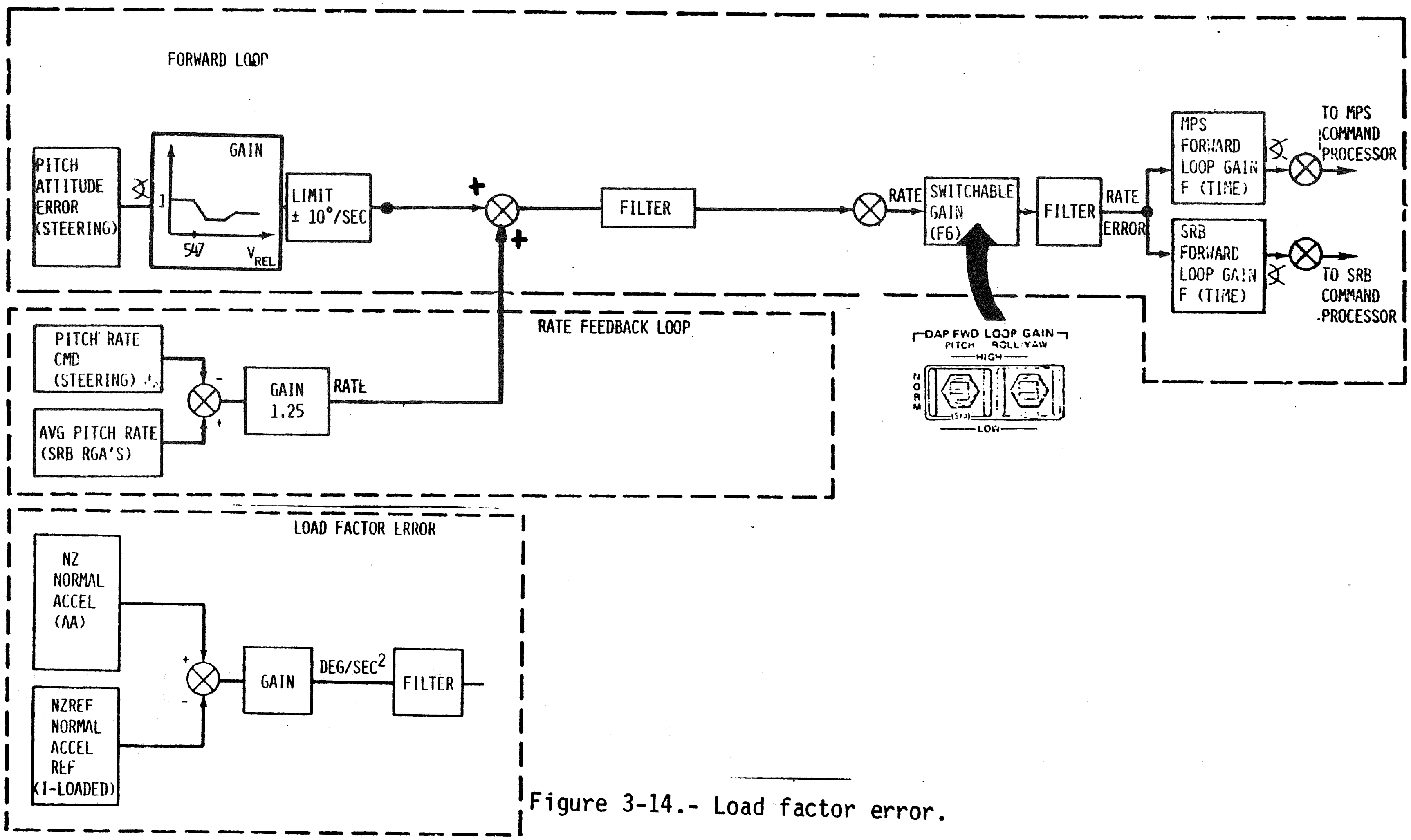


Figure 3-14.- Load factor error.

Note, again, that the polarity is actual minus commanded. The error signal is then gained and filtered.

Now life is not a bowl full of cherries, and be that as it may, the accelerometers were not able to be positioned at the center of gravity of the vehicle. This is mainly because the center of gravity moves throughout the entire flight due to the expulsion of propellant mass. Therefore, we must compensate for the fact that the accelerometers are not at the CG since we only want to measure normal acceleration, not rotational acceleration. This is done by bringing in the pitch RGA signal, gaining and filtering it, then differencing it with the acceleration error (fig. 3-15).

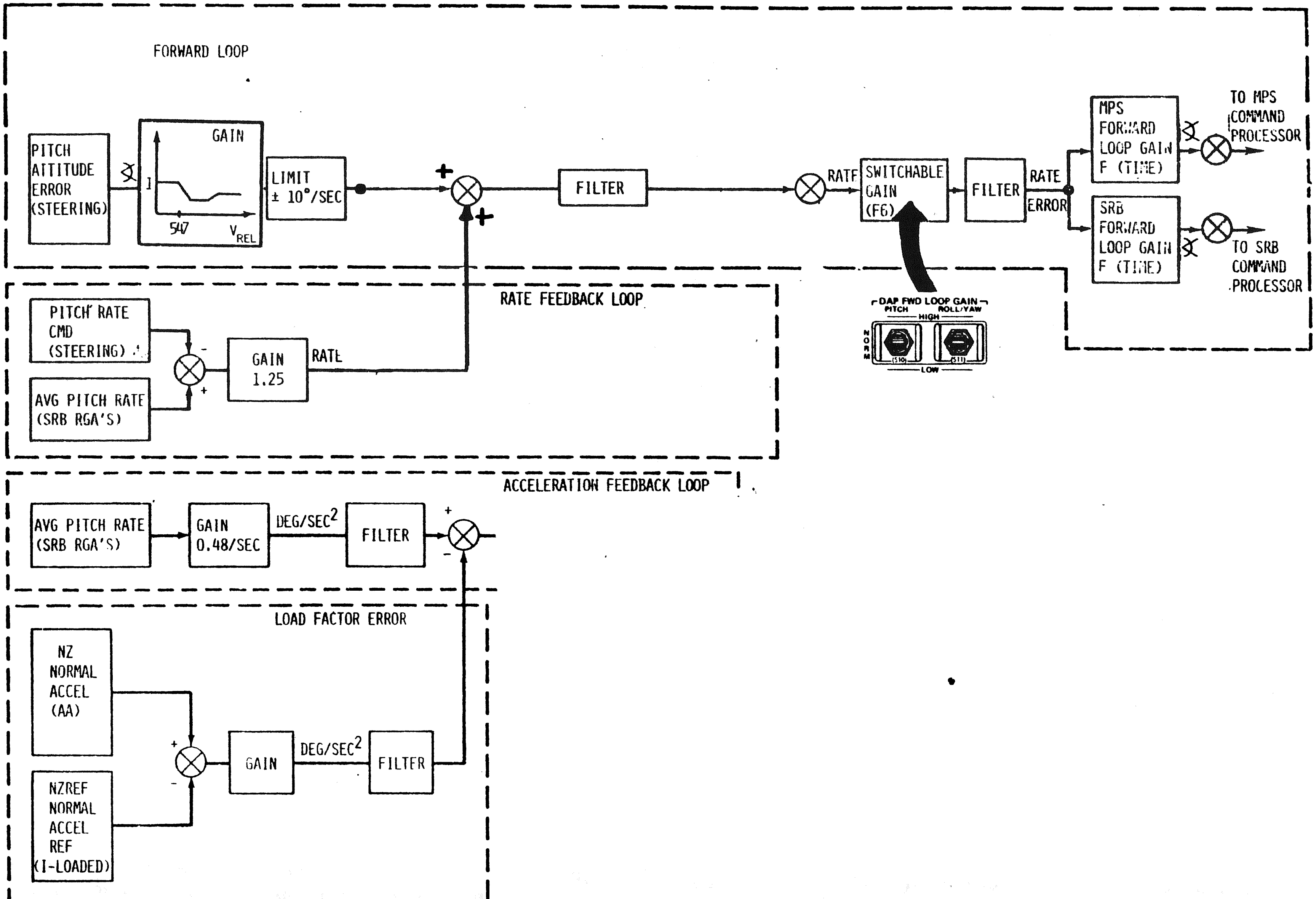


Figure 3-15.-Acceleration feedback.

The resultant signal, a compensated acceleration error, is filtered and then gained before it is added to the rate error signal that was generated from the IMU's and RGA's. The gain that the acceleration error is multiplied by is a sensitivity gain, comparable to the gain which multiplies the pitch attitude error in the forward loop. The gains play together, and by looking at figure 3-16, one can see that as the pitch attitude error gain decreases, beginning at V_{REL} = 547 fps, the acceleration error gain increases.

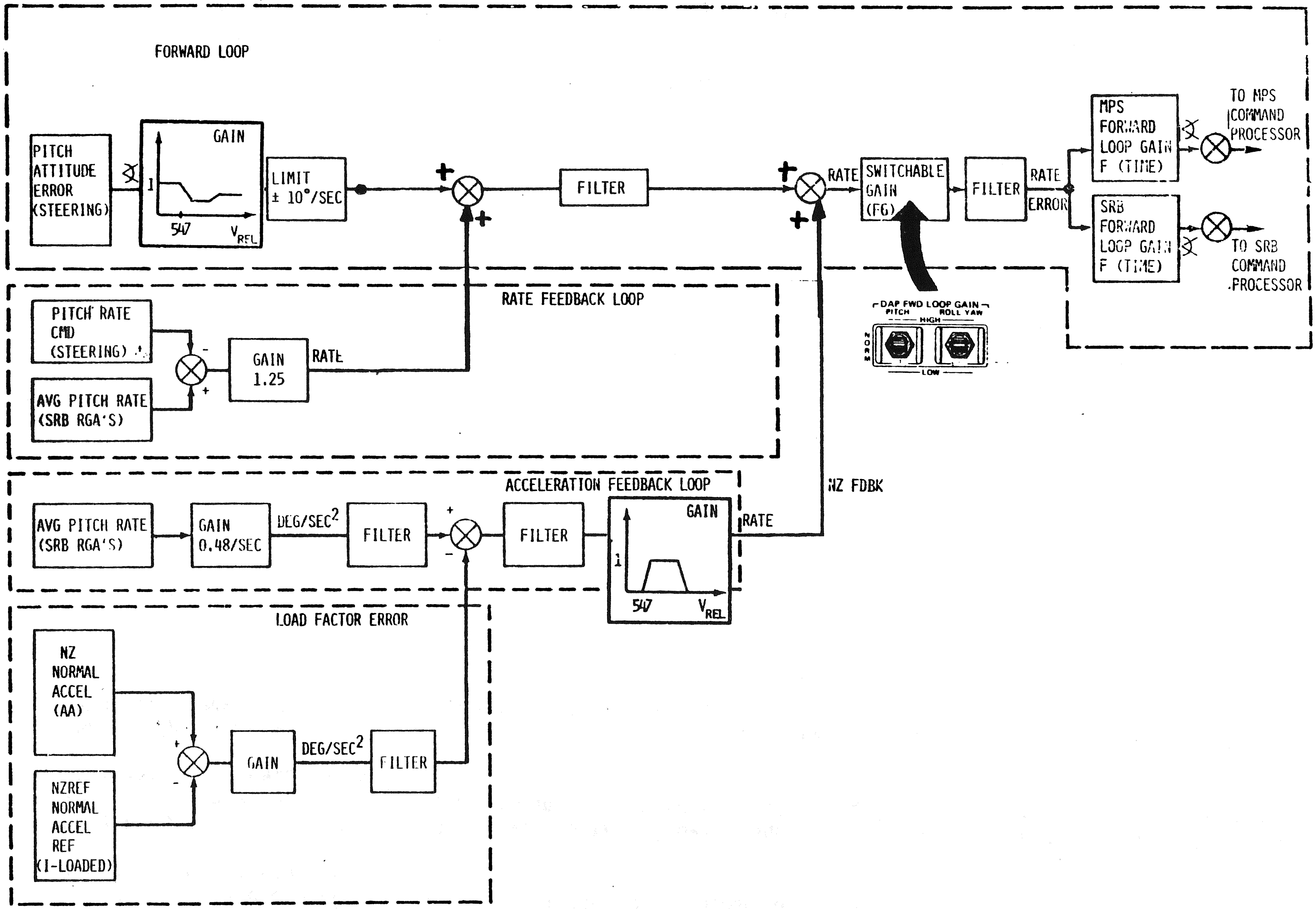


Figure 3-16.-Pitch loop.

(Note that it actually increases from zero, therefore the acceleration error signal was never "heard" until V_{REL} = 547 fps). This is how load relief does its thing; increase sensitivity to acceleration error and decrease sensitivity to attitude error. Therefore, the vehicle will actually steer off course in order to reduce acceleration error (a good determinant of loads) in the high q region where winds play havock with the vehicle.

What happens next? You guessed it, we add the signals together to come up with the total pitch rate error, commonly called the "pitch common error." (Note that the gains converted the signals to the units of deg/sec, or rate units.)

The common error signal is the signal that the crew has the capability of "playing with" by way of the "DAP FWD LOOP GAIN" switches on panel F6 (fig. 3-17).

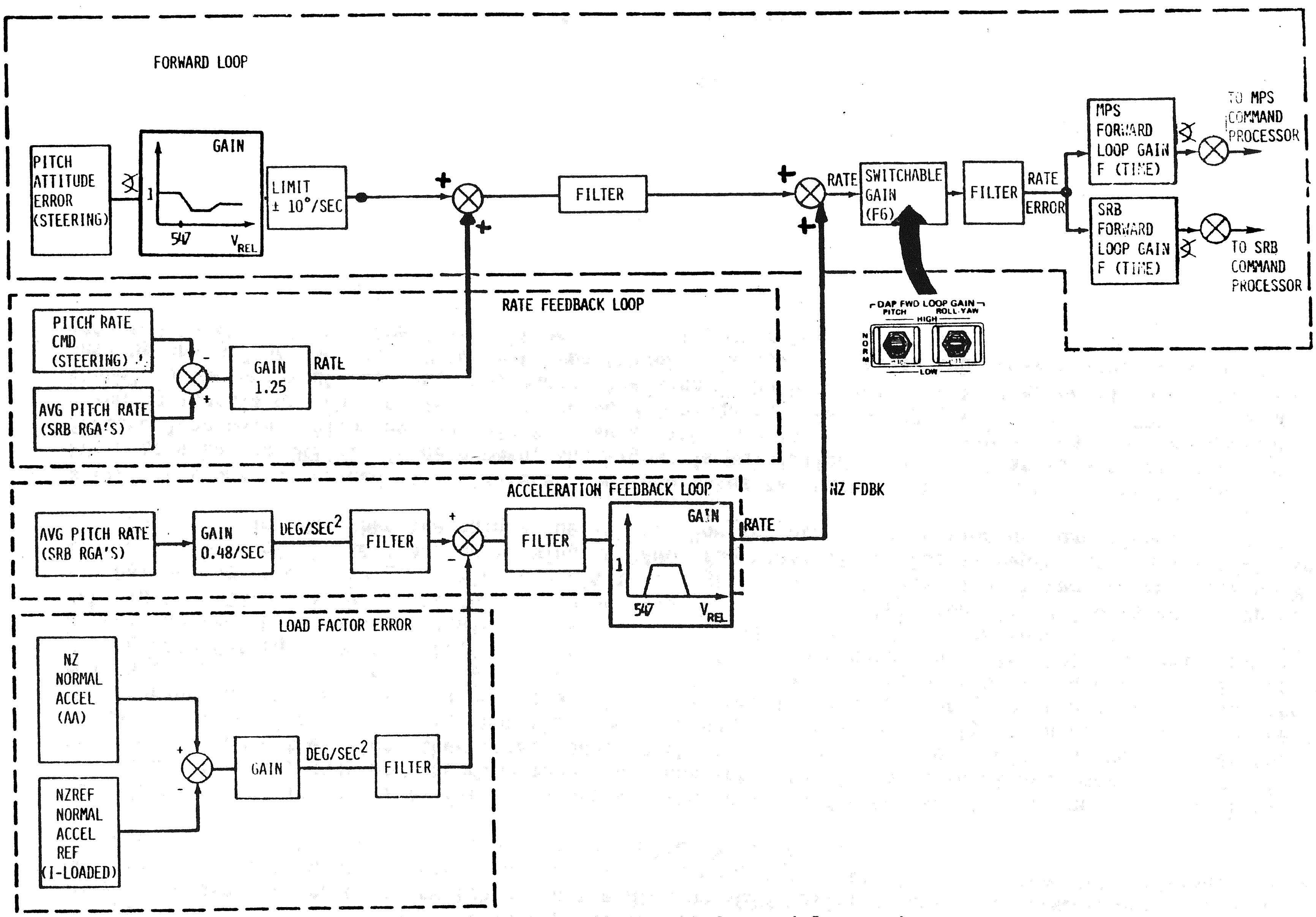


Figure 3-17.- DAP forward loop gain.

They are three position, triply redundant switches. However, the values of the HIGH and NORM position are the same, 1. The value of the LOW position is .5. What these switches allow the crew to do is change the values of the gain that multiplies the common error signal; in this case, pitch common error. So you can see, if the pitch switch is in the HIGH or NORM position, the pitch common error is being mulitplied by 1; so what, nothing happens. If, however, the switch is put in the LOW position, the gain is reduced to .5, therefore the signal is multiplied by .5 or halved.

At this point in time it is difficult to say why the crew would ever want to use the gain switches. It is thought that possibly body bending modes, not characterized in the simulations, may give cause for the crew to lower the gains, therefore, reducing the error signal, hence, making the vehicle respond slower and more sluggish. (Note that the crew does not have the capability to increase gains to speed the response of the vehicle). There is a little bit more to reducing the gain than just throwing the switch to LOW. First, the gain switch must be enabled. That is done by depressing either of the appropriate GAIN ENA (gain enable) PBI's on panels F2 or F4; in this case, the pitch GAIN ENA PBI. By depressing the PBI, the light behind it is lit. This allows the software to reduce the gain to .5 if the DAP FWD LOOP GAIN switch is set to the LOW position. However, if the light is extinguished by depressing the PBI another time, the gain value reverts to 1 regardless of the position of the DAP FWD LOOP GAIN switch. (Think of the software performing an "AND" function; if the PBI is depressed and lit "AND" the gain switch is in LOW, then the gain value is .5. For any other combination of conditions, the gain is 1).

After the signal is gained, it's filtered and then takes two paths, one path goes to the MPS and the other goes to the SRB's. In each path, the signal is multiplied by a gain that is a function of time, a forward loop gain. Allowing the gain to change with time is an accurate way of tracking the greatly changing inertia of the vehicle. These forward loop gains, one for MPS, the other for SRB's, change the signal to the units of degrees, thus making the signal the body axis attitude error signal to be "flown out" by the MPS and SRB's. (Note that even though this gain is called the forward loop gain, it is not affected by the DAP FWD LOOP GAIN switches.) See figure 3-18.

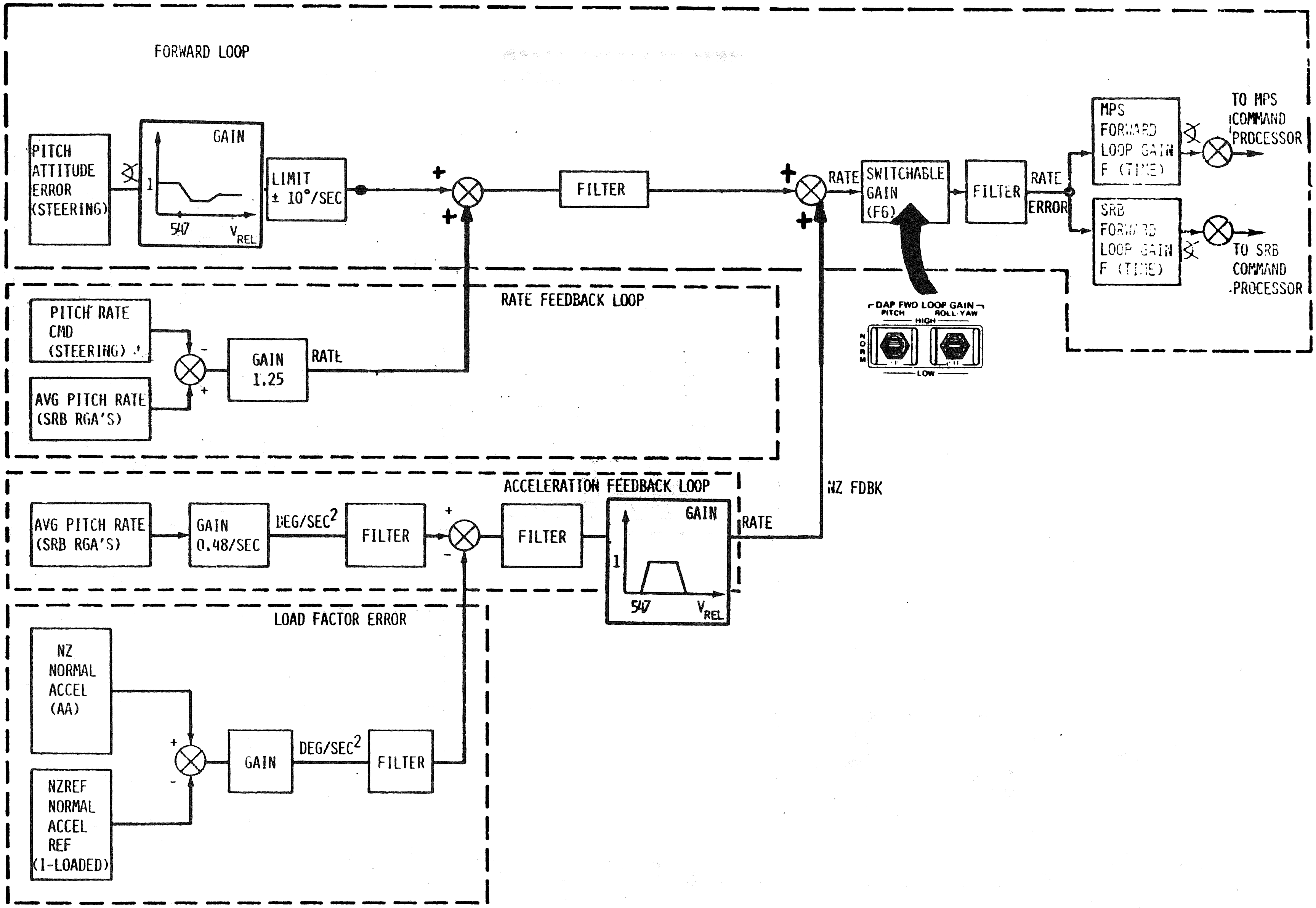


Figure 3-18.- Pitch loop.

Now we've completed the pitch flight control loop, almost. Yes, almost. There is one more addition to the loop called autotrim. It consists of an integrator and a limiter that acts on the pitch attitude error in the forward loop (fig. 3-19).

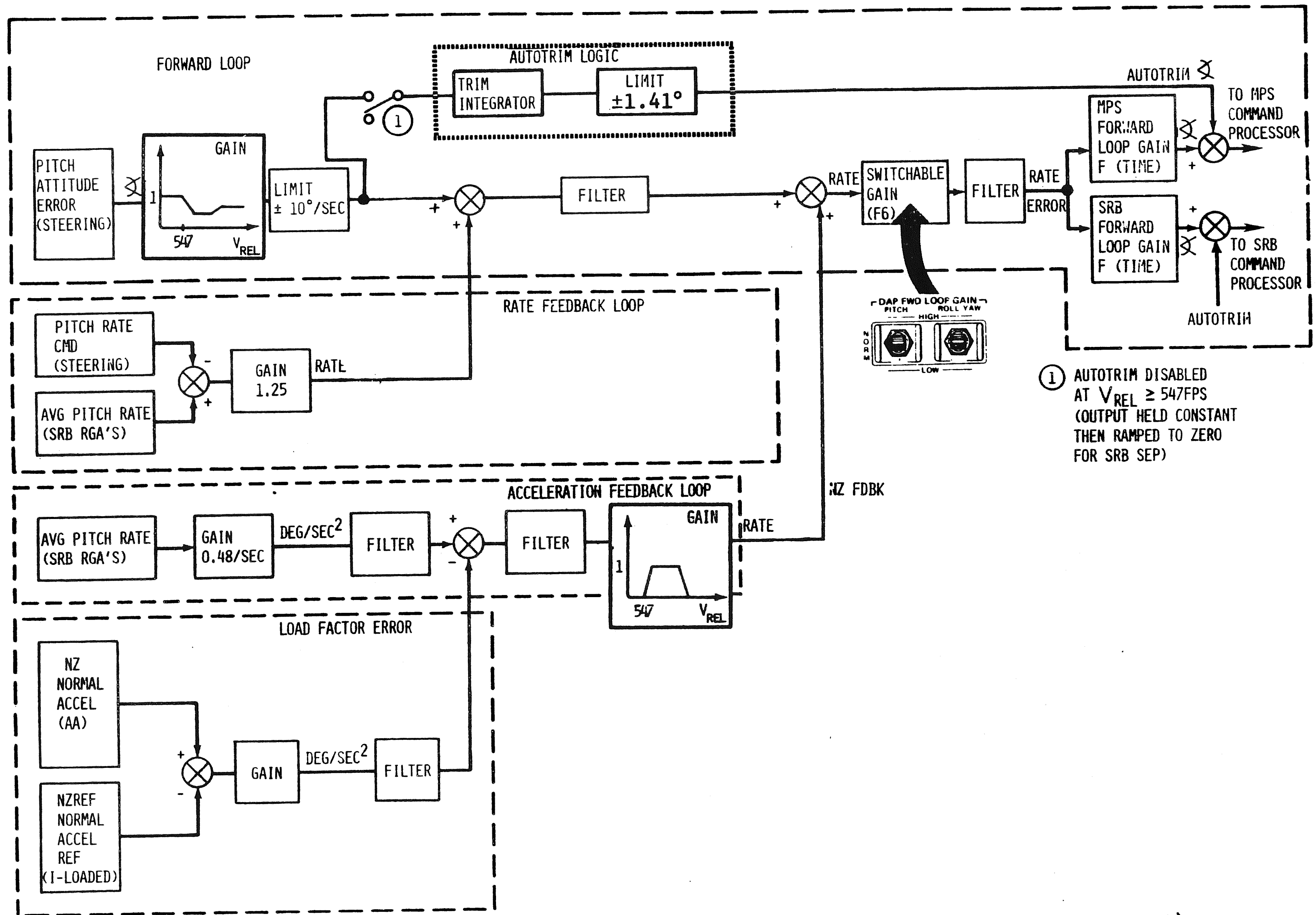


Figure 3-19.- Pitch control loop for first stage (liftoff to SRB null CMD).

The integrated error, maximum value of $\pm 1^{o}$, is summed "downstream" of the forward loop gain so as not to loose the maximum trim value in the gain multiplications (the forward loop gains are decreasing with time because of the decreasing inertia). Autotrim is provided to correct for thrust misalignment and small steady state errors. Note that the software switch indicates that autotrim is cut off when V_{REL} = 547 fps (beginning of load relief). The output of the integrator is held constant through load relief to help reduce loads by not "trimming," then the signal is ramped to zero prior to SRB SEP.

And that is the pitch control loop for first stage thrust vector control. We've ended up with two pitch error signals; one gained for the MPS and the other gained for the SRB's. Each signal next goes to its appropriate command processor to be converted to main engine bell and SRB nozzle deflection commands.

Before we get into the command processors, we first have to build the other two control loops for yaw and roll. Since the yaw loop is very similar in design to the pitch loop, there's no need to build it as meticulously as we did the pitch loop. Therefore, Figure 3-20 presents the yaw loop in the same format as the pitch loop. One minor difference occurs in the block that compensates for AA offset from CG. Yaw and roll rates are needed in this case to correct for this.

One other minor difference in the yaw channel occurs when MET reaches 90 sec (we're nearing staging). Switch 2 in Figure 3-20 shows what happens. The yaw accelerometers are used to aid in controlling the vehicle in yaw as it goes through SRB tailoff and prepare for separation. This is just an enhancement of the capability to control the yaw of the vehicle. This signal, called NY FDBK, is then ramped to zero when MET reaches 125 sec or when SRB P_C becomes less than 110 psi.

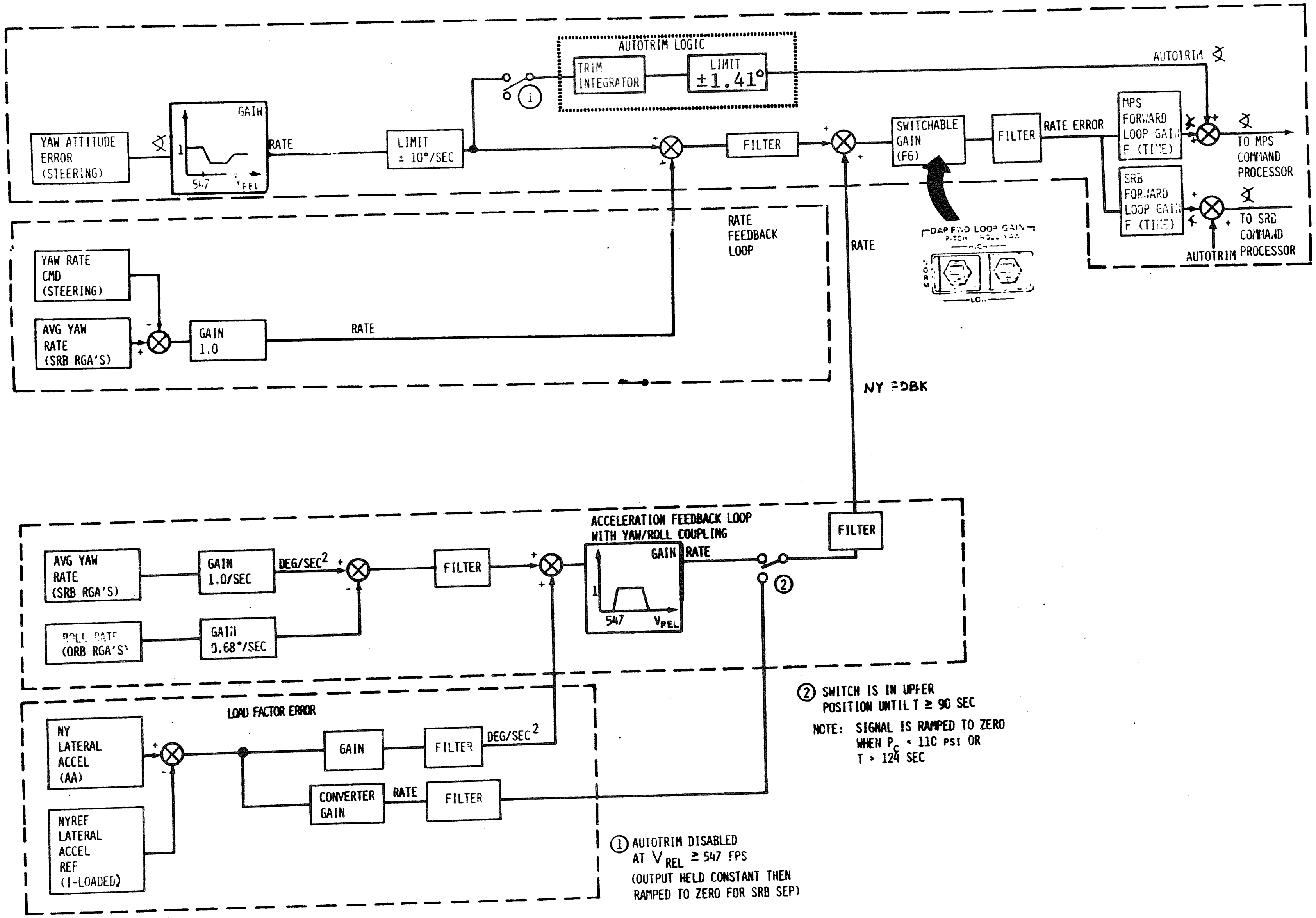


Figure 3-20.- Yaw control loop first stage (liftoff to SRB null CMD).

As in the pitch loop, the yaw loop generates the yaw common error which is then gained for the MPS and the SRB's and sent to the appropriate command processors for conversion to nozzle deflection commands.

Now let's look at the final loop, the roll loop (fig. 3-21).

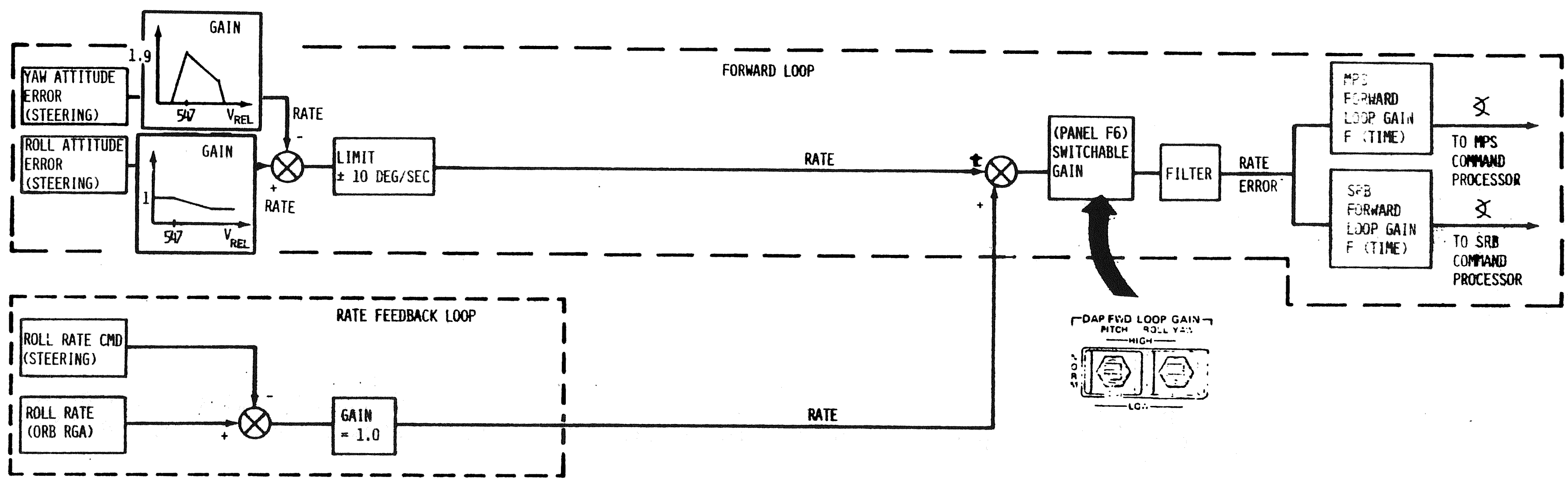


Figure 3-21.- Roll control loop first stage (liftoff to SRB null CMD).

This loop is different from the pitch and yaw loop in that accelerations are not used for load relief. Roll has a unique way of performing load relief. Basically the goal of load relief in the roll channel is to maintain the thrust vector in the desired trajectory plane while the yaw channel tries to reduce β. This sounds confusing, and it is at first. Think of it this way. The yaw channel, as we saw earlier, becomes more sensitive to the yaw acceleration error (by increasing the gain) and less sensitive to yaw attitude error (by decreasing the gain) during the load relief region. Keep in mind that by flying out the yaw acceleration error, we are trying to reduce β. So, by reducing β, we're flying into the winds that are causing the loads problem. But, by flying into the winds, be they shears, gusts, or whatever, we are actually flying out of the planned trajectory plane. Now, looking at figure 3-22 we can see that if we first yaw the vehicle toward the wind, we reduce β but rotate the thrust vector out of the desired plane. So if we subsequently roll the vehicle, we can put the thrust vector back into the desired plane. And that's exactly how load relief in the roll channel works. In the load relief region (between V_{REL} 547 and 2500) the yaw attitude error is gained higher than the roll attitude error so that the vehicle will roll the thrust vector back into the desired plane proportional to how much it had yawed it out of the plane. Amen.

So, back to the loop itself. After the error signal has been gained for the MPS and SRB's, it is ready to be sent to the command processors to be converted to nozzle deflection commands.

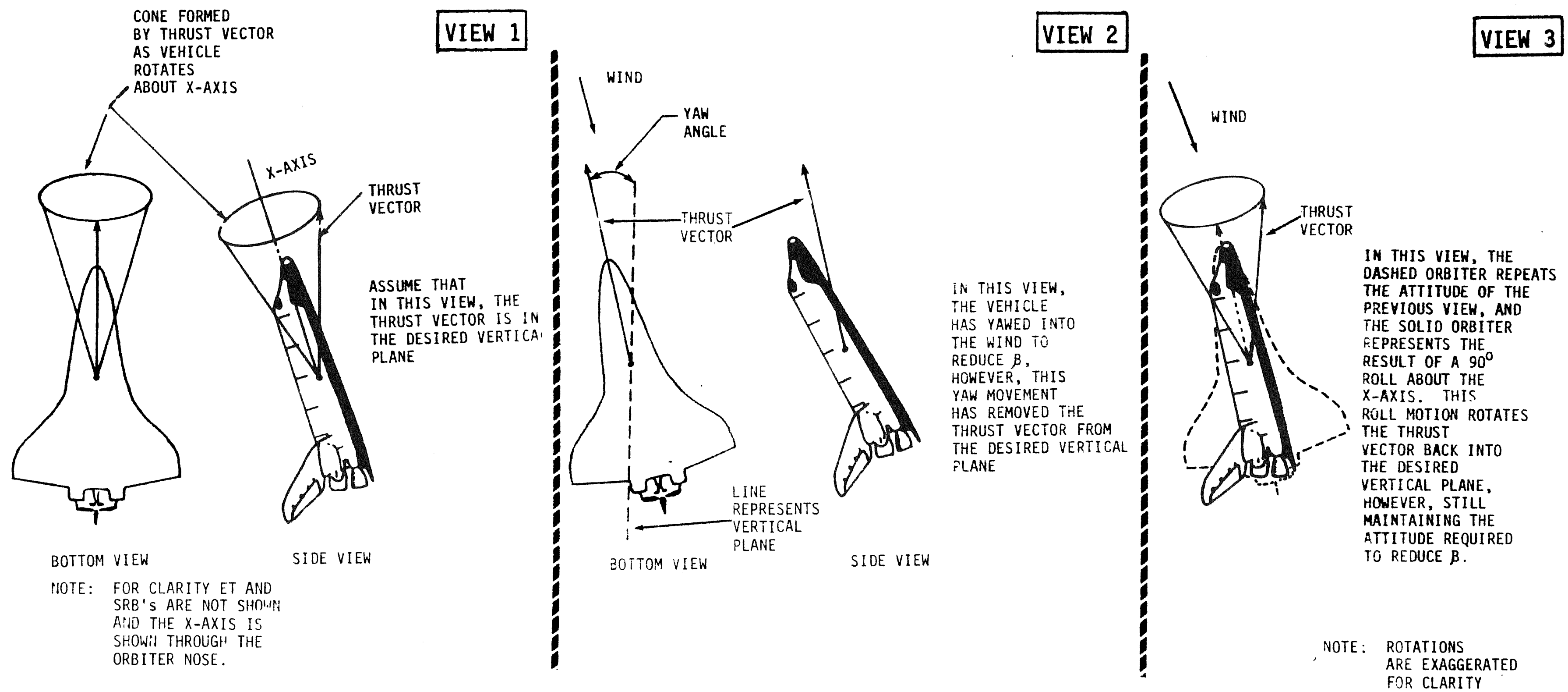


Figure 3-22.- Load relief operations.

EXERCISE

1. T/F. The DAP is an LRU located in the rear of the Shuttle. ______
2. The two feedback loops in the DAP are described as ________, and ________ feedback.
3. The inputs to TVC are:
   a. __________
   b. __________
4. T/F. The Shuttle flies upside down during ascent for structural purposes. ______
5. The sensor most valuable during load relief is __________.
6. T/F. The TVC software module generates engine bell roll, pitch, and yaw commands. ______
7. T/F. The autotrim function aids in removing large transient errors in the control loops. ______
8. T/F. Load relief ignores trajectory plane errors and is concerned only with reducing β. ______

EXERCISE ANSWERS

1. T/F. The DAP is an LRU located in the rear of the Shuttle. False

2. The two feedback loops in the DAP are described as rate and acceleration feedback.

3. The inputs to TVC are:
   a. attitude error
   b. rate command

4. T/F. The Shuttle flies upside down during ascent for structural purposes. False

5. The sensor most valuable during load relief is accelerometer.

6. T/F. The TVC software module generates engine bell roll, pitch, and yaw commands. False

7. T/F. The autotrim function aids in removing large transient errors in the control loops. False

8. T/F. Load relief ignores trajectory plane errors and is concerned only with reducing β. False

3.3.3.3.4 MPS Command Processor: The major function of the MPS command processor is to convert the body axis attitude error signals, that are generated in thrust vector control, into pitch and yaw engine bell deflection commands. (The pitch and yaw here describe engine bell deflection axes, not body axes). Before we go any further let's take a look at how the engine bells move.

In figure 3-23 a top, rear, and side view of the main engines is shown.

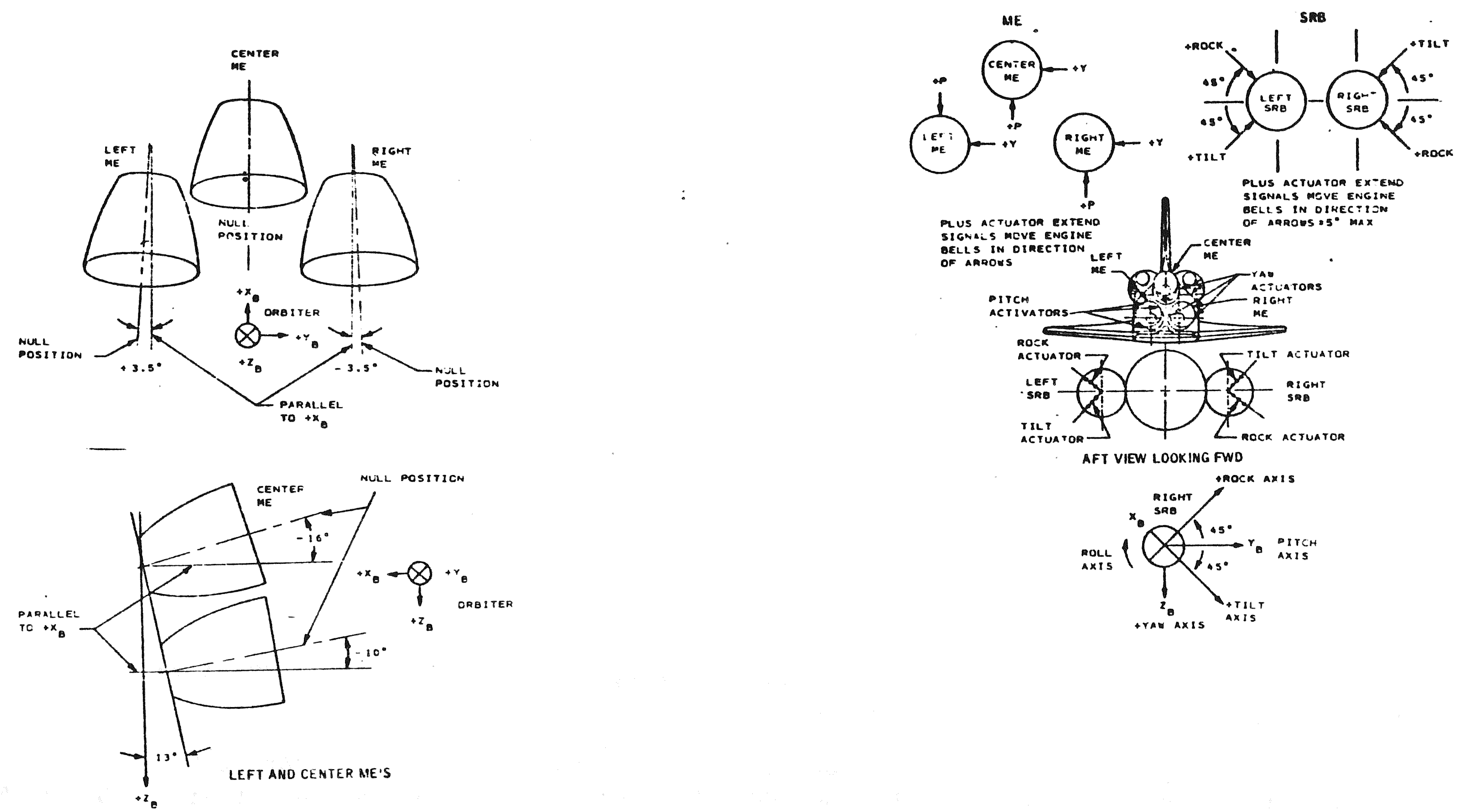


Figure 3-23.- Nozzle mounting information.

In the rear view, the arrows shown correspond to engine bell deflection in response to "positive" or "plus" actuator commands. Note that the left ME's pitch sign convention is the opposite of the other two. This is because of the physical location of the actuator. It is mounted so as to allow the ground crew to access the plumbing and other hardware in the rear of the vehicle. But back to the sign convention problem. The problem is rectified by changing the sign (multiplying by a "minus sign") in the MPS command SOP. It is the center and right engines, however, that have their sign changed; and we'll see why its done that way when we've learned a little more about the command processor.

Figure 3-24 is a functional view of the MPS command processor. The errors on the left are from TVC. Let's go through the figure by addressing the lettered boxes in order.

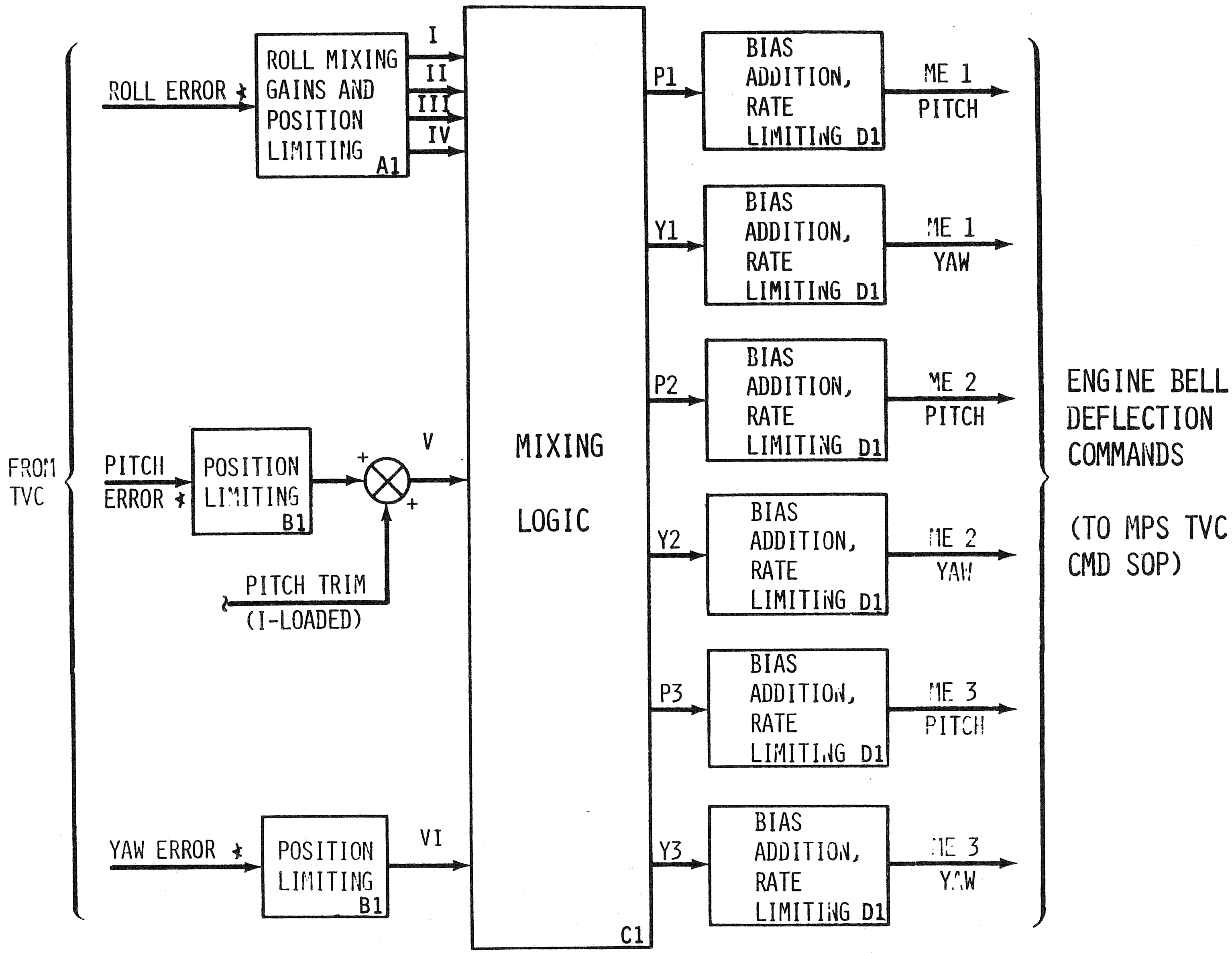
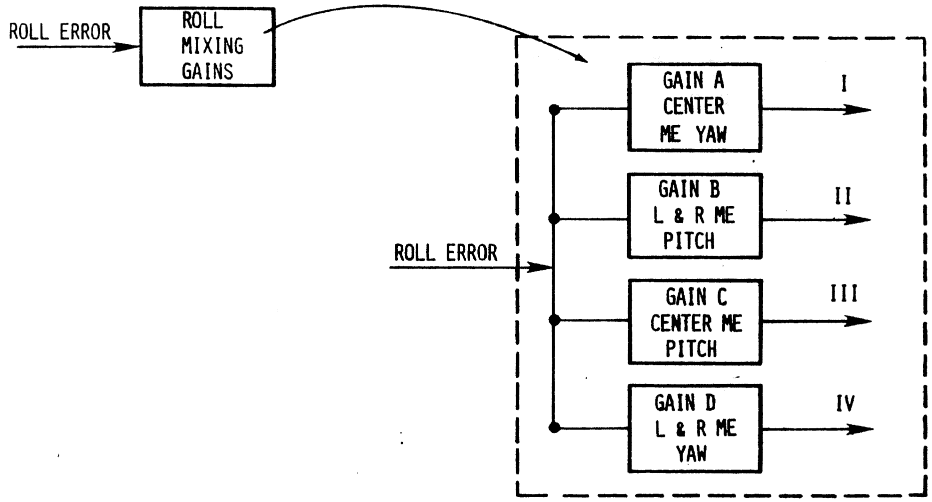


Figure 3-24.- Main Propulsion System command processing.

Starting with box A1, we see that the roll body axis error passes through mixing gains and limits. The mixing is required since to generate a vehicle roll, the engines must be deflected in some combination of (engine) pitch and yaw deflection. Therefore, the logic in box A1 converts the body roll error into the appropriate combination of engine bell pitch and yaw errors. This appropriate combination is dependent on the number of SSME's that are operating (i.e., not failed). When all three are operating, the roll error is converted to a left and right ME pitch error and a center ME yaw error. This is done simply by multiplying by various gains. This concept is easiest understood in chart form (fig. 3-25).



GAIN	CONDITION ①	CONDITION ②	CONDITION ③			④ TWO ME'S FAILED
			CENTER FAILED	LEFT FAILED	RIGHT FAILED	
A	0.75	0.75	0.75	2.0	2.0	0
B	1	2.5	2.5	2.5	2.5	0
C	0	0	0	2.5	2.5	0
D	0	0	0	2.0	2.0	0

CONDITION ① → (A) USE THESE GAINS WHETHER OR NOT THERE'S BEEN A ME FAILURE IF MET $<$ 90 SEC.

(B) USE THESE GAINS ONLY IF THERE IS NO ME FAILURE AND 90 SEC $\leq$ MET $<$ 120 SEC OR MET $\geq$ 90 SEC AND BOTH SRB PC'S $>$ 400 PSI

CONDITION ② → USE THESE GAINS IF THERE IS NO ME FAILURE AND BOTH SRB PC'S $\leq$ 400 PSI OR MET $\geq$ 120

CONDITION ③ → USE THESE GAINS FOR FAILURES AFTER 90 SEC MET

CONDITION ④ → USE THESE GAINS AFTER MET $\geq$ 120 OR BOTH SRB PC's $\leq$ 400 PSI AND TWO ME's HAVE FAILED

Figure 3-25.- Roll mixing gains.

Gains A through D each multiply the roll error from TVC. The resulting signals, I through IV, become pitch and yaw engine bell deflection errors due to body axis roll error. Sound like black magic? Well it isn't really. Let's go through the remaining logic and then do an example. Looking back at figure 3-24, the body axis pitch and yaw errors come in from TVC and pass through as ME deflection errors in pitch and yaw after being limited, (box B1). The limits are there to keep the amplitude of the error signal within a specified region. Limits are given on page 3-57.

So, now we have engine bell deflection errors for body axis roll, pitch, and yaw errors; a pitch and yaw error for body pitch and yaw error, and a pitch and yaw error for body roll error. Now the fun begins. Somehow all these errors must be combined (mixed) to form engine bell pitch and yaw deflection commands, box C1. And here's how it's done. With one big matrix! And here it is . . .

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Now, how do we use it, you ask. Well, its really quite simple - all we have to do is apply the basic rules of dynamics.

The matrix relates the computed errors to main engine deflection commands. However, the sign notation of the matrix is such that the signs are reversed so that when the matrix is used, deflection <u>commands</u> are generated. So, in order to compute the six (engine bell) deflection commands, you take the matrix and multiply it by the six error signals. It looks like this:

$$\begin{bmatrix} \text{CENTER ME PITCH CMD} \\ \text{CENTER ME YAW CMD} \\ \text{LEFT ME PITCH CMD} \\ \text{LEFT ME YAW CMD} \\ \text{RIGHT ME PITCH CMD} \\ \text{RIGHT ME YAW CMD} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{III} \\ \text{I} \\ \text{II} \\ \text{IV} \\ \text{V} \\ \text{VI} \end{bmatrix}$$

where the errors (designated by Roman numerals) correspond to those shown in figure 3-24.

What we have are six errors generated earlier. All we have to do is combine those errors appropriately and apply the proper sign convention. That's where the matrix comes in.

All we really have are six equations generating the appropriate deflection commands. Let's look at an example.

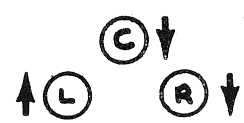
Let's assume, that the roll and yaw error from TVC are both zero. Let's next assume that the actual vehicle attitude in pitch is $+2^o$. The commanded vehicle pitch attitude is $+6^o$. Therefore, the body axis attitude error in pitch, assumed generated from TVC is actual minus commanded or $2 - 6 = -4^o$. So the pitch error coming into MPS Command Processor is -4^o. The mixing logic applies as follows:

$$\begin{bmatrix} \text{CENTER ME PITCH CMD} \\ \text{CENTER ME YAW CMD} \\ \text{LEFT ME PITCH CMD} \\ \text{LEFT ME YAW CMD} \\ \text{RIGHT ME PITCH CMD} \\ \text{RIGHT ME YAW CMD} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -4 \\ 0 \end{bmatrix}$$

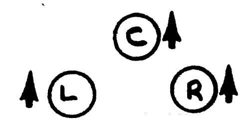
Therefore:

CENTER ME PITCH CMD = -4^o
LEFT ME PITCH CMD = -4^o
RIGHT ME PITCH CMD = -4^o

These deflection commands correspond to the following motions.



But recall that center and right pitch commands have their sign changed in the SOP, therefore the deflections become:



which creates a vehicle pitch up, which is the correct response to the guidance command.

The final column in figure 3-25 gives the roll mixing gain values when two ME's have failed. They are all zero. This means that there is no roll error signal going to the remaining ME. Instead, the roll error signal is fed into the RCS command processor. We'll see what happens in the RCS when we get to that chapter.

The blocks labelled D1 in figure 3-24 provide a bias addition and rate limiting of each of the six deflection commands.

Let's take a look at what happens (in box D1) to the pitch command of the center ME and call that the model since all the other commands are handled in the same manner. Refer to figure 3-26.

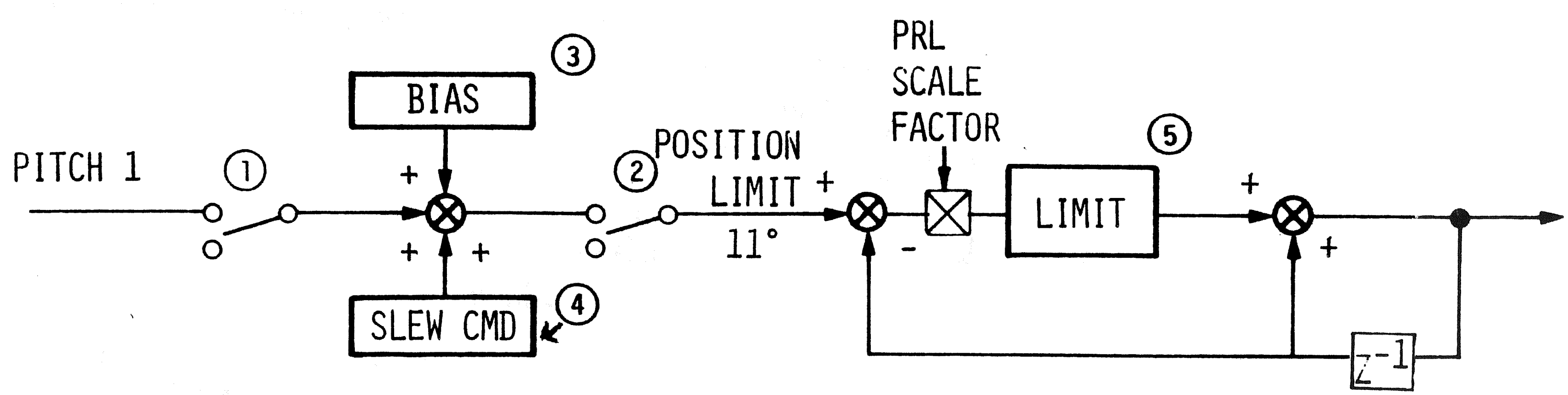


Figure 3-26.-Command biasing and limiting.

Switch 1 is closed (so-called "control loop closure") when lift-off has occurred, approximately MET = 0.32 sec. (Keep in mind these are software switches.) Switch 2 is normally in the upper position.

If the center ME were to fail, the switch would go to the lower position which would send a -3^{0} pitch position command. (Failed ME's are put in the failed position, (-3,0), (3,3), (3,-3) for ME's 1-3 respectively.) Save 3 for later.

The slew command, 4 , is used prior to lift-off to test the actuators. The slew command is $+2^{0}$ and -2^{0} deflection.

Eventually a pitch command comes out of switch 2 and into the 5 logic. This logic limits the rate of change of the command. Let's do an example. Assume the previous pitch command was $+7^{0}$. A second later, the command is -4^{0} (fig. 3-27). (For simplicity, 1 Hz Cycle time is assumed; however, the software is actually cycled at 25 Hz.)

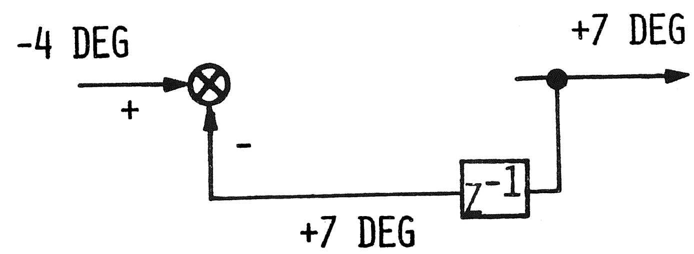


Figure 3-27

At the first junction we have: $-4 - 7 = -11^{0}$.

See figure 3-28. (The PRL scale factor is 1.0 in first stage.)

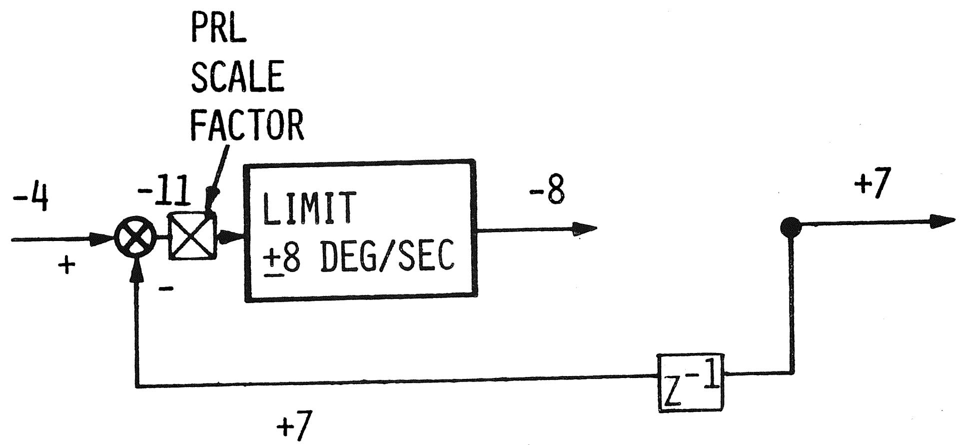


Figure 3-28

Then the rate limit is applied (limits depend on number of ME's failed and hydraulic system failed). Nominally the limit is ±8°/sec. So, since the -11° command exceeds the rate limit, only a -8° is passed through the limiter.

Finally, the signal is summed with the past value to generate the new command (fig. 3-29).

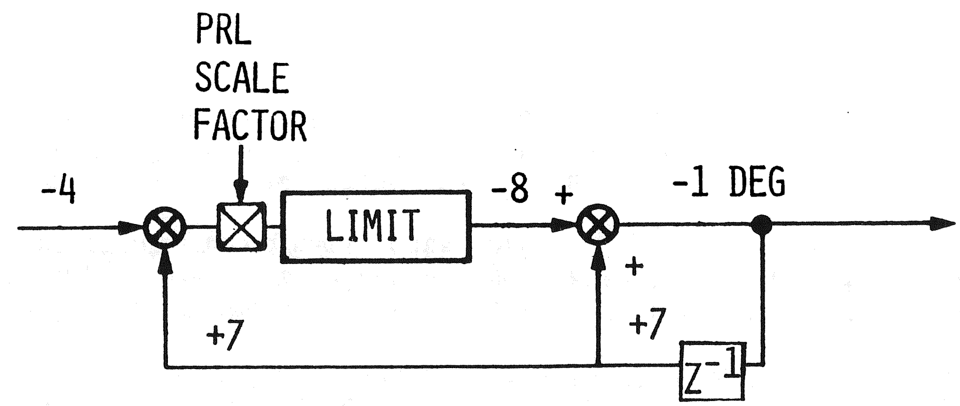


Figure 3-29

The new signal is -1° deflection in pitch, not the total -4° command since it would exceed limits. The slack in the command would be picked up in the next second. As was mentioned earlier, the rate limits depend on failures. The limits are:

8°/sec ∽ No failures

5°/sec ∽ Single hydraulic system failure

4°/sec ∽ Two hydraulic system failures

2.0°/sec ∽ In event of ME failure, limit failed engine only

No software limits on engine deflection are necessary to ensure that the commands do not exceed the maximum size (10 bits in 2's complement) which can be used by the MDM's. The software limits are ±11° in pitch, ±9° in yaw; the hardware limits are ±10.5° in pitch, ±8.5° in yaw.

Now, what about those biases (3) in figure 3-26. First of all, they are used prelaunch to command the ME's to the start position. Secondly, they are used after lift-off through MECO in order to parallel or unparallel the null axes in pitch and yaw. Now, what does that mean, you ask? Let's first look at yaw.

Note in figure 3-23 that the yaw null positions are offset from a line parallel to the body X axis by ±3.5°. To aid in thrust performance, a bias of 3.5° is added from the time the ME's are put in launch position until the MECO command. However, if the left or right ME fails, the bias is not added in second stage (between SRB SEP + 12 sec and MECO) so that total control authority is available. For the no failure case, the bias is removed at MECO CMD.

Now let's take a look at the pitch axes (fig. 3-23). The pitch axes are not parallel to the body X axis, they are canted up. (Canting helps rid heating problems on the belly of the Orbiter and also CG tracking problems.)

Therefore, a small amount of thrust performance is lost due to the angle generated by the null axis. Therefore, in the event of a left or right ME failure, and when in second stage, a bias is added to the center and remaining lower ME in order to parallel their null axes to each other. This effectively increases performance. (Bias is +3° for the top ME and -3° for the lower ME). Bias then must be removed when the vehicle mass gets below an I-loaded value (close to MECO CMD).

One other function within the MPS command processor is the addition of a I-load pitch trim (fig. 3-24). The trim is referenced to V_{REL}. The trim values are generated from simulation data. They take into account CG travel, winds, elevon movement, aerodynamic data, etc.

So, we've now generated the pitch and yaw engine bell deflection commands for the three ME's. The commands next go to the MPS TVC COMMAND SOP where the sign convention in pitch is rectified and the commands are scaled properly and converted to voltages for use by the actuators.

3.3.3.3.5 <u>SRB command processor:</u> This processor works in exactly the same fashion as the MPS command processor. Functionally, it takes the body axis roll, pitch, and yaw errors from TVC, adds a pitch trim (I-loaded), applies SRB mixing logic to generate SRB rock and tilt commands. Yes, rock and tilt! Here's why. The SRB's gimbal axes are at 45^o angles to the Orbiter body axes, therefore, the deflections are not pitch and yaw but they are tilt and rock (fig. 3-30).

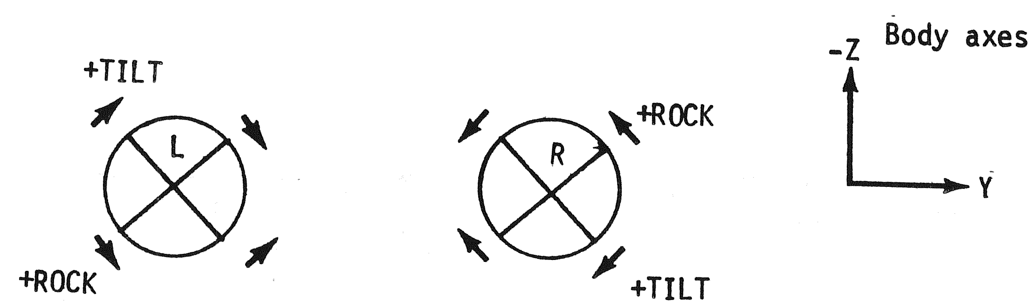


Figure 3-30

Figure 3-31 is a functional view of the SRB command processor. Since it is so similar to the MPS command processor, a detailed discussion will not be presented. The cos 45^o corrects for the gimbal axis rotation of 45^o. Just for your own curiosity, the mixing matrix is shown in figure 3-32.

The rate limit on the command signal is 7^o/sec. There are software deflection limits on the SRB's. They are:

$\pm 6.5^o$ ∽ Nominal

$\pm 2.8^o$ ∽ $P_C < 110$ psi (to prevent structural contact)

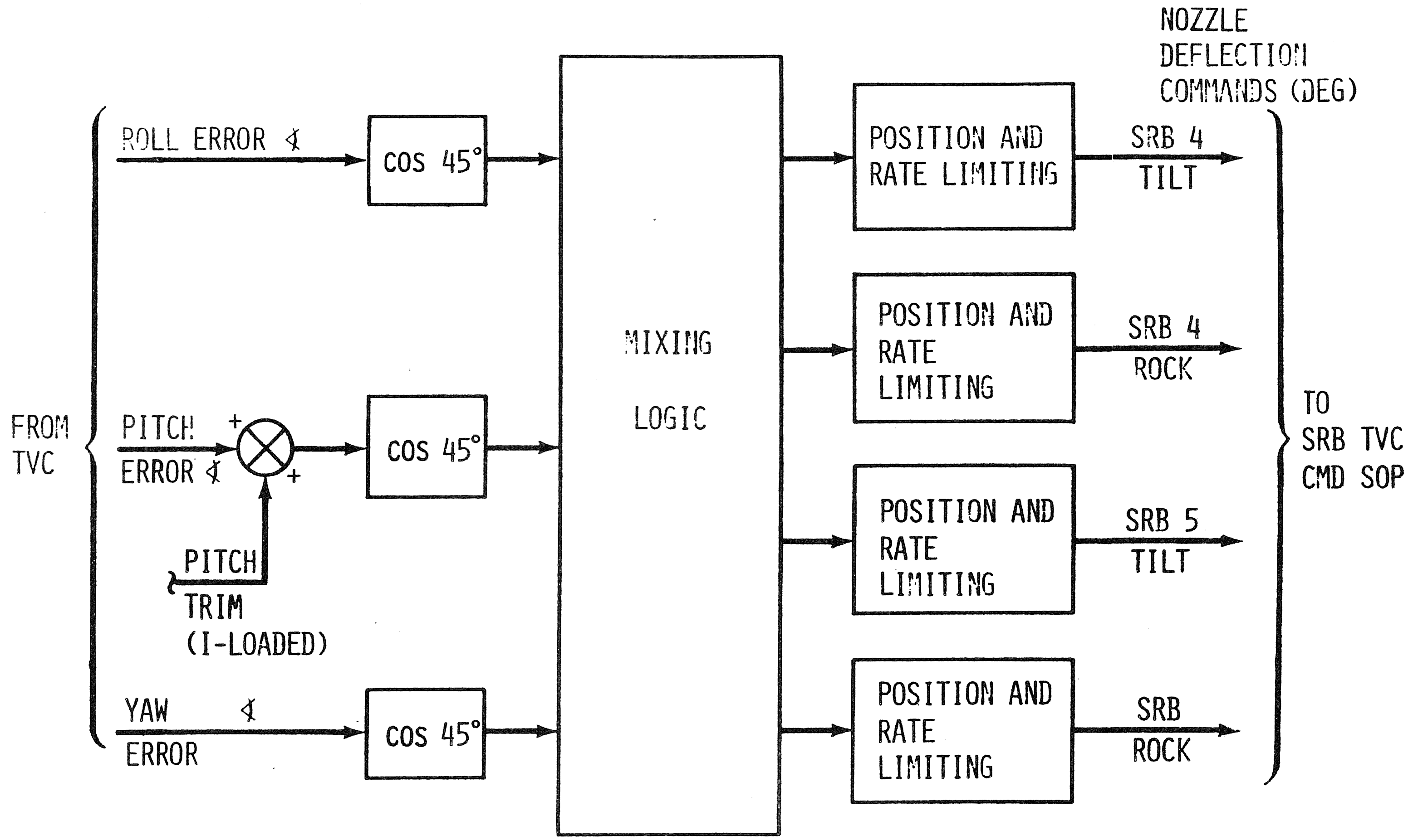
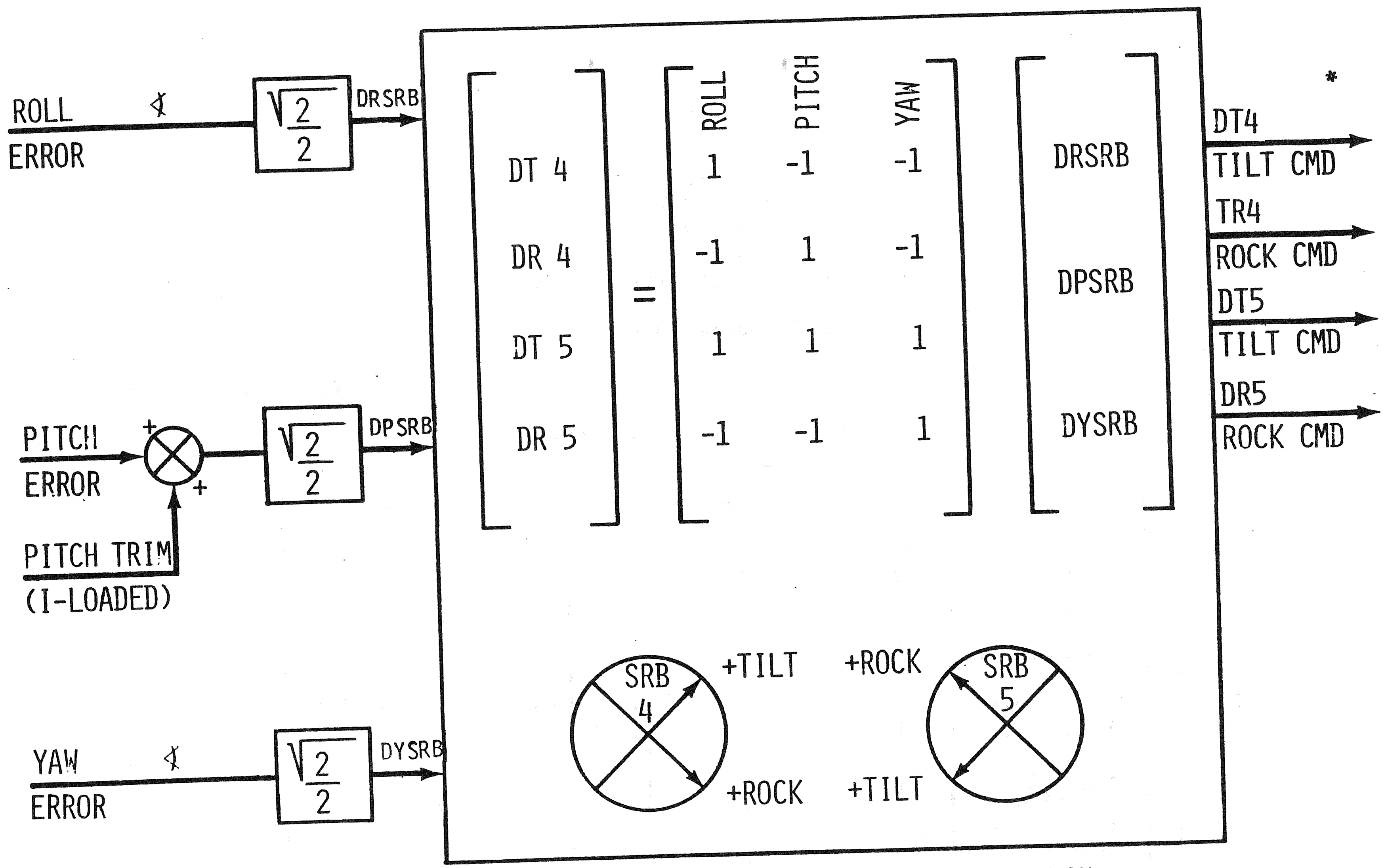


Figure 3-31.- Solid Rocket Booster command processing.



$$\begin{bmatrix} DT\,4 \\ DR\,4 \\ DT\,5 \\ DR\,5 \end{bmatrix} = \begin{array}{ccc} \text{ROLL} & \text{PITCH} & \text{YAW} \\ \left[\begin{array}{c} 1 \\ -1 \\ 1 \\ -1 \end{array}\right. & \begin{array}{c} -1 \\ 1 \\ 1 \\ -1 \end{array} & \left.\begin{array}{c} -1 \\ -1 \\ 1 \\ 1 \end{array}\right] \end{array} \begin{bmatrix} DRSRB \\ DPSRB \\ DYSRB \end{bmatrix}$$

Figure 3-32.- SRB command processing (mixing logic).

3.3.3.3.6 RCS command processor: The RCS is used to provide roll control during a single SSME maneuver (two SSME's failed). The roll error comes in from the MPS command processor and goes through a hysteresis loop with a deadband. If the error is outside the deadband, the appropriate RCS jets are commanded "ON" in order to null the roll error. The RCS logic is shown in figure 3-33.

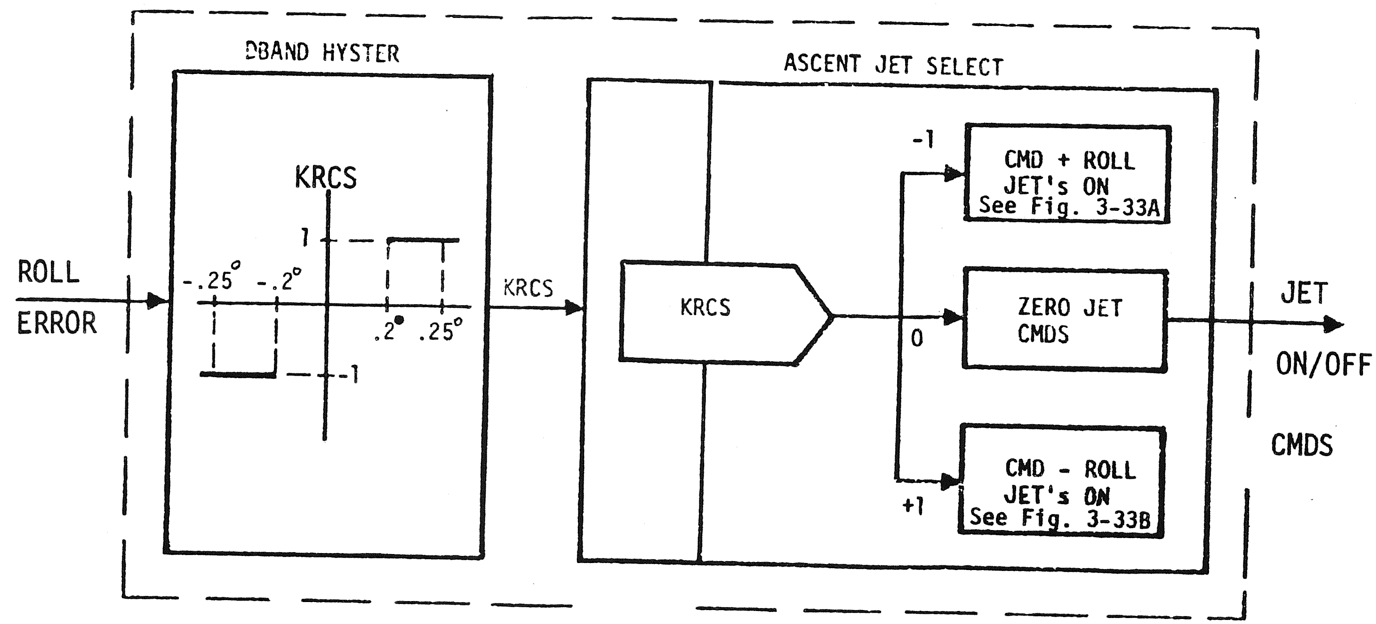


Figure 3-33.- RCS roll logic.

KRCS is computed as follows:

IF|Roll Error|≥|.25|
or
(KRCS=Sign (Roll Error)) and (|Roll Error|>|.2|)
then
KRCS = Sign (Roll Error) 1
otherwise
KRCS = 0

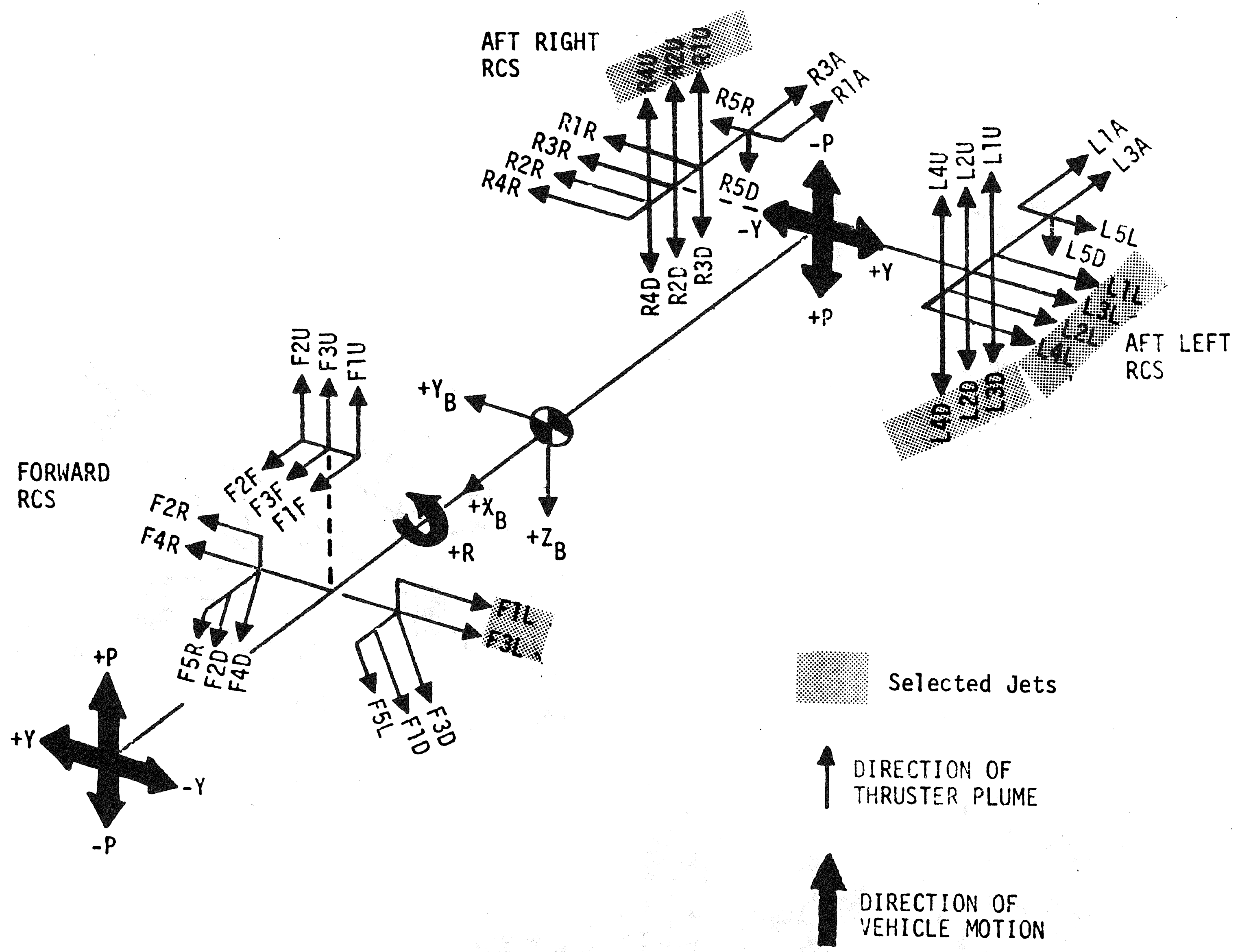


Figure 3-33A.- RCS +Roll jets for ascent.

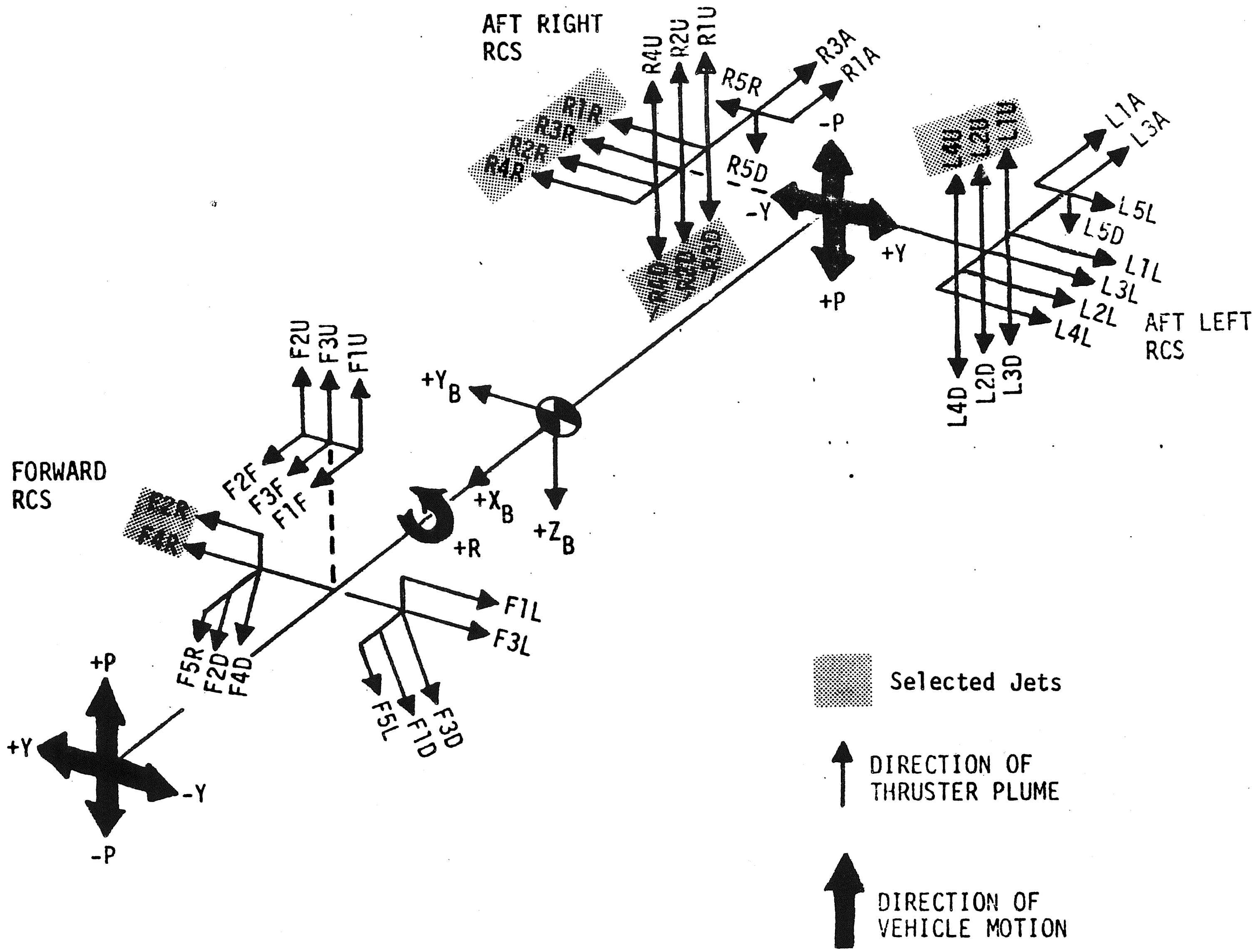


Figure 3-33B.- RCS -Roll jets for ascent.

3.3.3.3.7 Aerosurface control: This logic is responsible for holding the body flap and rudder/speedbrake in place during ascent. It also controls the elevons which are driven during the load relief region to reduce hinge moments. Since all surfaces but the elevons are held in place during ascent, only the elevons will be discussed here.

Basically, there is an I-loaded profile the elevons follow during first stage. They are not moved in second stage. There are also primary ΔP feedbacks which are utilized to perform real-time monitoring of the hinge moments just in case the I-loaded profile is not accurate. The elevons, when they are moved, are slaved. This means the left and right inboards are driven together, and the left and right outboards are driven together. The inboards are driven independent of the outboards, and visa versa.

Figure 3-34 is a functional view of the elevon logic. The same logic is done for the inboards and the outboards separately.

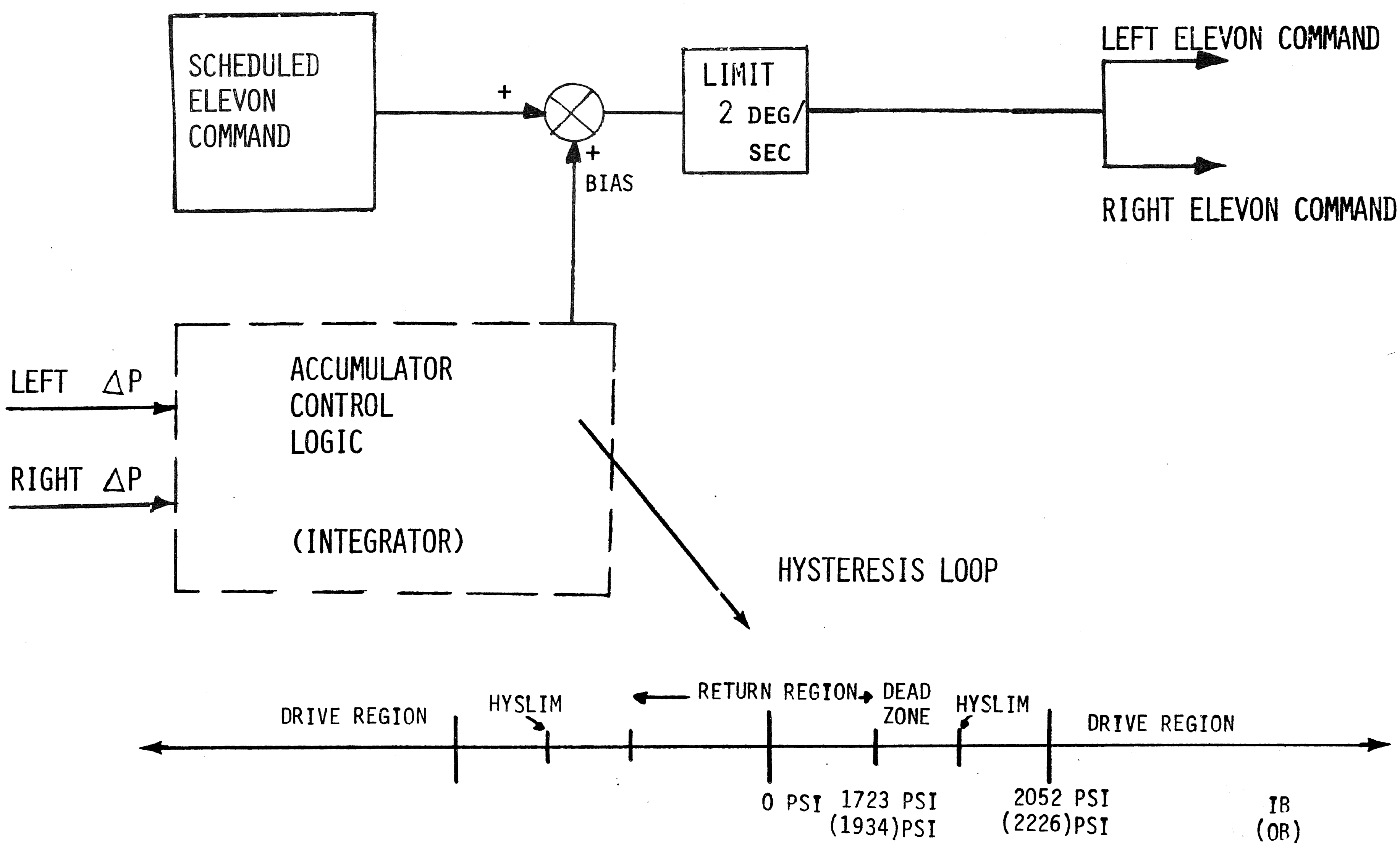


Figure 3-34.- Active elevon hinge moment load relief.

The left and right ΔP's go through a hysteresis loop with a deadband. If the ΔP's grow too large, the scheduled elevon command is biased so as to relieve the hinge moment. Let's take an example. Let's assume we're processing inboard elevon (IB)ΔP's.

Referring to the lower portion of figure 3-34, assume one or both ΔP's are large enough to be out in the drive region (>2052 psi). Therefore a bias is added to the scheduled command to reduce ΔP. As long as one or both of the ΔP's remain > than the HYSLIM, the bias is integrated over time. Finally, when both ΔP's are in the dead zone, the integration stops. When the ΔP's get into the return region, the bias begins being subtracted from the scheduled command. A similar example would apply to outboard (OB) ΔP's.

That concludes our discussion on the ascent digital autopilot and the automatic control system for the first stage.

EXERCISE

1. Place the number of the correct response in the lines provided.

   1. Reconfiguration logic
   2. Thrust vector control
   3. Attitude processor
   4. G/C STEER
   5. Linear interpolations
   6. Aerosurface control logic
   7. MPS command processing
   8. SRB command processing
   9. Autotrim
   10. Guidance
   11. RCS command processing

   _____ Mixes "body" roll, pitch, and yaw errors to provide deflection commands to thrusters.

   _____ Smooths automatic guidance commands.

   _____ Provides user with current M50 body quaternion.

   _____ Uses body mounted sensor data to compute body errors.

   _____ Provides appropriate trim values and scheduled gains to users.

   _____ Provides roll control in the event of two SSME failures.

   _____ Logic for handling errors such as thrust misalignment and mismatch.

   _____ Responsible for setting flags, lighting corresponding control mode lights, etc.

2. T/F. Roll body error is converted to pitch and yaw engine deflection errors in the MPS command processor. _______

3. T/F. The engine bells are never moved before lift-off. _______

EXERCISE ANSWERS

1. Place the number of the correct response in the lines provided.

    1. Reconfiguration logic
    2. Thrust vector control
    3. Attitude processor
    4. G/C STEER
    5. Linear interpolations
    6. Aerosurface control logic
    7. MPS command processing
    8. SRB command processing
    9. Autotrim
    10. Guidance
    11. RCS command processing

    7,8 Mixes "body" roll, pitch, and yaw errors to provide deflection commands to thrusters.

    4 Smooths automatic guidance commands.

    3 Provides user with current M50 body quaternion.

    2 Uses body mounted sensor data to compute body errors.

    5 Provides appropriate trim values and scheduled gains to users.

    11 Provides roll control in the event of two SSME failures.

    9 Logic for handling errors such as thrust misaligment and mismatch.

    1 Responsible for setting flags, lighting corresponding control mode lights, etc.

2. T/F. Roll body error is converted to pitch and yaw engine deflection errors in the MPS command processor. True

3. T/F. The engine bells are never moved before lift-off. False

3.3.4 Manual Control System (Manual Thrust Vector Control)

During first and second stage flight, the capability is available to manually control the vehicle's thrust vector. This is done by substituting the crew's inputs from the RHC for the automatic commands from guidance. The DAP still remains active, of course, to process the crew's inputs. We'll refer to the Manual Thrust Vector Control as MTVC.

MTVC is available beginning at lift-off (SRB IGN CMD + 0.32 sec). However, there is no "hot stick" capability. This means that any one of the CSS PBI's on panels F2 or F4 must be depressed before MTVC is available by way of the RHC. Once MTVC is activated (i.e., any CSS PBI depressed and lit) the vehicle is in a RATE COMMAND/ATTITUDE HOLD mode in all axes. In order to give control back to the automatic system, any of the AUTO PBI's must be depressed.

Now let's look at how RATE COMMAND/ATTITUDE HOLD is implemented. When the RHC is in detent, assuming MTVC has been selected, the vehicle is in attitude hold. This means that the DAP will hold the vehicle in the attitude it had when the RHC was put in detent. A rate command equal to zero replaces the rate command generated in the filter in G/C STEER. The attitude error for the DAP is computed by integrating the RGA measured rates and limiting the results. However, there are limits on vehicle rates, where, if the limits are exceeded when attitude hold is requested by placing the RHC in detent, attitude hold will not be initiated until the rates are within limits. The limits are:

10.0^{o}/sec ∽ Roll rate limit for initiation of attitude hold

3.0^{o}/sec ∽ Pitch rate limit for initiation of attitude hold

3.0^{o}/sec ∽ Yaw rate limit for initiation of attitude hold

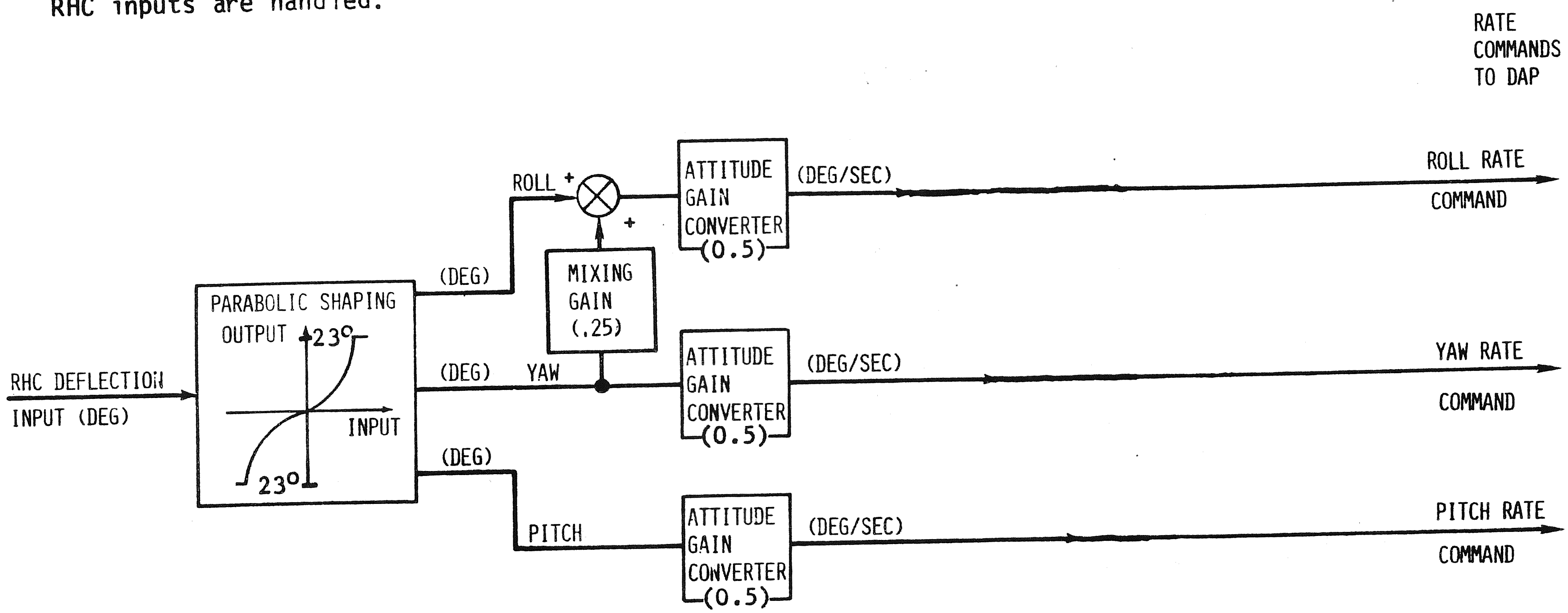
(The low pitch and yaw rate limits were set with RTLS powered pitchdown in mind.)

20.0^{o} ∽ Roll attitude error limit for initiation of attitude hold

20.0^{o} ∽ Pitch attitude error limit for initiation of attitude hold

20.0^{o} ∽ Yaw attitude error limit for intiation of attitude hold

As soon as the RHC is removed from detent, a rate command, proportional to the amount of deflection, replaces the rate command previously generated. The attitude error is zeroed. Figure 3-35 shows how the RHC inputs are handled.



- ATTITUDE ERRORS ARE ZEROED AS SOON AS RHC IS REMOVED FROM DETENT

Figure 3-35.- Manual steering - rate command/attitude hold (MM 102 and MM 103).

The crossfeeding of yaw into roll is necessary for stable controllability of the roll axis. (It has to do with the fact that the Orbiter/stack CG is located below the crew. It helps reduce pilot workload by providing the crossfeeding.)

3.3.5 Attitude Director Indicator

3.3.5.1 Attitude director (8-Ball).- The 8-BALL, a three dimensional enclosed ball, driven in three degrees of freedom, provides Orbiter roll, pitch, and yaw attitude information to the crew. The angles that are displayed are generated in the attitude processor from IMU data. For a detailed discussion of the ADI, see the GNC Dedicated Display Workbook.

3.3.5.2 Attitude error needles.- The attitude error needles, located on top of the 8-Ball itself, provide error information in roll, pitch, and yaw to the crew.

In first and second stage, the error needles are driven by the attitude error that is generated in G/C STEER.

There is a modification of the error needles during the load relief region. Since the needles are driven from G/C STEER data, and load relief acts on the signals in the DAP, downstream of G/C STEER, the needles wouldn't reflect what load relief was doing in the high q region. What would happen is that the attitude errors would grow very large because load relief would actually fly the vehicle away from the planned trajectory to relieve Orbiter loads. It is necessary to modify the error needles so they reflect the difference between the actual body attitude and load relief commanded attitude. This is done by summing the original attitude error with the compensated acceleration error (converted to the proper units) generated in the DAP.

This compensated acceleration error is the main driver of load relief in the DAP. So, now, the needles will reflect (by not displaying a large error) what load relief in the DAP is doing. Figure 3-36 shows how this is implemented.

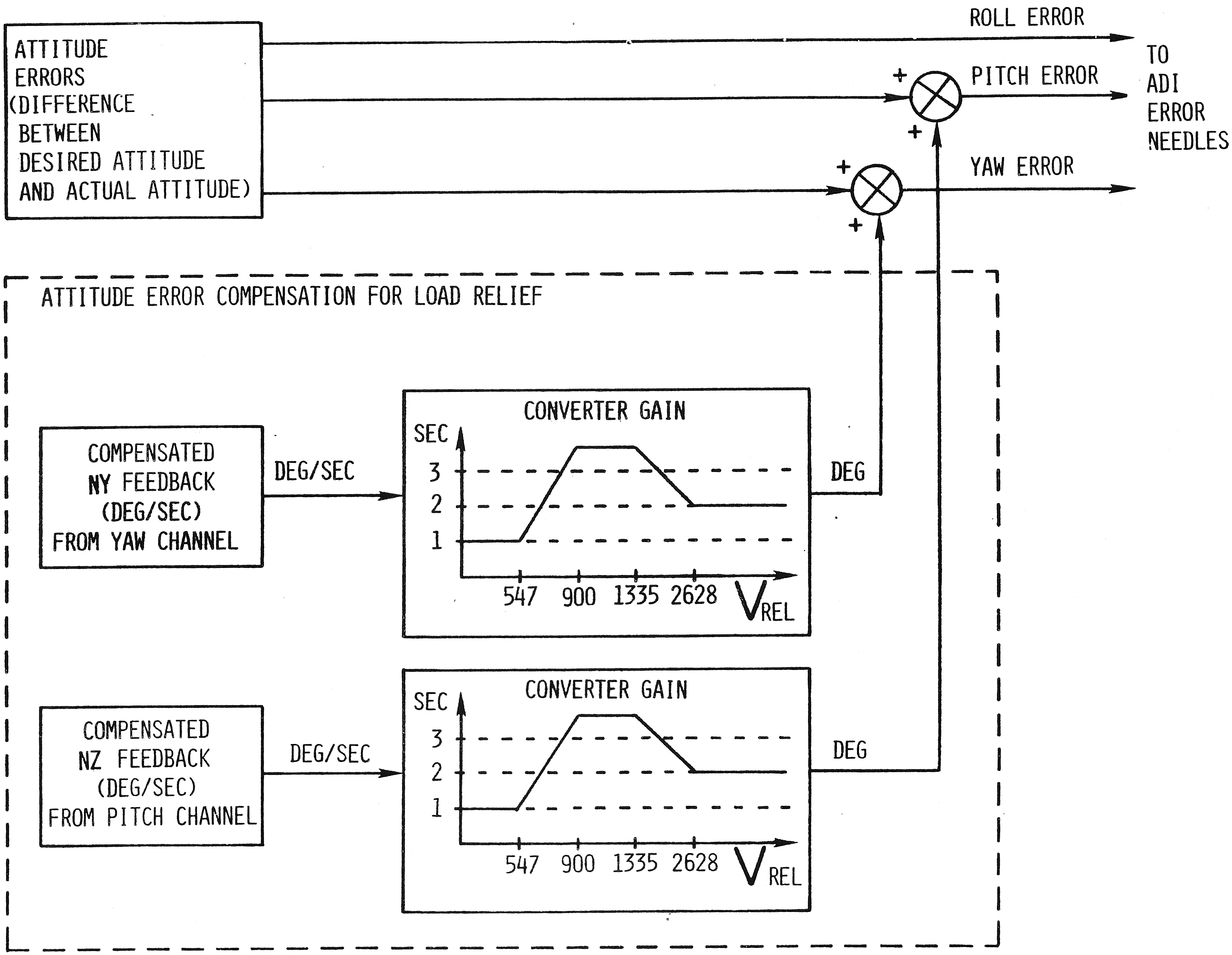


Figure 3-36.- ADI error needle computation.

3.4 ADDITIONAL SOFTWARE SEQUENCES

3.4.1 SSME Operations Sequence

3.4.1.1 Introduction.- The SSME operations sequence is initiated at T-0 (SSME 90 percent) and is used during the ascent phase to: 1) monitor the operating phase of each main engine, 2) issue inhibit commands to prevent a second engine from automatically shutting down if one shutdown has already occurred, 3) monitor the state of cockpit switches via flags from the switch processor and issue appropriate commands, 4) monitor a flag from GN&C software for proper time to check the LO2 and LH2 low level sensors, 5) monitor a cutoff timing request flag from GN&C software for proper time for MECO, 6) issue main engine shutdown commands when required, and 7) close LO2 and LH2 prevalves for each engine after shutdown has occurred. In addition, an SSME data path fail flag from the SSME SOP is checked and, if set and shutdown commands have been issued for that engine, then its prevalves are closed after a time delay. A MECO confirmed flag is set by the SSME Operations sequence after it has been confirmed that all engines have entered the shutdown phase. Also, a flag is set for the External Tank Separation sequence after the prevalves for all engines have been commanded closed. This is necessary since shutdown times may differ.

3.4.1.2 Description.- The SSME Operations sequence is initiated when the RS Launch sequence and sets the T-0 flag. At this point the main engines are at or above the required thrust, but the engine controllers will not enter the mainstage phase until 5.0 seconds after receipt of the start commands. The SSME OPS sequence operates cyclically from initiation at T-0 until MECO is verified and a few seconds after the prevalves are commanded closed. Under normal operation throughout ascent there are no commands to issue until the end of second stage when the main engines must be shutdown and certain MPS valves closed.

Continuous monitoring of certain inputs is required, however, in the event that: 1) an automatic shutdown by one engine occurs, 2) the crew operates any of the manual switches, 3) data from any of the main engines is lost, or 4) near the end of boost, either the LH2 or LO2 low level sensors indicate fuel or oxidizer depletion.

If an automatic shutdown of an engine occurs, for whatever reason, the LO2 and LH2 prevalves for that engine must be closed after an appropriate time delay, and the remaining two engines are inhibited from performing an automatic shutdown.

The switch processor software monitors the position of several crew station switches for the main propulsion system, and sets flags which are monitored by SSME OPS. The MPS switches being monitored are the three shutdown pushbutton switches and the limit shutdown switch, which can override the automatic inhibit logic and inhibit or enable engine automatic shutdown.

The SSME OPS sequence monitors the main engine operating phase; and, if any engine leaves the mainstage phase and enters the shutdown phase, the appropriate MPS valves are closed. The operating phase and mode within a phase is determined from the engine status word by the SSME SOP which gathers the engine data from the EIU, decodes, and sets applicable flags for the various user software packages. If valid data is not available from the engine, SSME SOP sets a data path fail flag. If this flag is set and shutdown commands have been issued, SSME OPS proceeds after a time delay to close the engine prevalves.

SSME OPS, upon receiving a flag from guidance, begins monitoring the fuel and oxidizer low level sensors. A check is made to see whether any sensors have been disabled during prelaunch checkout because of faulty indications. If any two LO2 low level sensors, which have not been disabled, indicate a dry condition, the logic will issue the MECO commands. Likewise, two LH2 low-level sensors, which indicate dry and have not been previously disabled, will cause the issuance of MECO commands.

Normal engine shutdown (MECO) is triggered by the vehicle achieving the desired velocity and a flag being set by guidance software. The MECO commands issued include shutdown enable and shutdown commands through the EIU's to each of the three main engine controllers. The MECO commands are repeatedly issued until it is determined that each engine has entered the shutdown phase. At this point a MECO confirmed flag is set for GN&C applications and to initiate the external tank separation sequence.

SSME OPS continues to operate until the required time after prevalves for each engine are commanded closed and then the close commands are removed. Since the shutdown for each engine may be either hydraulic or pneumatic with different time requirements for prevalve closures, a check is made of the shutdown mode for each engine; and, when LO2 and LH2 prevalves for all three engines have been commanded closed, a flag is set for the ET separation sequence. This is required to prevent initiation of ET disconnect valve closure prior to initiation of all prevalve closures. When the prevalves close commands have been removed and the flag set, the SSME OPS sequence is terminated.

3.4.2 Vent Doors Sequence

3.4.2.1 Introduction.- The Orbiter's vent and purge system is made up of 18 active ports and is divided into the following 6 groups: left and right ports 1 and 2, left and right port 3, left and right port 5, left and right ports 4 and 7, left and right port 6, and left and right ports 8 and 9 (fig. 3-37).

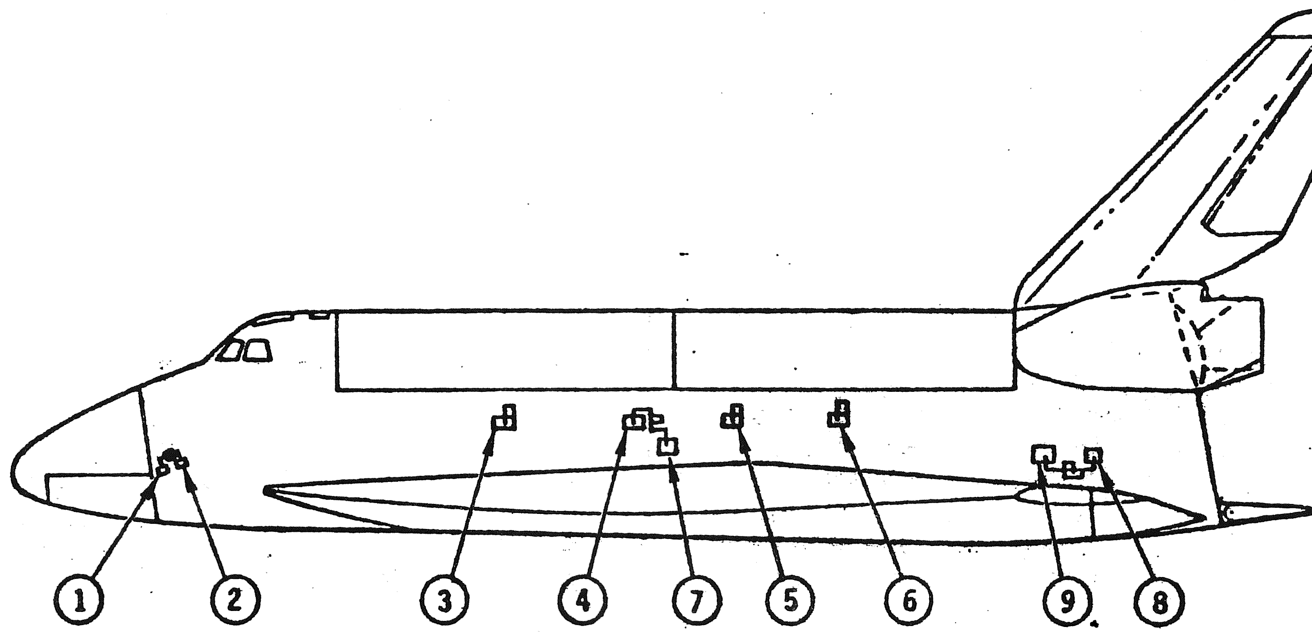
All vent ports have a purge position with the exception of left and right vent ports 3, 4, 5, and 7.

The vent and purge system is used for equalizing the pressure across the outer surface of the Orbiter and to permit molecular venting of the vehicle cavities and insulation blankets to achieve the required low internal blanket pressure. The purge position is required to maintain a positive pressure in the Orbiter's payload bay area to prevent contamination and to vent any residue gases in the payload bay area while in the ground-turnaround phase.

Operation of the vent and purge system is controlled exclusively through software; consequently, there are no dedicated manual controls or displays in the cockpit.

3.4.2.2 Description.- The sequencing of the active ports is by software program in the redundant set computer. The ports are cycled to the open, close, or purge position, as required, in each mission phase. Positioning of the active ports is performed by the software based on mission times or mission events during ascent, entry, aborts, and by keyboard entry during the on-orbit phase. Vent port status will be displayed on a CRT display page (on-orbit).

Upon receipt of a cue from the RS launch sequence to configure the vent ports to a launch configuration, vent ports 1 and 2 and 8 and 9 will be commanded to the OPEN position and all other vent ports will be commanded to the CLOSE position or all vents will be commanded to the "OPEN" position. The status of the vent ports position will be output to the RS launch sequence to determine that the vent ports have achieved the launch configuration within the specified time. The Orbiter will be launched with the vent ports in this configuration, and at T+10 seconds all vent doors will be commanded to the OPEN position. In a nominal mission, the vent ports will remain open until the crew closes the ports with a keyboard entry prior to deorbit. If during the launch phase (T-10 to T-0) a launch abort has occurred, the vent doors system will be reconfigured to the prelaunch configuration by the LPS.



VENT NO.*	COMPT VENTED	VENT DOOR SUBSYSTEM
1	FWD RCS	FORWARD
2	FWD FUS	
7	WING	PAYLAOD BAY AND WING
4	MID FUS	
5	MID FUS	PAYLOAD BAY
3	MID FUS	
6	MID FUS	
8	OMS POD	AFT
9	AFT FUS	

*LH AND RH

Figure 3-37.-Orbiter vent doors.

EXERCISE

1. An RHC input is seen as a __________ command in the DAP.
2. T/F. The attitude error needles reflect the action of load relief. ________
3. Normal MECO is triggered when the vehicle achieves _________________________.
4. Name the manual controls available for operation of the vent and purge system.

   ____

EXERCISE ANSWERS

1. An RHC input is seen as a rate command in the DAP.

2. T/F. The attitude error needles reflect the action of load relief. True

3. Normal MECO is triggered when the vehicle achieves targeted velocity.

4. Name the manual controls available for operation of the vent and purge system.
CRT page only

3.5 DEDICATED DISPLAYS/CRT DISPLAYS

3.5.1 Dedicated Displays

The dedicated displays that are driven in first stage are shown in figure 3-38. It may be beneficial for the student to read the Dedicated Display Workbook to get a general feel for how the displays work.

Note in figure 3-38 that the tape meters display parameters other than what is titled at the top of the meter.

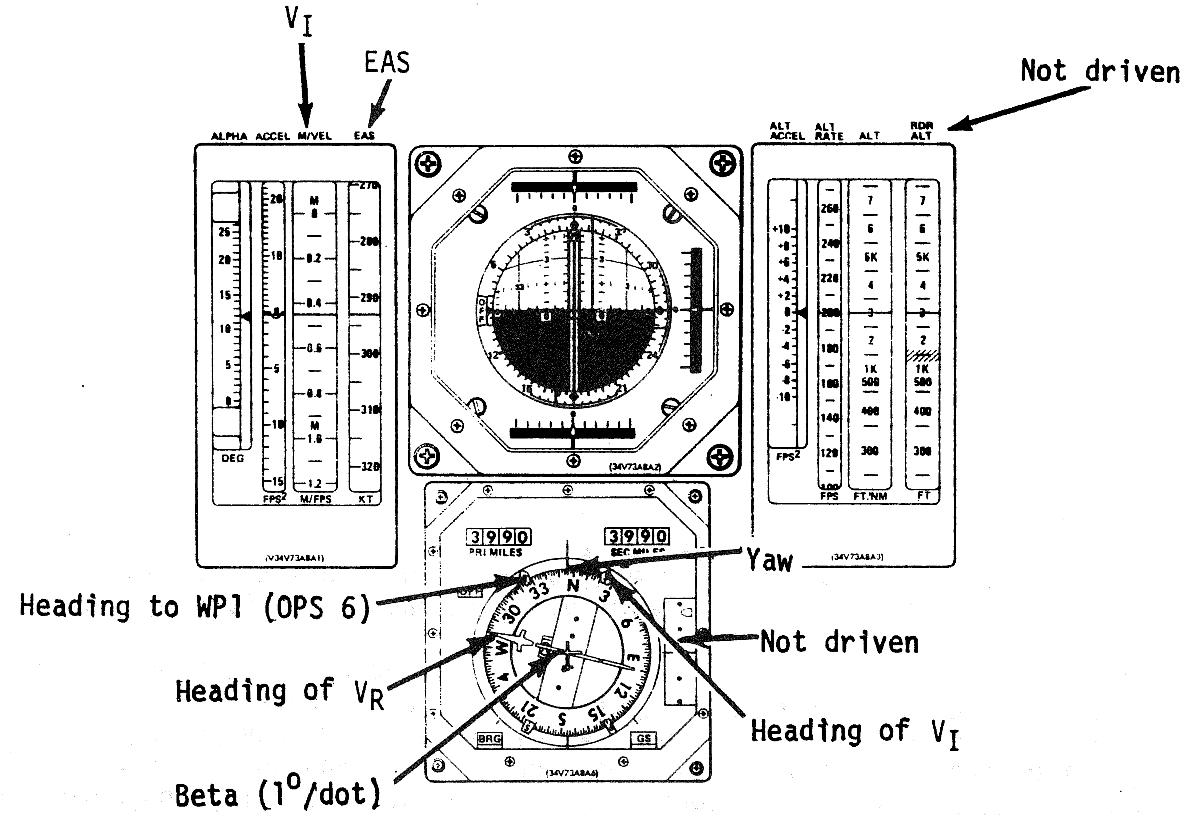


Figure 3-38.- Dedicated displays active in first stage.

3.5.2 CRT Displays

The OPS display that is available in first stage (it is also the OPS display in second stage) is shown in figure 3-39. The beginning of the plot is actually the staging point or SRB SEP. Since the display plot starts at staging, the plot is not really applicable to first stage. However, on the left hand side of the display, the words SEP INH and P_C <50 apply to SRB separation. P_C <50 appears on the display when that discrete is sent to the staging sequence. "SEP INH" is displayed flashing if the SRB SEP inhibit discrete is sent by the SRB SEP sequencer.

The part of the plot originating just above RTLS and going to the left is the plot for the latest RTLS. The horizontal line at the top is a plot of Δ range (i.e., glide range potential based on energy state minus present range from the landing site in nautical miles).

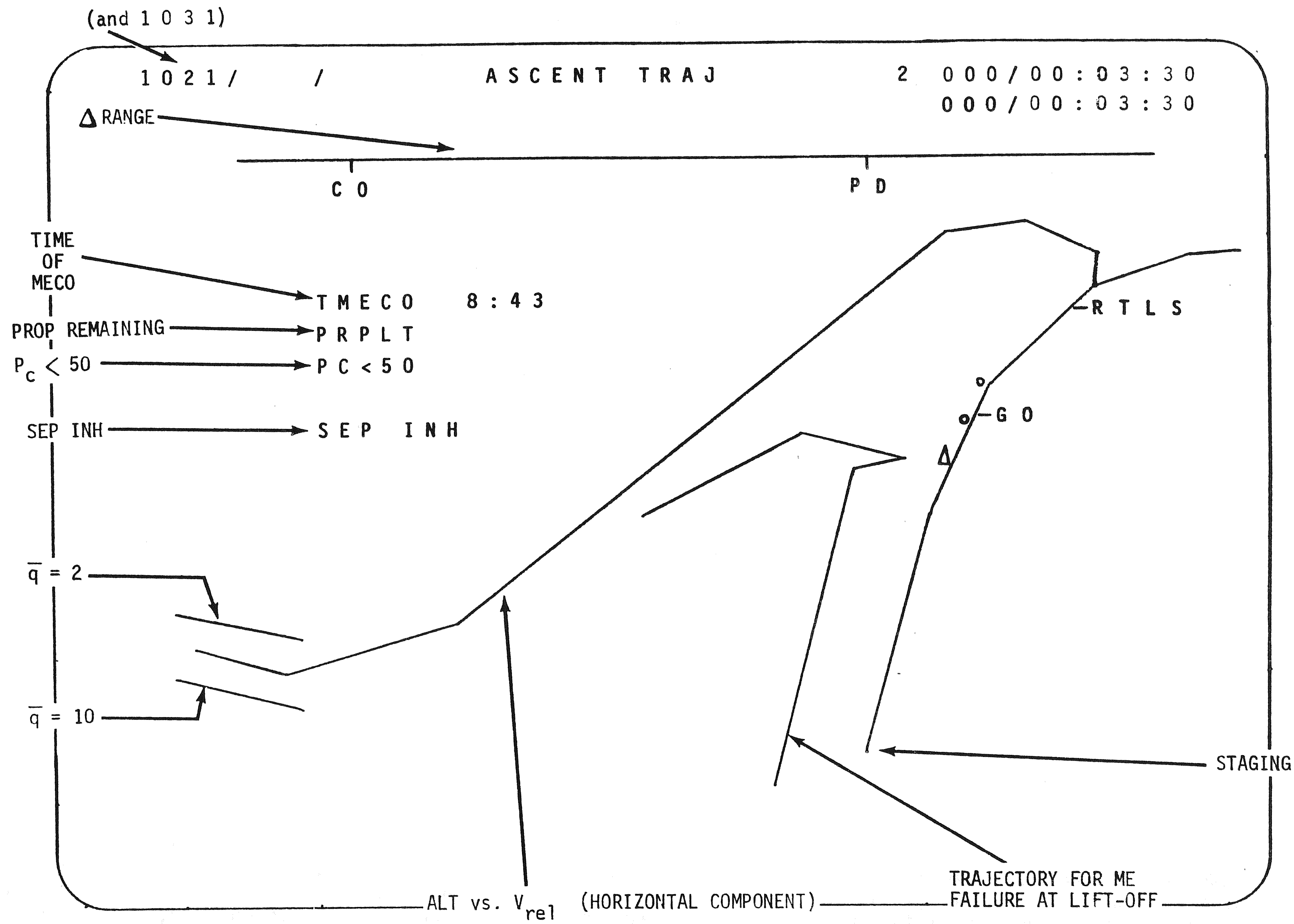


Figure 3-39.- First and second stage OPS display.

The BFS is supporting a TRAJ 1 display. Figure 3-40 shows this display. This plot starts from the lower left at lift-off.

The ADI roll, pitch, and yaw attitude errors are displayed so the crew can compare the BFS to the prime. (NOTE: The BFS is supplying the attitude errors on the CRT display, and the PFS is supplying the attitude errors to the ADI.) The attitude errors will be displayed on the CRT only preengage. As soon as the BFS is engaged, the errors are blanked.

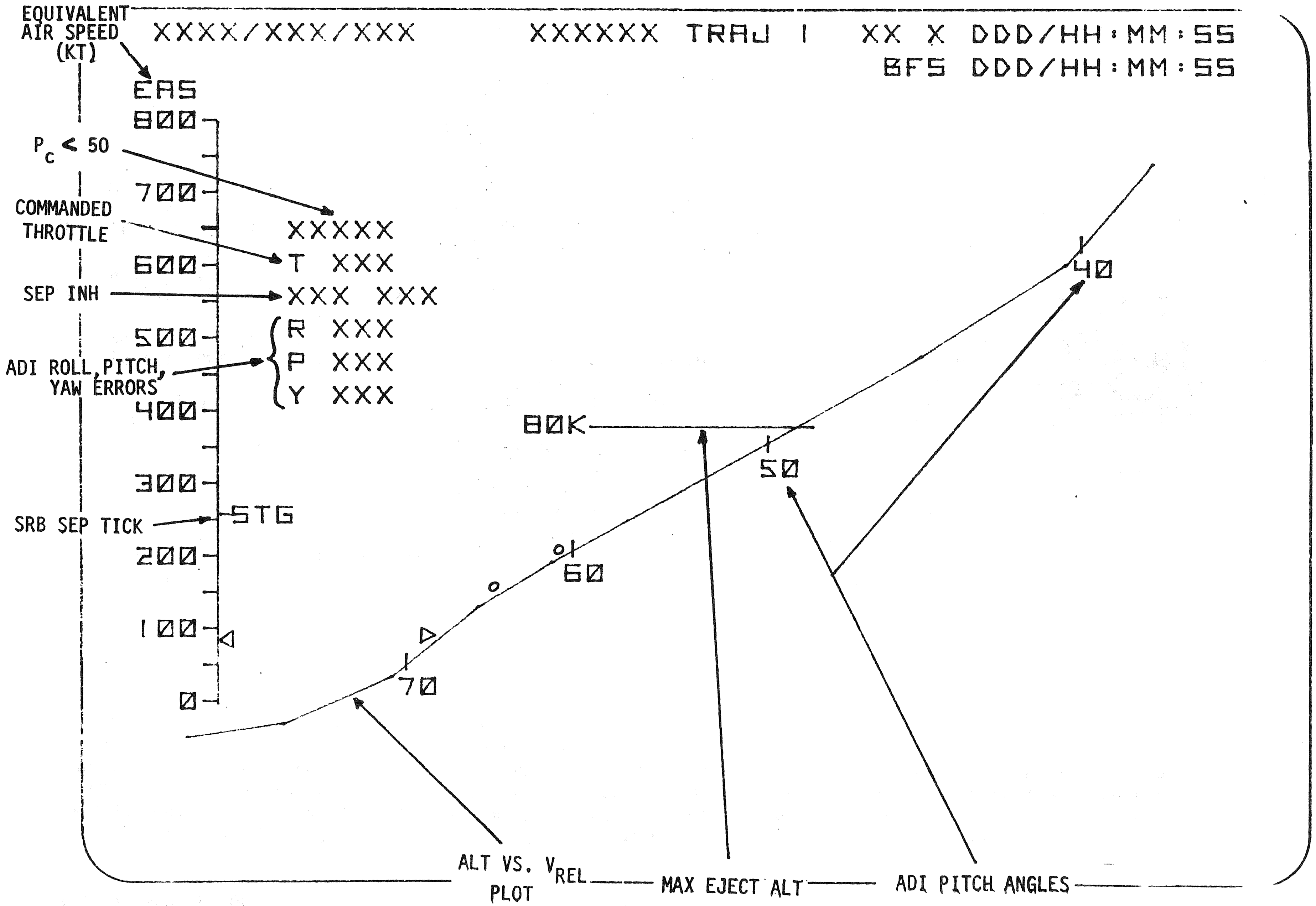


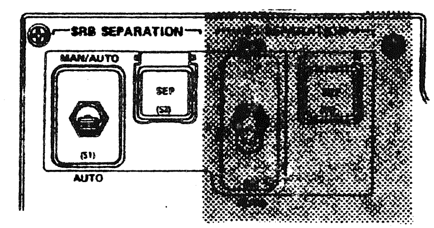
Figure 3-40.- BFS first stage display.

3.6 SRB SEPARATION

SRB separation is normally handled completely and automatically by the GPC's. However, the crew can command separation manually by way of the SRB SEPARATION switch and PBI located on panel C3. (Switches shown below.) When the switch is in the auto position, separation will occur as soon as the appropriate conditions are met concerning SRB chamber pressure, body rates, and dynamic pressure. When the switch is in the MAN/AUTO position, the crew can separate the SRB's by way of the SEP PBI. However, as soon as the appropriate conditions are met (mentioned above) the SRB's will separate automatically, hence the name MAN/AUTO.

SRB separation is controlled by an automatic software sequence in the GPC's. Since it is a sequence, let's go through it in chronological order.

(1)	MET $\geq$ 90 sec	: Enable NY feedback loop for pre-SRB separation control (fig. 3-20). : Disable elevon load relief.
(2)	If (1) above and both SRB P_c's < 400 psi or MET $\geq$ 120 sec	: Begin fading integral trim values to zero. : Begin fading I-loaded SSME pitch trim to -3.1^o.
(3)	If (2) above and both SRB Pc's < 110 psi	: Limit SRB nozzle deflection to $\pm 2.83^o$.
(4)	If (3) above or Met $\geq$ 124 sec	: Fade NY feedback to zero over 4 sec.
(5)	If Both SRB Pc's < 50 psi or MET $\geq$ 131.7 sec	: Initiate separation software. (See page 3-87 for the continuation of the separation logic)



(6)	When t > time at (5) above +4.3 sec	: Command SRB nozzles to the NULL position. : Mode FC to monitor only Orbiter RGA's.
(7)	When t = time at (5) above +6.0 sec and rates and $\bar{q}$ are within limits OR (6) above has occured and manual SEP has been selected	: MM103 : Fade pitch trim to -1.2°. : Initiate three axis attitude hold. : Issue separation command. : Initiate 2nd stage guidance for convergance.
(8)	When t > time at (7) above +4 sec	: Engage second stage guidance for control.
(9)	When t = time at (7) above +12 sec	: Engage trim integrators (±4°).

Figure 3-41 shows a nominal separation.

TIME	EVENT
105	1 MONITOR SEPARATION CUES
	Proceed to step 2 if:
122	a) Pc<50 psia on both SRB's and Δt between cues is < 4.2 sec. (If > 4.2 sec, then go to step 1 (b).)
	b) MET > 131 sec
	2 PREPARE FOR SEPARATION
122 (131)	a) Arm separation PIC's
	b) Safe range safety system (S&A device to SAFE)
	c) Remove range safety system battery power
	Wait 4.3 sec:
126.3 (135.3)	d) Null SRB TVC actuators
	e) Transition to stage II FCS configuration
	Wait 1.7 sec
128 (137)	f) Terminate SRB ATVC power
	g) Terminate SRB ATVC fault detection circuitry
	3 CHECK SEPARATION INHIBITS
	a) Monitor the following crew control positions SRB SEP sw - MAN/AUTO SRB SEP sw - AUTO SRB SEP pb - pushed
	Go to step 4 if: SRB SEP sw - MAN/AUTO SRB SEP pb - pushed otherwise continue step 3
	b) Monitor the following: Pitch, yaw, and roll rates Dynamic pressure (Q)
126* (135*)	If rates and Q within limits, go to step 4.
	If any rate or Q out of limits, then:
130* (140*)	SM ALERT lt - on SM ALERT tone - on SEP INH (CRT msg line) Return to step 3 (a)

SRB SEP SEQUENCE (Concluded)

TIME	EVENT	TIME	EVENT
	4 COMMAND SEPARATION		
128* (137*)	a) Issue SRB SEP FIRE 1&2 commands		
	***********************		
	SRB SEPARATION OCCURS		
	***********************		
	5 COMPLETE SEPARATION SEQUENCE		
132* (142*)	a) Turn off SRB power buses		
	b) Remove all MEC commands		
	Wait 0.04 sec		
	c) Reset MEC 1&2		
	d) Terminate SRB SEP SEQ		

*plus Δ TIME between sep inhibit and manual override of inhibits or restoration of inhibit within limits

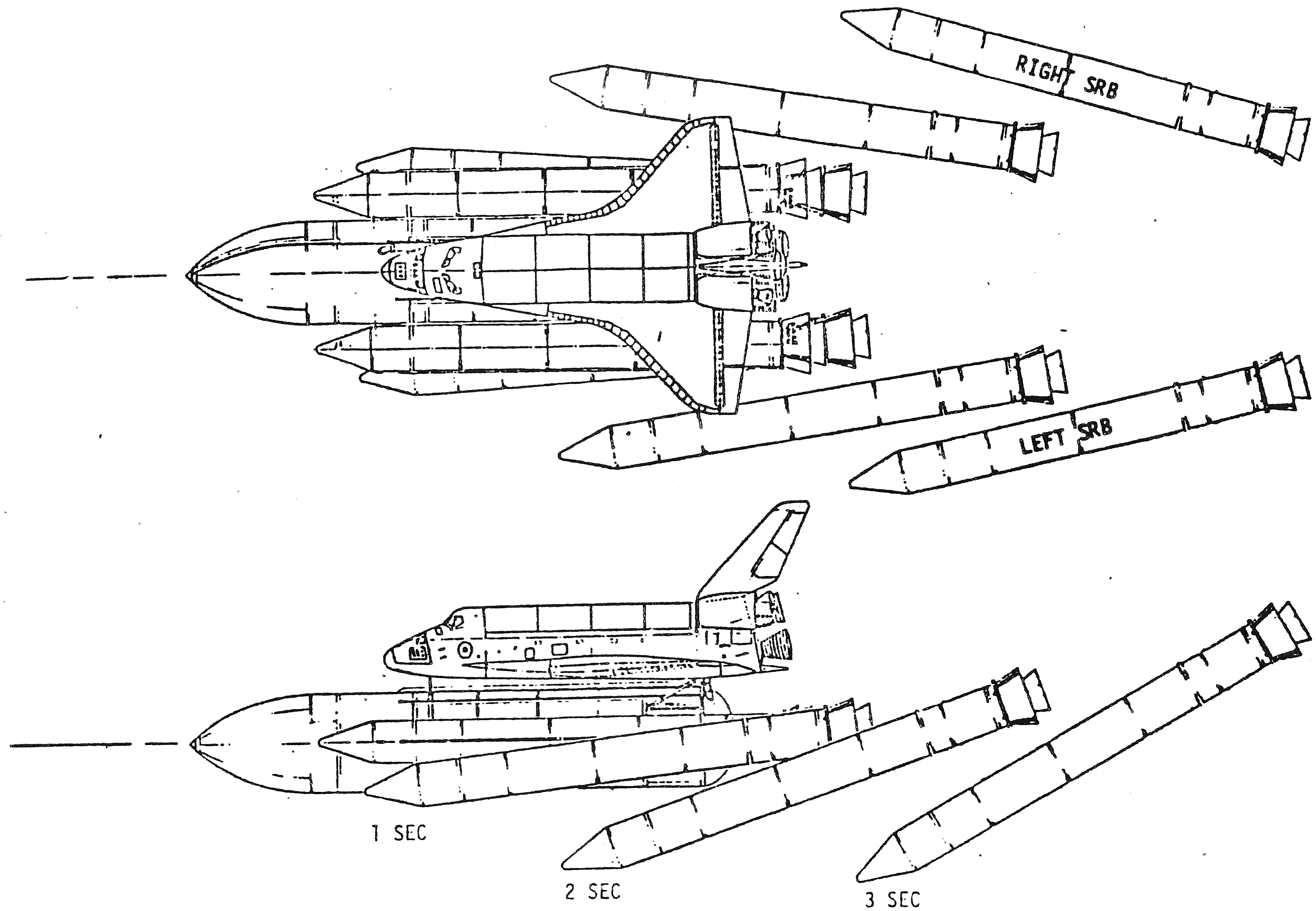


Figure 3-41.- Typical SRB separation movement.

EXERCISE

1. T/F. SRB SEP is always a manual procedure. _______
2. The actual separation command comes __________ the nozzles are nulled.

EXERCISE ANSWERS

1. T/F. SRB SEP is always a manual procedure. False
2. The actual separation command comes ___after___ the nozzles are nulled.

SECTION 4
ASCENT SECOND STAGE

4.1 NAVIGATION

Second stage navigation is exactly the same as first stage, so review first stage at this time if you think you need it.

4.2 GUIDANCE

Guidance, however, has completely changed from first stage. It is no longer an open loop program referencing an I-loaded profile. Now, guidance is a closed loop iterative function which computes each cycle where the vehicle should be in the sky so that an I-loaded MECO target can be reached.

4.2.1 Powered Explicit Guidance

Powered Explicit Guidance (PEG) is a guidance scheme consisting of four independent algorithms developed to handle all phases of Shuttle exoatmospheric powered flight. In this workbook only one of the four algorithms will be presented since it is the only one pertaining to second stage flight.

The algorithm which will be discussed here is called PEG 1 (sometimes shortened to PEG) or standard ascent guidance. The objective of PEG 1 is to generate commands to place the vehicle in a desired position with a desired velocity and with minimal fuel usage. The desired position and velocity are determined from a set of I-loaded target parameters. The target for PEG 1 is the MECO target. The parameters defining the MECO target are as follows (fig. 4-1):

V ∽ target velocity

λ ∽ flightpath angle

R ∽ target radius from earth center

IY ∽ desired orbital plane

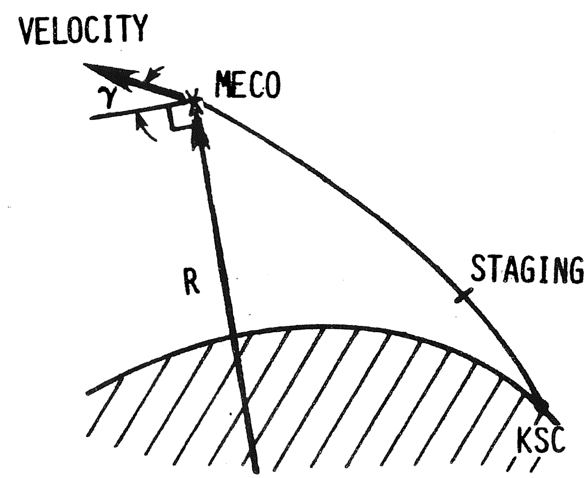


Figure 4-1.- MECO target.

The MECO target conditions are selected primarily to insure correct placement of the external tank footprint.

PEG DISCUSSION

In second stage flight, PEG assumes three distinct hypothetical phases will occur. PEG is initiated at SRB SEP CMD and sets up the three hypothetical phases as shown in figure 4-2.

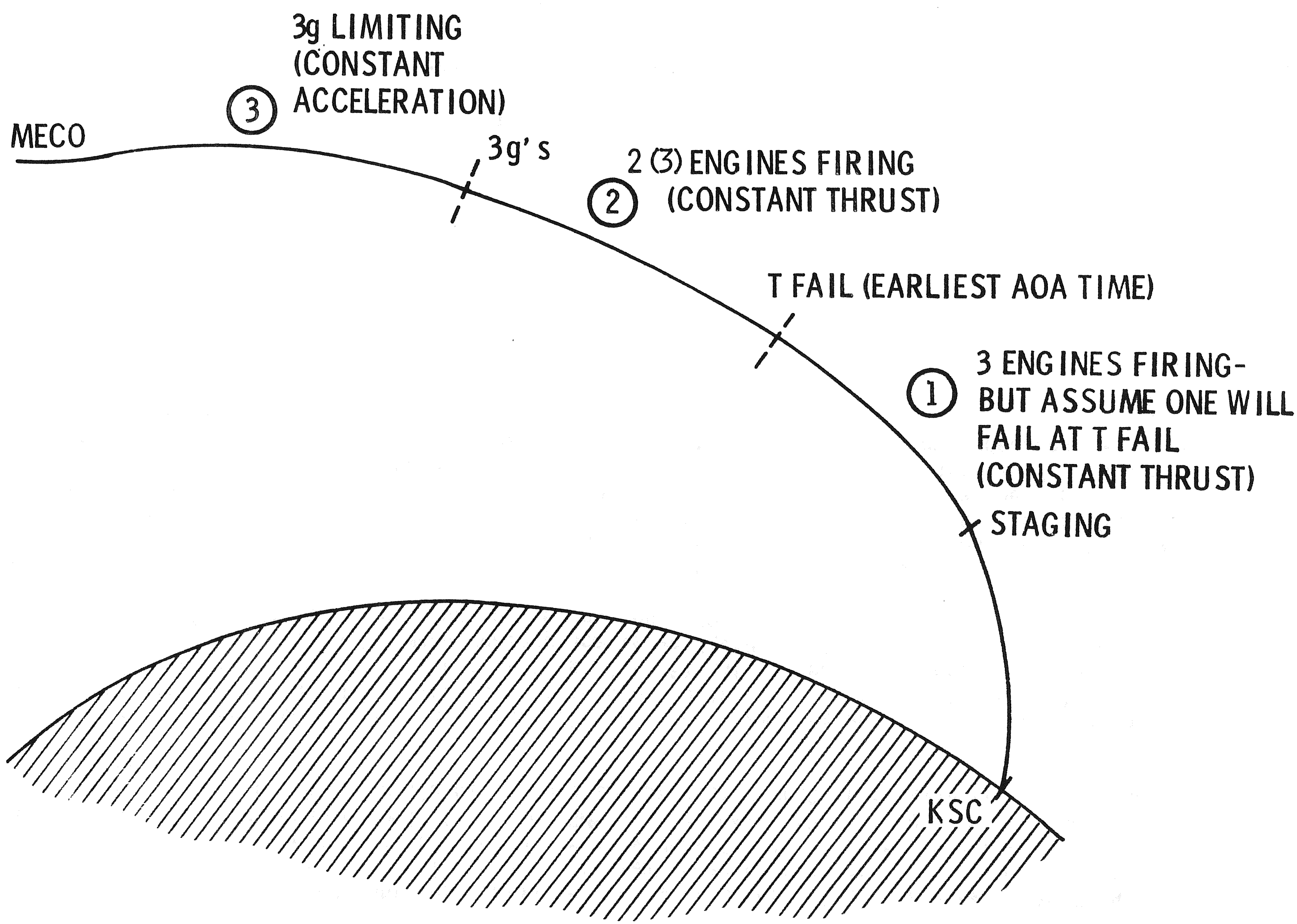


Figure 4-2.- Hypothetical guidance phases.

The first phase extends from the SRB SEP command until the earliest time at which an AOA abort is possible with a single SSME failure. This time is known as T FAIL. Throughout this first phase, PEG assumes that one SSME will fail at T FAIL. This assumption causes PEG to loft the trajectory during the first phase so that the MECO conditions can be reached using only two SSME's after T FAIL. This provision is included to prevent a gap between the latest RTLS and the earliest AOA.

The second phase begins at T FAIL and continues until an acceleration of 3 G's is reached. At T FAIL, PEG stops assuming that an engine failure will occur. The remainder of the second stage will be made assuming three active SSME's. (Unless, of course, one really does fail). So, what has happened at T FAIL is that PEG has gone from assuming only two good SSME's to assuming three good SSME's. The effect of this sudden change in assumed thrust would be to move the predicted MECO state away from and beyond the desired conditions.

This will cause a large correction to be made to the PEG steering parameters and a certain convergance condition (to be discussed) will not be satisfied. It should require two to four guidance cycles (one cycle takes 2 sec) for PEG to reconverge. During this period, Flight Control will continue to use the last steering parameter values received. When convergence is achieved, a significant pitch change (25° or so) will be commanded in order to steer down to the desired MECO conditions.

The third phase begins when a thrust acceleration of 3 G's is reached. At this point, a guidance task, external to PEG, will begin to adjust the SSME throttle command so that the acceleration does not exceed 3 G's. This phase continues until guidance is simplified* at MECO - 10 sec. If the vehicle weight is sufficiently heavy, or if SSME performance is degraded, the 3 G limit may not be reached before MECO. In this case, the third phase will not be entered and the second phase would continue until MECO - 10 sec. Also, if an engine failure were to occur during the third phase, the acceleration would decrease and the second phase would be reentered.

*to be discussed later

If an engine fails at any time other than T FAIL, PEG will temporarily lose convergence and a pitch attitude change will be commanded when it reconverges. However, if an engine fails at T FAIL, there would be no pitch transient because that is the situation that guidance is planning on.

Now that we know the phases that guidance assumes will occur, let's take a look at how guidance actually "guides" the Shuttle into space. It's important to know that the main objective of PEG is to compute thrust parameters for flight control. These thrust parameters are, more explicitly, a reference thrust vector in M50 coordinates, a thrust turning rate vector (unit vector), and a reference time associated with these vectors.

It can be shown that a fuel optimal trajectory can be achieved if the tangent of the thrust attitude angle is a linear function of time. The thrust attitude angle is the angle between the thrust direction and the vector velocity to be gained to reach the target conditions (fig. 4-3). This condition can be expressed as the equation of a line:

$$\tan \theta = At + B$$

where A and B are constants, and t is the time from the beginning of the maneuver; in this case, SRB SEP.

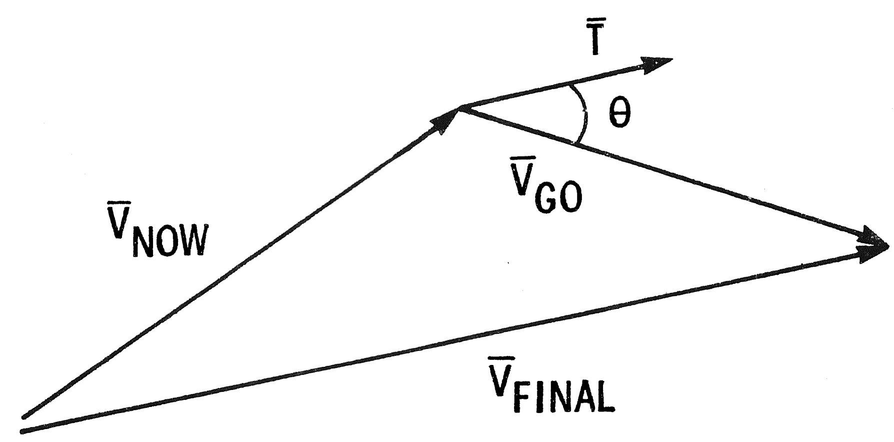


Figure 4-3.- Thrust attitude angle.

This condition will be satisfied if the commanded thrust direction is computed as follows: (See figure 4-4.)

$$T = \dot{T}\,(t - \tau_R) + T_R$$

where $\bar{T}$ is the commanded thrust direction vector

T_R is a reference thrust vector (unit vector). This vector is parallel to the velocity vector ($\bar{V}_{GO}$) which represents the velocity to be gained to meet MECO target conditions.

$\dot{\bar{T}}$ is the turning rate vector. It is perpendicular to T_R and indicates the direction and rate at which T will rotate during the maneuver.

t is the time from the beginning of the maneuver.

τ_R is a reference time which determines the point at which T and T_R are parallel.

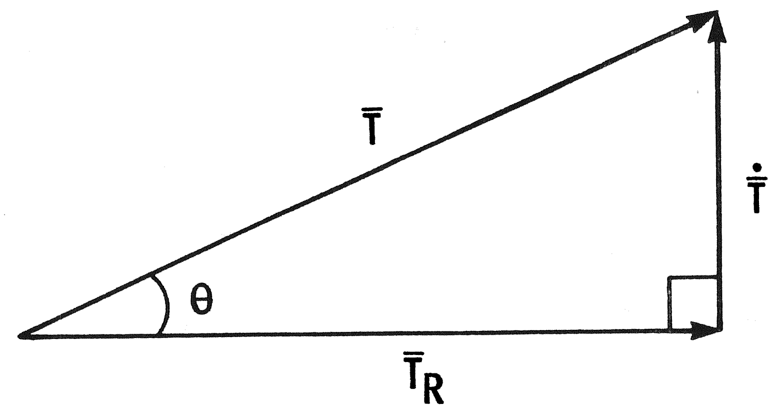


Figure 4-4.- Thrust steering law.

Now let's see where the thrust parameters (steering parameters) come from.

After PEG is initialized at staging it executes cyclically once every two seconds. At the end of each cycle, a velocity-to-go vector ($\bar{V}_{GO}$) is used to begin the next cycle. $\bar{V}_{GO}$ represents the vector difference between the desired (target) velocity and the current velocity (fig. 4-5).

$$\bar{V}_{GO} = \bar{V}_{DESIRED} - \bar{V}_{NOW}$$

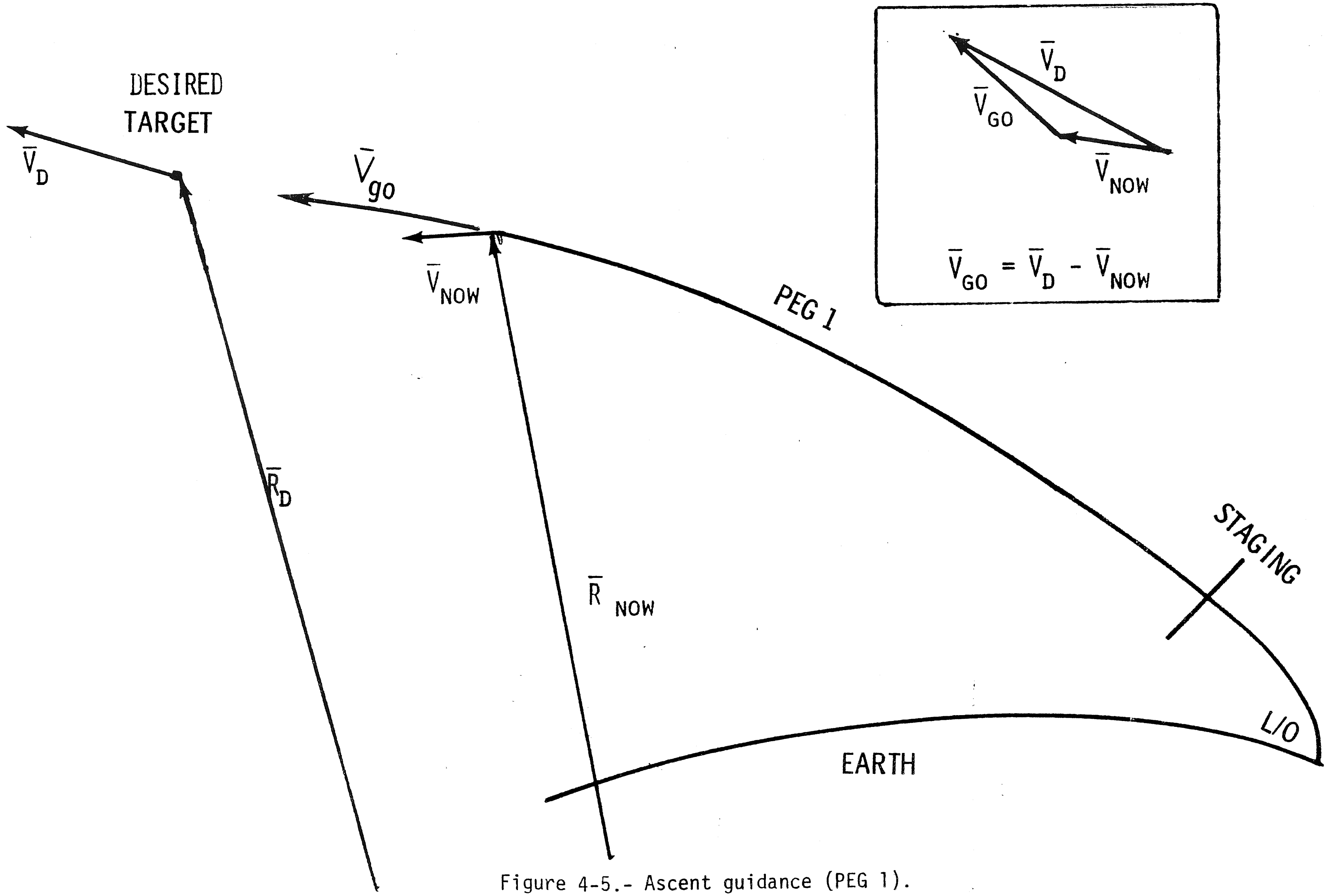


Figure 4-5.- Ascent guidance (PEG 1).

Once V_{GO} has been computed, we can define the reference thrust vector $\bar{T}_R$, mentioned earlier. $\bar{T}_R$ is a unit vector which really defines a starting point or reference point for computing changes in thrust direction.

$\dot{\bar{T}}$ is then defined as being perpendicular to $\bar{T}_R$. The τ_R was defined as a bias that specifies the time at which $\bar{T}$ and $\bar{T}_R$ are parallel. To see why this is necessary, examine figure 4-6. In this illustration, τ_R was assumed to be zero. The $\bar{T}$ and $\bar{T}_R$ are parallel at the beginning of the maneuver. The $\bar{T}$ turns at a constant rate throughout the maneuver so that the final thrust direction is as shown.

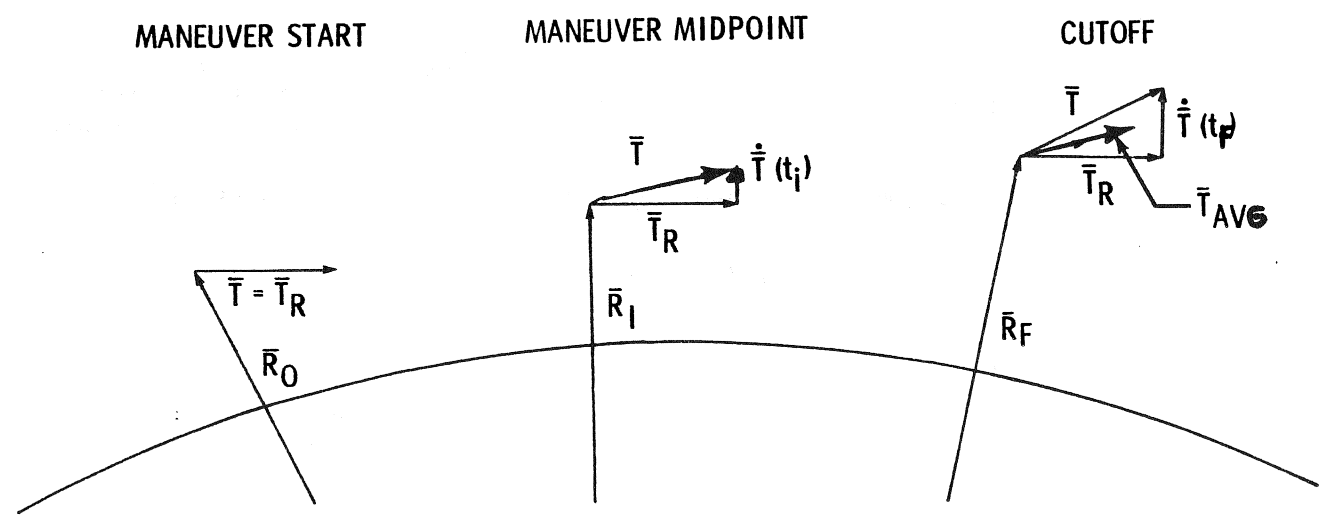


Figure 4-6.- Maneuver with τ_R = o.

The average thrust direction for this example is shown by $\bar{T}_{AVG}$. That is, a constant thrust direction parallel to $\bar{T}_{AVG}$ would achieve the same final state as the actual maneuver illustrated. Now, recall that $\bar{T}_R$ is the constant thrust direction that would satisfy the desired velocity. Since $\bar{T}_{AVG}$ is not parallel to $\bar{T}_R$, the maneuver illustrated in figure 4-6 would not achieve the desired velocity. For this reason, the value of τ_R must be chosen so that the average thrust direction is parallel to the reference thrust vector.

A maneuver made with the correct choice of τ_R is illustrated in figure 4-7. The thrust direction in this example turns "down" throughout the maneuver. Note that before $t = \tau_R$, the vector $\dot{\bar{T}}\,(t - \tau_R)$ points in the opposite direction of the rotation. This is due to the fact that $t - \tau_R$ is negative prior to $t = \tau_R$. After $t = \tau_R$, $t - \tau_R$ is positive and $\dot{\bar{T}}\,(t - \tau_R)$ is in the same direction as the rotation.

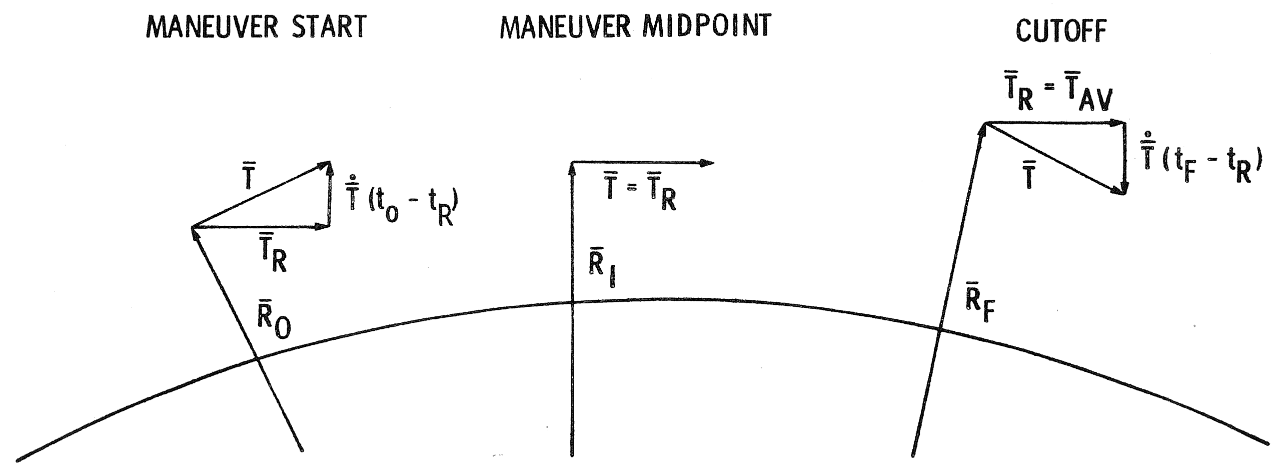


Figure 4-7.- Maneuver with τ_R chosen by PEG.

We have now defined the purpose of the three PEG steering parameters:

$\bar{T}_R$: reference thrust vector
Unit vector indicating the constant thrust direction which would achieve the desired velocity (recall, it is parallel to $\bar{V}_{GO}$).

$\dot{\bar{T}}$: turning rate vector
Vector indicating the direction and rate at which the thrust direction must turn to achieve the desired position (recall, it is perpendicular to $\bar{T}_R$).

τ_R: reference time
Time chosen so that the average thrust direction is parallel to $\bar{T}_R$.

One final point must be made regarding the PEG position targets (i.e., in this case, radius from the earth center). We have seen that the turning-rate vector, $\bar{T}$, is used to achieve the desired position. We have also seen that $\bar{T}$ is perpendicular to the reference thrust vector, $\bar{T}_R$. This means that $\bar{T}$ cannot be used to control the position change in the direction parallel to $\bar{T}_R$. Therefore, the PEG position targets may consist of altitude and/or crossrange constraints, the downrange component of position cannot be constrained.

The PEG steering parameters are defined with the assumption that the complete maneuver (in this case, second stage flight) will be computed at maneuver start (SRB SEP). In practice, the parameters computed at manuever start cannot be relied on to give sufficiently accurate guidance. To compensate for this problem, the guidance system is designed so that the computations are repeated periodically. Each pass through the guidance system is known as a guidance cycle. During each cycle, PEG computes new values for the steering parameters based upon current conditions. This allows the effects of propulsion system failure to be compensated for during the maneuver. It also reduces guidance errors since the PEG equations become more accurate as the maneuver time remaining decreases.

With all this in mind, it is easy to see why guidance incorporates a predictor/corrector algorithm in its cycles to aid in eliminating errors from its computations.

Predictor: The position and velocity at cutoff are predicted by using the current values of the steering parameters.

Corrector: The maneuver target conditions are used to compute the desired position and velocity at cutoff. The predicted and desired cutoff states are compared, and the differences are used to correct the value of $\bar{V}_{GO}$.

Following prediction and correction, the steering parameter computation algorithm uses a corrected $\bar{V}_{GO}$ and the desired cutoff position to compute new values for the steering parameters themselves. It would be possible to enter the loop at any of the three steps. The steering parameter computation step is chosen as the entry point in order to minimize transport lag for the steering parameters. Thus, the predictor and corrector steps are preparation for the next guidance cycle.

This scheme requires a method of obtaining a first estimate of $\bar{V}_{GO}$ during the guidance cycle following initiation of PEG. This first estimate will be in error by a substantial amount. For this reason, the first cycle steering parameters are not used for vehicle control. Instead, these parameters are used to start the normal predictor/corrector process.

The corrected $\overline{V}_{GO}$ computed in the second cycle will be better than the initial estimate since the errors produced by the first cycle steering parameters have been reduced. However, there is still no guarantee that this value is accurate enough for beginning active vehicle steering. The criterion used to enable PEG steering is that the correction of $\overline{V}_{GO}$ be sufficiently small. This condition implies that the predicted state obtained from the previous steering parameters be very "close" to the desired state. When this condition is satisfied, PEG is said to be converged.

During the period between PEG initiation (SRB SEP CMD) and satisfying the convergence criterion, the vehicle will be in attitude hold, using the last valid attitude command received. It should take no longer than four passes through the PEG algorithm to achieve convergence. When convergence is achieved, a change in thrust attitude is to be expected since the attitude commanded by PEG will not, in general, be exactly the same as the attitude at PEG initiation.

Let's look at the algorithm's of the predictor, corrector, and steering parameters and see how they compute the information needed to fly the vehicle to the MECO target. But first, a quick exercise.

A step by step cyclical process is performed each guidance cycle. Assuming the initialization cycle complete, the following sequence of guidance subtasks would be followed:

1. $\overline{V}_{GO}$ UPDATE - update $\overline{V}_{GO}$ by decrementing it by amount of velocity gained over last guidance cycle
2. $\overline{T}_{GO}$ - compute time to MECO
3. THRUST INTEGRALS - evaluate various time integrals of force over mass
4. REFERENCE THRUST VECTOR - determine reference thrust vector
5. RANGE TO GO - determine position to be gained
6. TURNING RATE VECTOR - determine thrust turning rate
7. STEERING INPUTS UPDATE - compute new steering inputs for G/C STEER
8. BURNOUT STATE VECTOR PREDICTION - compute thrust integrals needed to predict burnout state
9. DESIRED ORBIT PLANE CORRECTION - constrain vehicle position to be in desired orbital plane

10. DESIRED POSITION - correct magnitude of burnout position (altitude)

11. DESIRED VELOCITY - compute desired velocity at MECO

12. VGO CORRECTION - compute V_{MISS}, difference between $V_{predicted}$ and $V_{desired}$.

13. GUIDANCE CONVERGENCE - check if guidance is converged, i.e.

$$| V_{MISS} | \leq .01 \; | V_{GO} |$$

14. CUTOFF POSITION CONSTRAINT RELEASE - when TGO = 40 sec, release all position constraints for control stability reasons.

For those wishing to walk through some of the computations in the guidance cycle, the following pages have been provided. If you've had enough of PEG, feel free to move on to the next section; oh, after you've answered the following questions, of course.

EXERCISE

1. List the four MECO target parameters
   a. ________
   b. ________
   c. ________
   d. ________

2. What are the three hypothetical phases PEG 1 assumes will occur?
   a. ________
   b. ________
   c. ________

3. What is T FAIL?

4. T/F. V_{GO} is the difference between the final desired velocity and current velocity. ________

EXERCISE ANSWERS

1. List the four MECO target parameters
   a. target velocity
   b. flight path angle
   c. target radius
   d. desired orbital plane

2. What are the three hypothetical phases PEG 1 assumes will occur?
   a. assume engine failure at T FAIL
   b. 2 engine working until 3 G's
   c. 3G limiting

3. What is T FAIL?
   T FAIL is the time an AOA, with a single main engine failure, is available.

4. T/F. V_{GO} is the difference between the final desired velocity and current velocity. True

Assume we're entering the guidance cycle at the V_{GO} update task.

The vehicle has undergone thrust acceleration during the last computer cycle. Therefore, V_{GO} can be updated by subtracting the velocity change over the last cycle. This is illustrated in figure 4-8.

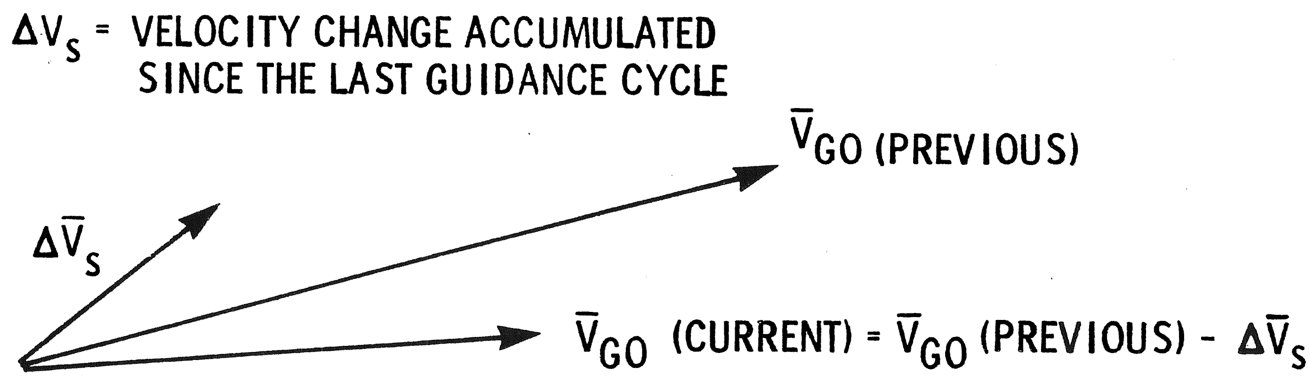


Figure 4-8.- $\bar{V}_{GO}$ update.

Now that $\overline{V}_{GO}$ is current, a total burn time remaining till MECO, t_{GO}, can be computed and is required for use in later calculations and for display. Recalling that there are three thrust phases (fig. 4-2), a time for each phase must be computed. Therefore

$$t_{GO} = t_{GO}\ (\text{PHASE } 1) + t_{GO}\ (\text{PHASE } 2) + t_{GO}\ (\text{PHASE } 3)$$

Specifically, these times are computed from the following equations:

$$t_{GO(\text{PHASE } 1)} = T\ \text{FAIL} - T_{NOW}$$

$$t_{GO(\text{PHASE } 2)} = \frac{M}{\dot{M}} - \frac{V_{EXHAUST}}{\text{Accel}_{MAX}}$$

$$t_{GO(\text{PHASE } 3)} = \frac{V_{GO}}{\text{Accel}_{MAX}}$$

Where M is current mass:

$\dot{M}$ is mass rate

$V_{EXHAUST}$ is exhaust velocity

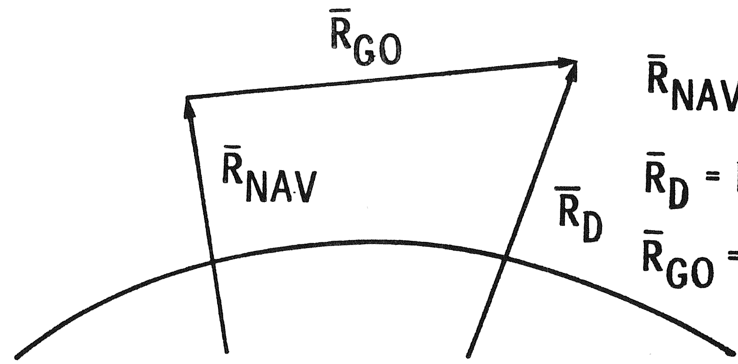
Accel_{MAX} is max acceleration

Now that we know t_{GO}, lets put it in our hip pocket until later.

The solution of the guidance equations requires that various time integrals of the thrust acceleration be evaluated. Two single integrals are used to compute the velocity change due to thrust. Two double integrals are used to compute the position change due to thrust.

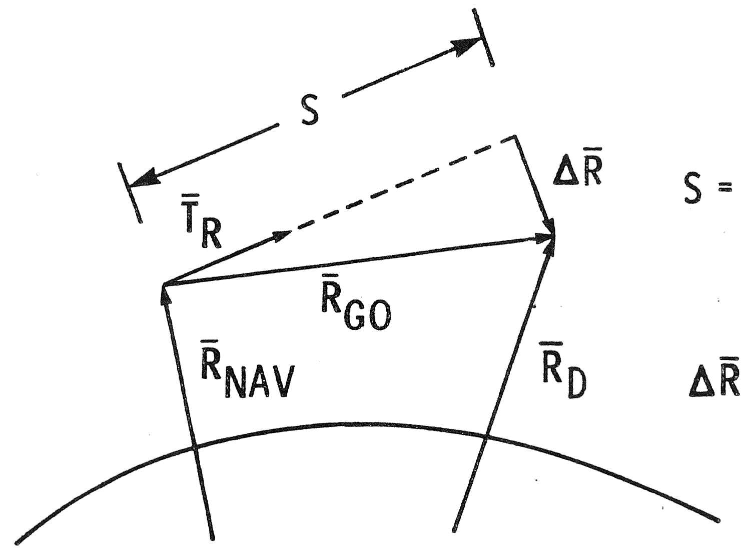
Two sets of thrust integral equations are provided. One set is used for the constant thrust phase. The other set is used for the constant acceleration phase. Each integral is computed for each phase and the parts summed to give the total integral for the maneuver required to reach MECO. The calculations for these integrals require the burn times for each phase (which we just computed) and basic thrust parameters provided by guidance tasks external to PEG.

Now we're ready to compute the steering parameters ($\dot{\bar{T}}_R$, $\dot{T}_R$, τ_R). The reference thrust vector, $\bar{T}_R$, recall is simply a unit vector in the direction of $\bar{V}_{GO}$. The reference time, τ_R, must be chosen so that the average thrust direction is parallel to $\bar{T}_R$. Analysis of the thrust equations of motion shows that the value of τ_R is given by the ratio of the two single integrals of thrust acceleration (have faith). The turning rate vector, $\bar{T}_R$, must steer the vehicle to the desired position, MECO. The computation of T requires an estimate of the position error at cutoff. This error estimate is computed from the thrust integrals and a position to be gained vector, R_{GO}. R_{GO} represents the position change that must be accomplished, by thrusting, in order to achieve the desired position at MECO. R_{GO} is illustrated in figure 4-9. R_{NAV} is the current position vector from navigation. R_D is the desired cutoff position vector computed in the corrector step of the previous guidance cycle. (In a minute, we'll see where R_D came from.) R_{GO}, $\bar{T}_R$ and the thrust integrals are used to compute the position error perpendicular to $\bar{T}_R$. This error is shown in figure 4-10.



$\bar{R}_{NAV}$ = CURRENT POSITION FROM NAVIGATION
$\bar{R}_D$ = DESIRED CUTOFF POSITION
$\bar{R}_{GO}$ = POSITION TO BE GAINED

Figure 4-9.- $\bar{R}_{GO}$.



S = ESTIMATE OF THE POSITION CHANGE PARALLEL TO $\bar{T}_R$ (FROM THRUST INTEGRALS)

$\Delta\bar{R}$ = POSITION ERROR PERPENDICULAR TO $\bar{T}_R$

Figure 4-10.- ΔR.

All data needed for the calculation of $\dot{T}$ is now available. The direction of $\dot{T}$ is given by ΔR. The magnitude of ΔR the thrust integrals, and the reference time determine the magnitude of T. The steering parameters can now be sent to G/C STEER (flight control) if the predicted cutoff time has not changed significantly since the previous cycle. The test condition is

$$|t_p - t_p'| \leq (EPS)\, t_{GO}$$

where t_p $\sim$ predicted cutoff time for the current cycle (current time plus t_{GO})

t_p' $\sim$ predicted cutoff time from the previous cycle

EPS $\sim$ I-loaded constant (TBD)

Failure to satisfy this condition implies that PEG is not converged. This test anticipates the result of the PEG convergence test which will be made at the end of the corrector step. This is done to minimize the delay in sending the new steering parameters to G/C STEER. If this test is failed G/C STEER will retain the last valid commands received. PEG processing will continue.

The next step is to use the predictor function to compute position and velocity at cutoff assuming the current steering parameters are used for the rest of second stage. But first let's see how the desired position, R_D, is computed.

(The desired position of velocity at cutoff must be constructed from the target conditions). The altitude and orbital plane targets define the radial and crossrange components of the desired position, R_D. The downrange position component is taken from the projection of the predicted cutoff position onto the desired orbital plane, figure 4-11. The desired velocity, V_D, is constrained to be in the desired orbital plane and is defined by I-loaded target parameters.

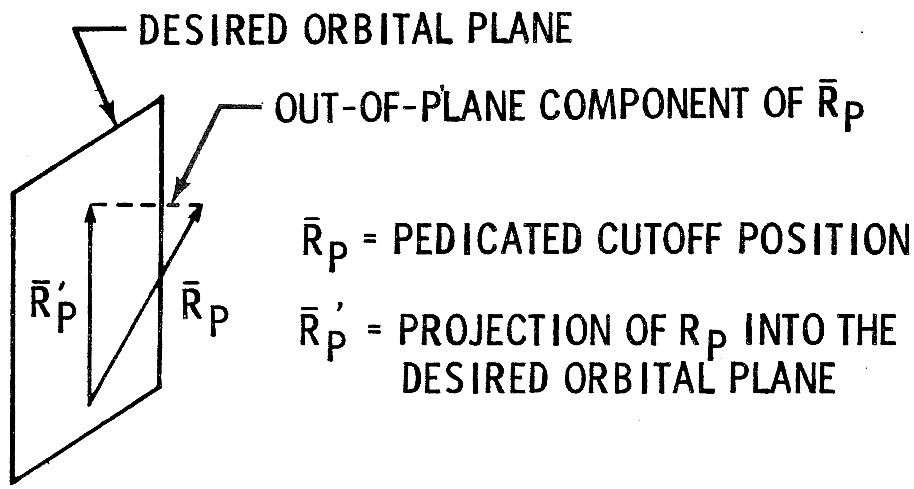


Figure 4-11.- Projection of R_p into the desired plane.

Now let's see where the predicted state comes from. The predicted state is computed from the current state and the changes due to thrust and gravity. The PEG design allows the effects of thrust and gravity to be evaluated separately.

The position change due to thrust is simply the R_{GO} computed previously.

The velocity change due to thrust, V_{THRUST}, is V_{GO} plus a term computed from the thrust integrals. This term is a small correction to V_{GO} and is used to improve the accuracy of the predicted state.

The position and velocity changes due to gravity are computed by the state prediction function in navigation using inputs provided by PEG. The inputs from PEG are the initial position and velocity vectors and the initial and final times. The initial time is simply the current time. The final time is current time plus t_{GO}. The outputs from the state predictor to PEG are final position and velocity vectors, $\overline{R}_F$ and $\overline{V}_F$. The position and velocity changes due to gravity are then, simply

$$\overline{R}_{GRAV} = \overline{R}_F - \overline{R}_{INITIAL}$$

$$\overline{V}_{GRAV} = \overline{V}_F - \overline{V}_{INITIAL}$$

The predicted cutoff state can now be computed:

$$\overline{R}_p = \overline{R}_{NAV} + \overline{V}_{NAV} (t_{GO}) + \overline{R}_{GO} + \overline{R}_{GRAV}$$

$$\overline{V}_P = \overline{V}_{NAV} + \overline{V}_{THRUST} + \overline{V}_{GRAV}$$

The end is in sight.

Now all we have to do is use the corrector function to compute V_{MISS}, where

$$\overline{V}_{MISS} = \overline{V}_D - \overline{V}_p$$

and then make the correction to V_{GO}

$$\overline{V}_{GO}\text{ (NEW)} = \overline{V}_{GO}\text{ (current)} + \overline{V}_{MISS}$$

and then test for convergence is $|\overline{V}_{MISS}| \leq .01\ |\overline{V}_{GO}|$
yes = convergence
no = not converged

The final function of the correction step is to release the position constraints on the target near the end of the maneuver (MECO). When t_{GO} is small, small change in position error can produce large changes in the turning rate vector. To prevent this problem, it is necessary to stop steering to the position target (radius in the case). The criteria for releasing the position constraints are that PEG be converged and t_{GO} be less than an I-loaded minimum (40 sec).

When these conditions are met, the desired position vector is simply set equal to the predicted cutoff position:

$\overline{R}_D = \overline{R}_p$ when position constraints are released.

One more thing; when t_{GO} reaches 10 sec, PEG guidance is terminated and the MPS guidance cutoff task is initiated. (This is the guidance simplification referred to earlier.) The purpose of this task is to compute a desired SSME cutoff time based on the desired cutoff velocity magnitude. The cutoff time calculation includes the predicted velocity change from the time the minimum throttle setting is commanded until cutoff and the predicted tailoff impulse from each active SSME. Another term for this task is "fine countdown."

Secondary functions of this task are to command the minimum throttle setting at such a time that ensures the minimum thrust level is applied for a desired time prior to MECO command, and to set a flag when desired MECO time is less than two guidance cycles for the SSME OPS principle function to initiate the nominal shutdown countdown timer.

Finis.

EXERCISE

1. T/F. $\overline{V}_{MISS}$, a small correction to $\overline{V}_{GO}$, represents the difference between predicted and desired velocity. ________

2. T/F. Guidance is active all the way to the MECO command. ________

3. T/F. $\overline{T}_R$ is a reference thrust vector in the direction of $\overline{V}_{GO}$. ________

4. The parameters sent to flight control are:
   a. ____________________
   b. ____________________
   c. ____________________

EXERCISE ANSWERS

1. T/F. $\mathbf{V}_{MISS}$, a small correction to $\mathbf{V}_{GO}$, represents the difference between predicted and desired velocity. True

2. T/F. Guidance is active all the way to the MECO command. False

3. T/F. $\mathbf{T}_R$ is a reference thrust vector in the direction of $\mathbf{V}_{GO}$. True

4. The parameters sent to flight control are:
   a. reference thrust vector
   b. desired thrust turning rate vector
   c. reference time

4.2.2 3 G Limiting

The 3G limiting software is simply responsible for generating the appropriate throttle command which will limit the average vehicle acceleration to 3G's. Note that algorithm limits to 3G's "on the average," therefore it will tend to oscillate, slightly, about the desired value.

4.3 FLIGHT CONTROL

4.3.1 Hardware

The hardware involved in second stage flight control is basically the same as first stage. However, the body mounted accelerometers are no longer used. Also, the elevons are no longer driven and are held in place.

4.3.2 Software

There are very few changes in the flight control software. The software recognizes that the SRB's have been separated (hopefully) and now the vehicle is controllable only by means of the main engines.

4.3.3 Automatic Control System

4.3.3.1 Guidance/control steering interface.- Now that guidance has moded to powered explicit guidance, G/C STEER also modes to be compatible with guidance. Figure 4-12 is a functional view of G/C STEER in second stage.

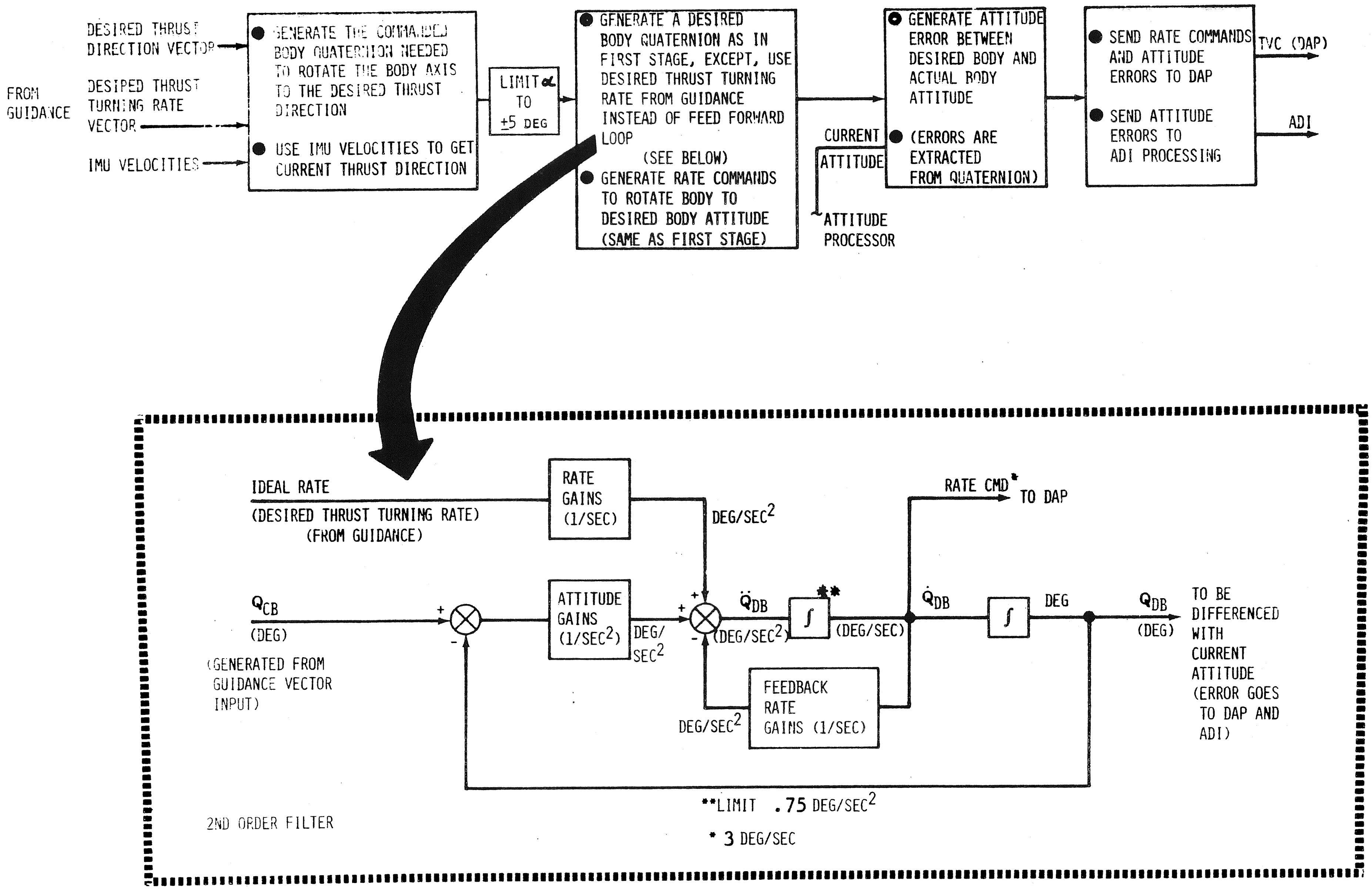


Figure 4-12.- Automatic steering (second stage).

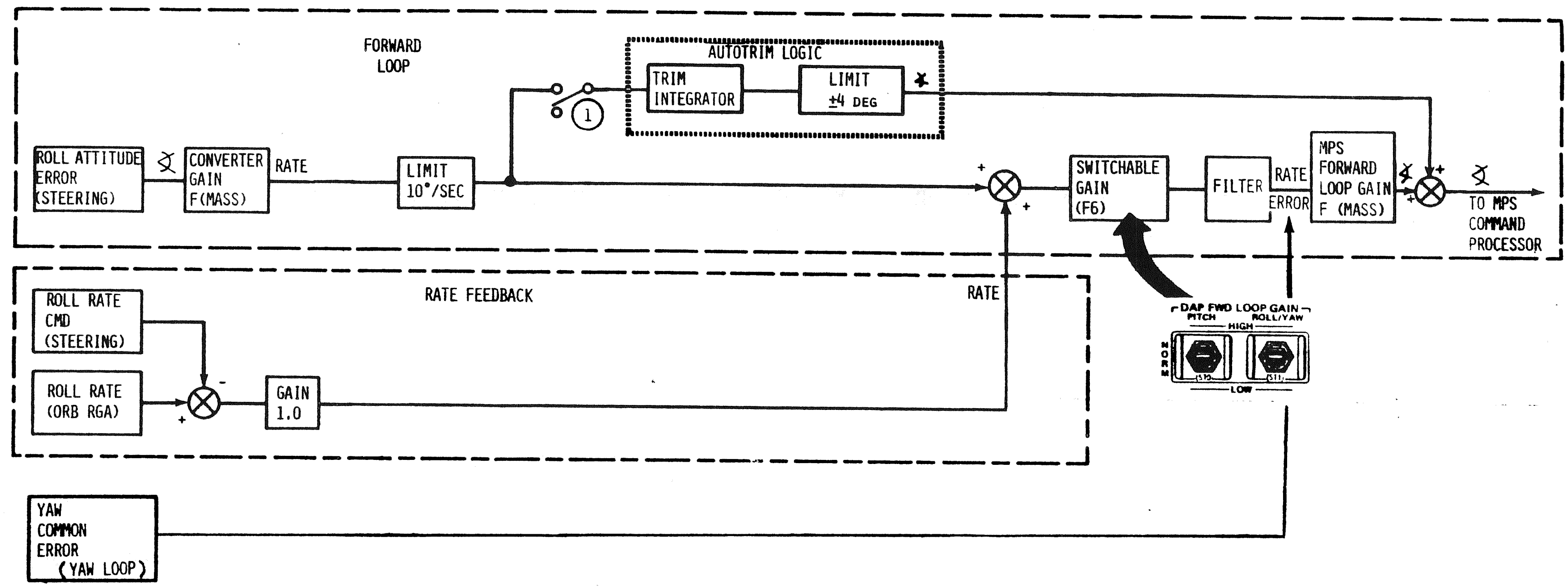
IMU ΔV's are now used in the computations which provide thrusting direction in the appropriate coordinate system. α limiting is performed so that heating problems with the ET do not occur. α limiting of $\pm 5^{\circ}$ is performed until MET = 150 sec or ALT = 200K ft.

Note in Figure 4-12, that the acceleration limit in the filter has changed to $.75^{\circ}/sec^2$. The remaining logic continues to function the same as in first stage.

Again G/C STEER generates rate command and attitude errors.

4.3.3.2 Digital Autopilot.- The few changes in the DAP are summarized on the following pages.

4.3.3.2.1 Thrust vector control: The accelerometers are not used. The gains that were referenced to V_{REL} and TIME in first stage are now referenced to MASS. MASS is a convenient parameter to track changing inertia. The limit on the autotrim output is raised to ±4 deg. Figures 4-13, 4-14, and 4-15 present the roll, pitch, and yaw channel for second stage. Note that there is crossfeed between yaw and roll for stability purposes.



1 AUTOTRIM ENABLED AT SRB SEP CMD + 12 SEC

Figure 4-13.- Roll control loop, second stage (SRB null CMD to ME zero thrust).

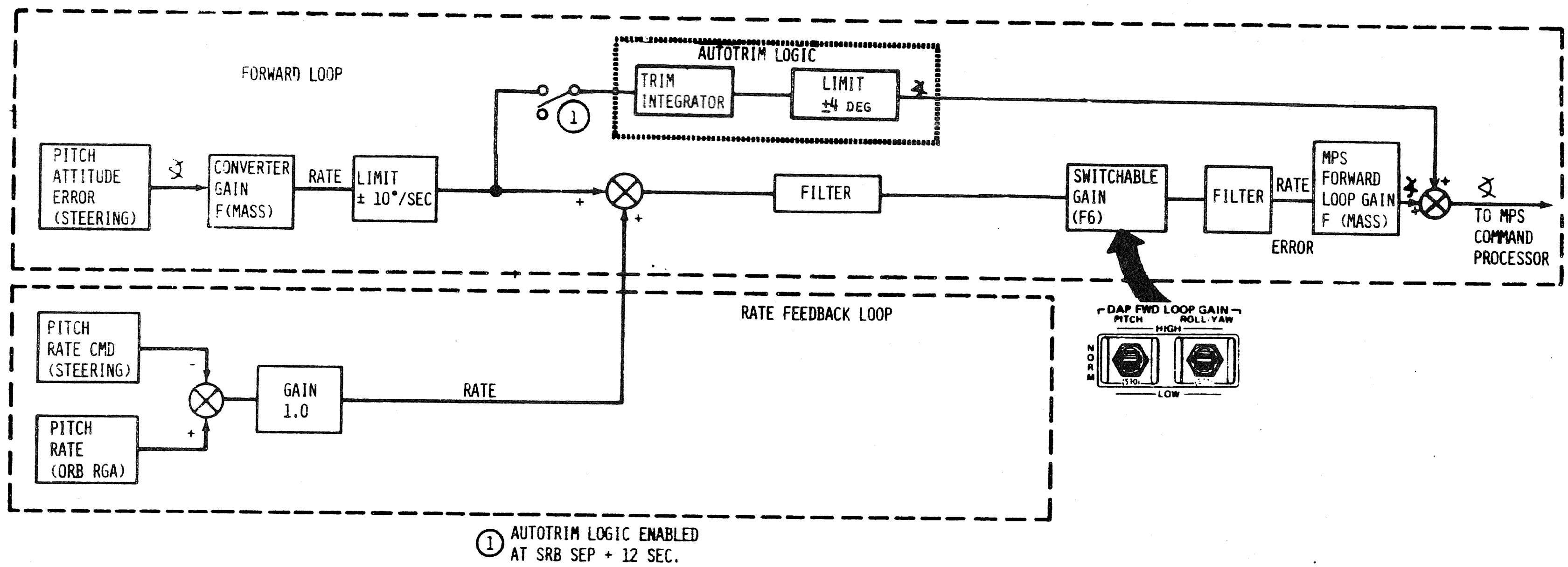


Figure 4-14.- Pitch control loop, for second stage (SRB null CMD to ME zero thrust).

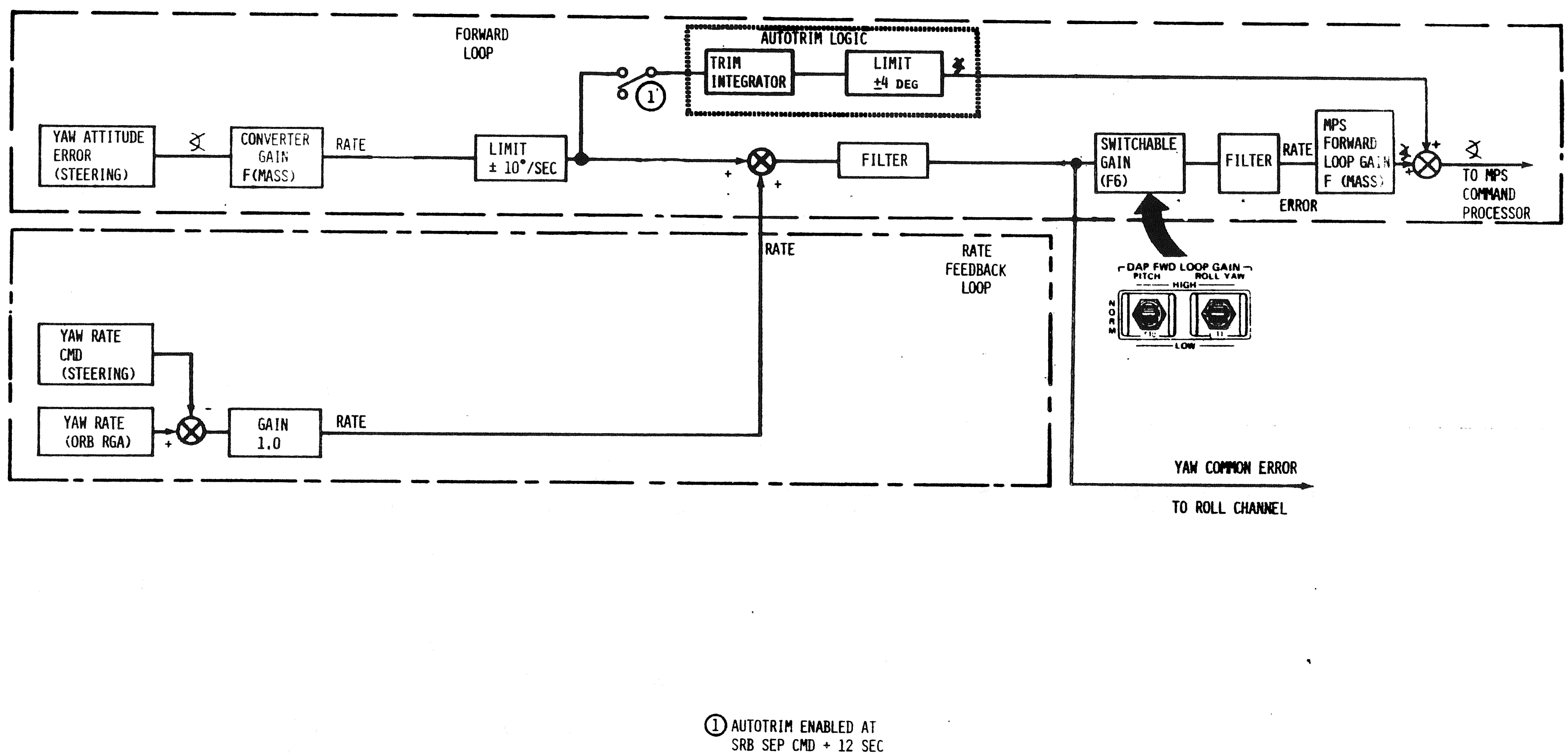


Figure 4-15.- Yaw control loop, second stage (SRB null CMD to ME zero thrust).

4.3.3.2.2 MPS command processor: Everything that was said in the first stage write up is applicable in second stage. There is one addition to the logic, however, in second stage. The software now performs priority rate limiting on the engine bell deflection commands. In first stage we saw that limits were applied to deflection commands with regards to hydraulic systems failures. Now, in second stage, the commands are scaled down proportionally so as to give "priority" to a larger command.
Here's how it works:

All of the demands (commands) on a given hydraulic system (there are three) are summed. If this sum exceeds an I-loaded rate capability, a scale factor is formed and multiplies all the demands on that hydraulic system so as to lower the total demand. For example; the actuators that are driven by hydraulic system 2 when there are no hydraulic system failures are:

Pitch actuator on ME 2 (left) deflection given by Y_2
Yaw actuator on ME 3 (right) deflection given by Z_3

The demand of the actuator is a rate. Therefore

$|\dot{Y}_2|$; $|\dot{Z}_3|$ are the demands on hydraulic system 2. Let's say, for example, that the DAP has command 14^o/sec from the left ME pitch actuator and 10^o/sec from the right ME yaw actuator.

We have a total demand of

$$14 + 10 = 24^o/\text{sec} = X_{CMD}$$

The I-load capability is 18^{o}/sec = X. Since 18 < 24, the commands must be scaled down. The formula is

if $X_{CMD} > X$, then the actuator CMD is; new CMD = $\frac{X}{X_{CMD}}$ x COMMAND RATE

or, in the example

$$\text{new } \dot{Y}_2 = \frac{18}{24} \times 14 = 10.5 \text{ deg/sec}$$

$$\text{new } \dot{Z}_3 = \frac{18}{24} \times 10 = 7.5 \text{ deg/sec}$$

so the demands have been scaled appropriately.

When there are failures, the remaining systems feed the appropriate actuators. The information used for priority rate limiting is provided in figure 4-16.

SYS FAILED	SYS OP	Actuator rate limit equations ($\dot{Y}_i$ = Pitch i actuator rate) ($\dot{Z}_i$ = Yaw i actuator rate)	RATE CAPABILITY X (^{o}/sec) SRB SEP CMD TO MECO CMD 100% APU-0,1 FAIL 110% APU-2 FAIL	RATE CAPABILITY X (^{o}/sec) MECO CMD 100% APU-0,1 FAIL 110% APU-2 FAIL
NONE	1	$1.24\|\dot{Y}_1\| + \|\dot{Z}_2\| = XC_1$	18.0	15.8
	2	$\|\dot{Y}_2\| + \|\dot{Z}_3\| = XC_2$		
	3	$\|\dot{Y}_3\| + \|\dot{Z}_1\| = XC_3$		
1	2	$\|\dot{Y}_2\| + \|\dot{Z}_2\| + \|\dot{Z}_3\| = XC_2$		
	3	$1.24\|\dot{Y}_1\| + \|\dot{Y}_3\| + \|\dot{Z}_1\| = XC_3$		
2	1	$1.24\|\dot{Y}_1\| + \|\dot{Y}_2\| + \|\dot{Z}_2\| = XC_1$		
	3	$\|\dot{Y}_3\| + \|\dot{Z}_1\| + \|\dot{Z}_3\| = XC_3$		
3	1	$1.24\|\dot{Y}_1\| + \|\dot{Z}_2\| + \|\dot{Z}_1\| = XC_1$		
	2	$\|\dot{Y}_3\| + \|\dot{Y}_2\| + \|\dot{Z}_3\| = XC_2$		
1 & 2	3	$1.24\|\dot{Y}_1\| + \|\dot{Z}_1\| + \|\dot{Y}_3\| + \|\dot{Z}_3\| = XC_3$	18.6	16.5
1 & 3	2	$\|\dot{Y}_2\| + \|\dot{Z}_2\| + \|\dot{Y}_3\| + \|\dot{Z}_3\| = XC_2$		
2 & 3	1	$1.24\|\dot{Y}_1\| + \|\dot{Z}_1\| + \|\dot{Y}_2\| + \|\dot{Z}_2\| = XC_1$		

Figure 4-16.- Prioity rate limiting logic, second stage.

The rate capability decreases at MECO CMD due to the demand on the hydraulic system for valve closures, etc. Notice, that if two hydraulic systems fail, the remaining APU is kicked up to 110 percent (shortens APU lifetime, but gets the crew home). Note also that the pitch actuator command for the center engine is multiplied by 1.24. This gives the command a higher priority since this pitch actuator is larger than the others because it has to drive the engine bell up into the air flow.

4.3.3.2.3 RCS command processor: The RCS continues to provide roll control when in single engine flight control. The RCS are controlled by the ascent DAP until ZERO THRUST.

4.3.4 Manual Control System

RATE COMMAND/ATTITUDE HOLD is available in second stage in the same manner as in first stage.

EXERCISE

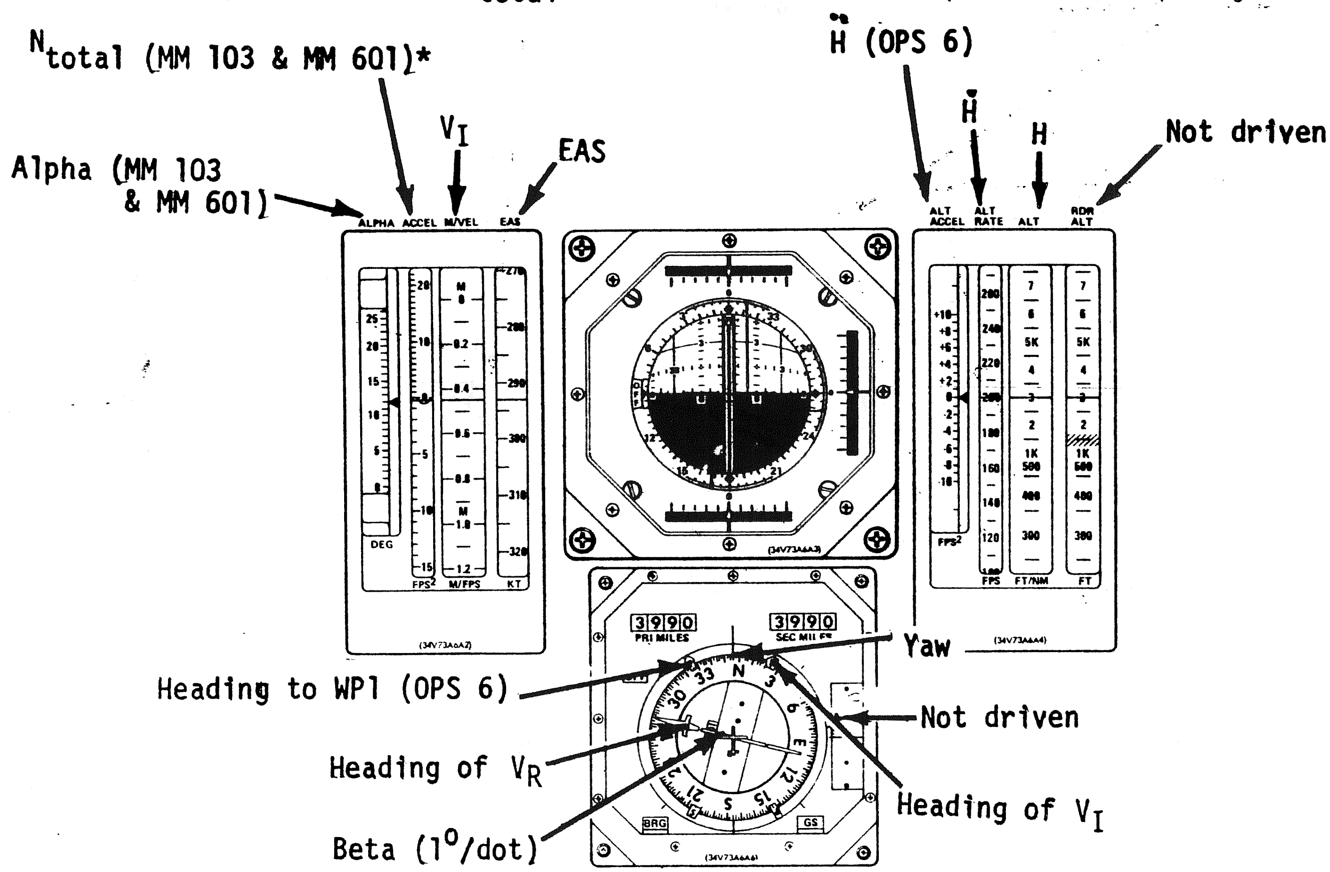
1. T/F. The elevons are not driven in second stage. ________
2. T/F. α is limited until MET = 15^{0} sec. ________
3. Priority rate limiting is performed on the ____________________.
4. T/F. The RCS aid in controlling the vehicle when a single engine has failed. ______

EXERCISE ANSWERS

1. T/F. The elevons are not driven in second stage. True
2. T/F. α is limited until MET = 15^{0} sec. True
3. Priority rate limiting is performed on the SSME actuator commands.
4. T/F. The RCS aid in controlling the vehicle when a single engine has failed. False

4.4 DEDICATED DISPLAYS/CRT DISPLAYS

The dedicated displays active in second stage are shown in figure 4-17. Two parameters have been added since first stage, Alpha and N_{total} (note that some of the parameters are only available in OPS 6)



*AMI ACCEL DISPLAYS
N_Z IN MM 602.

Figure 4-17.- Dedicated displays active in second stage.

The OPS Display is the same one as in first stage and is shown in figure 4-18 for review.

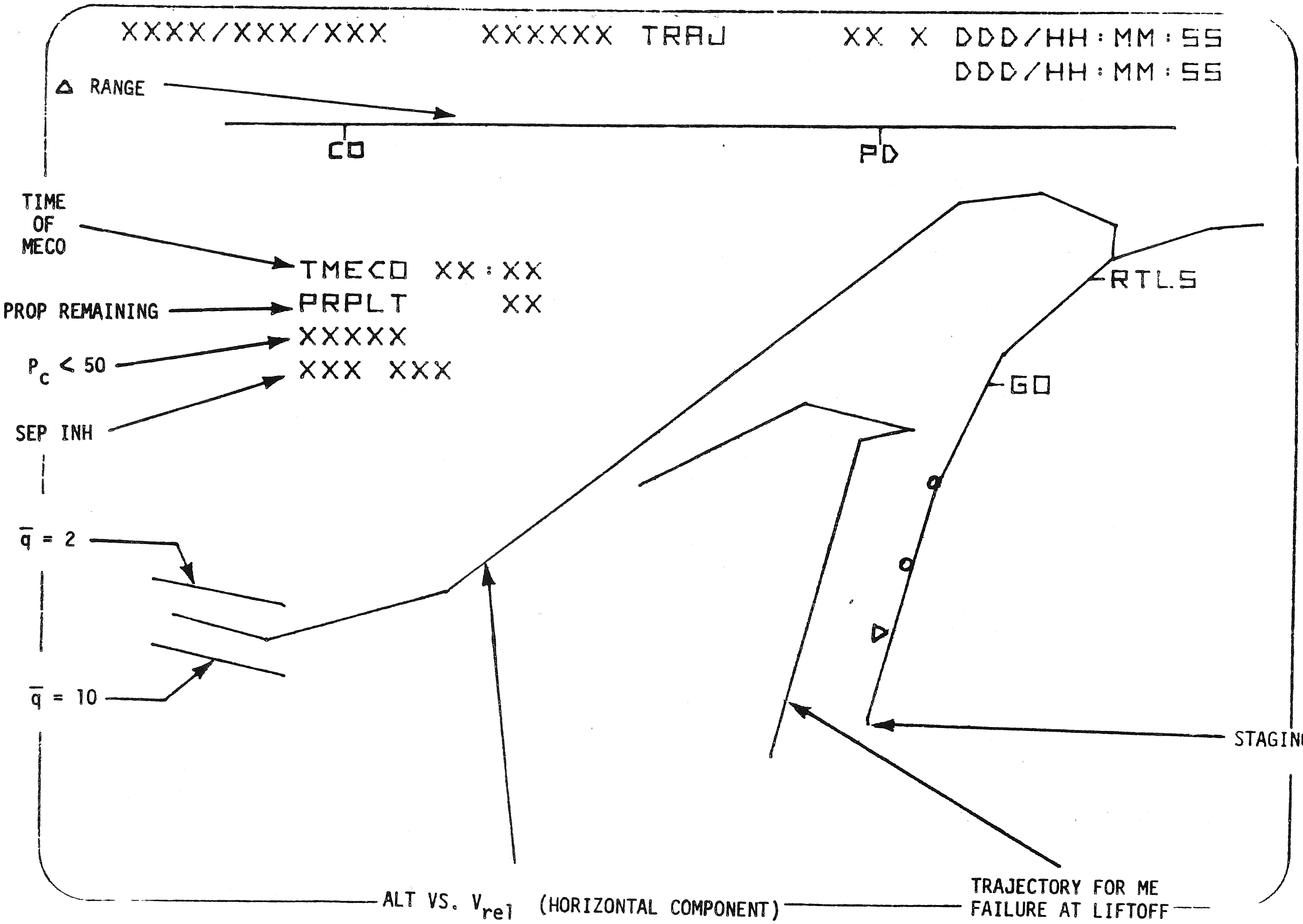


Figure 4-18

The BFS supports the TRAJ 2 display shown in figure 4-19.

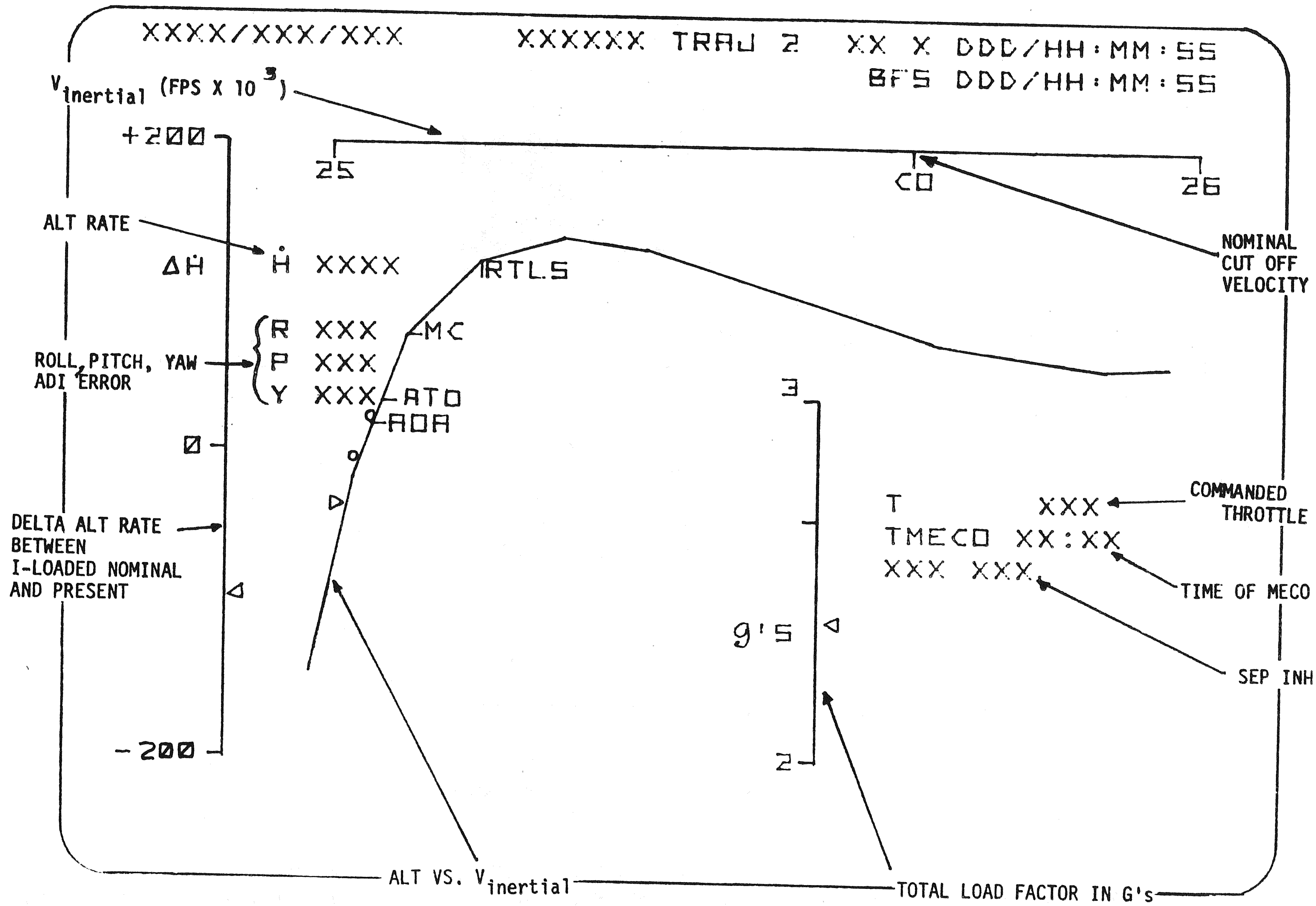


Figure 4-19.- TRAJ display.

4.5 ET SEPARATION

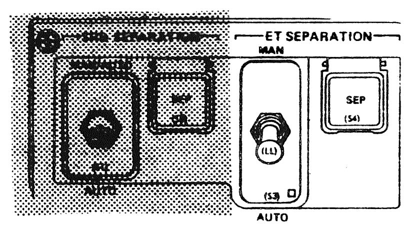
ET separation is normally handled completely and automatically by the GPC's. However the crew can command separation manually by way of the ET SEPARATION switch and PBI located on panel C3. When the switch is in the AUTO position, separation will occur as soon as the appropriate conditions are met concerning target parameters, body rates. When the switch is in the MAN position, the separation sequence will not continue until the SEP PBI is depressed. (See below.)

The external tank separation sequence is initiated by the GN&C moding, sequencing, and control function (MSC) when the SSME OPS sequence has determined that all of the main engines are in the shutdown or post-shutdown phase and sets the MECO confirmed flag.

The sequence has several major functions it accomplishes. First, it determines the mode of separation or if the separation is to be manually inhibited. It then arms the umbilical plate unlatch PIC's, arms and fires the tumble system after all of the MPS prevalves have been commanded closed, closes the feedline disconnect valves, gimbals the SSME nozzles to the proper position, deadfaces the ET/ORB interface, and unlatches and retracts the umbilical plates.

The sequence then arms the structural separation PIC's, performs some limit tests on certain body rates and tests for feedline disconnect valve closure before continuing with an automatic separation. If any of the tests are not satisfied, the separation is inhibited and can occur only if the out-of-tolerance parameter comes back within tolerance or if the crew elects to continue the separation by manually overriding the inhibit. When either of these conditions is satisfied, the structural separation PIC's are fired.

When the Orbiter has gained a velocity of 4 ft/sec vertically, the separation is flagged complete and the ground crew is responsible for issuing a GO/NO GO for the impending OMS 1 burn. OMS 1 TIG is currently set for MECO + 2 min (OFT 1) (fig. 4-20).



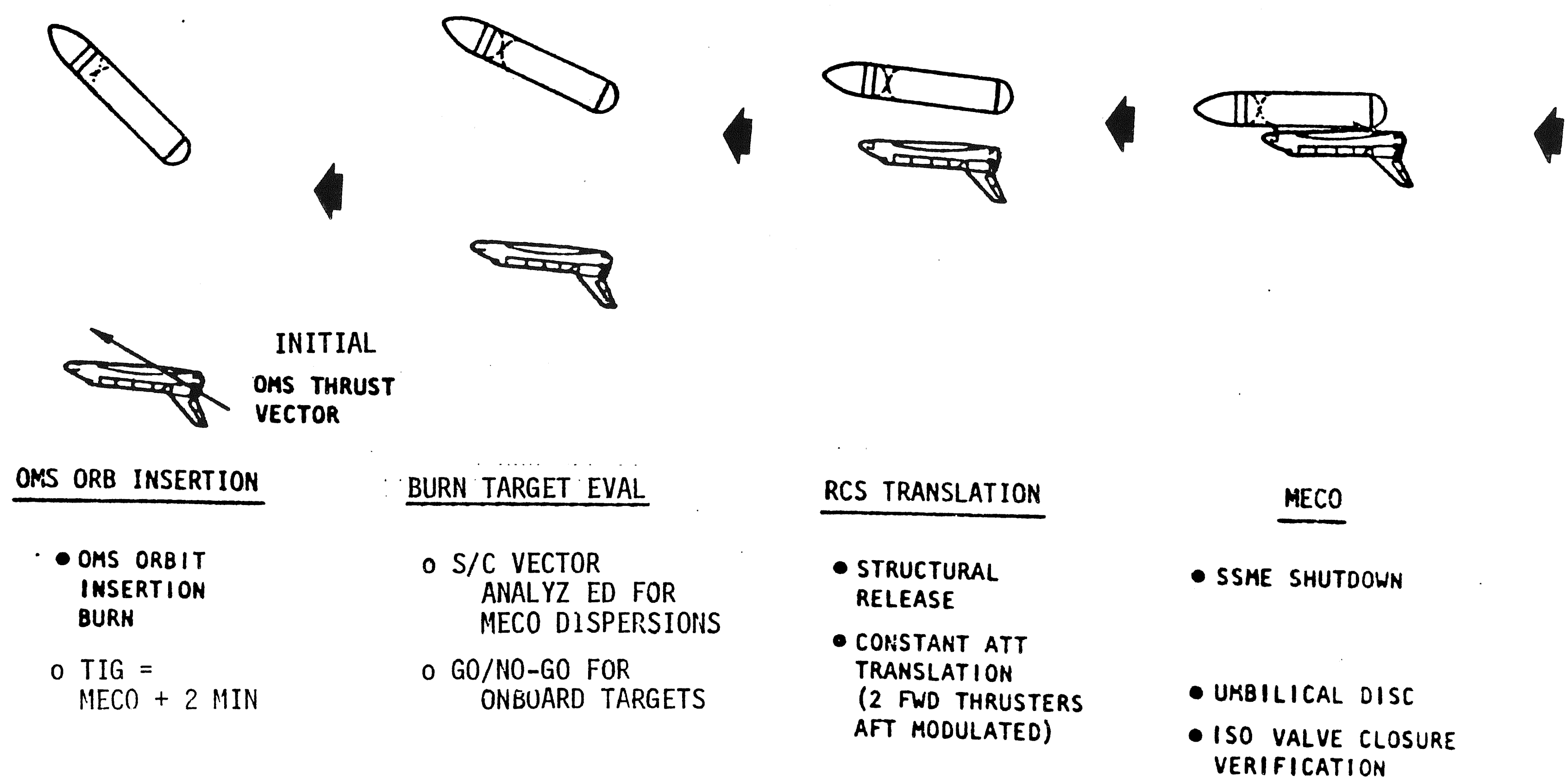


Figure 4-20.- Nominal and AOA separation maneuver sequence.

EXERCISE

1. T/F. The crew cannot separate the ET unless rate constraints are met. ______

2. T/F. OMS 1 TIG occurs when a 4 ft/sec velocity gain is measured after separation. ______

EXERCISE ANSWERS

1. T/F. The crew cannot separate the ET unless rate constraints are met. False

2. T/F. OMS 1 TIG occurs when a 4 ft/sec velocity gain is measured after separation. False

SECTION 5
BACKUP FLIGHT SYSTEM

Since the backup flight system is extremely similar to the prime system during mainstage flight, only the major differences between the two systems will be discussed here.

One major difference with the BFS is that there is no manual steering or manual throttling capability during first and second stage. Another major difference is that as of the publishing date of this document, there is no AOA or RTLS capability in the BFS. (However, these capabilities are expected to exist at a later date.)

Since there is no BFS takeover before SRB ignition the BFS does not have the capability to perform pre-launch slew checks of the aerosurface actuators, main engine actuators, or the SRB actuators.

There is no Fault Detection, Identification, and Recovery (FDIR), therefore redundant signals are mid-value selected (MVS) when triply redundant.

Majority vote is formed on triply redundant contact switches.

The "AND" function is performed on dual redundant contact switches. The software of the BFS will process only the left hand side for:

- ADI selection and REF pushbutton
- HSI source selection
- RHC trim

APPENDIX A
ACRONYMS AND SYMBOLS

ADI	Attitude Director Indicator
ALT	Altitude
AME/CMG	Abort Maneuver Evaluator/Contingency Maneuver Generator
AMI	Alpha Mach Indicator
AOA	Abort Once Around
ASC	Ascent
ATO	Abort To Orbit
ATT	Attitude
AVVI	Altitude/Vertical Velocity Indicator
BARO	Barometric
BFS	Backup Flight System
C1	Intercept of horizontal vs. vertical velocity line (PEG)
C2	Slope of horizontal vs. vertical velocity line (PEG)
CMD	Command
CRT	Cathode Ray Tube
DAP	Digital Autopilot
D/D	Dedicated Displays
DEU	Display Electronics Unit
DISP	Display
DRC/AH	Discrete Rate Command/Attitude Hold
EI	Entry Interface
EOM	Equations of Motion
ET	External Tank
FCS	Flight Control System
FPL	Fuel Power Level (100% SSME thrust)
GAMD	State Vector Flight Path Angle
GC STEER	Guidance/Control Steering
GNC	Guidance, Navigation, and Control
GPC	General Purpose Computer
GSE	Ground Support Equipment
HSI	Horizontal Situation Indicator
I-LOAD	Initial Computer Load (Prelaunch)
IMU	Inertial Measurement Unit
L/O	Lift-off
LPS	Launch Processing System
LTVCON	Linear Terminal Velocity Constraints (PEG 4)
M50	Mean of 1950 (Reference System)
MAN	Manual or Maneuver
MC	Mission Completion
MDM	Multiplexer/Dimultiplexer
ME	Main Engine
MECO	Main Engine Cutoff
MM	Major Mode
MPL	Maximum Power Level (109% SSME Thrust)
MPS	Main Propulsion System
MSBLS	Microwave Scanning Beam Landing System
MSC	Moding, Sequencing, and Control
NAV	Navigation
OMS	Orbital Maneuvering System
OPS	Operational Sequence
PFS	Primary Flight System
PRL	Priority Rate Limiting
RCS	Reaction Control System
RECON	Reconfiguration

REFMAT	Reference Matrix (stable member to M50)
RELMAT	Relative Matrix (between IMU's)
RGA	Rate Gyro Assembly
RPL	Rated Power Level (109% SSME Thrust)
RTLS	Return To Launch Site
SBTC	Speed Brake/Thrust Controller
SOP	Subsystem Operating Program
SPI	Surface Position Indicator
SRB	Solid Rocket Booster
SRM	Solid Rocket Motor
SSME	Space Shuttle Main Engine
TACAN	Tactical Air Navigation
TB	Time Base (Computer Clock)
TDRSS	Telemetry and Data Relay Satellite System
TGO	Time To Go
TGT	Target
TIG	Time of Ignition
TRAJ	Trajectory
TVC	Thrust Vector Control
UPP	User Parameter Processing

MAIL TO: Jenifer Herrman/DG6

LESSON CRITIQUE SHEET

LESSON CODE GNC ASC 2102 DATE OF MATERIAL 5/15/79

Please explain any negative answers to the questions below in specific terms. Use the back of the sheet if required.

1. Was the lesson material presented in a logical manner?

2. Comments on handout organization:

3. Were the figures and diagrams optimum to understanding the material?

4. Did the lesson start from your entry level?

5. In you opinion, was the level of detail about right?

6. Additions or deletions from the lesson?

7. Prior Knowledge: Considerable _____ Some _____ Very Little _____

8. Other comments (or questions):

MAIL TO: DG/TRAINING COORDINATION

WORKBOOK TRAINING REPORT

INSTRUCTIONS:

When you have completed this lesson, please fill this sheet out and return it to the Training Coordination;DG6 in order to have completion entered on your record.

STUDENT

STUDENT NAME

MAIL CODE

1	2	3

LAST NAME

6	7	8	9	10	11	12	13	14	15	16	17	18

INITIALS

20	21

LESSON CODE

ALPHA

G	N	C		A	S	C		
25	26	27	28	29	30	31	32	33

NUMERIC

2	1	0	2
37	38	39	40

41

ENTER TIME TO COMPLETE THIS LESSON

	:		
47	48	49	50

HR:MIN

DATE COMPLETED (M/D/Y)

		/			/		
52	53	54	55	56	57	58	59

WORKBOOK DATE (M/D/Y)

0	5	/	1	5	/	7	9
61	62	63	64	65	66	67	68

I certify that I have completed the above described lesson

STUDENT SIGNATURE ______________________________

KH/78/001 27 NOV

NASA-JSC

WORKBOOKS, FAMILIARIZATION MANUALS, TEXTS

DISTRIBUTION LIST

NASA MFSC

JA71/R. O. McBrayer

FORD AEROSPACE & COMMUNICATIONS CORPORATION

MS 30/1047/N. Schaffner

MS 1/205/D. Baerd

NASA JSC

AP/H. S. Stall

AT/J. P. Loftus

JM54/COD Pubs Center (3)

JM6/Tech Lib (2)

MR3/D. D. Ewart

FM8/P. Pixley