KIM Goes to the Moon

Jim Butterfield
14 Brooklyn Av
Toronto Ontario M4M 2X5 CANADA

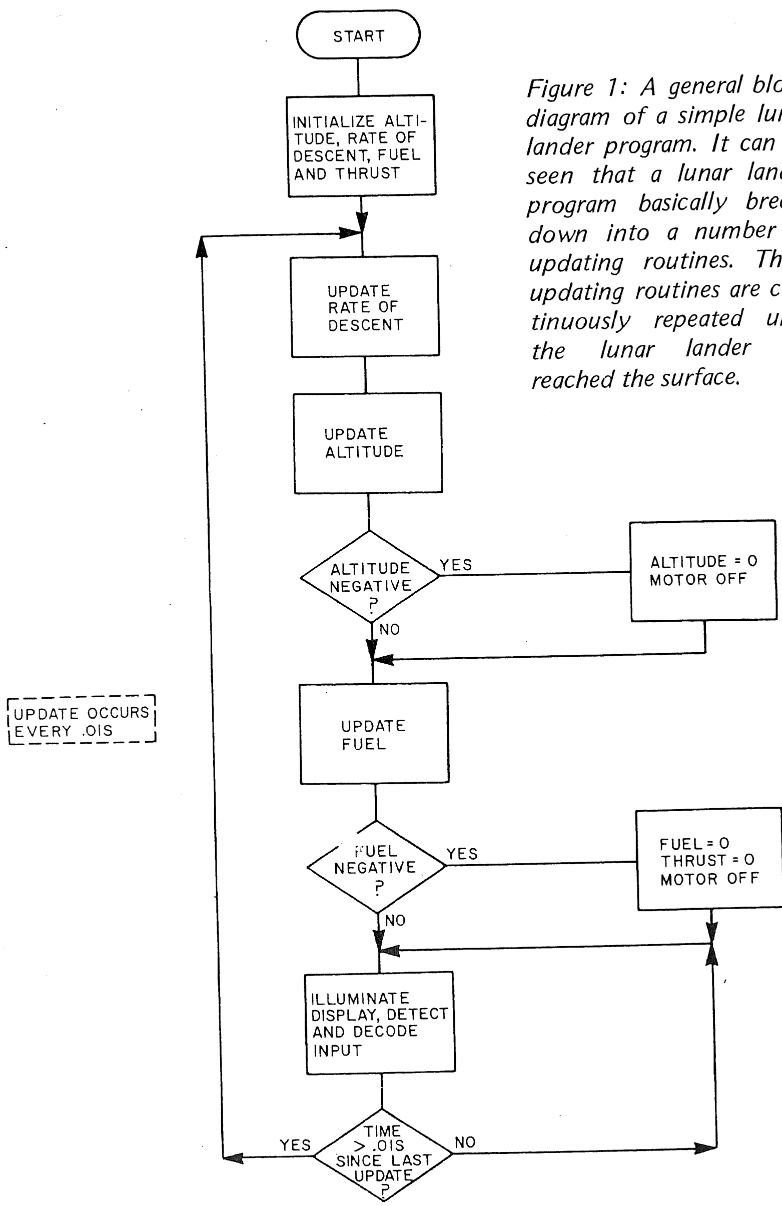

Figure 1: A general block diagram of a simple lunar lander program. It can be seen that a lunar lander program basically breaks down into a number of updating routines. These updating routines are continuously repeated until the lunar lander has reached the surface.

There are quite a few lunar landing programs available nowadays: some for pocket calculators, others using graphic displays. The one I wrote for my KIM-1, based on the MOS Technology 6502 microprocessor, illustrates many of the techniques needed to develop the program.

The KIM-1 comes with a six digit LED display, which can be accessed by the user. I used the first four digits to represent the craft's altitude, and optionally, the fuel remaining. The last two digits, which are slightly separated from the rest of the display, are used for rate of descent. Both values change continually as the craft moves.

The KIM-1 keyboard is used as the pilot's control panel. Thrust is set by pressing controls 1 to 9. A value of 1 is minimum thrust, and the craft's rate of descent will increase due to gravity. Nine is maximum thrust, which slows the rate of descent sharply. In addition to power control, the pilot can elect to view either current altitude, by pressing A, or remaining fuel, by pressing F.

The Equations of Motion

The craft, of course, moves in accordance with the forces acting upon it: thrust and gravity. A physics textbook shows some rather formidable equations. However, they can be boiled down to the following simple procedure:

Byte April 77

Every 0.01 second,
- add 0.01 of the acceleration to the velocity;
- add 0.01 of the velocity to the altitude;
- subtract 0.01 of the thrust from the fuel.

The acceleration is set equal to thrust minus gravity, and gravity is set at the constant value 5.

The time period of 0.01 s is arbitrary. Since KIM can operate in decimal mode, dividing by 100 becomes an elementary operation. Everything would work just as well if it were done in any other small time increment.

Figure 1 shows an elementary block diagram of the program. After setting the initial flight values, we settle into three main jobs: updating the flight, lighting the display, and detecting input from the pilot.

Setting Initial Values

An interesting flight can be obtained by starting the lunar module at a height of 4,500 feet with 800 pounds of fuel. That's more than sufficient fuel for a safe landing, but not enough to allow for prolonged hovering.

It's not difficult to set all the initial values by programming them individually. However, a faster method is to set them all together in memory and use a loop to initialize all of them. This is what I did as shown in listing 1 on hexadecimal lines 0000 to 0007.

Updating the Flight

Every 0.01 s we must update our rate of descent, altitude and fuel. As previously indicated, we have to add 0.01 of various values into the totals. We can accomplish this quite easily by using a gimmick. Instead of holding the altitude, for example, in feet, let's use two more digits and store it as multiples of 0.01 feet. Now we can add the rate of ascent directly into the six digit number; and the division by 100 happens automatically. For display purposes, of course, we drop the last two digits, so that we're back to height in feet. Using the same technique on the other parameters, we find that the updating job becomes relatively easy.

During the updating task, we must also detect two special conditions: touchdown and out of fuel. This seems fairly simple

Text, listing continued on page 132

Listing 1: An example lunar lander program written for the KIM-1 microprocessor that uses the flowchart of figure 1 as a base. The input and output of this program is handled by routines that are inherent to the KIM-1 system. The data display is seen on the keypad and LED display of the KIM-1 assembly. This display continuously shows the rate of descent, and on command will display either the amount of fuel left, or the altitude of the craft. Keys 1 through 9 are used to input thrust commands, while key A chooses the altitude display mode and the F key chooses the fuel display mode. All the numbers in this listing are in hexadecimal unless otherwise stated.

Address	Op	Operand	Label	Mnemonic	Commentary
0000	A2	0C	GO	LDX #0C	
0002	B5	B8	LP1	LDA INIT,X	initialize values;
0004	95	E2		STA ALT,X	
0006	CA			DEX	
0007	10	F9		BPL LP1	
0009	A2	05	CALC	LDX #05	X:=05;
000B	A0	01	RECAL	LDY #01	Y:=01;
000D	F8			SED	set decimal mode;
000E	18			CLC	clear carry;
000F	B5	E2	DIGIT	LDA ALT,X	
0011	75	E4		ADC ALT+2,X	add each digit;
0013	95	E2		STA ALT,X	
0015	CA			DEX	
0016	88			DEY	set up next digit;
0017	10	F6		BPL DIGIT	
0019	B5	E5		LDA ALT+3,X	
001B	10	02		BPL INCR	
001D	A9	99		LDA #99	
001F	75	E2	INCR	ADC ALT,X	
0021	95	E2		STA ALT,X	
0023	CA			DEX	counter:=counter - 1;
0024	10	E5		BPL RECAL	if counter positive go to RECAL;
0026	A5	E2		LDA ALT	else check if altitude is positive;
0028	10	0B		BPL UP	if altitude positive go to UP;
002A	A9	00		LDA #00	else altitude:=00;
002C	A2	02		LDX #02	X:=02
002E	95	E2	DD	STA ALT,X	
0030	95	E8		STA TH2,X	
0032	CA			DEX	else turn off engine;
0033	10	F9		BPL DD	
0035	38		UP	SEC	set carry;
0036	A5	ED		LDA FUEL+2	
0038	E5	EA		SBC THRUST	update fuel;
003A	85	ED		STA FUEL+2	
003C	A2	01		LDX #01	
003E	B5	EB	LP2	LDA FUEL,X	
0040	E9	00		SBC #00	
0042	95	EB		STA FUEL,X	check if fuel left;
0044	CA			DEX	
0045	10	F7		BPL LP2	
0047	B0	0C		BCS TANK	if fuel left go to TANK;
0049	A9	00		LDA #00	
004B	A2	03		LDX #03	
004D	95	EA	LP3	STA THRUST,X	else turn off engine;
004F	CA			DEX	
0050	10	FB		BPL LP3	
0052	20	AA 00		JSR THRSET	go to THRSET;
0055	A5	EE	TANK	LDA MODE	A:=display mode;
0057	D0	0A		BNE SHOFL	if mode not 00 go to SHOFL;
0059	A5	E2		LDA ALT	AX:=location of altitude;
005B	A6	E3		LDX ALT+1	
005D	F0	08		BEQ ST	go to ST;
005F	D0	06		BNE ST	
0061	F0	A6	LINK	BEQ CALC	
0063	A5	EB	SHOFL	LDA FUEL	A:=FUEL;
0065	A6	EC		LDX FUEL+1	X:=FUEL+1;
0067	85	FB	ST	STA POINTH	display values;
0069	86	FA		STX POINTL	
006B	A5	E5		LDA VEL	A:=velocity sign;
006D	30	06		BMI DOWN	if sign negative go to DOWN;
006F	A5	E6		LDA VEL+1	A:=/velocity/;
0071	F0	07		BEQ FLY	go to FLY;
0073	D0	05		BNE FLY	
0075	38		DOWN	SEC	
0076	A9	00		LDA #00	velocity:=/velocity/;
0078	E5	E6		SBC VEL+1	

Continued from page 9

Listing 1, continued:

Address	Op	Operand	Label	Mnemonic	Commentary
007A	85	F9	FLY	STA INH	
007C	A9	02		LDA #02	} DECK:=02; [counter]
007E	85	E1		STA DECK	
0080	20	1F 1F	FLITE	JSR SCANDS	look for depressed key;
0083	F0	06		BEQ NOKEY	if no input go to NOKEY:
0085	20	6A 1F		JSR GETKEY	else go to GETKEY;
0088	20	91 00		JSR DOKEY	go to DOKEY;
008B	C6	E1	NOKEY	DEC DECK	DECK:=DECK-1;
008D	D0	F1		BNE FLITE	if DECK not equal to 0 go to FLITE;
008F	F0	D0		BEQ LINK	else go to LINK;
0091	C9	15	DOKEY	CMP #15	A:=fuel mode?;
0093	D0	03		BNE NALT	if not fuel mode go to NALT;
0095	85	EE		STA MODE	else MODE:= fuel mode;
0097	60			RTS	return;
0098	C9	10	NALT	CMP #10	A:=altitude mode?;
009A	D0	05		BNE NAL2	if not go to NAL2;
009C	A9	00		LDA #00	else mode:=altitude mode;
009E	85	EE		STA MODE	MODE:=A;
00A0	60		RET1	RTS	return;
00A1	10	FD	NAL2	BPL RET1	return; [illegal mode]
00A3	AA			TAX	else X:=A;
00A4	A5	EA		LDA THRUST	A:=THRUST;
00A6	F0	F8		BEQ RET1	if thrust:=0 go to RET1;
00A8	86	EA		STX THRUST	else THRUST:=X;
00AA	A5	EA	THRSET	LDA THRUST	A:=THRUST;
00AC	38			SEC	set carry;
00AD	E9	05		SBC #05	THRUST:=THRUST - 05;
00AF	85	E9		STA TH2+1	TH2+1:=THRUST;
00B1	A9	00		LDA #00	} A:=00;
00B3	E9	00		SBC #00	
00B5	85	E8		STA TH2	TH2:=00;
00B7	60			RTS	return;
00B8	45		INIT		} [initial height]
00B9	00				
00BA	00				
00BB	99				} [initial speed]
00BC	80				
00BD	00				
00BE	99				} [initial acceleration]
00BF	98				
00C0	02				[initial thrust]
00C1	08				} [initial fuel]
00C2	00				
00C3	00				
00C4	00				[mode]

until we realize that both the altitude and the fuel gauge will probably go right past the zero mark, jumping directly from a positive to a negative value; so a zero test is out. Instead, we take action the instant the number goes negative, restoring it to zero and then taking whatever other action is called for.

Lighting the Display

The display is quite straightforward; in fact, the KIM-1 monitor program has a subroutine to do the job.

Depending on the display mode flag, all we need to do is to move altitude or fuel to the display area, together with rate of descent. Then we call the subroutine to transfer it to the LEDs.

Of course, we must remember to drop the last two digits from the displayed values (0.01 of units, remember?) and to negate the rate of descent, where necessary, so that it shows as a positive number.

Detecting Input

The KIM-1 monitor subroutine that lights the display gives us a free bonus: It also tells us whether or not a key is depressed on the keyboard. To find out which key, we must call another subroutine in the monitor program.

If we discover that the user has input a thrust command, buttons 1 to 9, we first check to see that the motor is on and that we have fuel. Then we set the thrust, and also calculate the acceleration as thrust minus 5, where 5 represents the force of gravity.

The two other legal keys, A and F, set the display mode to altitude or fuel. The program sets a memory location which will be tested by the display routine.

The program doesn't need to worry about when a button is released. Although the question can be quite important for programs that must distinguish between, say, 9 and 99 on the input, the lunar lander doesn't really care. If you leave your finger on the button, it will keep on setting the thrust over and over to the same value, without affecting the flight.

Coming Down

The program doesn't stop. If you run out of fuel, you will watch yourself freefall to the surface. When you land, with or without fuel, your rate of descent freezes so that you can see how hard you landed.

It would be easy to have the display change after you land, to show words such as "SAFE" or "DEAD." The KIM-1 display is segment driven so that you can easily produce special combinations.

The novice astronaut who would like to try his or her hand at flying this, or other, craft should keep the following rules in mind:

1. Always conserve fuel at the beginning by reducing power to minimum thrust.
2. Don't let your rate of descent get excessively high; with my program, it's wise to steady up with a thrust value of 5 when your speed gets over 90 feet per second.
3. As you get to lower altitudes, try to balance your altitude against your rate of descent. At 1000 feet, a rate of descent of 500 feet per second will bring you down in 20 seconds, which is reasonable. Keep that sort of balance.■