UNIVERSITY OF ILLINOIS

DIGITAL COMPUTER

LIBRARY ROUTINE F6 - 239

By A. T. Nordsieck

TITLE: Integration of a system of ordinary differential equations with automatic control of integration interval. (SADOI Only)

TYPE: Closed - no program parameters - entered with link in Q.

ENTRIES: Normal entry is at left of 0L. Special resetting entry: If this routine is entered at the right of 1L it will reset itself into the standard as-read-in condition, using the present values of the S6, S7 parameters and with the truncation accumulating locations reset. Use this entry only if a new sequence of integrations is to be started (different initial conditions or different parameters) without re-reading-in this routine; and then only for the first entry of the new sequence, otherwise much time will be wasted and accuracy will be lost. Special interval controlling entry: if we enter at left of 5L, truncation information is preserved but new values of parameters S6, S7 take effect. This entry is used when ℓ_0 is altered during a sequence of integrations. See S6 parameter.

NUMBER OF WORDS: 129

TEMPORARY STORAGE: a to a + 3n-1 for continuing temporary storage; 0, 1, a + 3n to a + 6n-1 for non-continuing temporary storage.

PRESET PARAMETERS: Locations 3 to 8 must contain the following parameters during input and operation of this routine.

LOCATION	CONTENTS	SIGNIFICANCE
3	00F 00aF	N(a+i), i = 0, 1, ... n-1, are the variables x (independent variable), y_1, y_2, ... y_{n-1}. Initial values placed here. Results appear here. Auxiliary subroutine reads from here. x must be in first location a.

LOCATION	CONTENTS	SIGNIFICANCE
4	00F 00(a+n)F	$n \geq 2$. n is the number of equations including the independent variable equation $\frac{dx}{dx} = 1$. N(a+n+i) are the scaled derivatives $2^{-m} f_i$. Auxiliary must supply these quantities for i = 1, 2, ... n-1. It need not supply $2^{-m} f_0 = 2^{-m}$. $a + 6n \leq 1024$.
5	00F 00mF	$1 \leq m \leq 37$. m is an integer such that $2^{-m}\|f_i\|$ are all guaranteed < 1 throughout integration range. m is provided so that the y_i and f_i may be scaled independently for efficient use of the a + i and a + n + i registers. NOTE: To change this parameter after program has been read in it is necessary to overwrite location 5 with new value of $m \cdot 2^{-39}$; overwrite location (a+n) with new value of $(2^{-m} - 2^{-39})$ and, if this routine begins at p, overwrite location (p+127) with new value 50(p+100) 10(38-m).
6	00F 00(ℓ_0-m)F	$m + 1 \leq \ell_0 \leq 38$. $2^{-\ell_0}$ is the largest interval (increment in x) which the user wishes to permit. Furthermore, no matter what interval is actually used, the points $x = (\text{integer})\ 2^{-\ell_0}$ are guaranteed not to be skipped over provided the initial value of x is such a point. NOTE: To change this parameter after program has been read in, change location 6 and use the appropriate special entry.
7	00F 00eF	$[3/4\ e] \geq \ell_0 - m+1$. Also $[3/4\ e] \leq 50-m$ approximately. Here [3/4 e] means the integral part of 3/4 e. The integer e specifies the accuracy desired by the user: Upon integrating over a range $\Delta x = 1$, the y_i's will be in error by: not more than about 1 in the e^{th} bit for "well-behaved" equations $\left(2^{-m} \left\|\frac{df_i}{dx}\right\| \cong 1\right)$; not more than about 1 in the [3/4 e]th bit for extremely "ill-behaved" equations

LOCATION	CONTENTS	SIGNIFICANCE
		$(2^{-m}\left\|\frac{df_i}{dx}\right\| \simeq 2^{38})$; and intermediate amounts for intermediate cases. For ranges of integration other than $\Delta x = 1$, the error is proportional to the range. If the above rules put the error beyond the 39^{th} bit, it will nevertheless be about 1 in the 39^{th} bit. The computation time depends on e as $2^{e/4}$, roughly. NOTE: To change this parameter after program has been read in, change location 7 and use the appropriate special entry.
8	00F 00bF	At location b, left side, is the entry to the auxiliary subroutine, which must be supplied by the user and which computes the scaled derivatives $2^{-m} f_i$ (i = 1, 2, ... n-1) from the y_i and places them in locations a + n + i.

DURATION

T = 1.20 [10.5n + 4 t] ms, where T is average time for one useful integration step; t is time to execute auxiliary once. The factor 1.20 is accounted for by discarded integrations incidental to keeping the interval correct. When modified into a frozen-interval program functioning like code F1, the time is reduced to: T' = 9.7 n + 4 t ms. per step.

To Freeze Interval: If this routine starts at location p,

	Overwrite:	With:
p+12:	L5 107L L0 78L	22 (p+21)F 00F
p+46:	L5 114L 40 49L	26 (p+34)F 00F
p+67:	40F L7F	26 (p+69)F 00F

(Interval will be fixed at the value $2^{-\ell_0}$, no tests will be made, no integrations discarded. e will be ignored.)

To Reverse Integration Sense: <u>After</u> read in, if this routine starts at p,

	Overwrite:	With:
p+38:	42 44L L5 F	42(p+44)F L1 ( )F
p+120:	40 a+4n+1 L0 a+5n+1	40 a+4n+1 L4 a+5n+1
a+n:	$2^{-m} - 2^{-39}$	$2^{-m} + 2^{-39}$

FF Stops During Read-in:

Order With Sexad. Address:	From Location:	Significance:
FF 044	7L left	a + 6n too large
FF 045	9L left	n too small
FF 046	12L left	m too large
FF 047	15L right	ℓ_0 too large
FF 048	16L left	m too small
FF 049	16L right	ℓ_0 too small
FF 04K	19L right	e too small

FF Stops During Operation:

Order With Sexad. Address	From Location:	Significance:
FF 04S	53L left	A test, so mild that any number whatever should pass it, is failing, hence program is out of order.
FF 04N	62L left	The interval must be reduced below 2^{-38} to achieve the e asked for. A single precision routine cannot do this. Reduce e and try again.

Information Available Upon Exit From This Routine:

Locations

a+i	Contain rounded values of variables at end of just completed step.
a+n+i	Contain approximations to scaled derivatives $2^{-m} f_i$ (not best values) pertaining to end of just completed step.
a+2n+i	Contain carried-over truncation information.
a+3n+i	Contain rounded values of variables pertaining to beginning of just completed step.
a+4n+i	Contain the averaged scaled derivatives $2^{-m} \bar{f}_i$ (best slope of chord) over interval just completed.
a+5n+i	Contain the 3rd approximations to $2^{-m} \bar{f}_i$, against which the N(a+4n+i) were tested to establish that the interval was adequately small.
78L, 80L	Contain the right address ℓ-m, specifying the interval last chosen for use.

Notes On Scaling and Choice of m, ℓ_0, e:

In general if less than full register precision is required, it is best to scale the y_i's so that they are small enough so that m can be taken small. This in turn allows ℓ_0 to be taken small if desired for increased speed. Truncation errors will not propagate more than a few places into the y_i registers because such errors are taken into account.

ℓ_0 should be chosen as small as possible for speed, consistent with printout or display intervals desired. If some $2^{-m} f_i$ has a relatively sharp fluctuation in value in a narrow range of x, then one must guarantee, by taking ℓ_0 large enough or otherwise, that at least one point x = (integer) $2^{-\ell_0}$ falls within the region of fluctuation, otherwise there is some risk of the program missing the fluctuation entirely.

e should be chosen after the scale for the y_i has been chosen. If the maximum value of $|y_i|$ is 2^{-d}, then there will be between e-d and [3/4 e]-d significant bits of the y_i developed (for $\Delta x = 1$), depending on how ill-behaved the equations are. As a further guide in the choice of e, we may remark that for extremely ill-behaved equations the interval will occasionally be forced down to but not beyond $2^{-[3/4\ e]-m}$. For well-behaved equations the interval is only forced down to about $2^{-e/4}$.

The auxiliary must be written to provide precision consistant with the precision demanded of this routine, namely e bits or 39 bits, whichever is the lesser, correct in $2^{-m} f_i$.

Method of Finding a Root of $F(x, y_i(x)) = 0$:

1. Integrate until F reverses sign.
2. Back off by copying N(a + 3n + i) back into a + i; read ℓ and put ℓ_0 equal to $\ell + 2$ or 3 (3 is probably the best); re-enter subroutine at right side of 5L.
3. When ℓ has increased to or beyond some chosen final value $\leq$ [3/4 e] - 1 + m and ≤ 38, terminate loop and read root x from location a = S3.

DESCRIPTION OF SUBROUTINE:

This program is designed to speed up the integration of ordinary differential equations (or of definite integrals) by choosing automatically, and independently for each integration step, a near optimum value of the "interval", i.e. the largest value of increment of independent variable consistent with the accuracy desired. The choice of interval is limited to inverse powers of 2 in order to make the arithmetic fast in a binary machine, but intervals all the way from 2^{-2} to 2^{-38} may be chosen. Thus if the optimum interval varies widely over the complete range of integration much time is saved compared to a similar integration in which the minimum interval is used throughout.

Whether or not the optimum interval varies widely, the user is also largely relieved of the necessity of estimating the correct interval for the accuracy desired.

The "price" for the above described automatic interval feature is 2n additional temporary storage locations and an average of 1 discarded integration step for every 4 useful steps.

The routine, upon being entered finds an optimum interval and executes one step at this interval and returns control to the link address. Thus in order to perform a sequence of integration steps a master routine must be employed and this master routine should monitor the independent variable, not a step counter, to determine how far the integration has proceeded.

We may integrate any set of n simultaneous first order ordinary differential equations (within the memory limitations of the computer) in which each derivative is expressed explicitly in terms of (or at least calculable from) the variables themselves:

$$\frac{dx}{dx} = 1$$

$$\frac{dy_1}{dx} = f_1(x, y_1, y_2 \cdots y_{n-1})$$

$$\frac{d\, y_{n-1}}{dx} = f_{n-1}\,(x, y_1, y_2 \cdots y_{n-1})$$

It is necessary to include the independent variable equation whether or not the other derivatives f_1 to f_{n-1} depend explicitly upon x because the interval will be unpredictable and in general variable. This rules out a counter for x; however for speed and accuracy the x equation is handled as an integer operation.

The computation of the functions f_1 to f_{n-1}, given their arguments, is performed by an auxiliary subroutine which must be supplied by the user since it depends on his particular problem. This auxiliary reads the variables from locations a+i, i = 0, 1, ... n-1 and writes the scaled derivatives $2^{-m} f_i$ into locations a + n + i, i = 1, 2, ... n-1. Note that the auxiliary need not supply 2^{-m} at location a + n. The auxiliary is called in 4 times in the course of any one integration step, whether this step is used or discarded because of failing the interval test.

Since in a sequence of integration steps the "net" change in y_i may be the sum of many numbers (one for each step), measures are taken to minimize the accumulation of truncation errors in the process. Locations a + 2n + i, i > 0, are used for this purpose and hold essentially 1/2 plus the least significant part of the current net increment of y_i. These locations require no attention on the part of the user. Locations a + i meanwhile hold the current rounded y_i.

The standard Runge-Kutta 4th order scheme* is used. In this scheme four approximations $2^{-m} f_i^{I}$, $2^{-m} f_i^{II}$, $2^{-m} f_i^{III}$, $2^{-m} f_i^{IV}$, each depending on the earlier ones, are successively computed and the best mean value $2^{-m} \bar{f}_i$ for the interval is then computed as a weighted average of these four with weights 1/6, 1/3, 1/3, 1/6. Now of the four successive approximations, the third $2^{-m} f_i^{III}$ is nearest the average $2^{-m} f_i$ for small enough interval and differs from it by terms of order $(\text{interval})^2$. Therefore this 3rd approximation when acquired is held and when $2^{-m} \bar{f}_i$ is acquired a test is made:

$$2^{-m} |\bar{f}_i - f_i^{III}| \leq 2^{-[3/4e] + \ell - m}, \quad \text{all } i \geq 1.$$

where $2^{-\ell}$ is the interval being tried. If this test fails the integration is discarded, the interval is halved and another integration tried. A failure of the test when $\ell = 38$ leads to an FF stop, since x can not be integrated properly with any interval smaller than 2^{-38} because the mid-point of the interval must be representable in the machine. If the test succeeds the integration step is accepted.

The preceding paragraph describes the rules by which the interval is decreased when necessary. The routine also automatically increases the interval when possible, according to the following rules: if an acceptable integration step has been performed with interval $2^{-\ell}$, if $\ell < \ell_0$, if the ℓth bit of x is 0 and if a delaying binary counter is right, the interval is doubled and another integration is tried, subject to discard if the test fails. The reason for monitoring the ℓ th bit of x is to insure that the points x = (integer) $2^{-\ell_0}$ will not be passed over. The reason for the delaying binary counter is to prevent excessive numbers of waste integrations in attempting to increase the interval

---

* See Milne: Numerical Solution of Differential Equations, p. 72 for a description of the 4th order Runge-Kutta method used.

too soon after it has had to be decreased. The counter plus the ℓth bit of x cause about four integration steps to have gone by since the last interval decrease, before an interval increase is tried.

When the routine is entered the first time it tries the invertal $2^{-\ell_0}$ and uses it or halves it and tries again in accordance with the outcome of the test. When the routine is entered by the normal entry after it has done some integrating, it remembers the interval last successfully used and tries that same interval first (or double that interval if all the conditions for attempting to double are satisfied).

DATE April 8, 1958 RT: 10/7/60
PROGRAMMED BY A. T. Nordsieck
APPROVED BY D. E. Muller

nj

LOCATION	ORDER	NOTES
	00 K(F6)	
0	L5 4F	Interlude
	L0 3F	
1	50 77L	
	00 1F	
2	40 F	
	00 1F	
3	L4 F	Test a + 6n
	L4 3F	
4	L4 63L	
	42 63L	
5	L0 63L	
	40 1F	
6	L3 1F	
	32 7L	
7	FF 68F	
	L5 F	
8	L0 77L	Test n
	32 9L	
9	FF 69F	
	L1 5F	Test $m \geq 1$
10	36 16L	
	L5 78L	
11	L0 5F	Test $m \leq 37$
	32 12L	
12	FF 70F	
	L1 6F	
13	32 16L	Test $\ell_0 \geq m + 1$
	F5 78L	
14	L0 6F	
	L0 5F	Test $\ell_0 \leq 38$
15	36 17L	
	FF 71F	
16	FF 72F	
	FF 73F	

LOCATION	ORDER		NOTES PAGE 2 F 6
17	L1 7F		Interlude
	10 2F		
18	L4 7F		Test e
	F0 6F		
19	36 20L		
	FF 74F		
20	09 1F		waste
	10 S5		
21	40 1F		$2^{-m} - 2^{-39}$ to location (a + n)
	F1 1F		
22	40 S4		
	F5 78L		
23	L0 5F		
	42 75L		
24	F5 3F		
	L4 F		
25	42 71L		
	00 20F		
26	46 71L		
	L0 79L		
27	46 63L		
	L5 4F		
28	L4 F		
	42 56L		
29	42 58L		Plant addresses depending on a, n, m
	00 20F		
30	46 64L		
	46 74L		
31	L4 79L		
	46 70L		
32	L5 3F		
	L4 F		
33	L4 F		
	42 59L		

LOCATION	ORDER		NOTES
34	F5 59L		Interlude
	42 60L		
35	42 66L		
	00 20F		
36	46 60L		
	46 66L		
37	46 68L		
	L0 79L		
38	46 65L		
	46 69L		
39	L5 4F		
	L4 F		
40	L4 F		
	42 57L		
41	42 61L		
	42 67L		
42	00 20F		
	46 61L		
43	46 67L		
	F5 67L		
44	42 62L		
	42 68L		
45	L5 55L		
	40 107L		
46	F5 45L		
	L4 79L		Copy constants into position
47	40 45L		
	L0 54L		
48	36 45L		
	L1 7F		
49	10 2F		
	L4 7F		
50	L0 6F		Construct and place test quantity
	42 51L		

LOCATION	ORDER		NOTES PAGE 4 F6
51	09 1F		Interlude
	10 F	by 50'	
52	40 104L		
	40 105L		Reset delaying counter
53	19 37F		
	26 80L		
54	75 76L		
	40 128L		
55	50 100L		
	10 S6		
56	N2 2L		
	49 F		
57	N2 80L		
	41 F		
58	L5 S3		
	40 F		
59	75 S4		
	40 F		
60	L4 F		
	40 F		
61	74 F		
	40 F		
62	L5 1S4		
	40 F		
63	75 F		
	40 F		
64	L4 F		Blank constants
	40 S3		
65	74 F		
	40 S4		
66	7J F		
	L4 F		
67	LJ F		
	L4 F		

LOCATION	ORDER		NOTES PAGE 5 F6
68	40 F		Interlude
	L0 F		
69	75 F		
	40 S4		
70	L4 F		
	40 1S3		
71	L4 F		
	40 F		
72	F5 1S3		
	40 1S3		
73	65 S4		
	40 S4		
74	L5 F		
	40 S3		
75	50 100L		
	10 F		
76	00 F		
	00 L		Relativizer
77	00 F		
	00 4F		
78	00 F		
	00 37F		
79	00 1F		
	00 F		
80	40 128L		2^{-38} to location 128
	50 76L		
81	L5 71L		
	42 82L		
82	42 2L		Reset locations a + 2n + i to 1/2
	49 F	by 80'	
83	F5 82L		
	40 82L		
84	L0 56L		
	32 82L		
85	26 999F		Ind of Interlude
	26 L		
	26 1N		

LOCATION	ORDER		NOTES
0	K5 F	←	Normal entry
	42 92L		
1	26 12L		
	F5 4F	←	initial entry
2	42 2L		
	49 F	by 2,3'	reset locations
3	F5 2L		a + 2n + i
	40 2L		to 1/2
4	L0 108L		
	32 2L		
5	L5 6F	←	interval modifying entry
	42 78L		
6	42 107L		
	K5 33L		
7	42 92L		
	L1 7F		
8	10 2F		
	L4 7F		
9	L0 6F		Construct test quantity
	42 10L		
10	09 1F		
	10 F	by 9'	
11	40 104L		
	40 105L		reset delaying counter
12	L5 107L		
	L0 78L		Test $\ell = \ell_0$
13	32 21L		
	L5 78L		
14	L4 5F		Test ℓ bit of x
	42 15L		
15	F1 S3		
	00 F	by 14'	
16	32 21L		
	L5 105L		Test counter

LOCATION	ORDER		NOTES
17	32 18L		
	41 105L		Advance counter
18	22 21L		
	L5 78L		
19	L0 100L		
	42 78L		double interval
20	L5 104L		
	10 1F		
21	40 104L		
	L5 110L		
22	40 49L		
	F5 21L		Copy N(a+i)
23	42 51L		to a + 3n + i
	F5 23L		
24	22 48L		
	L5 112L		
25	42 26L		
	F5 78L		Shift address for passes 2 and 3.
26	42 80L		
	41 F	by 25, 27'	
27	F5 26L		Clear locations
	40 26L		a + 4n + i
28	L0 109L		
	32 26L		
29	L5 101L		Reset pass counter j
	40 106L		
30	42 39L		shift address for j^{th} pass
	50 30L		
31	26 S8		Call in auxiliary
	F5 106L		
32	L0 102L		
	40 F		Test for j = 3.
33	L3 F		i.e. 3rd pass
	36 45L		

LOCATION	ORDER	NOTES
34	F5 4F	
	42 38L	
35	L5 112L	
	40 40L	Accumulate $3/4\ 2^{-m}\ \bar{f}_i$ in
36	F5 24L	locations a + 4n + i
	42 43L	
37	00 1F	
	F5 48L	
38	42 44L	
	L5 F	
39	L4 128L	Add 2^{-38} for average round-off
	10 F	
40	L4 F	
	40 F	Closed subroutine for
41	F5 38L	$X\ 2^{-q} + Y \rightarrow Z$
	40 38L	
42	L5 40L	
	L4 100L	
43	40 40L	
	L0 F	
44	32 38L	
	22 F	
45	L5 78L	Shift address for pass 4
	42 80L	
46	L5 114L	
	40 49L	Copy $2^{-m}\ f_i^{III}$ to locations
47	L5 82L	
	42 51L	a + 5n + i for test
48	L5 6L	
	42 52L	
49	L5 F	
	40 F	
50	L5 49L	Closed subroutine for copying
	L4 100L	

LOCATION	ORDER		NOTES
51	40 49L		
	L0 F		
52	36 49L		
	22 F		
53	FF 75F		test quantity has overflowed
	F5 106L		
54	40 106L		Advance and test for 4 passes completed
	F0 102L		
55	32 62L		
	L5 4F		
56	42 38L		
	L5 80L		
57	42 39L		Next intermediate values of x, y_i
	L5 116L		to locations a + i
58	40 40L		
	F5 57L		
59	42 43L		
	F5 59L		
60	26 38L		
	L5 106L		Shift address for passes 2, 3, 4
61	10 1F		
	26 30L		
62	FF 76F		Interval too small
	L5 118L		
63	40 65L		
	L5 120L		
64	40 66L		
	50 103L		
65	7J F	by 63, 71'	
	L4 F		
66	40 F	by 64, 70	$2^{-m}\bar{f}_i$ from
	L0 F		$3/4\ 2^{-m}\bar{f}_i$ and
67	40 F		compare with $2^{-m}f_i^{III}$
	L7 F		

LOCATION	ORDER	NOTES	
68	L4 104L		
	36 93L		
69	L5 66L		
	L4 100L		
70	40 66L		
	L5 65L		
71	L4 100L		
	40 65L		
72	L0 119L		
	32 64L		
73	L5 112L		
	42 77L		
74	L5 122L		
	40 79L		
75	L5 123L		
	40 81L		
76	42 83L		
	L5 124L		Double precision addition
77	40 84L	by 73', 85'	of increments
	L5 F		
78	50 100L		
	10 S6	by 5',19',96'	
79	L4 F	by 74', 87	
	40 F		
80	32 80L	waste	
	S5 F	by 26, 45'	
81	L4 F		
	40 F	by 75', 88'	
82	36 85L		
	36 115L		
83	L4 101L		
	40 F	by 76, 89	Carry into a_{39}
84	F5 F		
	40 F	by 77, 90'	

LOCATION	ORDER		NOTES PAGE 11
85	F5 77L		
	40 77L		
86	L5 79L		
	L4 100L		
87	40 79L		
	L5 81L		
88	L4 100L		
	40 81L		
89	42 83L		
	L5 84L		
90	L4 100L		
	40 84L		
91	L0 125L		
	32 77L		
92	00 1F		Waste
	22 F		Exit link
93	L5 104L		
	50 101L		
94	00 1F		Double and test the test quantity
	40 104L		
95	36 53L		
	40 105L		Reset counter
96	F5 78L		
	40 78L		Halve and test interval
97	F0 127L		
	36 62L		
98	L5 126L		
	40 49L		Copy back x, y_1 for new
99	F5 63L		trial integration
	26 23L		
100	00 1F		Address increment
	00 1F		
101	80 F		Starting constant for pass 1
	00 3F		

LOCATION	ORDER	NOTES
102	80 F	Test constant for pass 3 and end
	00 6F	
103	00 F	
	00 1832 5193 7963F	1/3
	Q1 129K	Causes SADOI to put following material on problem tape immediately after this program.
104	$-2^{-[3/4e]+\ell-m}$	Test quantity
105	counter	Used to delay interval doubling
106	$-1+(j+2)2^{-39}$	j=1, ... 4 for passes 1 ... 4
107	50 100L	
	10 ℓ_0-m	
108	N2 2L	
	49 a+3n	
109	N2 80L	
	41 a+5n	
110	L5 a	
	40 a+3n	
111	75 a+n	
	40 a+4n	
112	L4 a+4n+1	
	40 a+4n+1	
113	74 a+5n	
	40 a+5n	
114	L5 a+n+1	starting, end and test constants
	40 a+5n+1	formed by interlude
115	75 a+2n	
	40 a+6n	

LOCATION	ORDER		NOTES PAGE 13 F6
116	L4 a+3n		
	40 a		
117	74 a+4n		
	40 a+n		
118	7J a+4n+1		
	L4 a+4n+1		
119	LJ a+5n		
	L4 a+5n		
120	40 a+4n+1		
	L0 a+5n+1		Starting, end and test constants
121	75 a+4n		formed by interlude.
	40 a+n		
122	L4 a+3n+1		
	40 a+1		
123	14 a+2n+1		
	40 a+2n+1		
124	F5 a+1		
	40 a+1		
125	65 a+n		
	40 a+n		
126	L5 a+3n		
	40 a		
127	50 100L		
	10 38-m		
128	2^{-38}		