UNIVERSITY OF ILLINOIS

DIGITAL COMPUTER

ILLINOIS CODE L3 - 100

TITLE	The Complete Linear Equation Solver (DOI Only)
TYPE	Entire Program
ACCURACY	Depends upon the condition of the equations to be solved and the number of iterations carried out.
DURATION	(a) - about 15 seconds to input program.
	(b) - about $n^2/20$ (resp. $n^2/100$) seconds for each input of coefficients if coefficients are punched to 12 (resp. 2) figures; n is the number of equations being solved.
	(c) - about $n^3/1400$ seconds to solve equations for each iteration.
	(d) - about 2n/5 seconds to punch results on each iteration (to 11 figures, the maximum allowable)
	(e) - when n = 37 (the maximum allowable) the time is about 4 minutes.

PUNCHING DATA

To solve the set of equations

$$\sum_{j=0}^{n-1} a_{ij} x_j + a_{in} = 0$$

we proceed as follows:

(a) Scale the coefficients (usually simply by moving the decimal point) so that each coefficient is less than 1/2.

(b) Punch each scaled coefficient as a decimal with up to 12 places preceded by a sign, e.g., +016, -204, the decimal point being considered to follow the sign. (On some teletype keyboards + = K, - = S).

(c) Terminate each row $a_{io},\ldots,a_{in}$ of coefficients by punching the character N.

(d) Follow the last N by a sexadecimal character p which determines the number of decimal digits to be printed in the results. The character p can assume the values 1, 2, 3,...,9, K, or S where K = 10 and S = 11. Spaces (5 holes) may be punched at will in the data tape, and a group of spaces should precede the first coefficient.

MATHEMATICAL METHOD USED:

Let $x_0, x_1, \ldots, x_{n-1}, x_n$ be the machine representations of the correct solutions to the set of equations

$$\sum_{j=0}^{n-1} a_{ij}\, x_j + a_{in}\, x_n = 0; \quad i = 0, 1 \ldots, n-1$$

(x_n is a scale factor such that the numerical values $(x_i)_{num}$ of the solutions are

$(x_i)_{num} = x_i/x_n,\ i = 0, 1, \ldots, n-1)$

If a superscript k characterizes the set of solutions at the k^{th} stage of iteration, then

$$\sum_{j=0}^{n-1} a_{ij}\, x_j^{(k)} + a_{in}\, x_n^{(0)} \equiv \delta_i^{(k)}\, x_n^{(0)}$$

defines the residue at the k^{th} stage. (The scale factor $x_n^{(k)}$ is forced to remain $x_n^{(0)}$). If we define $\epsilon_i^{(k)}$, corrections to the solutions at the k^{th} stage, as:

$$x_i \equiv \epsilon_i^{(k)} + x_i^{(k)}$$

then the ϵ's are the solutions of:

$$\sum_{j=0}^{n-1} a_{ij}\, \epsilon_j^{(k)} + \delta_i^{(k)}\, x_n^{(0)} = 0 \qquad (1)$$

If these equations be solved, we obtain a better approximation to our correct solutions as:

$$x_i^{(k+1)} \equiv \epsilon_i^{(k)} + x_i^{(k)} \qquad (2)$$

The process is then repeated ad libitum. Instead of using equations (1) and (2) we use

$$\sum_{j=0}^{n-1} a_{ij}\, \epsilon_j^{(k)} + \delta_i^{(k)}\, x_n^{(0)}\, B = 0, \qquad (1a)$$

$$x_i^{(k+1)} \quad B^{-1} \epsilon_i^{(k)} + x_i^{(k)} , \qquad (2a)$$

where $B = 2^{6+4k}$; $k = 0, 1, 2, \ldots,$

Appropriate adjustments of scale factor are made and all numbers are tested for exceeding capacity in which case B is appropriately reduced.

METHOD OF USE

Case I: Problem placed on machine for first time, i.e., we have as yet no approximate solutions of any kind.

1 - Place program in reader and start with white switch.

2 - Bypass the stop that will occur about one-third of the way in by using black switch.

3 - When program is in, place coefficient tape in reader and start with black switch.

4 - Coefficient tape will be devoured and $x_i^{(0)}$ tape will be punched out; the machine will stop on an 80028 24025 order (sexadecimal).

5 - Rewind the coefficient tape and replace in reader. Upon starting (with black switch) the machine will punch out $x_i^{(1)}$ and stop on the 80028 24025 order.

6 - The above step may be repeated as often as desired producing $x_i^{(k)}$.

7 - If, during read-in of the coefficient tape, the machine should stop on a 24025 L5000 order (sexadecimal) before the tape is totally in, rewind tape and place in reader again. Start the machine again (using black switch). The process will be repeated using a new factor B, reduced from the last previous one by a factor 2^4.

8 - Step 7 can be repeated until the coefficient tape is read-in. If, in this process, the machine ever hangs up on a 19000 L2000 order (sexadecimal) from memory location OKN (sexadecimal), we have an indication that the equations are so badly conditioned that their residues will not accept a multiplication by any $B > 1$ without exceeding capacity. The programmer is advised to go elsewhere for his solution.

Case II: A continuation of iteration after any previous stage and after machine has been cleared.

1 - If after leaving the machine and examining his results at leisure the programmer desires several more iterations, the problem does not have to be repeated from the beginning (k=0).

2 - Read program into machine with white switch. When tape stops, remove it from the reader and move about three feet to the left setting it in the reader on the large group of spaces found there. Start machine (black switch) and read rest of program into machine.

3 - Place any previous answer tape containing $x_i^{(k)}$ in reader (on initial group of spaces) and start machine (black switch). This tape will be read into the machine.

4 - When machine stops, place coefficient tape in reader and start machine. At this point we are at the same stage as in step 5 above. $x_i^{(k+1)}$ will be produced, etc.

FORM OF RESULTS

1 - The results will be printed in a column with an extra space indicating the position of the decimal point. The last entry is the scale factor $x_n^{(0)}$ mentioned above. For example, if we print

12	04	the results are --	12.04
03	72		3.72
00	17		.17
01	00		

2 - The above column of results will be followed by 20 sexadecimal digits which needn't be printed but which are necessary on subsequent read-ins. The 20 digits are:

J2 a b c 500LJ10 d e f 66 g h i

where (a b c) = (10p + q) where 10^{-q} is the scale factor $x_n^{(0)}$ and p is the number of digits printed.

(d e f) = r where 2^r = B, the factor to be used on the <u>next</u> iteration;

(g h i) = (246 + n) where n is the number of equations being solved.

NOTE

Suppose k^{th} iterated solutions $x_i^{(k)}$ are obtained with Method I carried to the k^{th} stage and by Method II by inserting the $(k-1)^{th}$ solutions. These two will not agree exactly since iterations are performed in Method I using 12 place previous results stored in the machine, while the $(k-1)^{th}$ stage results with which Method II starts can be punched out to only 11 figures.

RT: 5/20/59

DATE November 1, 1956

CODED BY J.N.Snyder

APPROVED BY J.P.Nash

lgr

LOCATION	ORDER	NOTES
DECIMAL	ORDER INPUT	
	00 3K	
3	40 208F	Put $x_0^{(\ell)}, \ldots, x_n^{(\ell)}$ in $_0, \ldots, _n$
	50 3F	
4	26 136F	
	L5 144F	(λ_n+1) in R_1 (L.H.)
5	10 20F	(λ_n+1) in R_1 (R.H.)
	42 132F	
6	42 114F	
	L0 104F	(λ_n) in R_1 (R.H.)
7	42 133F	
	42 123F	
8	42 170F	
	L0 131F	(n) in R_1 (R.H.)
9	42 109F	
	00 20F	(n) in R_1 (L.H.)
10	46 109F	
	L4 203F	(λ_n) in R_1 (L.H.)
11	46 179F	
	L5 106F	
12	L4 109F	-- (r_n) -- (r_n) in R_1
	46 178F	
13	42 179F	
	46 82F	
14	42 90F	
	42 124F	
15	46 133F	
	L4 104F	-- (r_n+1)--(r_n+1) in R_1
16	46 105F	
	42 105F	
17	46 107F	
	46 132F	
18	L0 143F	(r_n-1) in R_1 (L.H.)
	46 61F	
19	L5 191F	
	42 100F	

LOCATION	ORDER		NOTES
20	L5 141F		Set switches for residue computation
	42 137F		
21	L5 155F		
	42 147F		
22	81 36F		Read proper proper print parameter from tape and set in printer
	46 29F		
23	L5 122F		Open print adjusting routine switch
	46 91F		
24	81 40F		Set shift orders.
	46 172F		
25	46 177F		
	46 124F		
26	24 37F		Stop for tape, then compute correction
	L5 131F	From 206	Set initial address in printer
27	42 28F		
	92 1017F		15 spaces
28	92 135F		3 line feeds
	L5 (λ_0)F	by 27,32	
29	J2 ( )F	By 22	Print $x_0,\dots,x_n$ from $\lambda_0,\dots,\lambda_n$
	50 29F		
30	26 180F		
	92 129F		
31	L5 28F		
	L4 104F		
32	42 28F		
	L0 114F		
33	32 28F		
	L5 29F		Put print order on tape
34	82 40F		
	L5 124F		Put shift order on tape
35	82 40F		
	24 37F		Stop for tape then compute correction
36	00 F		
	00 F		
	24 999N		

The following will overwrite the above if the last stop is bypassed.

LOCATION	ORDER	NOTES
	00 3K	
3	40 246F	Read in first row of matrix
	50 3F	
4	26 136F	
	L5 144F	(r_n+1) in R_1 (L.H.)
5	46 105F	
	46 107F	
6	46 132F	
	10 20F	(r_n+1) in R_1 (R.H.)
7	42 105F	
	L5 105F	
8	L0 104F	-- (r_n)--(r_n) in R_1
	46 133F	
9	46 178F	
	46 82F	
10	42 179F	
	42 90F	
11	42 124F	
	L0 104F	-- (ν_n-1)--(ν_n-1) in R_1
12	46 61F	
	L0 207F	--(n)--(n) in R_1
13	46 109F	
	42 109F	
14	L4 203F	λ_n in R_1 (L.H.)
	46 179F	
15	10 20F	λ_n in R_1 (R.H.)
	42 170F	
16	42 123F	
	42 133F	
17	L4 104F	
	42 114F	(λ_n+1) in R_1 (R.H.)
18	42 132F	
	41 3F	

LOCATION	ORDER		NOTES
19	L5 104F		Set print parameter to 1
	46 29F		
20	22 40F		Go to code 47
	81 4F	From 119	Compute printing parameter
21	10 39F		
	75 161F		
22	00 59F		
	L4 29F		
23	46 29F		
	L5 191F		Set switches for subsequent computation of residues
24	42 100F		
	L5 141F		
25	42 137F		
	L5 155F		
26	42 147F		
	L5 131F	From 206	Set initial address in printer
27	42 28F		
	92 1017F		15 spaces
28	92 135F		3 line feeds
	L5 (λ_o)F	By 27,32	
29	J2 ( )F	By 19,29	
	50 29F	135	Print $x_0,\ldots,x_n$ from $\lambda_0,\ldots,\lambda_n$
30	26 180F		
	92 129F		Line Feed
31	L5 28F		
	L4 104F		
32	42 28F		
	L0 114F		
33	32 28F		
	L5 29F		Put print order on tape
34	82 40F		
	L5 122F		Open print adjusting routine switch
35	46 91F		
	L5 124F		Print number of shifts
36	82 40F		
	24 37F		Stop for tape, then compute correction

LOCATION	ORDER		NOTES
	00 37K		
37 (0)	41 3F	From 35F	m = 0
	L5 1L		to set $M(a_{00})$ 0
38(1)	42 22L	By 44	
	50 284F		Address is used.
39 (2)	40 246F		
	50 2L		Input row m = $r_0, \ldots, r_n = a_{mi}$,
40 (3)	26 136F		$i = 0, \ldots, n$
	41 4F	From 20F	i = 0
41 (4)	L5 22L		
	42 5L		Make $a_{m0} = 0$ so that the virtual nth
42 (5)	49 2F		row will be interchanged with the
	$41(a_{io})$F	By 4	input row.
43(6)	L5 1L		
	42 22L		Set starting values to a_{00}.
44 (7)	42 27L		
	L5 22L	By 41	
45(8)	42 14L		Set address of $a_{i,i}$ in the division
	42 16L		and test orders.
46 (9)	42 18L		
	L5 4F		
47 (10)	L4 2L		
	46 15L		
48 (11)	46 16L		Set addresses of r_i in the division
	46 19L		and test orders, and initial values
49 (12)	46 21L		in the cycle.
	46 25L		
50 (13)	L5 67L		Set both addresses in the interchange
	42 21L		orders to 1.
51 (14)	46 23L		
	$L3(a_{ii})$F	By 8	Test and see which leading element
52 (15)	$L6(r_i)$F	By 10,26	r_i or a_{ii} is larger.
	36 18L		$= \|r_i\|$
53 (16)	L5 (r_i)F	By 11	$- \|a_{ii}\|$
	$66(A_{ii})$F	By 8	Divide and arrange no interchange

LOCATION	ORDER		NOTES
54 (17)	43 21L		
	26 20L		
55 (18)	50 66L		To take care of case when $a_{ii} = r_i$
	75(a_{ii})F	By 9	Divide and arrange
56 (19)	66(r_i)F	By 11	
	47 23L		interchange
57 (20)	S1 21L		
	40 5F		Store ratio = -k in 5
58 (21)	L5(r_s)F	By 12, 32, 31,	
	40 (1)F	By 13,17	Copy r_s and a_{is} into 0 and 1 <u>or</u> 1 and 0
59 (22)	50 5F		
	L5 284F	By 1,6,29	
60 (23)	40 (1)F	By 14, 19	
	7J 1F		
61 (24)	50(r_n-1)F	By 12	Replace r_{is} by $r_{is} - ka_{is}$ <u>or</u> $a_{is} - kr_{is}$
	L4 F		
62 (25)	40(r_{is})F	By 12,31	
	L4 2F	1/2	Test if result is greater than 1/2;
63 (26)	36 27L		if so record in address of 15.
	47 15L		
64 (27)	L5 1F		
	40(a_{is})F	By 7,29	
65 (28)	L5 22L		
	L0 66L		
66 (29)	42 22L		
	42 27L		Increase s from i to n.
67 (30)	L5 25L		
	L4 67L		
68 (31)	46 25L		Repeat until s = n+1
	46 21L		
69 (32)	L0 70L		
	36 21L		
70 (33)	L5 15L		
	L0 61L		Test if any result > 1/2, if so
71 (34)	32 39L		divide row by 10.
	L5 69L		Set starting value.

LOCATION	ORDER		NOTES
72 (35)	40 36L		
	50 39L	By 30,1/10	
73 (36)	7J(r_s)F	By 35, 38	
	40(r_s)F		
74 (37)	L5 36L		
	L4 67L		
75 (38)	40 36L		
	L0 68L		
76 (39)	32 35L		
	L5 4F	By 34	
77 (48)	L4 71L		Increase i
	46 4F		
78 (41)	L0 3F		Till m.
	32 7L		
79 (42)	L5 3F		Increase m.
	L4 67L		
80 (43)	40 3F		
	L0 72L		Till n + 1
81 (44)	36 2L		End of reduction
	L5 65L		$x_n = 1/10$
82 (45)	40(r_n)F	By 10	
	L5 22L		Set $a_{n-1, n+1}$
83 (46)	42 50L		
	L5 24L		Set address of x_{n-1} in comparison
84 (47)	46 73L		constant and substitution order.
	46 62L		
85 (48)	L5 68L		
	46 50L		
86 (49)	27 76L		
	S5 F		
87 (50)	50(r_s)F	By 48,55	Evaluate scalar product $\sum_{s=i+1}^{n} a_{is} x_s$
	74(a_{is})F	By 46,55	using words 50 through 55
88 (51)	L4 F		
	40 F	By 76	
89 (52)	L4 2F	1/2	
	32 54L		

LOCATION	ORDER		NOTES PAGE 8 L3
90 (53)	50 65L	1/10	If we are liable to exceed capacity
	7J(r_n)F	by 11	reduce x_{n-1} by 10.
91 (54)	26 96L	by 35	
	L5 50L	from 52	
92 (55)	L0 67L		Reduce s from n to i.
	42 50L		
93 (56)	46 50L		
	42 58L		
94 (57)	42 60L		Set a_{ii} in comparison and division
	L0 73L		orders
95 (58)	32 49L		
	L3(a_{ii})F	by 57	
96 (59)	L6 F		Test if we can divide a_{is} x_s by a_{ii}.
	36 53L		
97 (60)	L5 F		R_1 and R_2 $\sum_{s=i+1}^{n} a_{is}x_s$ Divide
	66(a_{ii})F	by 57	To
98 (61)	L6 2F		Constant From x_i
	S1 62L		
99 (62)	40(r_i)F	by 47	
	L5 62L		
100 (63)	L0 75L		
	36 115F	by 24	Switch
101 (64)	L4 74L		
	26 47L		End of back substitution
102 (65)	00F 001000		
	0000 0000J	1/10	
103 (66)	7L 4095F		
	LL 4095F	$(1-2^{-39})$	
104 (67)	00 1F		
	00 1F		
105 (68)	LJ(r_n+1)F	by 5	
	40(r_n+1)F	by 7	
106 (69)	7J 246F		Test constants
	40 246F		
107 (70)	N0(r_n+1)F	by 7	
	L4 2F		

LOCATION	ORDER		NOTES
108 (71)	80 1F		
	00 F		
109 (72)	80 nF	By 13	
	00 nF	By 14	
110 (73)	50(r_{ii})F	By 47	
	L6 F		
111 (74)	NO 1269F	= S	
	L6 F		
112 (75)	NO 246F		
	L6 F		
113 (76)	00 40F		Clear R_2, then control to 51'.
	23 51L		
114 (77)	12 136F	From 39	
	L5(λ_n+1)F		
115 (78)	L5 94L	From 63	Entry for zeroth solution
	40 79L		
116 (79)	00(L5 r_o)F	By 78,81	
	00(40 x_o)F	From 82	Place $x_0^{(0)},\ldots,x_0^{(n)}$ in
117 (80)	L5 79L		
	L4 67L		$\lambda_0,\ldots,\lambda_n$
118 (81)	40 79L		
	L0 95L		
119 (82)	32 20F		
	26 79L		
120 (83)	L5 94L	From 63	Entry for subsequent improved
	46 86L		solutions
121 (84)	46 89L		Set initial addresses
	42 88L		
122 (85)	50 82F		Waste (address used)
	42 89L		
123 (86)	50 (r_o)F	By 83,91 From 93	
	75 (λ_n)F	By 16	Compute corrected $x_o,\ldots,x_n$ and
124 (87)	10 (10)F	By 205	place in $r_o,\ldots,r_n$ and $\lambda_o,\ldots,\lambda_n$
	66(r_n)F	By 11	
125 (88)	S5 F		
	L4 (λ_o)F	By 84,92	

LOCATION	ORDER		NOTES
126 (89)	40(r_0)F	By 84,91	
	40(λ_0)F	By 85,91	
127 (90)	L5 89L		
	L4 67L		
128 (91)	40 89L		
	46 86L		
129 (92)	42 88L		
	L0 96L		
130 (93)	36 204F		
	26 86L		
131 (94)	L5 246F	r_0	Starting constant
	40 208F	λ_0	
132 (95)	L5(r_n+1)F	By 6	
	40(λ_n+1)F	By 17	End constants
133 (96)	40(r_n)F	By 6	
	40(λ_n)F	By 18	
134 (97)	L5 29F	From 54	Adjust printing parameter
	L4 67L		
135 (98)	46 29F		
	22 45L		
	00 136K		
136 (0)	S5 F		Set link address and storage address
	46 8L		
137 (1)	L4 4L		
	42 11L	By 25	Switch
138 (2)	81 4F		Read in 1st sign digit, a_0
	L0 25L	-10	
139 (3)	22 10L		
	40 2F		Store a_1 - 10
140 (4)	L5 F	+ M_1	$M_i = N_i - D_i/2$
	66 1F	$+D_i/2$	$M_i/D_i+2 = N_i/D_i + 2 - 1$
141 (5)	10 1F		$N_i/D_i - 1/2$
	SJ 42L		
142 (6)	40 F		
	L1 F	From 15	

LOCATION	ORDER		NOTES
143 (7)	40 1F		
	L5 (2)F	By 10'	Choose and store either + N_i/D_i or
144 (8)	40 (n)F	By 0'	$-N_i/D_i$
	L5 8L		
145 (9)	L4 4L		Increase storage address
	46 8L		
146 (10)	L5 2F	From 3	
	42 7L		
147 (11)	L0 23L	-2	
	32(q+1)F	By 1' and 26F	= N return to main routine or to residue computer (switch)
148 (12)	L5 24L	+ 5	
	40 1F		$D_0/2 = 5$
149 (13)	41 F		
	81 4F		Read in digit a_1
150 (14)	L0 25L	-10	
	40 2F		Store $a_1 - 10$
151 (15)	32 6L		Check for sign or terminating symbol
	L4 24L	+5	
152 (16)	40 F	From 22'	Store $a_1 - 5 = M_1$, M_i
	81 4F		Read in a_i
153 (17)	50 1F		$D_i/2$ in R_2
	40 2F		Store a_i
154 (18)	L0 25L	-10	$a_i = 10$
	32 3L		Check for sign or terminating symbol
155 (19)	75 25L	x 10	$D_i/2 \times 10$ in R_2
	S5 26L		
156 (20)	40 1F		Store $D_{i+1}/2 = D_i/2 \times 10$
	50 25L		
157 (21)	75 F	x M_i	$10 M_i \ 2^{-78}$
	00 39F		$10 M_i \ 2^{-39}$
158 (22)	L4 2F	+ 2_{i+1}	$M_{i+1} = 10 M_i + a_{i+1}$
	26 16L		
159 (23)	00 F		
	00 2F		
160 (24)	00 F		
	00 5F		

LOCATION	ORDER		NOTES
161 (25)	00 F		
	00 10F		
162 (26)	00 F		
	L5 203F		Set initial addresses
163 (27)	40 29L		
	41 F		Clear R_1, R_2, 0
164 (28)	50 F		
	S5 F	From 33	
165 (29)	00($50\lambda_0$)F	By 26	Accumulate scalar product in 0
	00(74 r_0)F	By 32	
166 (30)	L4 F		
	40 F		
167 (31)	L5 29L		
	L4 104F		Advance addresses and test for end of scalar product
168 (32)	40 29L		
	L5 43L		
169 (33)	L0 29L		
	32 28L		
170 (34)	L5 F		Compute δ_1 and place in 0
	66 (λ_n)F	By 16	
171 (35)	S5 F		
	40 F		
172 (36)	19 (10)F	By 205	
	L2 F		$2^{-11} - \|\delta_1\|$
173 (37)	32 40L		
	L5 36L		If $\|\delta_1\| > 2^{-11}$ reduce shift orders by 2 and stop to accept matrix tape again.
174 (38)	L0 182F		
	46 36L		
175 (39)	46 41L		
	46 124F		
176 (40)	24 37F		
	L5 F		Put $2^{10}\ \delta_1$ in r_n and leave
177 (41)	00 (10)F	By 205	
	50 F		Waste
178 (42)	40(r_n)F	By 9	
	22(q+1)F	By 2	

LOCATION	ORDER		NOTES PAGE 13 L3
179 (43)	50(λ_n)F	by 15	
	74(r_n)F	by 10	End test
	00 180K		
180 (0)	40 F		Store x at 0
	15 22L		
181 (1)	S4 F	by 1'	
	46 1L		
182 (2)	40 2F		
	92 961F		Space
183 (3)	L5 F		
	36 5L		
184 (4)	92 706F		- Sign
	22 5L		
185 (5)	92 644F		+ sing
	50 161F		10 x 2^{-39}
186 (6)	22 7L		Construct 10^p x 2^{-39} and place in 1.
	46 1L		
187 (7)	75 161F		10 x 2^{-39}
	L5 1L		
188 (8)	L0 15L		
	32 6L		
189 (9)	S5 F		
	40 1F		
190 (10)	50 L		From $\|x\| + 1/2 \times 10^{-p}$
	67 1F		
191 (11)	L7 F		
	S4 120F		
192 (12)	40 F		
	50 F		
193 (13)	75 161F		
	00 36F		Print one digit
194 (14)	82 4F		
	10 40F		
195 (15)	S4 10F		
	40 F		

LOCATION	ORDER		NOTES
196 (16)	L5 1L		
	L0 12L		Should we space?
197 (17)	46 1L		
	00 9F		
198 (18)	36 19L		
	92 643F		
199(19)	L5 2F		
	L0 15L		
200 (20)	46 2F		
	00 10F		Have we printed the last digit?
201 (21)	32 12L		
	22 30F		Return to program
202 (22)	02 1013F		
	00 1F		Constant
203 (23)	50 208F	λ_0	A starting constant
	74 246F	r_0	
204 (24)	L5 172F	From 130	Increase shift orders by 4
	L4 37F		
205 (25)	46 172F		
	46 177F		
206 (26)	46 124F		
	22 26F		
207 (27)	00 245F		r_0 -1
	00 245F		r_0 -1
	24 3N		