Digital Computer Laboratory
Massachusetts Institute of Technology
Cambridge 39, Massachusetts

SUBJECT: WWI Operating Speed

To: Mathematics and Applications Group, Systems Group, Block Diagram Group.

From: R. P. Mayer

Date: December 21, 1951

Abstract: Drawing SA-50304 (attached) shows much information which should help programmers calculate the speed of their programs, and which should be helpful to programmers and systems engineers in showing where to concentrate efforts to increase over-all computer speed. The drawing will be brought up to date from time to time, and the latest issue should always be used. This note explains how to use the information on the attached issue.

INTRODUCTION

Sections I, II, and III give a detailed account of WWI operating speed. Some suggestions for using this information begin on page 6. In many cases these suggestions will tell you all you need to know.

Section I. Equations and Symbols (See Section I of SA-50304, attached).

The two equations at the top of the section make use of the letter symbols defined in the remainder of the section.

The first equation shows the precise time required for any order, from time pulse 3 to time pulse 3 (or, speaking very generally, for any full time-pulse cycle, even with any future conditions), with the two following minor exceptions: (1) Restorer pulses ("RPF") are not now synchronous and so can be handled only on an "average" basis; the actual RP time can be somewhat more or less than this average figure, depending on a number of factors (such as the contents of the frequency divider at the start of an ES operation, the length of and spacing between, ES operations, etc.), (2), IO times are uncertain because of continuing modifications in the equipment, operating drift in the equipment, and the initial condition of, and previous orders given to, the equipment.

The second equation shows the precise time required for any program, with the above-mentioned minor exceptions. Orders which are repeated in a cycle must be counted as many times as they are performed; orders which are not performed (due to a cp which is inserted to handle an unlikely situation) must not be counted; if one or another set of orders is to be performed (based on cp) then the longer time, the shorter time, or some median value based on the probability of performing one of the sets, but not the sum of the orders performed in both sets, should be used; orders which do not originally exist but are later generated and performed must be counted. In short, count the orders which will actually be performed in the average (or maximum, or minimum) run through the program, regardless of the orders existing in storage. This equation can also be used for finding average order time in a program by simply dividing the total time by N. Actually, "d" is the only AE (Arithmetic Element) time that is not a function of the individual order being performed, so that "Dd" is the only correctly defined AE time. Each of the others (such as Mm) should actually be the sum of the individual processes performed, or $\left(\sum_{i=1}^{M} m_i\right)$.

The letter symbols, in general, should be self-explanatory. The sketch showing restorer pulses represents LFCP (Low Frequency Clock Pulses)by long upward marks, HFCP (High FCP) by short upward marks "half" of which exactly coincide with LFCP, RP (Restorer Pulses) by long downward marks, LFCP missed (during the RP wait) by high dots, and HFCP missed (during the RP wait) by low dots. It is assumed that the LFCP occur at 1 mc and the HFCP at 2 mc. "TRP" is the time for RP, or the number of LFCP missed. If no restorers occur at all, RPF=1. ES RP (the RP which occur at the start of each ES process) are not included in "RPF" but are treated separately, under ES. Each order group (P, A, T, H, -- F, as defined in the chart in section II) groups together those orders whose times are defined alike ("T" and "O" could be grouped together at present, but not if selective write is not used.)

The number of LFCP used in performing an AE operation (or ES or IO) equals the number of LFCP not used as time pulses, and so equals the amount of time that must be added to the order time. "Multiply" has 15 "shifts" plus as many "adds" as there are "ones" (of positive magnitude) in AC at the time mr or mh is ordered. This is done at 2 mc, and the rounding off required is discussed in connection with the "shift" time. "Divide" is performed by pulses from the divide pulse distributor which has, at present, a count cycle of 3 LFCP. The sl pulse occurs on one of these counts and adds to the step counter, which allows 17 shifts. The sl does not occur at the end of the count cycle so the last sl stops the "divide" before the cycle is completed. Thus only 17-1 full count cycles occur. The last, partial cycle uses 2 LFCP which must be included in the "divide" time. "Shift" (right or left) has as many "shifts" as the address of the order specifies (modulo 31). This is done at 2 mc, but the operation is not "completed" until the next LFCP occurs. If 0 or 1 shift is ordered, then no LFCP are lost due to the shift (the 1 HFCP used for the shift appears between the two LFCP). If 2 or 3 shifts are ordered, then one LFCP is lost, etc. So the time lost due to the shift is half the number of shifts, rounded off low. "Scale factor" has as many shifts (at 1 mc) as there are zeros (of positive magnitude) preceding the most significant "one" (which may be in AC or BR), not counting the AC sign (which always has a positive magnitude of zero). But these shifts are preceded by a sensing pulse, which may be taken into consideration by including the AC sign digit in counting the zeros. If there is no "one", the process is terminated when the step counter overflows, which occurs after 33 shifts, making 33 + 1 sensing = 34.

The selective write system of operating ES affects time only in that an ordinary "ES Read" is added preceding every "ES Write." The equation for any program is written so that an ES Read is included on all orders containing an ES Write if and only if selective write is used, with the exception that group "0" includes the ES Read regardless of whether selective write is used. The read (or write) process time can be found by adding up the ESC Reset times from "Read" (or "Write") to "F" (found on the latest drawings for ESC) as shown. The read process thus includes the rewrite. But the write process does not include the previous read, which is handled separately by use of "SW", the selective write factor.

The IO times are, at this time, so much subject to change that it is hardly worth-while discussing them. It might be pointed out that a program, or section of a program, with <u>any</u> present IO orders (including 'scope display orders) will sometimes be so short, compared to the IO time, that it is not necessary to consider the other order times at all, or in any great detail.

Section II. <u>Individual Order Times</u> (See Section II of SA-50304, attached).

The numerical values tabulated in the left half of the section result from using the equations (in Section I), with the assumptions shown in the remainder of this section. The symbols used are the same as in Section I except that ESR is written "R" when it obviously refers to ES rather than an order group.

The assumptions were chosen to represent conditions usually found in WWI these days, with the following exceptions: IO time is not included but should be handled separately (see discussion of Section I); no extra HG time is assumed although it may occasionally be used, in which case such time should be multiplied by (NR+NW) and added to the order time; likely AE times have been assumed, but if information about the AE time for the specific order is available it can be inserted as shown in the column labelled "composition" (note: "d" time can be assumed not to vary with the order). Restorers (5/16) are assumed (RPF = 16/11), except that two columns are labelled "No R.P." for convenience in investigating possible future operation of WWI without restorer wait periods. (Note that all columns assume 5μs of ES RP whenever ESR or W occurs). The order might be stored in ES or TS, and its address may refer to ES or TS; the four possibilities thus presented are tabulated in four columns, "TS Only" through "ES Only." For "ES Only" the ES processes involved are tabulated under "ES Used," (the first R is for the order, and the remainder is for the address) and the total ES times for any of the above four cases is listed under "ES Times Assumed."

Section III. <u>Average Order Times in Programs</u> (See Section III of SA-50304,attached)

The numerical values tabulated in the small black box in the upper right-hand corner of this section result from using the equations (in Section I), with the assumptions of section II and with assumed programs as shown in the upper left-hand corner of this section. The graphs at the bottom of the page show the same results, with and without RP times, and for a wide range of ES times. Note, in particular that NO IN-OUT ORDERS ARE INCLUDED in this section.

The values for assumed programs tabulated in "Percentages of orders used in programs" were obtained by counting orders performed (as discussed under Section I) in the following actual programs: (1) Selected programs written by the Application Study Group (C-62 to C-104), (2) R-155 (L-1), (3) E-161 by Orden, (4) R-156 by Adams, (5) a program written by John Dodd. (See E-267, "Time Saved by Simultaneous Operation of AE and ES.") Each of the last four programs consist of two distinct parts, and all the programs listed as "program (1)" are grouped together as one "part." Percentages were found for the orders in each part separately. The maximum (and minimum) percentages found, regardless of which part they occurred in, are tabulated under "Max" (and "Min"). The values tabulated under "Med" are approximate weighted median values between max. and min. These three columns are, of course, not normalized to 100%. From these values several kinds of programs were assumed: an average slow program was assumed to have a minimum use (see "min" column) of P, A, and S orders (which are fast orders -- see "ES Only" in Section II), a median use of M orders (which is medium speed), and a maximum use of other orders (which are slow). These values are tabulated under "Slow Ave." after having been normalized. An average fast program was assumed in just the reverse way and is similarly tabulated under "Fast Ave.". A medium speed program is assumed to have a median use of all orders, so the column "Medium" is merely a normalized version of "Med." The "Illustrative Sample Programs" show possible sequences of orders which fit these three assumed speeds fairly closely. The "Slow Limit" (or "Fast Limit") is the slowest (or fastest) the computer can be made to operate under the given assumptions (no in-out, etc.), and will never be reached (for any appreciable length of time) in a program without a conscious effort to do nothing but reach it. This is because the slow limit results from continually repeating the order sf WHEN AC PLUS BR EQUALS ZERO, which makes very little arithmetic or logical sense, while the fast limit results from continually repeating sp or cp, which also makes very little sense. If only test storage is used, the slow limit results from repeating order dv (which makes very little sense), while the fast limit results from using orders from only groups P, A, T, and O (which are all the orders considered which do not use AE) and can easily be reached in useful programs.

These tabulated values for assumed programs were then inserted in the equation for Average Order time (Section I), and the results are tabulated in the black box. The most important figure is the average order time for a medium speed (or "average") program which exists entirely in ES. This figure is emphasized by being placed in a special black box.

Values for the graphs at the bottom of the page were calculated in the same way, except that no RP were included (see below), and ES Read and Write times were made variable. It is assumed that test storage is not used, but if only TS is used, then this is equivalent to an ESR and W of zero, which can be found on the graph. The equation for average order time can be rewritten as:

$$\text{Ave. Time} = K_1 + K_2 \text{ (ESR)} + K_3 \text{ (W)}$$

where each K is a function of the number of various kinds of orders in the program, and replaces a number of symbols in the original equation in Section I. (For example, $K_2 = \left[N' + (A'+O'+M'+D') + (SW)(T'+F'+H')\right]$.) Each K can be found by inserting the various assumptions in the original equation. The graphs can then be drawn. Then if RP are assumed the ESR and W time must be changed, but also the value of K_1 changes by an amount equal to

$$K_4 = K_1 \text{ (RPF)} - K_1 = K_1\left(RPF - 1\right) = K_1\left(\frac{TRP}{ARP - TRP}\right)$$

so that the above equation can be rewritten as

$$\text{(Ave Time with RP)} = \text{(Ave Time)} + K_4.$$

Since K_4 is also a function of the number of various kinds of orders, K_4 for each assumed program is listed under each graph as a value to add to the value obtained from the graph.

The graph for the medium speed program is not shown in Section III, but is shown separately on SA-48264-G (attached). This graph is constructed in the same way as the others, except that K_4 is not merely stated but is used to provide an additional scale so that times with and without RP can be read directly (but different values of ESR and W must be used with and without ESRP).

All of the above graphs are combined on SA-48265-G (attached), but RP are assumed and ES times are assumed (but with variable HG time) as in Section II. These curves thus present the same information as the black box in Section III except that a wide range of extra HG time is shown. Note that

$$\text{(Ave. order time)} = \text{(Ave. order time with NO HG)} + K_2\text{(HG)} + K_3\text{(HG)}$$

which can be rewritten as

$$\text{Time} = K_5 + (K_2 + K_3)\ \text{(HG)}$$

or as

$$\text{Time} = K_5 + K_6\text{(HG)}$$

where K_5 is the value found under "all ES" in the black box of Section III, and K_6 can be found easily. The curves can then be drawn, placing "0" HG away from the vertical axis by ESR for convenience, and showing the "All TS" values also (which are NOT an extension of the simple equation above).

Suggestions for using this information (SA-50304, attached).

1. General

A. Consult the latest issue of the drawing (not attached).

B. The average over-all speed of WWI is given in the special black box of section III, but this does not include In-Out order times.

C. See section I for I - O times.

D. For some programs, the I - O time alone determines the speed.

E. For many programs, average over-all speed plus I - O time is sufficiently accurate.

F. Some programs, or sections of programs (e.g. those between I-O orders), make little or no use of I-O orders. They can be analyzed as follows:

a. If the program time is not critical the black box figures of Section III should be sufficient, and indicate the range of speeds to be expected. (Also, see Nomograph below.)

b. If the program time is somewhat critical, or rather critical in a very long program, the values of Section II should be sufficient--possibly modified by information about shift-order addresses. (Also, see Nomograph below.)

c. If the program time is quite critical, especially if it is short, the values of Section II should be modified by the m, d, s, and f times found in Section I and applied as shown in section II under "composition." (Remember that RP time can increase or decrease the order time, and is correct only on the average). (Also, see Nomograph below.)

G. If HG time is to be added, add it to every use of ES (see Section II under "ES used"), or see SA-48265-G (attached).

H. If any changes in timing or organization of WWI occur, refer to Section I.

J. Drastic, or unforeseen, changes in WWI may require complete revision of all sections.

\- - - -- --

2. Programmers

In addition to finding the time required for a particular program (see "Suggestions, 1. General"), a programmer can tell (usually from Section II) which orders he should use to provide the fastest program when several programming possibilities exist. From Section III he can get a feel for the range in average speed possible between "careless" programming and "efficient" programming (total speed is more important than average speed, but with a fixed amount of storage the two may be closely related). He can also get a feel for the number of dv orders, etc., he is likely to have in an average program.

3. Systems Engineers

In addition to estimating the times required for performing certain kinds of programs (see "Suggestions, 1. General"), the system engineer can estimate (usually from Section III) the over-all saving in operating speed that will result from making any timing change in any order, and thus can determine how much effort should be spent in making such a change (for instance, it would probably not be worth while to change dv). He can also tell what savings would result from reducing RP time, ES time, etc. Section III (Normalized table) indicates the duty factors of the different groups of orders and so can be used to find duty factors for some of the components in WWI, and may thus indicate the probability of certain failures occurring or [illegible] detected between periods of testing, etc.

<u>Nomograph</u> (SA-50530, attached)

This nomograph presents some of the information of Section II in a form convenient for finding total program time or average order time. Also, the AE times are presented in full and not merely assumed. The nomograph can be used by taking a strip of paper and marking off, end-to-end, the vertical distances taken from the nomograph for the number of orders of each group. The total length thus marked off can then be placed against the scale to find total or average time. A total of 100% must be marked off to find average order time, but any number of orders may be marked off to find total program time. The "number of orders" scale and the "total time" scale may be multiplied by any convenient power of ten to handle longer or shorter programs.

Signed: R. P. Mayer
R. P. Mayer

Approved: R R Everett
R. R. Everett

Drawings Attached:

SA-50304
SA-48264-G
SA-48265-G
SA-50530

SA-50304-2

WWI OPERATING SPEED

R.P. MAYER SEPT. 12, 1951

SECTION I — EQUATIONS AND SYMBOLS
(Numbers show present values only, some of which are subject to change.)

For Any Order: Time = [(TP)+(AE)][RPF]+[ESR](NR)+(W)(NW)+(IO)

For Any Program: (In counting orders, count number of orders performed.) (Average Order Time = [Total Time]/N)

Total Time = [N(TP) + Mm + Dd + (S+R)(s) + Ff][RPF] + [ESR][N′+(A′+O′+M′+D′)+(SW)(T′+F′+H′)] + [W][T′+O′+F′+H′] + [IO]

TP = Time Pulses = 8

NR, NW = Number of ES Reads, or Writes

RPF = Restorer Pulse Factor = $\frac{\Delta RP}{\Delta RP - TRP} = \frac{16}{16-5} = 16/11$; or $\frac{\infty}{\infty - 0} = 1$

AVERAGE FIGURE ONLY.

LFCP = 1 ms, HFCP = 2 ms, RP

ΔRP — TRP

N = Number of orders in program. N′ = Number of orders stored in ES.

P, A, T, H, O, M, D, S, R, F = Number of orders from specified group. (See chart below.)

A′, T′, H′, O′, M′, D′, F′ = Number of orders (" " ") with ES addresses.

AE = Arithmetic Element Time (See each order.)

m = Multiply Time = [15 + (# of positive "1"'s in AC before M order)] × ½; rounded off low 7 ≤ m ≤ 15

d = Divide Time = [(count cycle)([# of sl] − 1) + 2] = [(3)(17−1) + 2] = 50

s = Shift Time = [Address (modulo 31)] × ½; rounded off low. 0 ≤ s ≤ 15

f = scale Factor Time = [# of positive "0"'s (including signs), in AC & BR, preceding most significant "1"] 1 ≤ f ≤ 31 OR: f = 34 if AC+BR=0

SW = Selective Write Factor: "1" if SW is used, "0" if not.

ESR = ES Read = (Read Process) + (ES RP) + (Extra HG time)

W = ES Write: (Write ") + (" ") + (" " ")

Read Process (includes Rewrite) = ESC Resets: (Read + A + ⋯ D_R + ⋯ F)

Write Process (No previous Read) = ESC Resets: (Write + D_W + ⋯ F)

ES RP = 5 μs

Extra HG Time = Easily adjustable from 0 to 100 μs or more.

IO = In Out time (See each order.)

H orders: [HW − (time for orders since qd or qf) ± 15]; only if positive.

HW = "H" Wait = [H interlock time − (4 + w + ESR)] = [512 − 60] = 452.

Camera: 220 milliseconds.

Reader, Punch, Flexo: (140 millisec) + (flexowriter "Machine Function" time)

PETR: First qr: (1. sec, if more than 30. sec since qr.) PLUS (70 millisec per inch of blank tape.)
Other qr: 7 millisec.

NOTE: 1 × 16/11 = 1.454; 15 × 16/11 = 21.8; 31 × 16/11 = 45.1; 34 × 16/11 = 49.4

SECTION II — INDIVIDUAL ORDER TIMES

TIME FOR INDIVIDUAL ORDERS, WITH CERTAIN ASSUMPTIONS

ORDER GROUP	ORDERS	ES USED	NO R.P. TS ONLY	TS ONLY	ORDER ES ADDR. TS	ORDER TS ADDR. ES	ES ONLY	COMPOSITION
P	cp sp	R	8	11.6	29.6	11.6	29.6	BASIC P TIME
A	ca cs ad su cm sa ck	RR	8	11.6	29.6	29.6	47.6	BASIC A TIME
T	ts td ta							
H (SEE IO TIME)	qh qd qf	RRW	8	11.6	29.6	67.6	85.6	BASIC T TIME
O	ao qe							
M	mr mh	RR	20	29.1	47.1	47.1	65.1	A TIME + m TIME
D	dv	RR	58	84.4	102.4	102.4	120.4	A TIME + d TIME
S	sl sr	R	12	17.4	35.4	17.4	35.4	P TIME + s TIME
R (SEE IO TIME)	qr qp qs							
F	sf	RRW	16	23.3	41.3	79.3	97.3	T TIME + f TIME

ES TIMES ASSUMED:

R = READ + RP
W = WRITE + RP (NOT INCLUDING R)

R = 18 W = 38

R = 18
RR = 36
RW = 56
RRW = 74

(NO EXTRA HG TIME)
(SELECTIVE WRITE SYSTEM)

AE TIMES ASSUMED:

AE PROCESS	WITH NO R.P.	WITH 5/16 R.P.
m	12	17.4
d	50	72.7
s	4	5.8
f	8	11.6

OTHER ASSUMPTIONS:

NO IO TIME.
5/16 R.P. ARE USED UNLESS STATED OTHERWISE.

SECTION — IN PROGRAMS

PERCENTAGES OF ORDERS USED IN PROGRAMS.

ORDER GROUP	VALUES TAKEN FROM TOTAL SURVEY, NOT NORMALIZED: MAX	MED	MIN	ASSUMED KINDS OF PROGRAMS. NORMALIZED: SLOW LIMIT	SLOW AVE.	MEDIUM	FAST AVE.	FAST LIMIT
P	14	8	1	0	.9	7.7	13.2	100
A	50	40	30	30	28.3	38.6	47.2	0
T	40	26	14	0	37.7	25.1	13.2	0
O	17	6	0	0	16.0	5.8	0	0
M	37	15	0	0	14.2	14.5	14.2	0
D	2	1	0	(100)	1.9	1.0	0	0
S	13	7	0	0	0	6.8	12.3	0
F	1	0.5	0	(100)	.9	.5	0	0
TOTAL	174	103.5	45	100	99.9	100.0	100.1	100

ILLUSTRATIVE SAMPLE PROGRAMS

SLOW AVE.	MEDIUM	FAST AVE.
ca mr ts ts ca ad ts ts ao ao — 3 "A", 4 "T", 2 "O", 1 "M", 10 TOTAL	ca ad mr ts ca ad sr ts ao cp — 1 "P", 4 "A", 2 "T", 1 "O", 1 "M", 1 "S", 10 TOTAL	ca su cp mr sl ad su mr sl ad ts — 1 "P", 5 "A", 1 "T", 2 "M", 2 "S", 11 TOTAL

AVERAGE ORDER TIMES

ASSUMED AE TIMES, NO IN-OUT, SELECTIVE WRITE, 18 μs READ + RP, 38 μs WRITE + RP, NO EXTRA HG, WITH 5/16 R.P.

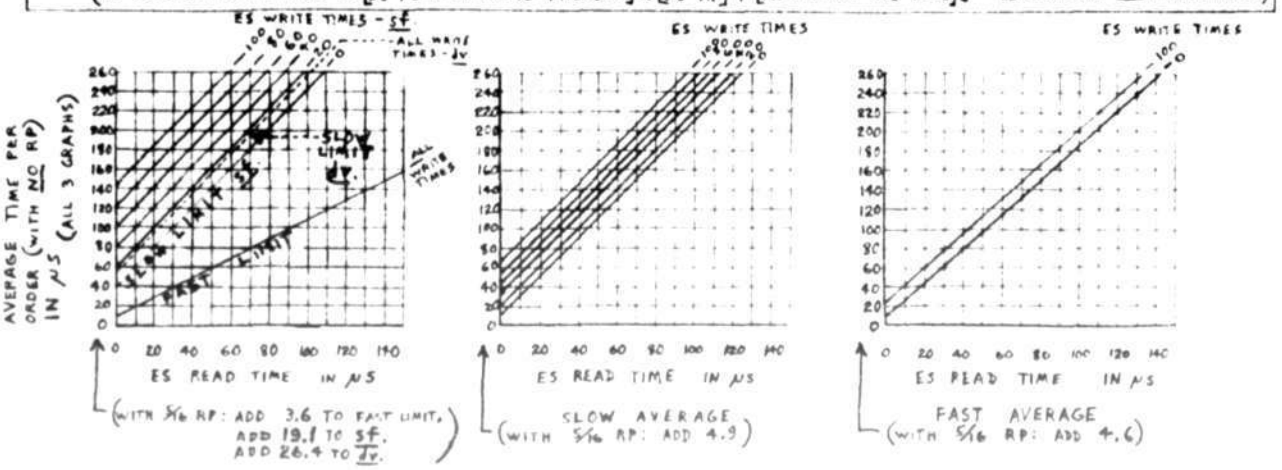
PROG. SPEED	ALL TS	ALL ES
S. LIMIT	84.4	135.1
SLOW	15.6	72.1
MEDIUM	15.3	60.6
FAST	14.8	50.4
F. LIMIT	11.6	29.6

OVER-ALL WWI SPEED (NO I-O)

SECTION III — AVERAGE ORDER TIMES

WWI JULY 1951 ORDER TIMES vs. ES TIMES FOR PROGRAMS OF VARIOUS SPEEDS

WITH: NO IN-OUT ORDERS; NO USE OF TEST STORAGE; SELECTIVE WRITE SYSTEM; ASSUMED AE TIMES.
(ES READ OR WRITE TIME = [ES READ OR WRITE PROCESS] + [ES RP] + [ADDITIONAL HG TIME]. ES WRITE MUST NOT INCLUDE READ.)



(WITH 5/16 RP: ADD 3.6 TO FAST LIMIT, ADD 19.1 TO sf, ADD 26.4 TO dv.)

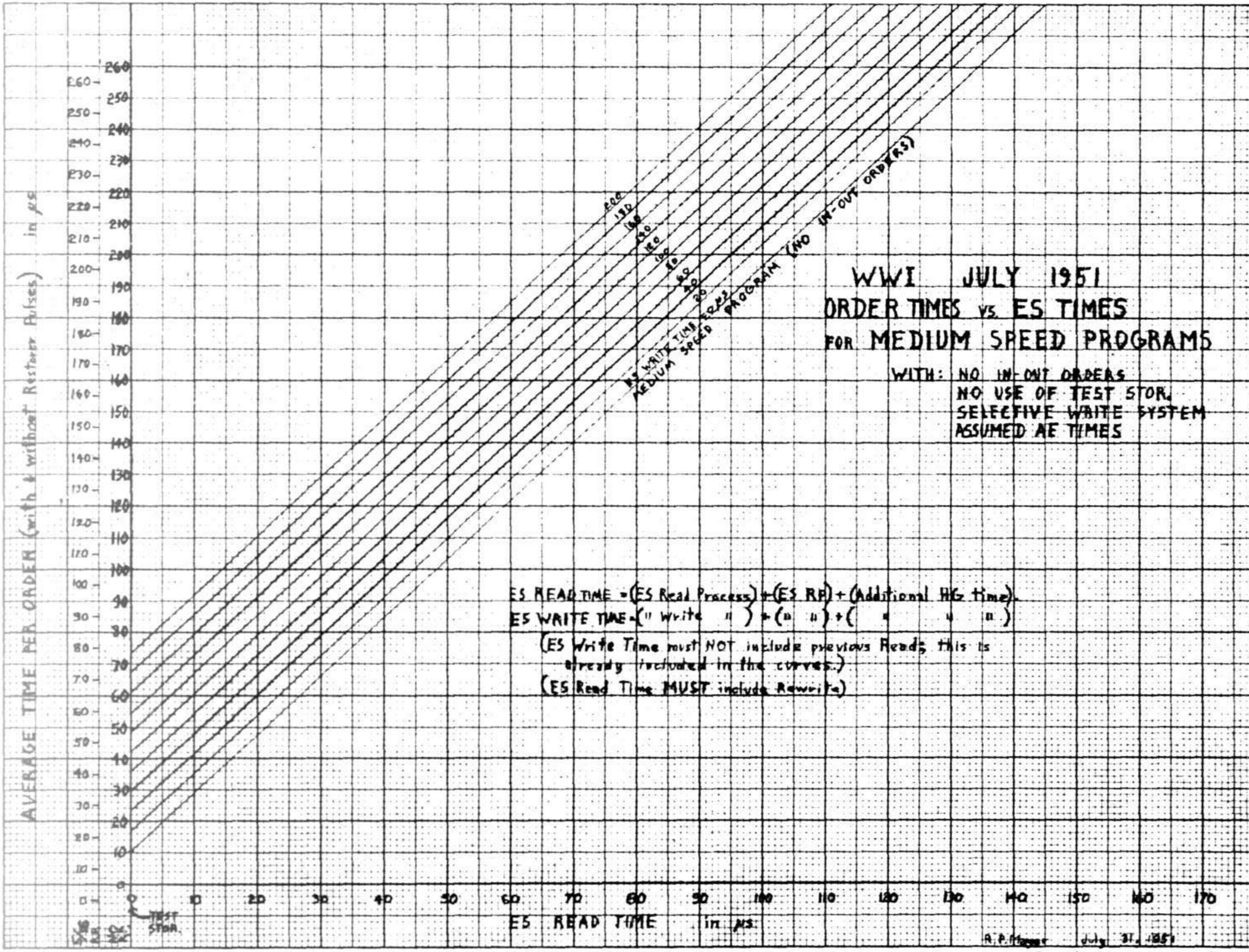
SLOW AVERAGE (WITH 5/16 RP: ADD 4.9)

FAST AVERAGE (WITH 5/16 RP: ADD 4.6)

SEE SA-48264-G FOR MEDIUM SPEED.
SEE SA-48265-G FOR ORDER TIMES VS. HG TIME (FOR VARIOUS SPEEDS).
SEE SA-50530 FOR NOMOGRAPH FOR AVE & TOTAL TIMES FOR ORDERS.
SEE NOTE E-440 FOR DISCUSSION.

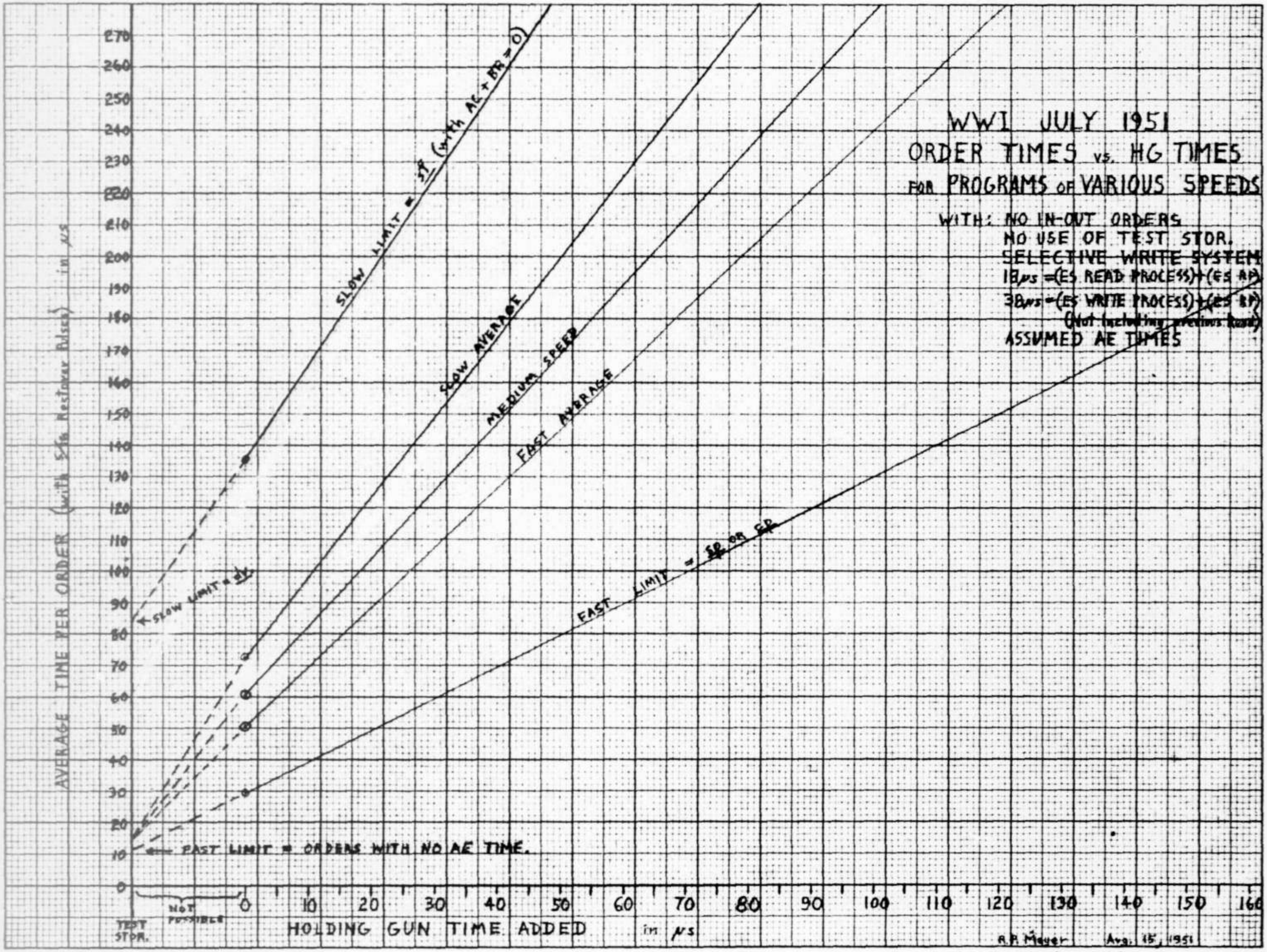
SA-50304-2

SA-48264-G-1

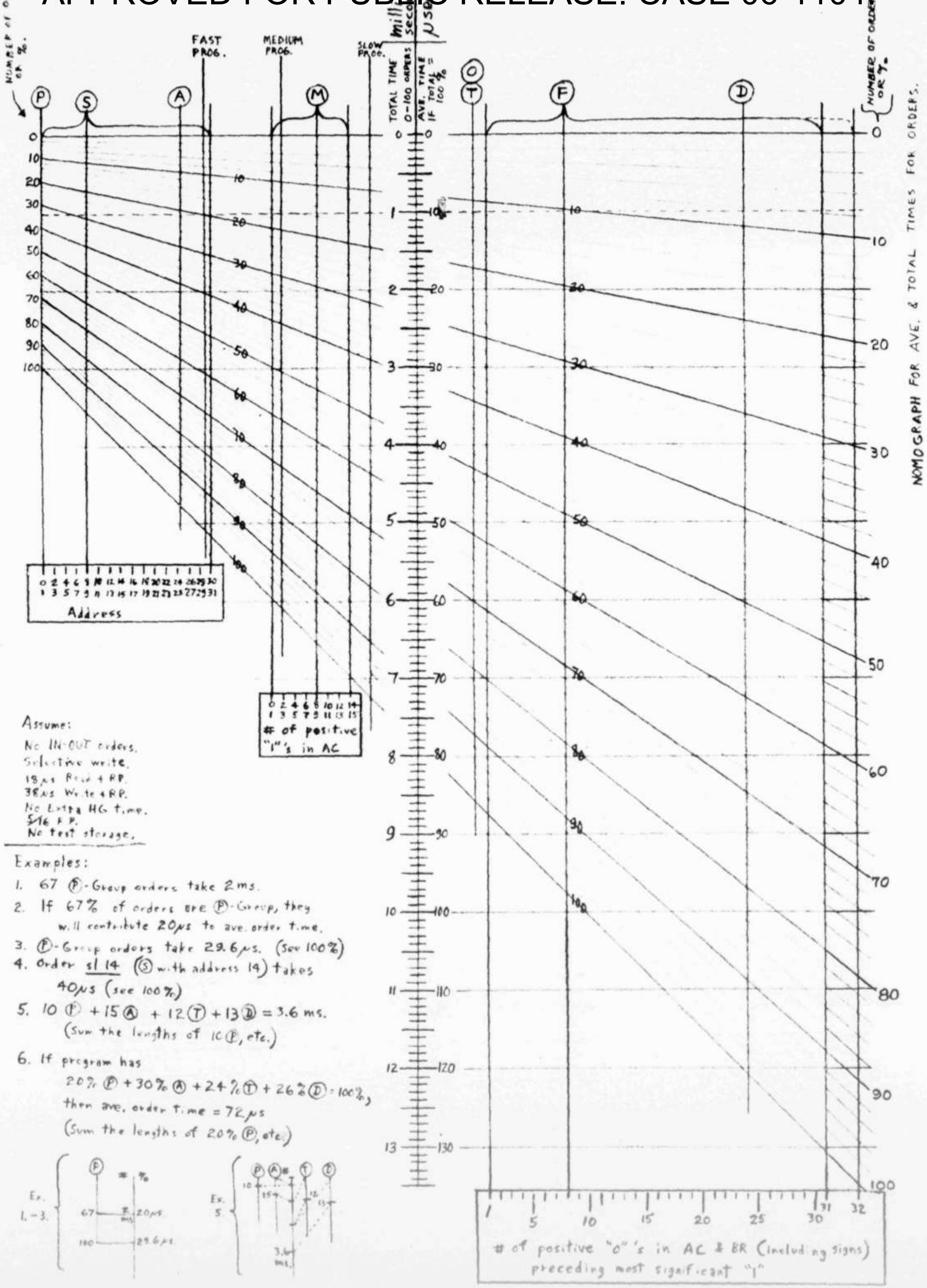


SA-48264-G-1

SA-48265-G-2



SA-48265-G-2



Assume:
No IN-OUT orders.
Selective write.
18µs Read + RP.
38µs Write + RP.
No Extra HG time.
5/16 FP.
No test storage.

Examples:

1. 67 Ⓟ-Group orders take 2 ms.
2. If 67% of orders are Ⓟ-Group, they will contribute 20µs to ave. order time.
3. Ⓟ-Group orders take 29.6µs. (See 100%)
4. Order sl 14 (Ⓢ with address 14) takes 40µs (see 100%)
5. 10 Ⓟ + 15 Ⓐ + 12 Ⓣ + 13 Ⓓ = 3.6 ms. (Sum the lengths of 10 Ⓟ, etc.)
6. If program has 20% Ⓟ + 30% Ⓐ + 24% Ⓣ + 26% Ⓓ = 100%, then ave. order time = 72µs (Sum the lengths of 20% Ⓟ, etc.)

NOMOGRAPH FOR AVE. & TOTAL TIMES FOR ORDERS.
R P MAYER 14 NOV. 1951

SA-50530-1