1900 Series Programming Standards Specification No 012
Range–Compatible Trusted Program Facilities
Issue 2

B.J. Moore
Basic Programming Dept. S.D.O.

20 October 1968

1 Introduction

1.1

The facilities described in this paper are mainly aimed at providing a suitable environment for operating systems, including GEORGE 1 and 2, MINIMOP and possibly PATSY. They are also expected, however, to cater for such software and trace routines as the FORTRAN Batch Monitor and any other system where it is necessary or expedient for one program to control the operation of another. Furthermore, it is intended that these facilities shall be made available to selected customers so that they may, if they wish, write their own operating systems. At present such customers will be severely restricted in number, and must be prepared to change their trusted programs as and when required by changes to this specification.

The specification is based on the original Single–Slot Trusted Program Scheme proposed for the 1902/3. Some changes from this proposal have been made in order to allow for multi–programming machines and some new facilities are included. Wherever possible an attempt has been made to base these new facilities on ones which already exist or for which there is a documented proposal.

1.2

Although this paper is a specification of Executive facilities some idea is given, here and there, of how it is intended the facilities might be used in the trusted program.

Elsewhere an indication of how executive might implement these facilities is given. Where this is done, it must not be taken as a definition of how Executive must be written. If, however, Executive can only be written in a way which alters the meaning of a definition, then the specification will be altered accordingly.

2 Implementation and Availability

The facilities described in this specification will be supplied optionally, as an enhancement of the executives listed below, for use with machines that have 16K words or more of core store. (There is no intention to preclude the possibility of implementation on 4K and 8K machines with disc, but this question will have to be considered separately. The aim of this specification, however, is the production of a system suitable for 16K machines and any complications which may arise in the implementation on smaller machines have not been considered). The facilities will usually be available only in specially compiled Executives and not as load–time modifications.

1902/3	EX2M
1902A/3A	E3TE, E3TM, E3DE and E3DM
1904/5	E4BM *(not together with the existing trusted program facilities but as an alternative which will eventually supplant them.)*
1906/7	E6BM

Standard Executives for all replacement machines for the above and possibly, at a later date, on the 1901 (EX1T and EX1V) and PF279.

On those executives which are overlaid, the trusted facilities will be within the overlays, thus giving no increase in executive size. However, time–penalties will occur for all the trusted features, and therefore overlaid versions are not suitable for off–lining peripheral transfers, existence of several P.U.C. datums, of time sharing by the use of direct response mode.

The interface will also be provided for programs running under the control of GEORGE 3, and a sub–set of the facilities will be available for programs working under GEORGE 1 and 2. The facilities are only available to programs having in the third word of their request slips bit 3 = 1. This bit was formally used to denote type R.

Type Q (File Reorganisation) and Type T (GEORGE 3) categories are not affected.

One or two facilities depend on the availability of suitable hardware and where this is so, the effect of attempting to use them on other machines will be broadly null.

The cost of these facilities, in terms of core store occupied by Executive, will also depend on the hardware, but it is expected that the cost on the 1902/3 (EX2M) will be about 400 words.

Trusted program facilities will be available to some Executives which do not provide subprogramming facilities. In such environments 162–164 instructions (except 164 X = 2 or 7) are illegal for TP and PUC, and references to multimember TP and PUC or members other than member 0 are irrelevant.

3 Program Control

The basic facility is to be able to set up one program, to inform Executive of its existence and then to monitor its progress. The trusted program (TP) and the program under control (PUC) must have different datums and, possibly, limits (the limit may be the size of the store with single–programming Executives) such that the TP will be free to change the contents of any location in the PUC's region of reserved core store. The TP and the PUC have the same program name, and console messages can only refer to the TP.

With a multi–programming executive it will be possible to time–share the number of programs specified for that executive, a TP and PUC combination counting as one program. With a dual–programming Executive for a processor without full program protection (no limit register) the TP facilities will be made available at the higher datum so that it will be possible to time–share one very well de–bugged program at the lowest datum with a TP and PUC pair at the higher datum. The limits of the TP and PUC will always be the same on dual–programming Executives.

It will be possible for the PUC itself to be a TP only if its TP will provide a suitable environment for it, and there is a difficulty here concerned with peripheral allocation. (See §4.1). When a PUC is running Executive, is only aware of one additional datum — that of the 'base' TP and control will always revert to this whenever an event occurs. The 167 and 164 (X = 7) orders are treated as illegal except when issued by a TP

3.1 Method of loading PUC

The TP can load the PUC itself if it wishes or, alternatively, it can 'single–shot' a 154 order planted in the PUC. (See §3.4.2).

3.2 Sub–Programming Restrictions

Facilities are provided for either for either the PUC *or* the TP effectively to have more than one member. Accumulator swapping takes place *either* at the TP datum *or* the PUC datum as determined by the 167 order (q.v.)

3.2.1

If the accumulator swapping level is set to the TP datum then all the usual subprogramming facilities, except priority members, are available to the TP and its member 0 can control a single–member PUC. Under these circumstances the effect of starting any other member at the PUC datum is undefined.

A multi–member TP must be careful to preclude events in the PUC which may cause operator suspension to be set in the slot, since the only way round this sort of suspension is for the operator to type GO AT, which is in general not safe for multi–member programs.

3.2.2

If the accumulator swapping level is set to the PUC datum then all the usual subprogramming facilities, except priority members, are available to the PUC except that 163 orders referring to disestablished members are monitored events. If the TP wishes the PUC to obey this order it can be 'single–shotted'. (See §3.4.2). The member referred to is then said to be established

in the PUC. It becomes disestablished after a 167 order (see §3.3) or a 164 (X = 7, N = 2) order (see §3.4.3).

A TP wishing to run multi–member PUC's will normally have priorities set for all members in its request slip. It can then change these priorities to conform with those requested by the PUC. This method has been adopted in order to avoid the necessity of modifying Executives to be able to delete members. A TP must not disentrust itself (via a 166 order) while monitor suspension is set on any member.

3.3 Extracode to set PUC Parameters

167 X N(M)

where N(M) addresses a word pair P, P+1:
word P contains the requested PUC datum relative to the TP datum;
word P+1 contains the requested store space to be given to the PUC.

X is either 0 or 1. If X = 0, the TP can be multi–member, but if X = 1, the accumulator swapping level will be at the PUC datum, i.e. it will be possible for the PUC and not the TP to have members. Executive will round up the requested datum and (implied) limit so that they are multiples of, for example, 64 on the 1904/5, 128 on the 1906/7 etc. It will then check that:

1. The resultant absolute value for the PUC datum is a permitted value on the particular machine (e.g. it cannot exceed 16K-256 on the 1902/3, 8K-128 on the 1901).

2. The relative datum and limit do not lie outside the limit of the TP. In some Executives the limit may be set to the size of store or the TP limit and the implied requested limit checked against this.

If any of these checks fail the instruction is illegal, otherwise the Executive will store in P and P+1 the actual relative datum and store size given.

Unless it is necessary to change the PUC parameters the TP normally needs to obey this extracode only once. If however the present PUC has members this extracode should *always* be obeyed prior to the loading of a new PUC.

In addition, if X = 1, Executive will make the current member 0 and set initial member and monitor suspensions on all other members, (i.e. state SL). All operator suspensions are cleared and all members disestablished and the slot is thus brought into a standard state. To this end the TP should always obey 150 orders on all the PUC peripherals and if necessary take them out of Direct Response Mode also *prior to* obeying this extracode.

Note, however, that it is not, in general, possible for a TP safely to switch a peripheral which the PUC is using from DRM to SM without cooperation from the PUC.

The effect of issuing this extracode from other than member 0 is undefined if the accumulator swapping level is set to the TP datum.

3.4 Extracode to Start a PUC Member

164 7 N(M)

Executive will activate the current member of the PUC and remove monitoring suspensions from all members. The order number, monitor modes etc., are taken from PUC datum + 8

and datum + 9, as follows:

datum+8, B0 = 1 V is set;

datum+9, B2 = 1 Extended Branch Mode is set;

datum+9, B1, B3, B5–B8 These bits are reserved for Executive or hardware use;

datum+9, B0, B9–B23 These bits are reserved for reply information;

datum+9, B4 = 1 22–bit address mode is set.

In addition, if B2 and B4 of datum+9 are both zero, the following fields have the given significance:

datum+8, B9–B23 order number relative to the PUC datum;

datum+8, B1–B8 these bits are reserved for Executive or hardware use.

Otherwise the following fields have the given significance:

datum+8, B2–B23 order number relative to the PUC datum;

datum+8, B1 reserved for Executive or hardware use.

3.4.1

Bits in datum+9 referring to facilities (22AM or EBM) which do not exist on particular machines will be ignored when the order is obeyed, and will be undefined when control is returned to the TP.

All bits reserved for Executive or hardware use should be cleared before first starting a new PUC and preserved during the life of that PUC.

Following this order, the PUC will run until an event occurs. This can be one of the following:

1. A 151, 152, 154, 155, 156, 157 (open, close, relabel, extend, contract, delete or rename and possibly some other modes, but *not* transfer modes or backspace or rewind), 160, 161, 163 (referring to a member not established at PUC datum, 164 (X = 7), 165 or 166 order in the PUC. A 156 order to an assigned peripheral with no additive mode set is not necessarily monitored, but mode #1000 is always monitored.
2. An illegal order in the PUC.
3. An input message from the console typewriter, the TP having previously called for one (see §5).
4. Periodic interrupt (only on machines with suitable hardware).
5. Event on a direct response mode peripheral (see §4.2). See §6 for the way in which events are notified.
6. In some executives, end of transfer which has been initiated on EDS by the TP.

In E4BM and E6BM, following a 164 (X = 7, N(M) = 0) order, monitor suspensions are set on all members of the TP.

3.4.2 N(M) = 1

As for N(M) = 0 but the next order (other than 117 or 123 orders) obeyed by the PUC must be one of the extracodes given in §3.4.1. (But note that extracode variants peculiar to TPs are not permitted and 163 orders are not allowed if the member is already established at the PUC datum). If the extracode is illegal then this order (the 164 order) is regarded to be illegal. (The TP can monitor this — see §7). (The stored link may point to the order after the 164). Software detected errors (e.g. checksum during 154) may cause illegal action, or else will cause the event to be monitored on the operator's console, in which case the TP probably cannot be restarted.

Monitoring will occur immediately after the extracode has been obeyed. (See §6.2.1). Monitor suspension on other members will not be released until the next 164 (X = 7, N(M) = 0) order.

Note that if the TP wishes to single–shot an extracode which has just caused an event in the PUC it can obey this order without touching PUC datum+8. If it is desired to skip the PUC extracode then this is achieved by copying X6 into PUC datum+8. (See §6.2.1).

There are two extracodes which have a special effect when single–shotted. The 155 order will not dump the request slip — this must be done by the TP; it will, however, dump the Supplementary Request Slip where necessary. On completion of a 154 order, Executive will not suspend the slot even if there is a suspend type entry block; the TP will not know what type of entry block was used. However, PUC+8 will contain the address from the entry block (type 2 or 3) or the address of the order after the 154 if entry block type 4. The effect of single shotting 165/4 or 166 or any non–applicable order is undefined.

3.4.3 N(M) = 2

Member suspension is set on the current member. It is disestablished and left in state SM. Monitoring suspension on other members is released. This extracode will normally be used to simulate the effect of a SUSTY or SUSWT order on a particular member of the PUC.

3.5 Extracode to read the Program Timer

165 X 10

Executive will store in accumulators X and X + 1 the approximate[1] number of microseconds used by the slot since the TP was loaded. On machines without a program timer, X will be set to zero, X + 1 being left untouched. The timer will only be cleared after the slot is deleted.

3.6 Special Effect of 165 (X = 4)

When a TP obeys a 165 (X = 4) order to change its core size, the difference between between the PUC limit and the TP limit after the order will bet the same as it was before. If the new value of the implied PUC limit is less than or equal to the PUC datum the effect is undefined.

[1]The actual number stored will vary not only from machine to machine but also depending what other programs have been running, what peripheral transfers have taken place and so on, since the order was last obeyed. It is intended only as a rough guide to the length (of the runs).

4 Peripheral Handling

There are two requirements. First, the TP must be able to assign peripherals to the PUC so that the peripherals available to the PUC are a sub–set of those available to the TP. The second requirement, mainly for the benefit of GEORGE 1 and GEORGE 2 is that the TP should be able to 'listen for' engage buttons on its slow peripherals. This is to enable these operating systems to allow the operator to load job descriptions and documents on the most convenient peripheral rather than one explicitly chosen by the operating system itself.

4.1

The first requirement is met by the introduction of a new mode in the 156 order, allowing the TP to change the unit number of an assigned peripheral or a file. (it is legal for the old number to be the same as the new number). The mode is additive #2000 and the new unit number is given in word 10. In addition, the TP will be permitted to use unit numbers greater than 31 for both peripherals and files. These two facilities together mean that the TP can initially reserve its peripherals and files with unit numbers between 32 and 47, (so that they are not initially available to the PUC) and then, if it wishes, assign them to the PUC by making the unit number that requested by the PUC, (or, in fact, any other unit number which is permitted in a TP). No message is output by Executive if the unit number is greater than 31 when assigning or releasing a peripheral. It is illegal for the PUC to attempt to assign a peripheral or a file with a unit number other than in the ranges 0–15 and 32–47 or an unassigned peripheral is undefined.

4.2

The second required facility is provided by an enhancement of the direct response mode of peripheral operation. The DRM facility is described in PSS 008. Briefly, the facility is as follows:

1. The normal state of a peripheral on assignment is 'suspend mode' (SM).
2. A peripheral may be switched into DRM by using additive mode #40000 in the 156 order, and returned to SM by additive mode #20000.
3. While running in DRM attempt to do transfers on the peripheral will rarely result in suspension. Executive will usually reply as follows:

   bit 0 = 0 transfer accepted and action completed.

   bit 0 = 1, bit 5 = 1 transfer rejected — inoperable.

   bit 0 = 1, bit 4 = 1, bit 5 = 0 transfer rejected — busy.

   bit 0 = 1, bit 4 = 0, bit 5 = 0 transfer accepted and action in progress.

4. Any event on a DRM peripheral will cause the program, or rather the member to be started if it had previously suspended itself using a 164 (X = 2) order. If the member is running, the event will be remembered in as much as the next 164 (X = 2) will not suspend the member.
5. Events on DRM peripherals, with unit numbers ≥ 32, whether they are operating in this mode for the benefit of the TP or the PUC, will always be notified to the TP. (See §6).

If the event is of no interest to the TP it can the use the 164 (X = 7) to start the PUC. This implies that Executive should maintain separate memory–status P for each level; or the equivalent.

6. The 164 (X = 7, N(M) = 0) order is equivalent to a 164 (X = 2) order in respect of awaiting peripheral events (see §3.4.1) provided the TP takes proper care of word 31 (see §6).

4.3 Assigning Magnetic Tapes by Serial Number

If in a magnetic tape open mode PERI order issued by the TP the second word of the control area is positive and non–zero, then only a tape with the serial number given in this word will be allocated (i.e. both name and serial number will be checked). This allows tapes with the same name to be distinguished with certainty by referring to the serial number. For non–TPs the second word remains 'non–significant' as stated in PAS 5/001, (including the case of a single–shotted PUC order). For mode #500 also, the second word is 'non–significant', and if non–zero has an undefined effect.

4.4 Concerning Flag–setting Peripherals

4.4.1

PUCs may not operate multiplexors.

4.4.2

On single–channel flag–setting peripherals, 157 orders subsequent to the mode #40000 157 order are only legal at the level (TP or PUC) from which that order was issued.

5 The MS Directive

5.1 Introduction

The MS directive allows for the input of messages from the console typewriter to a TP and also in certain other environments to programs not under the control of a TP.

It will be made available on all executives providing the Range Compatible Trusted facilities described here, and, in addition, on all Executives incorporating the multiplexor package.

The MS directive will not be available to a program using Double Slot Trusted facilities, but may be available to a non–trusted program running under an Executive providing Double Slot Trusted facilities or an Executive providing no trusted facilities.

5.2 Definition of the MS Directive

5.2.1 The Inverse Display Instruction (160 (X = 7))

When it is ready to receive a message, the program will obey a 160 (X = 7) order, N(M) has the usual meaning for 160 orders specifying a count and a starting address. The maximum value of the count is 38. Executive will remember the request and the address, and the program will be allowed to continue.

5.2.2 Input of Message by the Operator

When the operator wishes to send a message to the program the following must be typed:

MS #NAME =*message*

The underlined parts, except #NAME on a single programming Executive, are essential, but Executive will ignore additional alphabetic, comma or stop characters inserted in between these. A member number may not appear in association with the #NAME. The message may contain any of the 1900 6–bit 64 character codes except *23 (#) or *74–*77.

5.2.3 Executive Action on Receipt of an MS Directive

On receipt of an MS directive, Executive will check that it has a 160 (X = 7) order outstanding and that the format of the message is correct. It will generate an accept character (*76) and place this at the end of the message. It will then check that the length of the message (following and excluding the "=" and including the accept character) is not greater than the count specified in the previous 160 (X = 7) order.

If the message is not acceptable or is greater than the count specified, Executive will signal an error to the operator and the input will be ignored.

If the message is acceptable, Executive will reply affirmatively and the data will be transferred to the area specified in the preceding 160 (X = 7) order. The first character stored is that following the "=". If the message is shorter than the count specified the characters after the first *76 will be indeterminate. Cancelled messages are ignored.

When the data transfer is complete, Executive will notify the event by setting bit 6 of word 31. It will also release the program member that issued the 160 (X = 7) instruction if it is currently suspended in the state SMP (i.e. suspended on 164 (X = 2 or X = 7)) and set the memory state P if the member is not suspended. In a multi–member PUC environment, member 0 will also be de–suspended.

5.2.4 Effect of issuing a Second 160 (X = 7) Order

The effect of issuing a second 160 (X = 7) order prior to the receipt of the message corresponding to the first, indicated by bit 6 having been set in word 31, is indeterminate.

5.2.5 Programming Considerations for non–TPs

Since word 31 is used by trusted programs to indicate other conditions than the occurrence of MS directives, it is essential that any software that might be used in conjunction with trusted facilities should only change bit 6 of word 31.

5.3

Some issued Executives providing this facility do not strictly conform to this specification. They will eventually be brought into line.

6 Notification of Events

The various types of events are listed in §3.4.1. They fall into two categories — events in the PUC and external events. What happens when an event occurs depends on what category it belongs to, whether the PUC was running at the time and whether either the PUC or the TP is multi–member. In all cases, Executive will store some information for the TP to enable it to find out what the event was.

If member 0 of the TP is running when the event occurs, Executive will simply store the information (possibly only in TP word 31) and remember there was an event so that the next time the TP obeys a 164 (X = 7) order, it will not be suspended. If an event occurs which is not associated with a specific member, then at least member 0 will be activated. In all other cases (neglecting operator suspensions) the PUC will be running and Executive will have to restore control to the TP.

6.1 Restoring Control to the TP

The datum will be restored to that required for running the TP at the order following its last obeyed 164 (X = 7) order. The detailed action depends on the multi–member situation.

6.1.1 Multi–Member PUC

In the case of a multi–member PUC, monitor suspensions will be set on all members other than the current one, i.e. the one active (in the state NS and the highest priority) when the event occurs. The order number of the TP will be that of the order following its last 164 (X = 7) order, the control path of the current PUC member will be used. This is not to say that the TP must, or even can, be thought of as multi–member — it has only one order number and one set of accumulators. No accumulator swapping takes place when an event occurs and thus the information in PUC datum + 8 and datum + 9 refers to the current member and has the same significance as when the 164 (X = 7) order was obeyed (except, in fact, for the order number following a PUC event — see below).

6.1.2 Single–Member PUC

In this case, if the TP is also single–member, then control simply reverts to that member at the order following the 164 (X = 7). This is also the case if the TP is multi–member and the last member which was running at the TP datum was member 0. If it was some other member, then accumulator swapping takes place at the TP datum before member 0 is entered. The effect, so far as the TP is concerned, is that it enjoys the usual sub–programming facilities but its member 0 may be started for additional reasons.

In the case of events on flag–setting or Direct Response Mode devices (or any other event which in some environments would be notified to only one member) the peripheral event bit will be set in word 31 as usual and the datum for member 0 may or may not be reset. Executive will take its normal action, releasing member suspension on one or more members. This means that it is advisable for member 0 to have the lower priority.

After the event has been recognised, the member concerned should clear the flag in word 31 (bit 4, see §6.2.2) deal with the event and obey its 164 (X = 2) order as usual (and be returned to state SMP). A correct sequence of operations for any member of the TP in dealing with word 31 is given in the note following §6.2.2. If the flag is left set or if word 31 is non–zero

for any other reason, then member 0 will certainly be restarted at the TP datum, otherwise it may be restarted at either datum depending on the actual implementation in the particular Executive. As always, the rule is that each member must be prepared to drop through its 164 (X = 2 or 7) order even when there is nothing for it to do.

N.B. the 163 order has its usual meaning in this context. When referring to member 0, the decision whether to activate it at TP or PUC datum is decided, as always, on whether word 31 is non–zero or zero.

6.2 Information Stored by Executive

Word 31 is used as a flag word to indicate the type of event. Also, in the case of a multi–member PUC, bits 16 and 17 will contain the member number. Other information depends on the type of event.

6.2.1 Events in the PUC

PUC datum + 8 and datum + 9 have the same meaning as given in §3.4.1 except that the order number stored following an event in the PUC is the address of the order number which caused the event, or, if the order was accessed by an OBEY (023) order(s), the address is that of the first OBEY order, or, in machines which have a pre–modification facility, the address is that of the first order of the string which caused the event.

Information is also stored in the TP's words 4, 5, 6 and 31. What is stored depends on the event as shown in the following table:

Event	Word 31		Additional Information
	Bits 0 to 3	Other Fields	
Extracode (where Exec has had the function available) (including some illegals, in particular 150 and 157 on unassigned peripherals)	1100	bits 18 to 23 = l.s.6 bits of the function	word 4 = X of entry order; word 5 = N(M) of entry order (including premodification but relative to PUC datum); word 6 = address of next order which would have been obeyed had the event not occurred (Datum + 8 format, see §3.4.2)
Other illegals	1010	bits 18 to 23 may contain an error character	none (words 4,5 and 6 indeterminate)
Completion on a single–shot (see §3.4.2)	Undefined	Not written to	none

6.2.2 External Events

If the PUC is running, control must be restored to the TP and the contents of word 31, bits 4 to 6 set as given below. For a multi–member PUC, bits 16 and 17 of word 31 always contain the number of the member in use. Words 4, 5 and 6 are unchanged.

Event	Bits 4 to 6 of Word 31
Event on a peripheral (see §4.2)	`1yy`
Spasmodic Interrupt (only available with processors having a real time clock)	`y1y`
Console typewriter message (MS)	`yy1`
All other occasions on which the 164 (X = 7) drops through	`yyy`

Where "`y`" means not written to by Executive as far as the particular event is concerned.

6.3 Note

If the TP is running when one of these events occurs then Executive will remember that there was an event and *or* the appropriate bits into word 31. Then when the TP next obeys the 164 (X = 2 or 7) order, it will be allowed to continue.

The condition for this may be that word 31 is non–zero i.e. Executive may itself merely test word 31 to see whether the 164 (X = 2 or 7) should drop through. The TP should therefore be strict in the way it uses word 31. One way which is always guaranteed to work is for the TP to copy word 31 into a working accumulator and then to non–equivalent word 31 with this accumulator. After the events now flagged in the accumulator have been dealt with, the TP can return to the 164 (X = 7) order and then repeat the process if and when it continues.

Executive does not guarantee to notify any event immediately or within a given period of time following its occurrence. Furthermore, events which occur while the PUC is suspended awaiting operator action following a peripheral failure may be ignored altogether, otherwise events will always be notified, except perhaps for spasmodic interrupts.

7 Self Monitoring of Illegal Orders in the TP

The action taken by Executive when the TP goes illegal depends on the content of word 6 (i.e. the 7^{th} word) of the request slip. This is either zero — meaning that self–monitoring is switched off, take normal action — or it contains an address P. In this case, when an illegality occurs, Executive will set the order number of the failing member to P (with V clear) and store information in this member's words 14 and 15 as follows:

Word 6 must contain an address only. The effect of bits set outside the TP's limit is undefined.

Word 14 B0 = 1 if V was set.

B1–B8 indeterminate (compact mode)

B9–B23 (compact mode)

B2–B23 (EBM or 22AM) the address of an order in the illegal string, always the first in a group of group 15 orders.

word 15 B22–B23: member number. All other bits are undefined.

After storing this information the TP is allowed to continue as if the member concerned had merely branched to P (with V clear).

Postscript by John Hunter

This text has been transcribed from a Roneo-ed text supplied by ICL to the Defence Automatic Data Processing Training Centre (DADPTC) at Blandford in late 1968. The copy used had many minor typing errors and some notational inconsistencies. I have attempted to "correct" these shortcomings without changing the wording or sense of the original. I would like to apologise in advance for any errors inadvertently inserted by me.

The original document was obviously drawn up for the benefit of the GEORGE 1 and GEORGE 2 programmers. Neither GEORGE 1 nor GEORGE 2 implemented any form of "filestore". The peripheral handling section (§4) assumes that that the PUC is allowed direct access to slow peripherals. The more advanced TPs written using these guidelines essentially ignored §4 by never allowing a PUC to ALLOT a slow peripheral. Input requests were satisfied by the TP by supplying text obtained from a nominated subfile of a filestore. Output requests were satisfied by spooling output to a cache file which could be printed (or punched) at the operators' convenience.

Magnetic tapes were a special case. There were times where tapes were simulated by subfiles of the filestore, There were other times where this was inappropriate and PUCs would be permitted access to real magnetic tapes.

This approach gave rise to two important advantages:

1. *the number of operators in the computer room was halved;*
2. *in a peripheral bound environment, throughput was, in some cases, tripled.*

This paper has a detailed discussion on the effects of multi-member TPs and PUCs. From experience, ignoring flag–setting peripherals, it was straight forward for a single–member TP to control the common peripherals using DRM. This is not necessarily true for multiplexors. However, it was found that register swapping (by Executive) could absorb an appreciable amount of processor power; consequently, the use of multi-member TPs was avoided where possible.