A NOTE ON STAR-FREE EVENTS

By

Albert R. Meyer

Carnegie-Mellon University
Pittsburgh, Pennsylvania

February, 1968

This work was supported by the Advanced Research Projects Agency of the Office of the Secretary of Defense (SD-146) and is monitored by the Air Force Office of Scientific Research. Distribution of this document is unlimited.

ABSTRACT

A short proof of the equivalence of star-free and group-free regular events Is possible if one is willing to appeal to the Krohn-Rhodes machine decomposition theorem.

1. INTRODUCTION

The star-free events are the family of regular events expressible in the extended language of regular expressions (using Intersection and complementation, as well as union and concatenation of events) without the use of the Kleene star (closure) operator. The equivalence of the star-free and group-free events was first proved by Schutzenberger [1966]. Papert and McNaughton [1966] show that the star-free events are precisely the events definable in McNaughton's L-language, and are thereby able to establish the above equivalence without extensive use of the properties of finite semigroups. However, if one is willing to appeal to the machine decomposition theorem of Krohn and Rhodes,the equivalence of star-free, group-free, and also noncounting regular events can be proved more simply. We present such a proof in this note,

2. PRELIMINARIES

We assume the reader is already familiar with regular events and finite automata. Our notation follows Yoeli [1965] and Ginzburg [1968]. In particular, if f and g are functions from a set S into itself, arguments are written on the left (so that sf = f(s)), and the composition f *o* $_{g}$ $_{me}$ans that f is applied first (so that s(f ° g) = (sf)g).

A <u>semlautomaton</u> is a triple A = <Q^{A}, £^{A}, M^> with Q^{A} a finite set (of <u>states</u>), E^{A} a finite set (of <u>inputs</u>), and M^{A} a set of functions M^{A}_{a}: Q^{A}->Q^{A}indexed by $_{CT}$ £ E^{A}. The mapping M^{*}_{a} is abbreviated "$_{a}^{A}$". The element qo-^{A} £ Q^{A} is the <u>next state</u> of q £ Q^{A} under input $_{CT}$ £ 2*. For x € (F^{A})* the mapping x^{A}; Q^{A}-> Q^{A}is defined inductively: A^{A} is the identity map on Q^**wliGre** A is the null word in (£^)*, and if x = y$_{CT}$ for y **e** (E^)" and $_{CT}$ £ £^, then x^ is y^ o $_{CT}$^. Hence, (xy)^= x^ ° y^ for all x, y € (£^)*. For x £ and integers k £ 0, x^{k} is the concatenation of x with itself k times; x° = A by convention. Clearly, (x^{k})^ = (x^)^{k} = the composition of x^ with itself k times. The (necessarily finite) set of distinct mappings x for x € (E)* form a semigroup G^ under composition. G^ Is called the <u>semigroup</u> of A.

Let A and B be semiautomata. B is a <u>subsemiautomaton</u> of A providing S^{B} c E^, Q^{B} <= Q^ and the mapping $_{CT}$^{B} is the restriction of $_{CT}$^ to **Q**^{B} for each a € £^{B}. B is a <u>homomorphic image</u> of A providing that S^ = E^{B} and there is an onto mapping T(: **Q**^ -» Q^{B} such that T[° = cr^° T| for each & £ £^, The mapping **H** i^{s} called a <u>homomorphism</u> of A onto B. A covers B, in symbols "A ^ B" if and only if B is a homomorphic image of a subsemiautomaton of A.

An <u>automaton</u> Is a quintuple A^{A}— *CQ^{A}, ^{A}, s^{A}, F^{A}, ^* wnere A = <Q^, £^, *H^>* is a semiautomaton, called the <u>semiautomaton</u> of &$_{ps}$^ is an element of Q^ called the <u>start state</u>, and F^ is a subset of Q^ called the <u>final states</u>. The <u>event accepted</u> by_ **A** is {x € **(£^)* j** s^x^ 6 F^]. This definition of automaton is merely a notational variant of the usual finite state acceptor (cf. Rabin and Scott [1959]), and the events accepted by such automata are precisely the regular events.

3. STAR-FREE AND NONCOUNTING EVENTS

The star-free events are defined inductively as follows:

<u>Definition 1</u>. Let J be a finite set (of inputs). The singleton *{o}* is a star-free event over E. If U, V c j * are star-free events over £,

then U U V, U (the complement of 0 relative to £*), and UV (the concatenation of U and V) are star-free events over £. An event is star-free over E only by implication from the preceding clauses.

By DeMorgan's law, U fl V = 0 U V and so star-free events are also closed under intersection. Since the regular events over £ include the singletons and are closed under union, relative complementation, and concatenation, it follows that every star-free event is regular.

Definition 2. (Papert-McNaughton) A regular event U c j * is a noncounting regular event over £ if and only if there is an integer k_y S 0 such that for all x, y, z € £*

$$xy^{k\,U}z\ e^{r}\ U'' \text{ » } x\ y^{k\,u+1}z\ {}^{r}\text{£}^{TT}U.$$

Intuitively, an automaton accepting a noncounting event U need never count (even modulo any integer greater than one) the number of consecutive occurrences of any word y once consecutive y's have occurred in an input word.

Lemma 1. (Paper-McNaughton) Every star-free event is a noncounting regular event.

Proof. The singleton $[_{Q})$ is trivially a noncounting regular event for every $_{a}$ £ £ (choose =2), so it is sufficient to show that if U and V are noncounting regular events over £, then so are U U V, U, and UV.

Let k = maxfky.kyj. Then for any x, y, z € £*, $xy^{k}z$ £ U U V

$$\text{»}\ xy^{k_U}(y^{k-k_U}z)\ \text{€}\ 0 \text{ or } xy^{k_V}(y^{k-k_V}z)\ \text{€}\ V \text{ « } xy^{k_U+1}\ (y^{k-k_U}{}_{z})\ 6\ U \text{ or}$$

$xy^{k.,-H}$ $(y^{k-k}$ z) £ V » xy^{-} z 6 U U V. Thus, U U V is a noncounting regular event with k = maxtky, k_y\.

Similarly, $xy^{k.j}$ z € U «* xy^{kir} z j£ U « xy^{k_n+1} z j£ U » xy^{ktj+1} z € U, so that 0 is a noncounting regular event with *k^* = *k^.*

Finally, let k = 2-maxfk ,k \+ 1 and suppose $xy^k z$ £ U V. Then •$_y$\ U' V

$xy^k z$ = uv for some u £ U, v € V, and it must be the case that either u = $xy^{k//2}w$ for some w £ £*, or that v = v ' y ^ z for some w'€ £*. In the first case, u = $xy^{k\wedge 2}w$ = $xy^{ku}(y^{k/2-ku}$ w) € U implies that $xy^{ku+1}(y^{k/2"k}$U w) » $xy^{k/2+1}$ w € U since U is noncounting. In the second case, v = $w'y^{k/2}z$ £ V similarly implies that $w'y^{\wedge 2+1}z$ € V. Hence, in either case $xy^{k+1}z$ € UV. Conversely, if $xy^{k+1}z$ € UV the argument can clearly be reversed to show that $xy^k z$ g UV. Thus, UV is a noncounting regular event with k^ = 2•maxfk ,k_y} + 1. Q.E.D.

^{k}U

If U is a noncounting regular event over £ and a €2, then *a* € U implies that U contains all words in o* of length at least k^. Therefore, either U fl cr* or U fl a* is a finite event. The regular event (era)* is neither finite nor has finite complement, which proves:

<u>Corollary 1</u>. The noncounting (and hence the star-free) regular events are a proper subfamily of the regular events.

4. GROUP-FREE EVENTS

Associated with any event U c S * is a congruence relations (mod U), on 2* defined for w, y £ 2* by:

$$w = y \text{ (mod U) « (Vx, z £ 2*)[x w z £ U » x y z € U].}$$

Noncounting regular events are thus those regular events U such that $y^{k_{TT}}_{U}$ s $y^{k_{„}+l}_{U}$ (mod U) for all y € S*.

The relation between this congruence and automata is an immediate consequence of the familiar theorems of Nerode and Myhill (cf. Rabin and Scott [1959]): if U is a regular event, then there is an automaton ft accepting U (viz., the reduced automaton accepting U) such that $x = y$ (mod U) » $x^{A} = y^{A}$.

Definition 3. A subgroup of a semigroup S is a subsemlgroup of S whose elements form an abstract group under multiplication in S. A semigroup is group-free if and only if all its subgroups are isomorphic to the trivial group with one element. A semiautomaton is group-free if and only if the semigroup of the semiautomaton is group-free. A regular set U is group-free if and only if there is an automaton ft accepting U such that the semiautomaton A of ft is group-free.

Lemma 2. Let S be a semigroup. If there is an integer k S 0 such that s = s for all s € S, then S is group-tree.

Proof. Let G be a subgroup of S, and let g be an element of C. Then $g^{k} = g^{k+1}$ implies e ^ g V V ^ V ^{1}) ^ g where g"' is the inverse of g in G and e is the Identity of G. Hence, G = [e) is the trivial group. Q.E.D.

Corollary 2. Every noncounting regular event is a group-free regular event.

Proof. If U is a noncounting regular event, then $y^{ky}_{U} = y^{kn+1}$ (mod U) implies that $(y_{u}^{kfi})^{A} = (y^{kU+1})^{A}$ in the reduced automaton $\overset{A}{A}$ accepting U. Hence, $(y^{A})^{klJ}$ « $(y^{A})^{ku+1}$ for every element y^{A} € G^{A}, and G^{A} is group-free by lemma 2. Q.E.D.

5. DECOMPOSITION INTO RESETS

The machine decomposition theorem of Krohn and Rhodes supplies the key step in the proof that group-free events are star-free.

Definition 4. Let A and B be semiautomata and w: Q^{A} x $£^{A}$ - $_{s}{}^{B}$. The

0

cascade product A to B of A and B with mapping cu is the semiautomaton C with $\overset{c}{Q} = \overset{a}{Q}$ x ^{b}Q , $\overset{C}{£} = {}^{a}£$ $\overset{c}{a}$nd o for $_{CT}$ € $\overset{C}{£}$ defined for all s ^{A}£ Q^{A}, s^{B} £ Q^{B} by:

$$_{<\,s}\backslash \quad S{*}> {}_{CT}{}^{C} = < S^{A}{}_{0}{}^{A}, \quad s \;W\; u \;O\; V$$

A cascade product of three or more automata is defined by association to the left, e.g., a cascade product of semiautomata A, B, and C is any semiautomaton (A *^^3)^2^* for any mappings u)^ and with appropriate domain and range.

Definition 5. A semiautomaton R is a reset providing Q^{R}= f1,2|, and $£^{R}$ is the union of three mutually exclusive sets $£^{R}$, $£^{R}$, $£^{R}$ such that: cr € $£^{R}$ => range ($_{CT}{}^{R}$) = fl 1; cr € range (cr^{R}) = T 2>; and $_{a}$ £ $£^{R}$ -> CT^{R} = the identity on Q^{R}.

The following weak form of the decomposition theorem is sufficient for our purposes (for a constructive proof of the general theorem see Ginzburg [1968}):

Theorem. (Krohn-Rhodes) Every semiautomaton A is covered by a cascade product of semiautomata $A_{]}$, such that for 1 <; i £ $_{n}$, $A_{\pm}$

^ I
IS **3** reset o**it** **6**1 s£ G I. s & oorx t**IT** 1.v**!L3**X tiornoinorptii.c lirts^s of s GROUP OF G .

SINCE THE TRIVIAL GROUP HAS ONLY ITSELF AS A HOMOMORPHIC IMAGE, THE FOLLOWING LEMMA IS IMMEDIATE:

LEMMA 3. EVERY GROUP-FREE SEMIAUTOMATON IS COVERED BY A CASCADE PRODUCT OF RESETS.

COROLLARY 3. EVERY GROUP-FREE REGULAR EVENT IS ACCEPTED BY AN AUTOMATON WHOSE SEMIAUTOMATON IS A CASCADE PRODUCT OF RESETS.

PROOF. LET ft, WITH GROUP-FREE SEMIAUTOMATON A, BE AN AUTOMATON ACCEPTING A GROUP-FREE REGULAR EVENT U. BY LEMMA 3 AND THE DEFINITION OF COVERING, A IS THE IMAGE UNDER A HOMOMORPHISM *t[* OF A SUBSEMIAUTOMATON OF A CASCADE PRODUCT C OF RESETS. THERE IS NO LOSS OF GENERALITY IN ASSUMING THAT $£^{A} = £^{c}$, SINCE THE SUBSEMIAUTOMATON OF C OBTAINED BY

...... T A , . . . **X T** . , „

RESTRICTING E TO £ IS ALSO A CASCADE PRODUCT OF RESETS WHICH COVERS A. CHOOSE ANY S^{c} € Q^{c} SUCH THAT S^{c}T| = S^{A} (THE START STATE OF ft) AND DEFINE F° = {Q € Q^{c} | Q H € F^{A}J. THEN FOR ANY X £ $(£^{A})*$, X € UO

A A , C A A C C , A C C „ C
SX £' F O S TL X £ F « S X T| £ F O S X € F" . HENCE, THE AUTOMATON
A C ^ C
C WITH SEMIAUTOMATON C, START STATE S , AND FINAL STATES "F 'IS THE REQUIRED AUTOMATON ACCEPTING U. Q.E.D.

6. THE MAIN THEOREM.

THE BEHAVIOR OF CASCADES OF RESETS CAN BE DESCRIBED IN TERMS OF STAR-FREE EVENTS USING

Deftnitlon 6. For a semiautomaton A and states p, q € Q^A, the set A of p-g-inputs is fx £ $(S^A)^*$ | px^A = q).

Lemma 4. Let C = B u> R with B a semiautomaton, R a reset, and UJ: Q^B x S^B ^2^R. If B_{pq} is a star-free event (over $£^B$) for all p, q € Q^B, then C ' Is a star-free event (over $£^C$ = $£^B$) for all a, b £ Q^C.

Proof. Write "S" for the (equal) sets $£^B$ and By the definition of cascade product, the first component of <p, 1> y Is simply py for

B *

any p € Q , y € S • Since R Is a reset, in order for the second component of ^Cp, 1> y to be 2, R must receive an input <r, cr^ € £2 for some r € Q^B, <j € S-

Suppose x £ $C_{<p]><(i 2>}$. Then px^B = q and so x £ B_{pq}, but also x must equal y_a z for some y, z £ £*, $_{CT}$ € S such that: py^B= r for some r £ Q^B and <r, $_{CT}$> $^{1)0}$ € E*. c h o o s e t n e shortest z such that x = y $_{CT}$ z for y and $_{CT}$ satisfying the preceding conditions. Then no prefix of z causes R to receive an input <s, 5> ^ € E_1^R (where s £ Q^B, 5"£ £), ^{J} i.e.,

$_B$ > $_B$ '$_6$ r*
* r_a ,s

Conversely, if py^B= r for <r, o> a> £ $£^{R\ and\ z}$ j£ $^B{}_{r(J}$B $_s$ 6 for any <s,6> «> € E^R, then y_az € C providing y_{CT}z £ B ' Altogether one li 3 5 *

^{C}<$_{Pj}$1$_{xq|2}$>$^{B\ B}{}_{pq}$ H [U B_{pr} or (U B^$_{BjS}$ 6 S*>]

thelefthand union being over all r £ Q^B, $_Q$ 6 E,such that <r,$_a$>u £ ^' and the righthand union being over all s £ Q^B, 5 £ *Z* such that <s,8>u € *Z^.*

The unions in the expression for C „ _ are finite, and v* is
<pj><q,2>
a star-free event & = £ and ,£ = f$_a$}n fat), so that C ^ ^ is a
<p, lXQ,Z>
star-free event. The set of x £ C . is precisely the set of
<p,lXq,l> * ^{J}

x such that $px^B = q$ and x t^L ^{c}<p,lXq,1>
B fl C ^{andsoc}^ $_K$i ^{saisoa} star-free event.

Since the argument is symmetric in states 1 and 2 of Q , $*^s$ a star-free event for all a, b £ Q^Q . Q.E.D.

<u>Lemma 5</u>. If C is a cascade product of resets, then C_Q * is a star-free event for all a,b £ Q .

<u>Proof</u>, Let R be a reset and B a semiautomaton such that $Q^B = \{p\}$ and $E^{BR} = £$. For cr € H^B , define u>: Q^B X Z^B ->$£^R$ by the condition <p,o>u) = <j. In this trivial case of cascade product, R. . = (B <B R) for all i, j € Q^R • Since B^B = $(S^B)^*$ is star-free, lemma 4 implies that R,, is star-free.

The rest of the proof follows immediately by lemma 4 and induction on the number of resets in C. Q.E.D.

C<u>orollary 4</u>, Every event accepted by an automaton *, whose semiautomaton A is a cascade product of resets, is a star-free event.

<u>Proof</u>. Let a £ Q^A be the^A start ^{A}state of A, and F the final states. The event accepted by * is $\bigcup_{b£F^A} A$, which is a star-free event since the union is finite and A* is star-free by lemma 5. Q.E.D.

This completes the proof of the following

<u>Theorem</u>. (Schutzenberger, Papert-McNaughton) The following are equivalent for events U C £ :

1) U is a star-free event,

2) U Is a noncounting regular event.

3) U is a group-free event.

4) U is accepted by a cascade product of resets.

References

1. Ginzburg, A., Algebraic Theory of Automata, to appear (1968).

2. Papert, S. and R. McNaughton, "On topological events", Theory of Automata, University of Michigan Engineering Summer Conferences (1966).

3. Rabin, M. 0. and D. Scott, "Finite automata and their decision problems", IBM J. Res. and Dev., 3,2 (April, 1959), 114-125.

4. Schutzenberger, M. P., "On a family of sets related to McNaughton's L-language", Automata Theory, Caianiello, E. R. (ed.), Academic Press, N. Y, Xl966)7~320^324.

5. Yoeli, M., "Generalized Cascade Decompositions of Automata", JACM, *12,* 3 (July, 1965), 411-423.

SCOPE USER MANUAL

By

Alan H. Bond

Carnegie-Mellon University
Pittsburgh, Pennsylvania
February 6, 1968

This work was supported by the Advanced Research Projects Agency of the Office of the Secretary of Defense (SD-146) and Is monitored by the Air Force Office of Scientific Research.

(0

CONTENTS

FIGURES

1. PREFACE

THIS MANUAL DESCRIBES HOW TO USE THE SCOPES. IT IS CONCERNED MAINLY WITH SOFTWARE, AS THE HARDWARE IS TREATED DEFINITIVELY IN THE CARNEGIE TECH, DOCUMENTI »A VISUAL DISPLAY SYSTEM SUITABLE FOR TIMESHARED USE' BY QUATSE, JESSE T., LATEST VERSION DECEMBER 1966, OBTAINED FROM THE COMPUTATION CENTER DOCUMENTATION STAFF, IN ORDER TO USE THE SCOPES, IT IS SUFFICIENT TO READ THE QUATSE DOCUMENT AND THE SCOPE USERS MANUAL. THE QUATSE MANUAL SHOULD BE READ FIRST, A BRIEF DESCRIPTION OF HARDWARE CONCEPTS AND TERMS IS GIVEN IN SECTION 3 OF THIS MANUAL^

THE SCOPES ARE SITUATED IN ROOM PH18A, COMPUTATION CENTER, TELEPHONE EXTENSION 27. THEY ARE" ON WHEN TELETYPES ARE ON, USUALLY 10100 AM TO MIDNIGHT AND HAVE NORMAL TELETYPE TURN-ROUND TIME, EXCEPT THAT PROGRAMS SUBMITTED FROM SCOPES RUN AT THE BEGINNING OF THE WAIT TIME, 1,6.* IMMEDIATELY, TO ALLOW THE USER TO BE PRESENT AT RUN TIME AND TO INTERACT WITH HIS PROGRAM, AT PRESENT ONLY 3 MINUTE PROGRAMS CAN BE RUN, AND ONLY PROGRAMS SUBMITTED FROM SCOPES^CAN INTERACT WITH THE SCOPES;

ONLY ALLOWED USERS CAN USE THE SCOPES. IN ORDER TO BECOME AN ALLOWED USER, ONE SHOULD CONTACT A. H. BOND, C. C> EXTENSION 66, THE MAIN USES OF THE SCOPES ARE EXPECTED TO'BE FORI (A) PROGRAMS NEEDING ON-LINE DYNAMICAL: GRAPHICAL DFSPLAYL AMD' (B) INTERACTIVE PROGRAMS, THAT IS, PROGRAMS WHICH COMMUNICATE WITH! THE HUMAN WHILE RUNNING, AND CAN BE GUIDED AND INFLUENCED BY THE HUMAN, THE VERY GENERAL DISPLAY EQUIPMENT ALLOWS A GREAT VARIETY OF METHODS OF MAN-PROGRAM INTERACTION.

THE SYSTEM IS STILL UNDER DEVELOPMENT AND ATTEMPTS TO USE "SOME TEATURES WTLL YTELD THE E^RO^ MESSAGE 'SORRY, NOT YET IMPLEMENTED'. HOWEVER, THE SCOPE USERS MANUAL WILL' BE KEPT STRICTLY UP TO DATE WITH CURRENT IMPLEMENTATION, THFCI DATE OF REWRITING IS SHOWN ON THE FRONT COVER, BETWEEN REWRITES, ANY CORRECTIONS TO THE MANUAL ARE KEPT ON AN AND~FIA~*D~1Zk!T~>IF OBTAINED BY EXECUTING

USER CR38AB14; FILE 81/P; GET TO $| RUN,AND,TAPEI
ALLOW 5 PAGES AND 2 MINUTES.

FURTHER COPIES OF THE FULL MANUAL CAN BE OBTAINED BY EXECUTING

USER CR38AB14; FILE 82/PJ GET TO SJ RUN,AND,TAPEI
ALLOW 60 PAGES AW 6~MINUTES. ;~

OR FROM A H BOND.

2. INTRODUCTION AND SUMMARY

THE SCOPES CAN BE USED OFF-LINE, THAT IS, WITHOUT USING THE CENTRAL_PROCESSOR_OF_THE G-21, _I'N_FACT,___0NLY_USING ONE BK MODULE OF MEMORY, OFF-LINE, ONE CAN ENTER CHARACTERS OMTO THE SCOPE FACE FROM THE KEYBOARPS AND ENTER VECTORS (LINES). ONE CAN ALSO ALTER EXISTING DISPLAY BY DELETION AND INSERTION OF ELEMENTS, AND ONE CAN__TRANSLATE___(MOVE) PARTS _OF THE DISPLAY TO OTHER _PARTS OF_JJHE SCREEN. THE REST OF THE G-21 CAN OPERATE NORMALLY. THE OFF-LINE" CAPABILITIES_ARE THE SUBJECT_OF J. QUATSE^S MANUAL'. THEY CAN ALSO BE USED WITH THE SCOPE MONITOR LOADED. THE SCOPE MONITOR IS AUXILIARY TO THE MAIN G-gj MONITOR ANp WORKS ON AN INTERRUPT BASIS, NORMAL USER PROGRAMS CAN BE PROCESSED BY THE G-21 AND WHEN SOM E SCOPE CO M PUT A T IQ N I_S_ N E ED EJ3 ,,, _THE__U S E R PROGRAM_IS_ INTERRUPTED FOR A FEW MILLISECONDS.

IN THIS WAY THE SCOPE MONITOR CAN SNATCH BRIEF SPELLS OF COMPUTATION TO CARRY OUT MANAGERIAL FUNCTIONS AS DESIRED BY THE ySER.__T_HIS_I_S DONE _BY PR_ES_SjNG T HE_ A P P R 0 P R I_AT E__I N T ER R UP T BUTTONS^ THE MEANINGS CURRENTLY ASSOCIATED WITH THE BUTTONS ARE SHOWN BY AN EXPLANATORY DISPLAY. _THE F_AC I L I TI ES_PROV_I DED_ BY THE SCOPE MONITOR ARE DESCRIBED IN DETAIL IN SECTION 4. THEY INCLUDE STORAGE OF DISPLAY MATER I AL ON *SCQPE FILES', SUBMISSION OF PROGRAMS TYPED ON THE SCOPES, THE PERUSAL AND EDITING OF TEXT, AUXILIARY DRAWING" OP E R AT I DNS _L I K E_L_I GH T_- PEN _IR. A C KI NG. THERE A RE_D E3UGG I NG_F A Cj L_I TIE S WITH A DYNAMIC CORE DISPLAY AND ON-LINE PATCHING AND TRANSFER FACILJTIES,

IN ADDITION TO INTERRUPTS PRODUCED BY THE INTERRUPT BUTTONS, THE SCOPF MONITOR RECEIVES INTERRUPTS ONCE EVERY SECOND, TRIGGERED BY THE G-20 REAL-TIME CLOC K.'" RELY I NG ONLY ON T HE S E CLOCK P U L S E S TO PROCESS REQUESTS WOULD LEAD To TOO LONG A RESPONSE TIME._THE_CLOCK PULSE "ENABLES THE' SCOPE MONITOR TO PR 0V I DECONTINUOU'S MODE OPERATIONS SUCH AS THE DYNAMIC CORE DUMP, THE ROTATION MODE AND THE CURVE DRAWING MODE.

INTERACTIVE PROGRAMS CAN BE WRITTEN___IN NY PROGRAMMING LANGUAGE. THEY C A N C 0 MM U NI GATE WITH THE SCOPES 3V US ING THF •B ROUTINES' PROVIDED BY THE SCOPE MONITOR {THESE ARE LIKE I ROUTINES HTTHE MAIN MONT"**TOTr;** USING THESE, A PROGRAM CAN SET UP A GENERAL GRAPHICAL DISPLAY AMD CAN EXAMINE A GRAPHICAL DISPLAY ENTERED BY A HUMAN. THE HUMAN AND PROGRAM ARE TREATED MORE OR LESS EQUIVALENTLY BY THE SCOPES, AND THE SCOPES PROVIDE A GENERAL, =UPID AMD ""TRANSPARENT INTERFACE BETWEEN THEM TO PERMIT MAN-MACHINE COOPERATION ON A PROBLEM.

INTERACTION WITH THE PROGRAM CAN CONSIST OF EACH READING DISPLAY MATERIAL SET UP BY THE OTHER j AND * IN ADD ITI ON.THERE ARE 8 'STATE SWITCH FSl AND^ 2 IN A L OG KNO 0 S » (G IV J NG_ A_Q UA S I - C 0 NT INUO US_ VARIABLE), WHICH CAN BE SET BY THE HUMAN AND READ BY THE PROGRAM USING B ROUTINES. ALSO THE U_SER_ CA N_DEF_I_NE_H_I_S_ 3W_N INTERRUPTS AND THE SCOPE MONITOR WILL PASS CONTROL TO THE DEFINED POINTS IN HIS PROGRAM,WHEN HE PRESSES THE APPROPRIATE BUTTON.

THE 8 ROUTINES ARE DESCRIBED IN SECTTOM 7. THE*E IS A ' B-PROCEDURE ' _IN _ALGOL_AND F0RML, WHJ CH CALLS T_HE 8_R_0UJM M_ES, AND ALSO MANY USEFUL SUBPROGRAMS TN THESE LANGUAGES AND IN SPITE. THESE ARE KEPT ON AND FILES AND ARE DESCRIBED IN SECTION 8, SIMILAR SUBPROGRAMS CAN BE WRITTEN IN ANY LANGUAGE AVAILABLE ON THE G-2l_.

UTTERACTION wTTH USER PROGRAMS CAN ONLY OCCUR DURTN"G~THE" ⁻ " SHORT RUN TIME OF THE PROGRAM,BUT WE ARE TRYING TO MAKE IT EASY FOR ANY USER TO WRITE A 'SUBSYSTEM•, WHICH WOULD BE ESSENTIALLY PART OF THE SCOPE MONITOR AND OPERATE ON AN INTERRUPT BASIS. MODULESOF CODE ARE KEPT ON SCOPE FILES AND~ SWAPPED IN BY THE SCOPE MONITOR AS NEFDED AND AS SPACE ALLOWS. IT IS ONLY POSSIBLE "TO WRITE SUCH MODULES IN ASSEMBLY LANGUAGE AND THEH SfZF IS RESTRICTED TQ <3K; HOWEVER, A SUBSYSTEM CAN CONSIST OF AN ARBITRARY NUMBER OF LINKED REENTRANT MODULES. SUBSYSTEMS ARE DISCUSSED IN _SPCT 10 N_9 .

G-21 SYSTEMSAUXILI 4RY~"T0 ' THE SCOPE Mo NITO 3 ARE DES C ft I BED" ""IN" SECTION lQ) FnR EXAMPLE, A USER SYSTEM IS NEEDED TO MOVE MATERIAL BETWEEN AND FILE'S AND SCOPE FILES". "IN SECTION u, *E OUTTTNE THE INTERNAL WORKING OF THE SCOPE MONITOR PROGRAM.

3. OFF LINE USE, TERMS AND CONCEPTS

THERE ARE 3 SCOPES* NUMBERED 1, 2,3 FROM THE LEFT OF THE ROO_M._THE SCOPE_ FACE JS 10 INCHES BY 10 INCHES AMD HAS 1024 X 1024 RASTER POINTS. THE 32 BUTTONS ALONGSIDE THE FACE ARE THE STATE SWITCHES AND CONSTITUTE THE STATE WORD. THE LOWER 2 ROWS ARE COLORED GREEN AND ARE FOR USE BY A USER PROGRAM, WHEN A SWITCH IS ON IT IS LIT AND THE VALUE OF THE SWITCH_IS 1, 0_N_THE LE_F_T_OF THE_ SCREEN ARE 2 'ANALOG KNOBS' KNOB 1 ABOVE KNOB 2, THESE CAS 3S USED BY A_ USERJPROGRAM;____THEIR VALUE VARIES FROM 0 TO 63 AND THE FULL RANGE IS OBTAINED IN HALF A TURN,

_______ON THE DESK, THERE ARE TWO KEYBOARDS, USED EQUIVALENTLV. AND A_ CONTROL PANEL CONSISTING OF:

(I) 20 INTERRUPT BUTTONS NUMBERED 0-19

(If) A CURSOR CONTROL CONSISTING OF 4 BUTTONS T~0 INDICATE WHICH DIRECTION___TO MOVE THE CURSOR ON THE SCREEN.____THE CENTRAL BUTTON fN THE CONFIGURATION MAKES THE CURSOR MOVE FASTER, AND THE SLEW BAR MAKES IT MOVE EVEN FASTER.

(III) THE MARK BAR

TO ENTER DISPLAY MATERIAL ON THE SCREEN ONE FIRST NOTES FROM THE 6TH ROW OF STATE SWITCHES THAT THERE ARE 4 'PAGES' FOR USE. THIS____MEANS THAT ONE CAN HAVE j4 DIFFERENT INDEPENDENT DISPLAYS AVAILABLE, WHICH CAN BE MADE "VISIBLE BY USIMG THE APPROPRIATE STATE SWITC H_ AND SUPERIMPOSED AS D ESI RED. HOWEVER, ONE SHOULP ONLY ENTER MATERIAL"INTO ONE PAGE"AT AT I ME.

_______BEFORE ONE CAN ENTER MATERIAL* ONE MUST _USFJ___TJHE SCOPE MONITOR, DESCRIBED LATER, TO RESERVE SOME MEMORY SPACE FOR THE MATERIAL__AND _TO DEFINE_THAT SPACE TO CORRESPOND TO THE REQUIRED PAGE NUMBER. " ALSO ONE MUST ENABLE THE PAGE ONE IS USING AND TURN THE PAGE STATE SWITCH ON. THIS ALLOWS DISPLAY MATERIAL TO BE ENTERED MANUALLY. WHEN A PAGE IS ENABLED, IT HAS A CURSOR VISIBLE AND TH]S_DEFINES THE PLACE ON THE SCREEN WHERE ATTENTION IS FOCUSED." CHARACTERS MAY NOW BE ENTERED FROM THE KEYBOARD IF OME SETS THF STATE SWITCHES, TO_ENTER AND CHARACTER.

_______VTCTORS___(LTTiETJ____MTY~TE~ERTE~RTD MANUALLY AMD THE DISPLAY CATT BE MANIPULATED WITH____CORRECTI0NS,_____DELETI 3MS,______INSERTIONS, TRANSLATIONS, ETC«, AS DESCRIBED IN J.Q.H, HOST OPERATIONS ARE DONE BY SETTING THE STATE SWITCHES TO THE APPROPRIATE VALUES, P0SITIJ)NING THE CURSOR AND PRESSING THE MARK_BAR.J_ONE CAN INPUT DOUBLESIZE CHARACTERS BY SETTING A STATE SWITCH AND ONE CAN GET SUBSCRIPT_ SIZE CHARAGTERS_ BY PRESSING_^j_0NE QEJTS BACK TO NORMAL_ SIZE BY PRESSING , MARGINS ARE ENTERED BY POSITIONING THE CURSOR AND SETTING THE MARGIN STATE SWITCHES AND PRESSING HARK, MARGINS AND ALL OTHER CONTROL SYMBOLS CAN NOT ONLY BE MADE VISIBLE BY A STATE SWITCH, THEY CAN ALSO_BE MANIPULATED IN EXACTLY THE SAME WAY AS NORMAL SYMBOLS.

TO CLEAR A PAGE OF ALL MATERIAL, IT MUST BE VISIBLE AND ENABLED, AND "~ONE THEN SETS THE CLEAR STATE SWITCH. THE SPACE RESERVED FORTHE PAGE IS STILL THERE AND IT IS STILL ENABLED AFTER THE CLEAR OPERATION.

ONE SHOULD ONLY HAVE ONE PAGE ENABLED AT ANY ONE TIME, AS THERE IS ONLY ONE CURSOR POSITION.

MARGINS CONTROL ONLY TEXT ENTERED AFTER THEM AND ONE CAN HAVE SEVERAL MARGINS ON' ONE PAGE. IN THE ABSENCE OF MARGTNFTHE END OT THE SCREEN IS AN EFFECTIVE MARGIN WHEN ONE DOES A RETURN CHARACTER. IF DISPLAY MOVES OFF THE SCREEN IN ANY DIRECTION IT *WRAPS ROUND' AND APPEARS ON THE OTHER ' EDGE OF THE SCREEN) SIMILARLY FOR THE CURSOR POSITION, ~ " ~~

THE LIGHT PEN CAN BE USED TO POSITION THE CURSOR AT AN EXISTING DISPLAY ELEMENT. ONE POINTS THE LIGHT ?EN AT THE ELEMENT AND TH^jcURS0R_W_ILL_AUT0MATICALLY MOVE THERE. IT MAY BE NECESSARY TOTNCREASE THE BRIGHTNESS TO" GET IT TO WORK. "

THE SCREEN FACE CAN BE PHOTOGRAPHED IN COLOR OR 3LACK AND WHITE WITH AN ORDINARY CAMERA. THE ENGINEERING GROUP HAS A POLAROID CAMERA FOR_ THIS PURPOSE, ALSO THERE IS A SPECIAL HARD CO'PY DEVICE, UNDER DEVELOPMENT" BY THE ENGINEERING GROUP, WH TCH" TAKES AN_ ACTUAL SIzE_ NEGATIVE IMAGE OF THE SCREEN DIRECTLY ONTO PHOTOSENSITIVE PAPER"," SO THAT THE "LINFS AND" CHARACTERS A'RE' BLACK ON WHITE. ENQUiRIES ABOUT THIS EQUIPMENT SHQULD BE DIRECTED TO BEAU BRINKER,C,C, EXTENSION 75. OPINIONS AND IDEAS OW THE HARDWARE SHOULD BE_SENT TO THE ENGINEERING GROUP , THERE IS NOW_ AVATLABLE A RAND TABLET,WHICH CAN BE ATTACHED TO EITHER SCOPE 1 OR SCOPE 2~. IT CAN BE USED AS A POINTING DEVICE LIKE THE LJQHT PEN, BUT IN AUDITION IT ACTS 1KB THE MARK BAR. FURTHER, IT WILL EMTER LINES C0NTINUOUSLY INTO THE PAGE GIVING CURSOR TRACKING AND CURVE DRAWING, INSTRUCTIONS ON ITS USE ARE TO BE FOUND IN A FOLDER WITH THE EQUIPMENT. ENQUIRIES ABOUT IT SHOULD BE SENT TO DICK SHOUP.

4. THE SCOPE MONITOR

A. GENERAL LAYOUT AND OPTION STATE

_______THE_S_COPE MONITOR PROVIDES A^ RANGE OF FACILITIES WHICH ARE LINKED TO THE INTERRUPT BUTTONS. THE MEANING OF THE BUTTONS IS ^E D EFIN'ED AS 0 NE USES VARIOUS * STATES' OF THE SCOPE MONITOR. WHEN ONE FIRST APPROACHES A SCOPE, AFTER THE SCOPE MONITOR HAS BEEN LOADED^ 1T__HAS__1IHIS SPACE RESERVED FOR SYSTEM MESSAGES' ON THE BOTTOM OF THE SCREEN. IN THIS STATE, EVERY INTERRUPT BUTTON LEADS JO THE LOG-IN_STATE JFIG. 1> AND THE USER _MUST ENTER,HIS FULL G-?l USAGE NUMBER AT THE POSITION OF THE CURSOR, THE CURSOR IS SET 9Y~ THE SCOPE MONITOR AND THE STATE SWITCHES ARE SET TO ENTER, CHARACTER, PAGE 1 (IF THIS DOESN'T HAPPEN, SET THEM 3Y HAND). AFTER TYPING THE USAGEJJUMBER, PRESS_RETURN, _JH= _RETU_RN CHARACTER IS USED BY THE SCO~PE MONITOR AS A COMPARE INTERRUPT, AND TELLS IT TO READ IN_J-HE_CHAR_A_CTER JUST TYPED BY THE USER. IF THE NUMBER WAS MISTYPED, OR DOES NOT BELONG TO AN ALLOWED USER, THE MESSAGE 'SORRY NOT ACCEPTABLE' WILL APPEAR. OTHERWISE. IT WILL GO TO OPTION STATE AND DISPLAY THE MEANINGS OF THE INTERRUPT BUTTONS IN THIS,, _STAXE_i____THE OPTION STATE !S_jrHE. TOP-LEVEL OF A HIERARCHY OF STATE'S AND WITH IT ONE SELECTS ANOTHER STATE.

NOTE THE WORD 'STATE' IS USED TO DESCRIBE THE CONDITION OF THE SCOPE MONITOR AND THE DEFINITION OF INTERRUPTS IN THAT CONDITIONj,______EACH HAS AN ASSOCIATED SYSTEM 'PAGE' AND SO SOMETIMES THE WORD 'PAGE. REFERS TO A 'STATE'. OCCASIONALLY, THE CONDITION OF THE SCQPE_MflNJI0R IS_ DESCRIBED AS A 'MODE', ESPECIALLY IF IT IS DOING AN OPERATION CONTINUOUSLY. THE USE OF THESE WORDS SHOULD BE DISTINGUISHED FROM THEIR USE IN OFF-LINE_USE. THERE IS SOFTWARE STATE, PAGE AND MODE DISTINCT FROM HARDWARE STATE, PAGE AND MODE. IT IS HOPED THAT NO CONFUSION WILL ARISE. _T_HUS THE MEANINGS OF THE BUTTONS IN THE OPTION STATE ARE ALL 'CHANGE STATE TO. . . . STATE'. THE_VARJOUS_STATES ARE DESCRIBED BELOW. IN EVERY STATE, INTERRUPT 0 ALWAYS MEANS GO BACK TO OPTION STATE. INTERRUPTS 17, 1.8, AND 19 ARE CURRENTLY USED FOR SYSTEM MAINTENANCE AND SHOULD NOT 3E USED.

________fHE—DISPLAYS USED BY"'THE" SCOPE MONITORCAMNOT BE" ALTERED 9Y THE USER AS THEY ARE IN ALTERNATE MODE, EVEN THOUGH PAGE 1 IS USED BY THE SCOPE MONITOR, IT CAN ALSO BE USED BYTHE USER AS A NORMAL PAGE.____•___________________

WHEN TYPING IN MORE THAN ONE VALUE TO THE SCOPE MONITOR, DO A RETURN AFTER EACH VALUE AND THE SCOPE MONITOR WILL REPOSITION THE CURSOR","

```
          LOG - IN
          ENTER YOUR USAGE NUMBER HERE






THIS SPACE RESERVED FOR SYSTEM MESSAGES
```

FIGURE 1

IN EACH STATE, THE MEAJyj 1 NG_ OF THE INTERRUPTS ARE DISPLAYED 9Y **A** SYSTEM PAGE, THIS DOES NOT INTERFERE WITH THE USER DISPLAY AND CAN BE TURNED ON OR OFF (MADE VISIBLE OR INVISIBLE) IN ANY STATE BY USING INTERRUPT 1. ON PRESSING AN INTERRUPT BUTTON, ITS NUMBER IS DISPLAYED IN THE BOTTOM RIGHT H**A**NQ CORNER OF THE SCREEN, DURING THE PROCESSING OF AN INTERRUPT THE NUMBER IS MADE TO FLASH, THE USER SHOULD^ NOT PRESS ANOTHER INTERRUPT BUTTON UNTIL THE NUMBER HAS STOPPED FLASHING. USUALLY THE OPERATION IS VERY QUICK AND THE US-ER DOESN'T _Sj=E_ ANY FLASHING_JHOWEV|R, OPERATIONS REQUIR I NG_THE SCOPE FILES INVOLVE THE USE OF THE DISC AND ONE MAY HAVE TO WAIT FOR THE DISC TO BECOME AVAILABLE FOR A SECOND OR TWO. THE NUMBER WILL ALSO FLASH WHILE TYPING IN VALUES OF PARAMETERS TO THE SCOPE MONITOR, IN THIS CASE. ONE CAN CONTINUE TO„ENTER PARAMETERS.

LOG OUT

PRESSING INTERRUPT 8, ON THE OPTION PAGE, LOGS THE CURRENT USER OUT AND THE MESSAGE 'LOGGED OUT' IS DISPLAYED.

B. MANAGEMENT STATE

THE MEANING 0?~THTTNTERRUPTS IN THIS~STATE A"RT~SFIIOWBY~THE SYSTEM DISPLAY, REPRODUCED IN FIGURE 2.

TK ALLOWED USER HAS RESERVED FOR H1H #T£M "$P*8WfS?" 1 TO 20 WHICH ARE ARBITRARY IN SIZE. HE CAM SAVF C I ^ *ATfffMt_ ON THESE FILES PERMANENTLY BY USING INTERRUPT 2, £A* *SV§ F«f CONTENTS OF A PREVIOUSLY STORED FILE TO A PAGE OISPLAYFFF USiHQ INTERRUPT 3. WHEN USING 3. SPACE DOES NOT HAVE TO BE""RffWIf)TO* THE PAGE. IT IS DONE AUTOMATICALLY, INDEED ANY MATGRFAJ!M THAT PAGE BEFORE IS CLEARED. ONE CAN GET A DIRECTORY OF THE '(FIt-f* BY PRESSI_NG_INTERRUPT 5. THE DISPLAY_IS_L_IKE FIGURE 3, jf THE BASE (RECORD NUMBER) AND LENGTH OF THE RECORD ON THE DISC. THIS IS NOT OF MUCH USE TO THE NORMAL USER EXCEPT TO SEE THAT A FILE IS PRESENT OR HAS CHANGED IN LENGTH.

INTERRUPTS 4, 6-9 HANDLE THE RESERVED SPACE FOR THE PAGES, ilnWijyT™6Tn^sTRvTs~soME SPArcEToP~A~QTVEN PAGE, THE UNIT USED IS THE BLOCK, WHICH IS 16Q» W0RDS, THERE ARE 30 BLOCKS AVAILABLE FOR USE BY 3 SCOPES. A PAGE PACKED SOLID WITH DISPLAY PROBABLY NEEDS 4 BLOCKS OF SPACE.

INTERRUPT 7 ENABLES A PAGE,"AND 8 DISENABLES A PAGE,

INTERRUPT 9 DELETES A PAGEI I.E., IT REMOVES THE SPACE RESERVED FOR THAT PAGE AND MAKES IT AVAILABLE FOR OTHER USE, USING 6 MERELY DISENABLES A PAGE AND KEEPS THE SPACE RESERVED.

```
MANAGEMENT PAGE 1

PRESS INTERRUPT NUMBER

2.  SAVE PAGE AS SCOPE FILE

3.  READ IN SCOPE FILE AS PAGE

4.  APPEND PAGE TO .PAGE

5.  DISPLAY DIRECTORY OF SCOPE FILES

.6. GET BLOCKS FOR PAGE

7.  ENABLE PA,GE. . . . . . .

8.  DISENABLE PAGE

?.  DELETE PAGE
```

FIGURE_2

```
           DIRECTORY  FOR  LC02
FILE              BASE          LENGTH

00.               000            000
01^               576           0  02_
0 2.              535           00  2"
03.               570            004
04.               533            002
05.            _ 530             003
06.         "   525"           0  03
07.               00 0          00  0
0 8."             00 0          0 0 0
09.               0_O0           000
10.               000            000
11.               000            000
12.               000           00  0
13^               0 0J)          000
14.               0 0 0         0 0 0
15.               p_O0           000
16.               000            000
17.               0 0 0         0  o x
18.               000            000
19^               0 0 0          000_
20.               00 0           000
```

THIS SPACE RESERVED FOR SYSTEM MESSAGES

FIGURE'3

mi utwv

INTERRUPT 4 WILL APPEND ONE PAGE TO ANOTHER SO THAT THE SECOND PAGE THEN HAS THE DISPLAY MATERIAL OF BOTH, AND THE FIRST IS UNCHANGED

BELOW IS GIVEN THE SEQUENCE OF ACTIONS SGQUIRFU TO L&8 IN *HO SET UP THE SCOPE FOR ENTERING._CHAR*CTRS.F AND L MIH.DN. TM0 SGRGGN.

1. IF NO ONE IS LOGGED IN YET, THERE WILL JUST BE THE ONE LINE MESSAGE ON THE BOTTOM OF TH<= SCREEN, OR ELSE THE MESSAGE •LOGGED OUT*. IN THIS CASE, PRESS INTERRUPT 0. THIS GIVES THE LOG¬IN PAGE. TYPE *IU* YOUR USER NUNBGFT A*0 PRESS RETURN, THIS WILL GIVE THE OPTION PISE.

2.„JF SOMEONE ISJ.OGGED IN ALREADY,_PRESS INTERRUPT 0 -• THIS GIVES **THE** OPTION PAGE,

3. „JJN THE OPTION STATE, PRESS INTERRUPT ?_. THIS GIVES THE MANAGEMENT PAGE.

4. IN THE MANAGEMENT STATE, PRESS INTERRUPT 6. THIS PUTS THE CURSOR AFTER 'GET' AND DISPLAYS THE NUMBER 6 BLINKING IN THE BOTTOM RIGHT.HAND CORNER, TYPE THE FIGURE 2 FROM THE KEYBOARD AND PRESS RETURN. THIS RESETS THE CURSOR TO AFTER 'PAGE', TYPE 2 AND RETURN. **YOU NOW** HAVE RESERVED 2_BJ_0CKS **OF** SPACE **ON** YOUR ?A3E *2.*

5. PRESS INTERRUPT 7, THE CURSOR WILL APPEAR AFTER 'PAGE' ON LINE 7_OF_THE M_ANAOEMENT_PAGE. TYPE 2_AND.....RETURN. PAGE ? IS NOW ENABLED, AND WILL ALLOW DISPLAY MATERIAL TO BE ENTERED FROM THE CONSOLE,. . . . .

6, PRESS TNTERRUPTT"! THIS MAKES **THE** MANAGE~MTTNT~TXGE^THLS"PLAY " INVISIBLE.. . .. .

T, PRESS"THE STATE SWITCH FOR PAGE 2.YOU'SHOULD SEE A CURSOR. USE THE CURSOR CONTR0L TO POSITION THE CURSQR. TP TYPF IN CHARACTERS, PRESS STATE"SWITCHES ENTF~R AND CHARACTER AND THEN TY°E FROM THE KEYBOARD. TO DRAW LINES, PRESS STATE SWITCHES_ENTER AND VECTOR AND USE THE CURSOR CONTROL AND THE MARK BAR.

C. THE PROGRAM STATE

FIGURE 4. " THIS STATE ORGANIZES THE INITI'ATION "OF USER PROGRAMS AND USER SYSTEMS FROM THE SCOPE MONITOR. WHEN A PROGRAM IS ACTUALLY INfERACTING "WITH fHE SCOPES,"THE "SCO'E~MONITOR SHOULD BE PUT IN USER PROGRAM INTERACTION STATE OBTAINABLE FROM THE OPTION STATE, HOWEVER, ALL' ORGANIZATION PRIOR TO AND AFTER THE RUN IS DONE WITH THE PROG RAM STATE

Tl- SUBMIT a PROGRAM, ONE SHOULD GET" SOME BLOCKS~FOR "A PAGE AND ENABLE IT, THEN TYPE THE PROGRAM ONTO THAT PAGE. NOTE THAT THERE ARE N0~TA8 SETTING"S~ON THE~SC'OPESl EVEPTytTTING MUST BE SPACED BY HAND, ONE CAN KEEP PROGRAMS ON SCOPE FILES ALSO AND PUT THEM ON THE PAGE THAT" WAV, ONE WOULD USUALLY SET UP TH= PROGRAM ~WITH THE" PROGRAM PAGE SYSTEM DISPLAY TURNED OFF. THEN OMg SHOULD TURN OFF THT"" ~~PAGE~'~ANd"TulRrrGNTHiE~SYSTEM "6TSPTa"Y~A"G Ai"N~UsTnG~TNTERH! UPTTT THE SUBMISSION OF A PROGRAM TAKES PLACE IN TWO STAGES. FIRST IT MUST BE MOVED TO THE 'INPUT FILE'. THIS IS NOT TO BE CONFUSED WITH A SCOPE FILE. IT IS A PSEUDO TELETYPE BUFFER» SECOND. THE INPUT FILE " KUST BE'SUBMITTED"» " TO RUN ON THE G-21. TO "OVE IT "TO THE ~ INPUT FILE ONE SHOULD USE INTERRUPT 2. THIS CONVERTS THE PROGRAM TO (UPPER CASE) G-21 CHARACTERS AND PUTS IN A " BLA^K J09 CARD AT THE TOP. INTERRUPT 3 MOVES A PAGE WITHOUT CONVERSION AND.IS BARELY USED.

USING INTERRUPT 4, ONE CAN NOW SUBMIT THE INPUT FILE. THE VALUFS OF TIME, PAGE S~/AN D^J" SYSTEM "REQUESiTED~ AR=~TYPTD! N 1TN D PUT INTO THE JOB CARD, AND THE JOB IS PLACED IN THE G-21 QUEUE TO 3E RUN.

WHEN IT RUNS, ANY TELETYPE OUTPUT IS PUT IN THE 'OUTPUT FILE' , 0NE CAN LOOK "AT" THE" INPUT FILE OR THE OUTPUT Fl L"E-9Y""USINQ IN TERRUPTS 5 AND 6. TH ESE M0VE THEM T0 ADESIGNATE D PASEJ SPACE TJOES NOT HAVE TO BE RESERVED F"OpTThE^GXTN~THT"r~015^aTI0N,

INTERRUPTS 7 AND 8 ARE NOT YET IMPLEMENTED BUT WILL PERMIT A PERUSAL OF THE" INPUT" 0'R~0'UTPUT" FILE. THESE FILES ARE VERY "MUCH" LARGER THAN CAN BE FITTED ONTO A PAGE, AND INTERRUPTS 5 AND 6 JUST ToWinH-HE^FT^fTF^ ONLY LOOK AT THE REST OF ONE'S OUTPUT BY.GETTING THE LINE PRINTER OUTPUT. THE SCOPES 1, 2, AND 3 ARE EQUIVALENT TO TFLETYPES MUMBFR 5, 6, AND" 7" RESPECTIVELY, AND LINE PRINTER OUTPUT IS NUMBERED WITH THESE REMOTE NUMBERS. ALSO THE JOB CARD HAS THE WORDS SCOPES AND .COURIER., WHFN THE COURIER SERVICE IS IN OPERATION; OUTPUT IS PLACED ON THE TA8LE HN ~P0RT£R HALL PASE MEN~TNEA~R ~ T~H_F~SC0'ES-R00M*~ OTHERWISE, ASK FOR IT AT THE I/O COUNTER.

WHILE A PROGRAM I_S_INTERACUN6 WITH THE SCOPES, THE SCOPE MONITOR CAN STILL BE USED IN ANY STATE. THE INTERRUPTS DEFINED 9Y THE_ USER WILL ONLY BE PASSED TO THE USER PROGRAM WHEN THE SCOPE MONITOR IS IN THE USER PROGRAM INTERACTION STATE.

```
                                                        SCOPES

                    PROGRAM PAGE

PRESS INTERRUPT NUMBER

   2^ CONVERT PAGB AND MOVE TO INPUT TILE

   3. MOVE PAGE (UNCONVERTED) TO INPUT TILE

   4. SUBMIT INPUT FILE I TIME PAGES SYSTSM

   5^ DISPLAY INPUT FILE AS. PAGE

   6. DISPLAY OUTPUT FILE AS PAGE

   7. FORWARD TEN LINES

   8. BACK TEN LINES

   9. LOAD MONITOR MODULE OF USER

  10. TRANSFER TO ENTRY POINT .°F M?PULE- OF USER

  11. RELEASE MODULE OF USER

  12. ALLOW PROGRAM FROM SCOPE TO INTERACT
```

FIGURE 4.

THE _|NPUT_FIL_E IS MOVEDTO ANOTHER INACCESS18LE INPUT FILE ON SUBMISSION, AND THIS LATTER "INPUT FILE CANNOT BE LOOKED AT OR ALTERED^ HENCE,__JT YOY_HAVE _MADE_A_MISTAKE_IM YOUR PROGRAM AMD HAVE ALREADY SUBMITTED IT, YOU CANNOT RECALL IT|T IF WILL 3E RUM. IF YOU RESUBMIT* PROBABLY BOTH WILL RUN.

SCOPE PROGRAMS ONLY HAVE THE SAME PRIORITY AS NORMAL TELFTY'E PROGRAMS, AND_ THEY CAN ONLY RUN FOR 3 MINUTES) HOWEVER,THE WAITING IN HANDLED DIFFERENTLY, TO MAKE IT EASIER FOR THE USER TO 3E PRESENT WHILE HIS PROGRAM IS RUNNING. ON SUBMISSION OF THE PROGRAM IT GOES TO THE TOP OF THE QUEUE (SM PRIORITY) AND WILL PROBABLY RUN WJTJILN 10 MINUTES OF SUBMISSION. JHE_S_C0PE MONITOR COMPUTES, AT THIS TIME, THE ALLOWED TIME OF NEXT SUBMISSION, ALLOWED TIME = [REAL TIME -• (TIME OF SUBMISSION OF CURRENTLY RUNNING PROGRAM)} + R"EAL TIME.

A SUBSEQUENT_ATTEMPJ_TO SUBMIT A PROGRAM WILL YIELD THE_ERROR MESSAGE 'SORRY NOT ACCEPTABLE'*t* IF THE TIME THEN IS BEFORE THE ALL_0_WED TI_"E WHEN A_PROGRAM IS_QUJ=UEP THERE IS_N0_INDICATIJ)N THAT IT IS QUEUED. WHEN IT"FINISHES, THE SCOPEM0NIT0R'~DISPLAYS THE MESSAGE 'OUTPUT READY' AND THE USER CAN FIND TELETYPE OUTPUT IN THE OUTPUT FILE.

INTERRUPTS 9 _~ 12 ARE NOT YET DEBUGGED AND ARE FOR WRITING •USER SCOPE MONITOR SUBSYSTEMS' OR 'USER MODULES', MODULES ARE DISCUSSED IN SECTION 8.

D. THE DEBUG STATE

SEE FICURE 5. THIS DISPLAYS A DYNAMIC CORE DUMP OF **A** MY REGION OF CORE OF THP G-?I. THE REGION DISPLAY IS SELECTED HY TURNING THE ANALOG"KNOBS AND SETTING THE STATE SWITCHES. <N03 i IS THE LAST " TWO OCTAL DinITSJ __KNQB *2* THE MIDDLE TWO) AND THE 80TT0**M** ROW OF STATE SWITCHES "IS THETOP 4 BITS OF THE ADDRESS, WHILE THE DUMP IS BEING DISPLAYED, IT IS TYING UP THEG-21, AND THEUSER PROGRAM IN LOWER" CORE IS NOT 9EING PROCESSED; H6WEVER." INT*RRUPTS CAN 3E PROCESSED. THUS THIS FACILITY SHOULD BE_ USED SENSIBLY AND CERTAINLY NOT LEFT DISPLAYING FOR **A** LONGTIME,

THE INTERRUPTS ^LLOW ONE JO PATCH THE COR_E_. _ THIS **JS_**00NE_9Y **PUTTING A NUMBER INTO THE INPUT BOX. THE DEBUG STATE IS ENTERED IN CORRECOODE. AND THE CURSOR IS_ENABLED, ONE MOVES THE CURSOR TO THE INPUT BOX AND CORRECTS THE CONTENTS OF IT; THEN ONE SHOULD GET OUT OF CORRECT MODE.**

I fitERRUP T"**T**~C**L**EAR'STHE CONTENTS OF THE INPUT BOX" TO ZERO.

INTERRUPT *2*_ STORES THE CONTENTS OF THE INPUT ROX IN THE LOCATION OF "THE OCTAL DUMP WHICH IS UNDERLINED,

INTERRUPT 3 PUTS THE CONTENTS OF THE UNDERLINED LOCATION INTO THE INPUT"BOX'; "

INTERRUPT 4 SWAPS THE CONTENTS OF THE INPUT BOX WITH THOSE OF THE UNDERLINED LOCATION. " "™ " "

INTERRUPT 5 **ALLOWS ONE TO TRANSFER TO ANY LOCATION) ONE PLACED" TH£ "LOGATI** 0**N**~PTf**H**E ~ ™UT~801T**A**ND~TFI**E N**"FRTS"**S"E"S**~m=R^J**P**T-ST **THIS DOES A TRM WITH CONTROL OFF) HOWEVER, NOTE THAT CE AND PE ARE "SET FOR** THE **SCOPE MONITOR,** so **THAT**

11) **THE USER HAD BETTER RESET THEM**
TO HIS OWN" VALUES, ~ "' " " ~

_(11) **HE _MUST_** KE**EP_^.ONTROL_**0**F_F_.**

ALSO NO"fl **THAT *'**

(111) **HE MUST RETURN THROUGH HIS MARK.** _ ^

IF (I) (II) OR (MI) "ARE VIOLATED, YOU WILL" PROBABLY DESTROY JH_E_ENT_I RE_WORLD. . . . .

THE USER CAN LOOK AT ANY REGION OF COPE) HOWEVER, HE CANNOT ALTER OR TRANSFER TO AN ADDRESS IF IT IS NOT IN USER CORE, I.E., IN 170 TO /730n0. IF HEJTJ?IES_Tp DO SO, THERE WILL BE NO _RE_SP_ONSE FROM THE SCOPE MONITOR.

DEBUG PAGE

```
_      STORE  XNPTJT       ~                 ~

3T⁻10AD   INTUT^mSH^IEWRY"

T.     sWa"P~"in"put                     :
______

               oooooooooooo

 005344    0 0 0 0 0 0 0 0~6~7    Oil 0 0 0 0 73653    00000001453    040500Q56JZ
 005350     01550 0 0 010 (1     01730005632    00050000100    05550006732
- 0 05354    0T7T0T0673?~0177UT07546  0  WbTOTO11"(i"~~TO0500"0W04  "
 005360    00170005353    00050000002    01770076666    01770005300
"TO5364    OOOTOOOOOrib"   0177000"310 6" "' 01770003106   "00170004312
```

THIS SPACE RESERVED FOR SYSTEM MESSAGES

FIGURE 5

TEXT HANDLING HOPE

PRESS INTERRUPT NUMBER

2. SELECT PAGE

3. SELECT FILE

4. FORWARD TEN LINES

5. BACKWARD TEN LINES

6. GET TO S

7. DUMP

8. NAME CURSOR POINT TO BE
STRUCTURE POINT

9. UNNAME STRUCTURE POINT

10. GET TO POINT

11. DISPLAY DIRECTORY OF STRUCTURE POINTS

12. READ BLOCKS AT BLOCK FILE TO BLOCK- PAGE

13. WRITE BLOCKS AT BLOCK FILE TO BLOCK PAGE

FIGURE 6

E. TEXT HANDLING STATE

SEE FIGURE 6. THIS STATE IS NOT YET DEBUGGED. IT **DOES** THE MOyEMENT_AjvlD___SCROLLJ NG__(' ROLL ROUND »J OF_TE_XT ,_ J T_I S DISTIMCT FROM THE TEXTEDITING SYSTEM WHICH IS BEING DEVELOPED BY MIKt COLEMAN AND IS CONCERNED WITH TEXT MANIPULATION ON THE PAGE TO AUGMENT THE FACILITIES PROVIDED BY THE HARDWARE.

TO PERUSE SOME TEXT,_ IT MUST BE ON A SCOPE_ FILG_._ *IT* CAN BE MOVED ONTO A SCOPE" FILE FROM AN AND 'FILE BY JSINQ AN AUXILIARY " SYSTEM (Q,V. >______AUXI'LURY SYSTEMS W*IL*L MOVE TEXT FROM AN ANO FILE IN G-20 CHARACTERS AND CONVERT AND MOVE TO A SCOPE FILE AND WILL MOVE IT BACKHAND J20NVERT IT BACK.______OR WE CAM MOV_E_JT INSCOPE CHARACTER'S UNCONVERTED BETWEEN AND FILE AND SCO=>E FILE A MO ALWAYS KEEP IT IN SCOPE CHARACTERS, _UNTIL_J_T I S_ NEC ESS ARY_ TO ᴮ*INT_I_T_ OUT. ' IT IS SUGGESTED THAT DOCUMENTATION USE THE LEAD SYSTEM. (SEE SEPARATE WRITE-UP); IN WHICH ONE INSERTS TYPESETTING COMMANDS INTO THE TEXT, SO if IS PRINTED OUT IN A PRESCRIBED FORMAT. fffE"LEAD COMMANDS COULO_JF_KEPT_I_N_ALL JHE TIME AS PART 0FJTH6 TEXT______IT IS HOPED EVENTUALLY TO BE ABLE TO OUTPUT ON THE LINE PRINTER OF THE _360 _WN_ICH _HAS_ UPPER _ AND LOWER CASE CHARACTERS, THE 3-20. _OF__ COURSE, HAS ONLY *64* CHARACTERS, INCLUDING ONLY UPPER CASE LETTERS'. HAVING GOT THE DOCUMENT INTO A SCOPE FILE, ONE SELECTS .THAT FILE TTS'ING IN TERR UP t 3 AND SELECTS A PAGE TO WORK ON USfNO INTERRUPT 2. THIS WILL AUTOMATICALLY GET *5* BLOCKS (AS MUCH AS CAM REASONABLY 3E SEEN ON ONE "PAGE)" TOR THAT PAGE AND ENABLE IT.'" THERE IS A SPEC I ALLY RES F.RVED___F I LE___USED FOR_A JSCRATCH AREA _ANDONE CAN N0W_ ROLL THROUGH T HE TEXT USING INTERRUPTS 4 AND"5. THIS SUCCESS IVELY B*H*\ NGS IN TEXT FR 0 M___THE___SELECT_E n__*fjl* E_0 N T 0__TH E_ 8 0 J T 0 M OF THE SELECTED PAGE AND MO'VES THE TOP OF THE PAGE fNTQ THE SCRTT'CH AREA, ONE_ CAN JJSE THE HARDWARE FEATURES TO ALTER_THE T=XT,_ AND ALSO THE SOFTWARE "TEXT EDITING "FEATURES PROVIDED BY THE" TEXT EDITING MODE." FINALLY, TO PUT THE EDJJED TEXT ONTO A FILE (WHICH CAN BE THE SAME ONE) " ONE EXECUTES G£T TO <fi, WHICH PUTS EVERYTHING IN THE SCRATCH AREA £ S EL EC T S_A _F I L E_, jAND_ EXECUTES DUMP_______ONE MAY NOT BE ABLE TO BACK UP' "THE" TEXT"" ONTO 'THE SAME FILE AS *if* MAY~H~ATE ALTERED IN LENGTH; HENCE_ THE DUMP PROCEDURE SHOULD ALWAYS BE FOLLOWED. IN ORDER TOWORK MORE EASILY, ESPECIALLY WITH LONG FIL=S, IMTERRUPTS 8__JT0_ _PROVIDE THE FACILITY OF IMPOSING STRUCTURE ONDTHERWISE AMORPHOUS TEXT. THE ' TEXT IS TREATED AS A V=RY LONG STRING OF" ᶜ_ᴴARACTERS_AND CONTROL _CHA_RAC_TERS_.____THE USERCAN NA_M_E__A_NY POINT IN THE TEXT, BY "A 6 CHARACTER "NAME OF HIS"" OWN CHOICE, RY G'ETTING THE TEXT ONTO_ THE SCREEN, PLACING THE CURSOR AT THE POINT ANT) USIMG INTERRUPT 8. ONE CAN MOVE THE POINT REFERENCED 3Y 4~GI VEM NAME 9Y SIMPLY USING *8* AGAIN. ONE CAM REMOVE THE NAME ALTOGFTHER 3Y US ING 9, AND ONE CAN DISPLAY ADIRECTORY OF NAMED POINTS CURRENTLY USED BY_PRESSING INTERRUPT 11._ ONE_CAN _THEN GO JMMEDI ATELY TO ANY NAMED POINT" AND"" WORK "FROM THERE WITH 4 AND "5". A'S THE" TEXT" MOVES RTC'K WAR D~ AND FORWARD, THE SCOPE MONITOR KEEPS TRACK OF TH= LOCATIOMS OF THE

NAMED POINTS! H ACTUALLY PUTS A SCOPE NO-OP COMMAND (NO OPERATION COMMAND) AT THE NAMED POINT. THE USE OF LINE NUMBERS IS CUMBERSOME TO PROGRAM»_ WASTEFUL OF STORAGE SPACE, BUT, MORE IMPORTANT, VERY MISLEADING IF BACKWARD AND FORWARD MOTION AND ARBITRARY INSERTION AND DELETION ARE ALLOWED. HOWEVER, SOME STRUCTURE IS NEEDED, AND THIS HAS BEEN MADE AS FREE AS POSSIBLE.

F USER MANUAL

IT IS HOPED THAT THIS USER MANUAL WILL BE DISPLAYABLE FROM THE SCOPE MONITOR) HOWEVER, THIS IS NOT YET IMPLEMENTED.

G. DRAWING STATE

SEE ~F"iCURE 7. THIS STATE IS INTENDEO TO PROVIOE EXTRA _FACILITIES _FOR__C0NSTRUCTINQ DISPLAY MATERIAL_. _N0ME___0Fj_IT_! S DEBUGGED. ""

INTERRUPT 2 SELECTS A PAGE FOR ATTENTION.

rRTERRUPT"3"TIJT^0NE fNnT"R^TAriW"M'aDr."~rirTITI"SrHIJTTe~~«""'Cr5IE~' TURNS ANALOG KNOB *I,* THE VECTORS ON THE CURRENTLY SELECTED PAGE ARE ROTATED ABOUT THE POSITION OF THE CURSOR.

INTERRUPT 4 PUTS ONE IN TRACKING f|0D<E. THIS PUTS A TRACKING FIGURE ON THF SELECTED PAGE. ONE CAN THEN USE THE LIGHT PEN TCf MOVE THE CURSOR AROUND.

IN fER R "IP T~~T~P UT S~0 N E~ ***HT~CU*** F»VE "D RA WTI W~ff OD^ TirTWTS-ffISETTSr ONE MOVES THF CURSOR WITH THE LIGHT PEN, A CURVE IS DRA *)N PERMANENTLY INTO THE PAGE. -"

DRAWING MODE

PRESS INTERRUPT NUMBER

2 SELECT PAGE

3 ROTATIONAL MODE

4 TRACKING MODE

5 CURVE DRAWING MODE

FIGURE 7

H. USER PROGRAM INTERACTION STATE

ITi—T**'HTS**~STATE; "THE MEANING OF" THEINTERRUPTS"ARE' **AS"D**5**T**.N**?D"** BY THE USER PROGRAM, THE_US PROGRAM DEFINES THEM_8Y CALLING_R25, AND GIVING THE" INTERRUPT ENTRY POINT IN THE PROGRAM. THIS"IS EXPLAINED IN SECTION 6. ONE CAN ONLY GET INTO USER MODE "HILE THE PROGRAM IS ACTUALLY RUNNING.

I. TEXT EDITING STATE

THIS IS A SUBSYSTEM BEING DEVELOPED BY MIKE COLEMAN.

J. ERROR MESSAGES

~ ERROR MESSAGES FROM THE SCOPE MONITOR ARE FEWANO "U'NHELPFUL. IT IS USUALLY POSSIBLE TO RECOVER AND JUST_CARRY ON FROM THE OPTION STATE AFTER AN ERROR. ' " " ' " ' " ":

1. SORRY ROUTINE NOT YET IMPLEMENTED.

2. SORRY NOT ACCEPTABLE."INDICATES AN ARGUMENTIS NOT ACCEPTABLE. USUALLY OUT OF BOUNDS. ATTEMPTS TO USE A PAGE WITH NUMBER NOT IN [1, 4). ATTEMPTS TO READ IN A SCOPE FILE WITH NOTHING__ON IT; ^ATTEMPTS TO ALTER CORE_LOCATIONS NOT IN JSER_CORE WILL .EVOKE THIS MESSAGE. THE STACK IS CLEARED,

3. UNSPECIFIED INTERRUPT. IF ONE PRESSES BUTTONS NOT DEFINED BY THE SYSTEM DISPLAY,

4. MULTIPLE INTERRUPT ERROR WILL OCCUR IF MORE THAN ONE ' INTERRUpT IS REQUESTED: FOR' EXAMPLE," IF °NE IS REQUESTED 3=F0RE"A PREVIOUS ONE HAS BEEN PROCESSED. , ALL INTERRUPT REQJESTS ARE REMOVED, AND YOU MUST REREQUEST.

5. **PANIC. THIS INDICATES THAT YOU HAVE RUM OUT** Or **SPACE, EITHER CORE** sPATE^7TTSc"Tr^fAck~spACE"". rf'TNTt**IALTZE^T-HE^ST**X**CK^** AND **REMOVES CONTINUOUS MODE OPERATIONS. YOU SHOULD BE ABLE** TO **RECOVER, IF IT IS CORE SPACE, DELETING UNWANTED CORE BLOCKS WILL HELP.**

6. ADDROP" <ADDRESS>. THI S SH 0 ULDN' T EVER '~HX=> "PEN". IF IT DOES, WRITE DOWN THE VALUE OF THE ADDRESS AND SEND IT TO **A.** H. BONO, YOU MAY WELL BE ABLE TO RECOVER FROM THIS ERROR CONDITION,

7. USER ERROR* THIS INDICATES AN ERROR HAS OCCURRED IN THE CALLING OF A S ROUTINE BY THE USER PROGRAM. "YOU CAN REMOVE THE ERROR MESSAGE DISPLAY BY GOING BACK TO OPTION STATE MOMENTARILY. AN ERROR CONDT**TTO**TTTS INDICATED TO THE PROGRAM AMD AN **ERR**3R NUMBER IS PASSED TO IT. A LIST IS GIVEN AT_THE END OF_CHAPTER 7. '

8-(SOMETIMES,- AS A RESULT OF A SERIES OF ³**A**RTIALLY RECOVERABLE ERRORS, THE SCOPE MONITOR GRADUALLY DEGENERATES AND EXHIBITS" AN0MALOUS~TE"HAVTOR^———L1KE~SETTiNG RANO0M **PA**TTERTSTTJN T"HE STATE LIGHTS, ETC. IN THIS CASE, IT IS TIME TO RELOAD......ALSO, IF

YOU DO NOT RECOVER CORRECTLY FROM ANY OF THE ERROR CONDITIONS, YOU CAN RELOAD. YOU RELOAD 8Y PHONING THE MACHINE ROOM (EXT, 60) AND ASKING FOR A R.ELOADJ)F THE SCOPE MONITOR AT THE: NEXT C0NYGNIENT TIME. HARDWARE ERRORS OR FAULTS SHOULD BE REPORTED TO THE RESIDENT PHILCO CUSTOM ENGINEERS, C. C. EXT,, 59, WHO ARE IN CHARGE OF HARDWARE MAINTENANCE.

5. PROGRAMMING FOR GRAPHICS.

A TYPICAL OUTPUT DEVICE,LIKE A DISC OR " PRINTER,WITH AN AUTONOMOUS CONTROL UNIT. WORKS AS_FpLLOWS« THE OUTPUT MATERIAL IS PLACED IN A BUFFER WHICH IS PART nF THE ADDRESSABLE CORE»IT MAY J±AVE TO BE A SPECIAL AREA OR CAN BE ANY LOCATION. THIS MATERIAL IS IN BIT PATTERNS CORRESPONDING TO OPERATIONS PERFORMED 9Y THE OUTPUT DEVICEj____THE ^OBV^OUS CASE_ IS_THE CHARACTER.___J_HERE MAY3E OTHERS WHICH CONTROL THE OPERATION OF THE DEVICE LIKE NEW LINE ETC.____THE__TRANSFER___IS THEN_INITIATEDBY THE C^HICH CARRIES ON WITH OTHER TASKS WHILE THE I/O CONTROL UNIT PERFORMS THE I/O TRANSFER,____THE CP AND THE I/O CONTROL UNIT COMMUNICATE EITHER BY MUTUALLY ALTERABLE SENSE SWITCHES OR BY INTERRUPT. THE I/O UNIT WILL SET AN INTERRUPT BIT WHEN READY TO _S.TARJ__AN_D WHEN TRANSMISSION IS COMPLETE TYPICALLY. A CP CAN COMMUNICATE WITH SEVERAL_DEViCES___EACH___HAVING ITS 0WN__CHARACTER SETj _S0_THAT THE INTERNALLY STORED VALUES DO NOT HAVE ANY INTRINSIC EXTERNAL REPRESENTATION,___SUCH REPRESENTATIONS ARE PROPERTIES OF THE I/O DEVICE. INPUT FROM A TYPEWRITER USUALLY TRANSFERS A SINGLE CHARACTER__OR_L11E_OF CHARACTERS TO A BUFFER AND INTERRUPTS THE CP WHICH READS FROM THE BUFFER TO A PACKING" AREA, SCOPES ARE SOMEWHAT MORE GENERAL. THE I/O MATERIAL IS PLACED_ IN _A_BUF_FE_R_ FOR T_HE_SC0PE CONTROLLER TO DISPLAY. MOST OF THIS MATERIAL HAS TO BE"INTERPRETED AS___COMMANDS TO THE SCOPE. MOST SCOPES ARE RANDOM SCAN TUBES. MEANING THAT THE BEAM CAN MOVE EQUALLY EASILY TO ANYWHERE ON THE TUBE FACE ANT) FURTHER IMPLYING THAT ONLY THOSE POINTS EXPLICITLY MENTIONED WILL BE SCANNED. THIS IS IN CONTRAST TO A TELEVISION WHERE EVERY POINT ON THE FACE SCANNED IN TURN. OUR SCO°ES ONLY HAVE TWO LEVELS OF BRIGHTNESS, BUT SOME HAVE FIVE OR MORS, TELEVISION HAS A LARGE RANGE OF BRIGHTNESS AVAILABLE. THJS THF 1/ 0 MATERIAL CONSISTS OF A SERIES OF COMMANDS TO THE' BEAM TO MOVE TO A CERTAIN POINT, DRAW A_ LINE TO ANOTHER POINT, NQW MQVE SOMEWHERE ELSE, NOW DISPLAY A CERT AIN CHARACTER AND SO ON. THERE MAY 3E SPECIAL BITS FOR BLANKING CERTAIN ELEMENT S, ALTERING THE SIZE ETC. ALSO THE SCOPES OUTPUT FUNCTION IS A REGENERATIVE PRO CESS AND WE HAVE TO INSTRUCT THE BEAM TO DO THE SAME SCAN SEVERAL TIMES A SECOND TO GIVE A CONTINUOUS DISPLAY. THJS A TYPICAL BLOCK USUALLY OF WORDS OF GRAPHIC I/O MATERIAL IS SOMETHING LIKE THISI

```
     AI     START SCANNING HERE
            MOVE TO X0/Y0
            DRAW LINE TO X1.Y1
            DRAW LINE TO X2,Y2
            MOVE TO X3,Y3
            DRAW CHARACTER NO 32
            LOQP_BACK jOJf.1
```

OUR SCOPES HAVE A VERY NICE WAY 0F SCANNING» THc DISPLAY MATERIAL }§ SETUP AS RP L0CATA8LE BLOCKS WITH TRANSFER COMMANDS MHICH MUST CONNECT UP TO GIVE A LOOP AROUND WHICH THE SCOPE SCANNER OPERATES, THF DISPLAY MATERIAL MUST_BE IN ONE SP ECIAL REGION OF ADDRESSABLE

G-2Q__CORE__VI_7_.____/16000p_TO /J.7J22⁷i___THE_ADOR=SSES USE D__BY THE_ SCOPE SCANNER " ARE RELATIVE TO"/160~o6o AND THEREFOR£ RAN3E FROM 0 ᵀᵒ_ /17 777. EACH OF THE THREE SCOPES CAN HAVE 4 PAfleg AND INDEED EACH PICTURE IN THE CORE IS A SEPARATE MODULE OF DISPLAY MATERIAL. THE LAYOUT IS SOMETHING LIKE THISI __ _

```
            Al        "DELIMIT A2 PAGE 1. SCOPE "i
_____________________DISPLAY_MATERIAL_____________________
                      STORE COMMAND
__________A2          DELIMIT_A2 PAGE 2 SCOPE 1 AND 3       _   _
                      DISPLAY MATERIAL
_______               STORE COMMAND
            A3CYCLE TO AI "  '  "  "  '
```

THE SCANNER ENTERS A M_ODUL_E,__ REMEMBERS_T_HE FIRST WORD, UNTIL IT HITS A STORE COMMAND, THEN JUMPS TQ~ THE ADDRESS MENfI0N50 IN THE FIRST kiORD. FACH DISPLAY MODULE CAN BE DISPLAYED OJ^ ONE OF PAGES iTHROUGH 4 ON ANY COMBINATION OF SCOPES 1, 2, AND 3. THr "DISPLAY MATERIAL _CA_N BE CHANGED BY A PROGRAM FREELY ALTHOUGH ONE SHOULD ALWAYS PRESENT A WELL-FORMED DISPLAY TO THE SCANNER", INPUT OR CORRECTION OF DISPLAY MATERIAL FROM_THE HUMAN Aj; THE SCQ³E CAN BE ACHIEVED USING' THE KEYBOa"RD OR RAND TABLET THE INPUT 07 Rf'TS INTO THE ADDRESSBLE MEMORY IS DONE BY THE SCANNER AS IT SCANS ROUND, ONE OF THE ADVANTAGES OF THE MODULAR LAYOUT IS THAT'NEW MATERIAL IS SIMPLY APPENDED TO THE END OF THE APPROPRIATE MODULE AND f"H"E"""STO" RE" COMMAND MOVED DOWN, THE SCANNER WILL KE=p ADDING _NEW___MATERIAL AS_REQUESTED U_NTH. IT HIJTS AG/UNST THJ? NEXT DELIMIT AT WHICH TIME" 11 WILL GENERATE A MEMORY F"ULL INTSRRUPT, ~~TO~~TIFYING THE SCOPE MONITOR, AND WILL_REFUSE TO ENTER ANY MORE. INPUT ACTUALLY WILL BE PLACED IN ANY DISPLAY MODULE DeS13NATED AS ENAB_LED FOR THa T SCOPE AND THAT INPUT DEVICE. THE DESIGNATION IS BY MEANS OF CERTAIN BITS IN T HE DELIMIT WORD. "THERE IS ONE 31T TO ENABLE___THE MODULE FOR ALL_ENTRY, VECTORS_AND CHARACTERS _FROM_ANY_ ~OF"""THE 'SCOPESDESIGNATED, AND" TWO OTHER BITS "FOR" THE" KsYSJSA RDS FOR THE PARTICULAR!_SCOPE_.ₗ_ THE FULL DELIMITCOMMAND IS____________

1 2> 0		Abbess		A	LLt'		SCOPE	DC LIMIT

10 i B 4' m- 2 %. 1 0

THE PAGE FIELD IS 2 BITS,SO CAN BE 0=4,1,2,3, A INDICATES ALTERNATE MOOE-USUALL"Y ONLY USED BY SCOPE MONITOR," = IS THE GENERAL ENABLE BIT. KEY IS 2 BITS ONE F OR EACH KFYRO*=?D, FOUR CONSOLES ARE PROVIDED FOR BUT ONLY 3 INSTALLED. IF SEVER AL MODULES A.RE _FNABLED FOR THE_SAME DEVICE^ THE _HPUT MATERIAL_WJLL BE ENTERED IN ALL OF THEM. THE NORMAL USER NEVER" SEES OR HAS TO BOTHER WITH THE DELIMIT, STORE OR CYCLE COMMANDS, THESE ARE MANAGED FOR HIM BY THE B ROUTINES. IT IS ARRANGED AS A SET OF STRINGS,EITHER CHARACTER STRINGS OR VECTOR STRINGS WITH A HEADER COMMAND AT THF FRONT TO INDICATE THE STARTING POINT ON THE SCRFEN. THUS A DISPLAY OF LINES AND CHARACTERS IS JLI_KE _TH*IS*________________

```
        ..................mfader"Yo"xo
                          VECTOR INCREMENT Yj. XI
```

VECTOR INCREMENT Y2 X2
HEADER Y3 *3
CHARACTERS CI C2 C3__
CHARACTERS C4 C5 C6 ~
STORE _
THE ACTUAL FORM OF THESE WORDS fS AS FOLLOWS

	11	1	Y_0		**HEADER**

10

	1 1 0	|*Y		T ! 8	A Y	A X	**Vecron.** XNCIL_«e>«RR

1\ 1* XI T.1 ID n Id •i 0

SY, SX ARE SIGN BITS, 8 IS THE BLANKING BIT, IF SET THE VECTOR 1'NC RE MEN T IS 1 N V ~ I SI SLE. T IS T H E TAG B IT, IF S_T THEN W ILL~BLIN K " OR INTENSIFY IF BLINK OR INTEN SIFY SWITCHES ARE SET,

10	0 1	Ti	£|	C2	It	Si	C 3	.3		

3T 1* %\ in \% II to 4 * f 0

T IS TAG BIT'S IS SIZE BIT, IF SET CHARACTER IS D0U3LE SIZE, CHARACTERS ARE SCOPECHARACTERSET AS ~ GIVEN I\ THE HARDWARE MANUAL. IN A ALGOL-20 A WAY TO SET UP THESE WORDS IS SOMETHING LIKE

```
HEADER «• 8L3 + YO * 8R 4000 + X0 >
VECTJNC * 8L120 + SY « 2t23 + SX*2t22
          •• T # 2*21 + B#2t20
          + DELX»2tiO + DELXl
CHARS - 8L2 • <C1 * 4 + Tl*2+Sl>
          »2t20
          * (C2»4 *T2»2 +S2) »2tlQ
          * TCJiT + T3*2*§3TV
```

SUCH MATERIAL SHOULD BE PACKED INTO AN ALGOL LOGIC ARRAY IN NORMAL "CORE—AND -THEN—83 C AL~LED "Td MOVE I T~T0~~A"TTEST37\ITTETTTTalT TN~TH~F DI SPLAY AREAi OTHER SCOPE OPERA TIONS AVAILABLE ARE

	1 3 0				X	LfcFr

		1 0		6		K	

a_0

ijthTch Tet mTrglns"fo^m^ after them in the module.

	r o		C		COMFTW.fr

2i 1 I <?

THIS SETS___A_ COM PAR E_T R AP. J_N A JC E RTAIN CHARACTER KEYBOARD ENPINQ AFTER THIS WORD IN THIS MODULE. IF THIS CHARACTER IS TYPED IN, IT IS ENTERED -IN .THE MODULE AND AND INTERRUPT IS GENERATED BY THE SCANNER. THIS TS PROCESSED BY THE SCOPE MON.TO R AND CAN 3E PASSED TO A USER PROGRAM ENTRY POINT IF DESIRED.

NO-oP

0 I

31 11

NO-OP_ HAS NO EFFFCT ON THE SCANNER, CAN BE USED FOR KEEPING INFORMATION AROUT TH E PICTURE WITH THE PICTURE TOR CONVIENT PHO_LSS_G,_ NOTE THAT__DOUB_LE____SIZE CHARA CTERS A^E OBTAINED 3Y SETTING THE S17E BIT, BUT SUBSCRIPT SIZE CHARACTERS BY INC LUDING THE^ SUBSCRIPT SHI-T CHARACTER IN THE STRING. AN EXAMPLE OF A PROGRAM SEG MENT IN ALGOL-20 FOR DISPLAYING A S'JUARE SIDE 100 START ING_ AT (200,500).

```
                LOGIC ARRAY Ati:20M
                A C.1.J.-.HEA0ER(20n».5.nO).I
                A C2I-VECTORtIOO,0 , 0 ) j
......          A [3.1-VECTOR t 0,100,0) I
                At4].-VECTOR(-100#Of O) t
..   _ _        AI53-VECTOR(0,-100,0) J
                B(3,L0C(AtI)* PAGE. 0,0,0);
```

TO ALSO PUT THF. WORD 'SQUARE' AT (200,300) WE CAM U5£ BO .TO CONVERT TO THE SCOPF CHARACTER CONVENTIONS

```
                CtIJ-'SOUA' ;
                C(2I-'RE ' ;
                B(0»LnC(CriI ), LOC(At_7JI_6,0»0)I
                A [6 I *-HEADER ( 20 0 , 300 ))
              _B<3> L0CtA_t61 )j_P_AGE,J),_0)j
```

USING THE SCOPE ALGOL LIBRARY, ONE DOESN'T HAVE TO USE THIS GENERALITY IN SETT IN G UP. A DISPLAY..... ..THE S AHr . DISPLAY COULD. 3E. ACHIEVED BY

```
      ___        LINE< 200 , 300 , 300 , 50 0 )1
                LI NEC 30 0,50 0 , 30 0 , 40 0 )\
                LINE (.100., 400 ,.300^400 )j
                LINE(200,400, 3 00,500)J'
            _    AtU«-'SQUA'J
                A(2I«-'RE 'i
....            B(2,L0C(AtH >, 2. 200 , 300 ,PAGE) J
```

BUT WOULD BE MORE WASTEFUL, AS EACH CALL OF LINE PRODUCES A HEADER _AN_D__VECTOR J N j.REMEM T. _WP_R D_

6. WRITING INTERACTIVE PROGRAMS.

A• TH^ P *OUT*NES

ALL INTERACT ION ' BETWEEN PROGRAM AND ' THE""SCOPES IS ACCOMPLISHED BY USING THE B ROUTINES. THE B ROUTINES ARE ACCESSED THROUGH A SPECIAL INTERFACING ROUTINE. THESE MECHANISMS NEED NFVER BOTHER___THE___AVERAGE- USER, IF HE SJMP_LY_USFS JHE_C_OPJES 0/ THE 9^ ROUTINES IN THE SCOPE SUBPROGRAM LIBRARY (SECTION 7> . " THUS IN _ALQOL-eo_01_ FORMULA ALGOL *OHE* SIMPLY WRITER _ ________

B (BNUM, ARG1, ARG2, ARU3. ARQ4, ARG55J AND IN SPITE

__________B BNUM, ARGj, ARGg, 8NUH IS THE NUMBER OF THE 3J*OUTP(E _ REQUIRED. NOT ALL ARGUMENTS ARE USED FOR ALL B ROUTINEST MOST OF~ _____THEM HAVE TO DO WITH PASSING INFORMATION FROM THE PROGRAM TO THE SCOPES, BUT A FEW GO THE OTHER WAYI E.G., 88, WHICH rTa*D~S THE ANALOG KNOB_S_._ IN THEJ3ESCR I_PT I ON_OF THE 8 ROUTINES, ARG1, ETC, . ARE ogNOTED BY R52 ETC. I T I S TO BE NOTED THAT THE VALUrS OF THE ___ARGUMENTS _AR| CHANGED BY A CALL ON A B ROUTINE, AND THIS CAN LEAD TO HAVOC T E.G.,"" CALLING 88 WITH ZEROSFOR A R G U vj ENTS" *A* AND 5 WI L L _____CAUSE THE ALGOL CONSTANT D TO BE REPLACED BY ANOTHER VALUE. TO AVOID THIS ONE CAN USE A GLOBAL BOOLEAN VARIABLE OUT, TF 0TJTT5 F A L_S_E1__T HEN_NO__OUTPUT_OF VALUES WILL_OCCUR,_ A ME>_N0 OV ERWRI T | N Q _ WILL OCCUR, IF' OUT IS TRUE, OUTPUT WILL OCCUR, AND, IN H i S CASE, ____ONE CAN PUT SOME DUMMY ARGUMENTS_ IJ<_T J E _PARAMETER LIST. IF THE___ LOCATION OF SOME DATA IN AN ARRAY' OR SCALAR IPENTTFTER IS NEEDED, _____ONE MUST USE LIBRARY INTEGER PROCEDURE LOC IN ALGOL OR FQRML, WHTfCH FINDS THE ADDRESS WHERE THE ACTUAL VALU£TTRT"!fTI5"RTin TTfTJS ____LOC (A til) IS THE _AD_DRE_SS_ CONTAINING _THE_VA_LUE OF_AriJ^LOC IS__IN___ THE "S COPE LIB RaRY FOR A L G 0 L " 0 R " FO R M L ." I F YO U NEED THE L 0"CATI ON OF A PROCEDURE ENTRY PO_I_NT __0R LABEL, YOU USE _T_HE L1 9R AR YJ NTEGER PROCEDURES PROpLOC oR LABELLOC RESPECTIVELY IN" AC30L-20.ALL' THE " ______ARGUMENTS TO 9 A R_E_J_NT EG E_RS,_IF AN ERROR OCCURS ON CALLING B DUE___ nr^IN'CORRE'cT ARGUMENTS". 9 wTIL PRINT AN ERROR M^JJ'SA~GT~TfJD SET BNUM* -1. THE REASONS FOR ERRORS ARE DESCRIBED IN D E T A I L_ I N THE DESCRIPTION OF THE B ROUTINES IN SECTION'6. THE DETAIuS OF THE INTERFACE ARE GIVEN IN SECTION 10. IN ADDITION, AND ON A HIGHER___ ———Level than the b routines."there are several useful subprograms in ALGOL,___FORML AND SPITE IN THE LIBRARY FOR DOING HIGHER LFVEL___ ———TASKS"'; FOR F"XA"MPLET"" PR0"CEDURE~NUM (X, Y, N> "WILL'TaTp A REAL VARIABLE N ANO DISPLAY IT AT X,Y IN -5D.3Z (OR 58.3) FORMAT. THE FULL I/O FACILITIES OF ALGOL-20 CAN RE USED IN READING "ROM AN0 •PRINTING* TO THE DISPLAY PAGE. THIS IS SIMPLY ACHIEVED JSING THE "' SCOPE LIBRARY PROCEDURES READ.PAGE AND PRINT,DN,3aGE ,4HICH ARE EXACTLY ANALOGOUS TO READC<W>) AND PRINT(<W>) . E.G._ _ R E A D . P AJ36___ ——READS A" CARD" ^ROM "fHE ' SCOPE FACE INTO A BUFFER, "WTCH" "CaN"THEN 3E READ IN THE USUAL WAY WITH A READ STATEMENT. AT PRESENT, A PROGRAM jCAN___ONLY INTERACT WITH THE SCOPES IF IT HAS BEEN SUBMITTED FROM A SCOPE AND IF THE" JOBCARD- USER IS LOGGED IN ON THAT SCOPE. AT THE TERMINATION OF A USER PROGRAM CONTROL GOES, TO Ifl AND THENCE BACK TO SCOPE MONITOR TO ALLOW IT TO UNSET ALL THE SWITCHES SET BY THE ..PROGRAM . HENCE.....THE USER SHOULD NOT PATCH I 0. . . . .

BEFORE INTERACTION CAN OCCUR THE PROGRAM MUST ANNOUNCE ITSELF BY CALLING.B-I,. . . . . - .

TO DisPLAYTEXT. "" ONE'S PROGRAM WILL NORMALLY SET IT **UP** IN 0-2.0____JiHAJ^CTERjSj^JJO___QNE_ HAS TO CONVERT TO SC3PE CHARACTERS AND MOVE IT TO THE SCOPE DISPLAY REGION.

_B_ 0 AND_9 1 WILL_C__NVERT TEXT QETWEEN_G-20 CHARACTER SET AND SCOPE CHARACTER SET.

B 2 CONVERTS TEXT___A_ND_ MOVES *IT_J0J11* S?LAY _REG_I_OM IN ONE OPERATION.

B 3___MOVFS A REGION ALREADY IN SCOPE FORMAT TO TH_ DISPLAY REGION.

TO DISPLAY VECTORS, ONE MUST SET THEM U' IN A LOGIC ARRAY AND USE B 3. **ONE CAN EASILY SET UP A DESIRED LOGIC ARRAY USING PROCEDURES HEADER, VECTOR, LINE, CURVE, ETC** Big, B16, B17, B18

oKrE - ¥ Q st - - ReseRvE SPACE IN THE DISPLAY AH_A SY "CAC'LINS" **B15**J THE PAGE DOES NOT NEED TO BE ENABLED FOR THE PROGRAM TO ENTER DISPLAY MATERIAL BUT NEEDS TO BE ENABLED FOR THE HUMAN USER TO ENTER DISPLAY MATERIAL.

" " B16, 617. B18 ENABLE, DISENABLE AND DELETEA PAGE RESPECTIVELY. . . . .

B19 APPENDS ONE PAGE TO ANOTHER

820 DI**SENABLESALL PAGES.**

B"**2** AND **B3** ACTUALLY APPIND NEW DfSPLAY"~"MXTERITL TO fHE EXISTING PAGE,. . . . .

_B28^CLEARS *A* PAGE. **B4** AND B5 PERFORM RECIPROCAL OPERATIONS TO 8**2** AND B3 IN COPYING DISPLAY MATERIAL FROM A GIVEN PAGE INTO A GH/EN ARF_AY_ I NTHE USER PROGRAM^

B4 CONVERTS ALL TEXT TO G-21 CHARACfE~R~STT AMO IGNORES ALL VECTORS. THE.ARRAY ;C0ULD THEN BE PRINTED OUT IN A FORMAT,

Qg - c 0 p l ? s WITHOUT "CONVERS ION, A PROGRAM"CAN "ONLY DEDUCE INFORMATION AROUT THE PI SPLAY BY COPYING IT INTO AN ARRAY AND SEARCHING "THE AREA FOR FEATURES LIKE KEYWORDS,

86, B7, B8. Blf], 911 PROVIDE COMMUNICATION WITH THE CURSOR, ANALOG KNOBS AND USER STATE SWITCHES,

Bft READS THE CURSOR.

B7 SETS THE CURSOR.

"B* READS THEANALOG KNOb"S AND STATE

SWITCHES.

"BIT'READS THE STATE SWITCHES ONLY.

BIl SETS THE STATE SWITCHES,

B. USER INTERRUPTS

B12,813,822,824 ARE FOR COMPARE INTERRUPTS, BL3 DEFINES THE USER ENTRY POINT TOBE ENTERED WHENA COMPAREINTERRUPT OCCURS ON ANY CHARACTER. THIS OCCURS IN ANY STATE OF THE SCOPE MONITOR, EXCEPT DURING TYPING INTO THE SCOPE MONITOR, WHICH USES A COMPARE CHARACTER. B12 SETS COMPARE INTERRUPT ON A SPECJFIED CHARACTER FOR A SPECIFIED PAGE. B22 RESETS THE COMPARE ROUTINE TO THE STANDARD SCOPE MONITOR ROUTINE. B2L REMOVES COMPARE ON A SPECIFIED CHARACTER ON A SPECIFIED PAGE, 824 SETS AN ENABLED CURSOR AND IJNT_ERRUPT R0UTINE ON A SPECIFIED CHARACTER ,

< I I) B14", 823 ARE FOR THE MEMORY FULL INTERRUPT. "B~ 14" SETS THE USER ENTRY °QLNT WHICH IS ENTERED ON MEMORY FULL. 923 RESETS MEMORY FULL ROUTINE TO THE STANDARD SCOPE MONITOR ROUTINE.

< I 11) B2_5_ DEFINES _THE_ USER ENTRY POINT_ FOR THE_INTERRU^{3}_T BUTTONS 1-15. AFTER B?5 HAS BEEN EXECUTED AND PROVIDED THE SCO'E MONITOR _IS__IN_USER MODE^ THE INTERRUPT BUTTONS WILL_CAUSE AN INTERRUPT IN THE USER PROGRAM AND FOR" CONTROL TO BE PASSED TO THE SPECIFIED_PR0_CPJDU_RE OR ENTRY POINT.

INTERRUPTING USER PROGRAMS

ALGOL PROGRAMS

IN ALGOL-?0I THE ENTRY POINT OF_ A PROCEDURE_OR THE LOCATION OF A LABEL CAN BE USEDAS THE USER INTERRUPT ENTRY POP:T, THE CODE FOLLOWING WI LL_ N0RMALLY_MAKE_ DECISIONS_A BOUT THE C0MPUTATION AND CAN BE CALLED "THE USER 1NTERRUP'T SERVICE ROUTINE (UFSR). A SIMPLE ^{WA}_Y_ TO ADEFINE THE IMTERRUPTS AND ENTRY POINT IS:

```
WH              L8L        T3;       "                                        "
WH              CLA   0    TIL
AL    V5*ACC}                                      "   ~   "
AL B<25, V9,ETC) t
```

THE INTERRUPT ENTRY POINT WOULD THEN OCCUR AT THE BOTTOM OF THE PROGRAM AND BE

```
WH  TL          ENT       J        """"
AL    ETC
```

THIS CAN 8E DONE SEPARATELY FOR INTERRUPTS FROM THE BUTTONSAMD FROM_ COMPARE CHARACTERS. THE ENTRY_POINT IS_TRANSFERED TO WITH A TRE INSTRUCTION SO THAT CONTROL WILL BE ON IN THE"~U ISR "UNLESS THE FIRST INSTRUCTION AFTER THE ENTRY POINT IS

WH EXR Q /77776,CE*

IF YOU WISH TO RETURN TO THE INTERRUPTED COMPUTATION* YOU HAVF TO POP THE MAIN MONITOR INTERRUPT STACK. THIS CAN BE DONE BY RETURNING THROUGH YOUR MARK(A PROCEDURE END IN ALGOL)i SINCE THE SCJJPE MONITOR PUTS ITS OWN INTERRUPT ENTRY MARK IN THE USER INTERRUPT ENTRY POINT. IF CONTROL HAS WANDERED AROUNTHFORE THE DECISION TO RETURN TO THE INTERRUPTED COMPUTATION HAS BEEN MADE. VOL) MAY EFFECT THE RETURN BY RETURNING TO THE SCOPE MONITOR* HOWEVER, THE _SWITCHES WILL BE UNSET BY THE CLOCK INTERRJJPTS WHILE IN THE UISR, SO WE PROVIDE A B ROUTINE (327) TO SET UP THE SWITCHES CORRECTLY AND RETURN CONTROL TO THE MAIN MONITOR. IF YOU WISH TO PASS^ CONTROL J0~AN0THER~"P01 NT TN"Th1~PR0QRAM7~XG0"To STATEMENT WILL EFFECT ALL THE STACK PUSHING REQUIRED SO THAT THE VARIABLES ARE CORRECTLY REDEFINED.

PRINTING HAS TO BE CAREFULLY CONTROLLED IN ALGOL PROGRAMS WHICH ALLOW INTERRUPTS. IFAN INTERRUPT OCCURS DURING PRINTING AND THE UISR PRINTS, THEN IT IS MORE OR LFSS IMPOSSIBLE TO RETURN TO THE ORIGINAL LINE OF COMPUTATION. ONE CAN SAVE THE INTERRUPT PRINT LINE AND PRINT LINE POINTER BY ; _ _ _ _

LOGIC ARRAY AI-2H201; TEMpV|25Il BUFFERSET(•PRINTi,A/0I)I

MP_ RESTORE IT LATER. HOWEVER, IF THE NAME AND PRINT STATEMENTS IN OPERATION HAVE 8EEN CHANGED» ONE CANNOT RECOVER *fHWW]* TF f*E UISR PASSES CONTROL TO ANOTHER PART OF THE PROGRAM AND NEVER WISHES TQ CONTINUE AT THE INTERRUPTED POINT, THEN THE PRINTING WILL WORK OUT ALL RIGHT. SOME: SAFE RULES ARE (1) DON'T PRJNT OUT IN THE UISR, (2) TURN OFF THE CONTROL SWITCH DURING PR"INTINQj BUT, AS DISCUSSED, THESE ARE NOT RIGID RULES.

EXAMPLE **OF** INTERRUPT DEFINITION IN ALGOL

```
I.  USING A LABEL
    BEGIN INTEGER ENPT,CSW,IN,SN,CC»
Wfl                 LBL         tn
WH               •  CLA    0    TI;
AL—ENPT-ACCJ
    B(25,ENPT>L0C(CSWi,L0C(I   N),LOC(SN),LOC<CC))J
————LOOPI  CONTINUOUS  ACTIONS>~J  GO~TO~LOOPI

WH TI               ENT         ;                     INTERRUPT ENTRY POINT
AL   ,                                                                    1
————<ACTIONS>,   QO^O~NEl*WrONl
WH                  TRA    1    Til                   RETURN TO INTERRUPTED
                                                      C0HPUTATIW
AL  i                            ;              ;
    END*
```

```
I 1. .USING A PROCEDURE
     BEGIN
 ____LIBRARY^PROCEDURE PROCL.OCJ
     PROCEDURE UJSRI
   _<ACTI0NS>J GO TO NEWACTION)       _          _   _
     END GOES BACK TO INTERRUPTED ACTION)
_____BI?JL.PKOC^
     <CONTINOUS ACTIO^S> }
```

£CR_kA___ALGOL PROGRAMS

FORMULA _ALGOL COMPILES CODE WHICH IS HEAVILY DEPENDENT ON RUN-TIME ROUTINES. IF ANY RUN-TIME ROUTINE IS INTERRUPTED BY THE _SC0PE___MONITOR___WHICH__THEN_CALLS THE UISJ_WHICH_IN TURN CALLS THE INTERRUPTED RUN-TIME ROUTINE,THEN GLOBAL PARAMETERS^LI<E RETURN MARKS, INDEX REGISTERS AND TEMPS) ARE SOON FORGOTTEN, THE_R_EFORE THE_ ONLY CODE WHICH CAN BE USED WITHOUT DRASTIC SAFEGUARDS IN THE UISR IN FORMULA ALGOL IS CODE WHICH DOES NOT CALL ON RUN-TIME_ ROUTJMES__ HOWEVER IF THE UISR AND THE CODE FOLLOWING THE CALL ON 325 ARE COMPLETELY___INnEPENpEjNT___AND_ DO NOT CALL ON THE SAME ROUTINES THEN ONE HAS MORE FREEDOM. 0~PE~RATIONS WHICH DO NOT USE THE RUN TIME ROUTINES INCLUDE STORING.AND ACCESSING OF SIMPLE^VARIABLES(BUT N0T_ ARRAY ELEMENTS),AND THE OPERATIONS +,-,»,/»*,*,-,IF THEN ELSE, J3IGN,ABS,ENTIER, AND GO TOtLOCAL BACKWARD___TRANSFERS___ONLYJ^^ HOWEVER, WITH INTIMATE KNOWLEDGE OF FORMULA ALGOL AND A LISTING OF ITS RUN-_Tj_ME_R0UTINES_,__THE_EXPER.LENCED_USER_CAN BUILD HIS UISR SO THAT IT CAN CALL ON ANYTHING, THIS WOULD PROBABLY BE: DONE 3Y WRITING_ SMALL MACHINE CODE ROUTINES, CALLABLE ONLY_WITH_ CONTROL __ OFF, WHICH WOULD SAVE AND RESTORE THE CONTENTS OF A LIST OF" J1AC_H_INE_ LOCATIONS. THE UISR WOULD PROBABLY LOOK LKE TH13^____

EXAMPLE OF INTERRUPT DEFINITION IN FORMULA ALGOL

```
SN  CDLC   0

     PROCEDURE  INTERACT}  BEGIN  INTEGER »LOCISR)

SN  CMPL   07200          ERA       NC    READ NEXT COMMAND REGISTER

SN  CMPL   0050000000     ADD   0   5     NUMBER OF INTERVENING COMMANDS

SN  CMPL   1330011000     STI           UISR  SAVE LOCATION OF UISR

~S"N"~CMPT 1337700Oil'    "Sff          LOCTSfTuSED IN~CALL IN𝐼𝐼 325
```

```
      GO AROUND'

SN CMPL  Q                  UISR  ENTRY  POINT

SN CMPL   0760067776        EXR  Q     /77777-S13-$0   t
                                         TURN OFF CONTROL. AND H MOD
SN CMPL   3770011001        JJRM        SAVE    SAVE VARIABLES

      CODE PREFERABLY -HITH CONTROL OFF

SN CMPL   3770011002        TRM         RESTORE  RESTORE VARIABLES

SN CMPL   6370011000        TRE  3   UISR  GO BACK TO MONITOR

   GROUND I JF J < 25,' LOC I SR» E TC )  THEN  PRINTt . CANT- . INTERACT) t
      END IS TO INTERACT;
```

THE ABOVE CODE AND PARAGRAPH ON THE INTERRUPTION! OF FORMULA ALGOL PROGRAMS IS BY RUDY KRUTAR WHO SHOULD BE CONSULTED ON ALL' RELATED MATTERS. IN FORML, THE PRINT ROUTINES ARE RECURSIVE AND THEIR VARIABLES ARE IN THE GENERAL COMMUNAL RECURSION STACK, THUS, IT SEEMS THAT ONE CAN ONLY PRINT IF THE UISR DOES NOT PRINT, AND IF IT ALWAYS RETURNS TO THE INTERRUPTED COMPUTATION. ONE SHOULD NOT INTERRUPT DURING CALLS ON MAIN MONITOR ROUTINES, IF ONE IS GOING TO USE THEM IN THE UISR, AND THEN TRY TO RETURN TO THE INTERRUPTED COMPUTATION.

C. INTERACTION WITH MORE THAN ONE SCOPE

TO INTERACT WITH A DIFFERENT SCOPE FROM"THE ONE SUPMjTTED FROM, A PROGRAM SIMPLY USES THE B ROUTINES AS USUAL, BUT IN ADDITION SETS THE "SCOPE NUMBER BY USING ALGOL PR0CEDURE~ SETSCOPENUM(N) . IT DOES NOT NEED TO BE SET BEFOR. EV5RY CALL OF A BPOUTINE, JUST ONCE.
THUS, TO READ THE STATE SWITCHES ON SCOPE 2. ONE PERFORMS
NSAVE-SCOPENUM. SETSC0PENUMC2) J ZERO.-0I OUT.-TRUEI
_B(10_, ZERO, S_TSw, DUM, DUM, DUXJJ_ 0UT-FALSEJ SETSC0PEMUM(MSAVE);

rvOTE WE SAVED TH. NUMBER OF THE SUBMISSION SCOPE BY USING ROUTINE _SCOPENIJM. SETSCQPENUM AND SCOPENUM MERELY SET AND _ READ INDEX REGISTER -51.

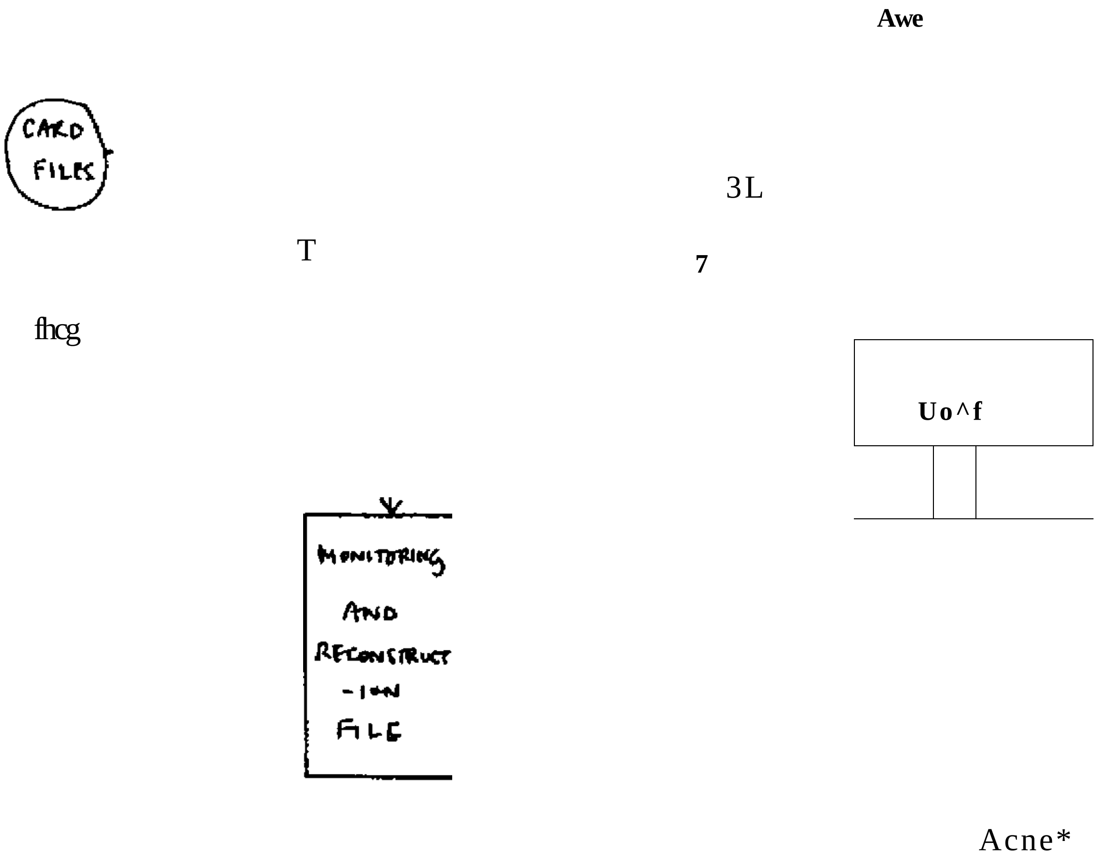
ONE CAN THUS DO ALL THE USUAL INTERACTIONS WITH'ANY OTHER ~SCOPE, HONEVER' ,THE B R0UTINE CALLS""WILL ALL GIV= " ERROR' EXITSUNLESS PERMISSION To INTERACT HAS BEEN GIVEN_3Y THE USER AT THE SCOPE TO BE INTERACTED WITH, BY USIN3 I\'T5RRU^{3}f _J-3„.IN_JHE PROGRAM STATE. .
USER INTERRUPTS FOR ANOTHER SCOPE ARE DEFINED BY USHG B25. UPON ANY USER INTERRUPT, THE SCO_P_E_ NUMBER IS PASSED TO.....TH___USER PROGRAM,

F_Ij_ALLY_ ONE_ CA_N DJSjPLAY jf__QIVEN_P_AGE ON MORE THAN ONE SCOPE, BY USING R26. THIS TAKES PARAMETER RS2. WHICH "'TS THE BIT PATTERN AT THE END OF THE DELIMIT CONSISTING OF 4 91T5 FOR SCOPES 3.2,1 AMD 4 RESPECTIVELY,SN CORRESPONDING TO SCOPE M. THUS IF A PROGRAM WANTED TO DISPLAY A PaGF ON TH= SUBMISSION SCOPE AND ANOTHER SCOPE N, IT WOULD DO SOMETHING LIK^ THIS
NI«-SCOPENUM; _BITPaTj-21 N1-?tN1 B(26_J_AGE_RITp_A_T, 0, 0_jn_

OF" COURSE", IF PERMISSION HAS NOT REEN GIVEN, IT WJ.LL.._NOT. D.I SPLAY,

D. OVERALL CONTROL STRUCTURE.

MIGHT BE LIKE THIS.



7. DESCRIPTION OF THE B ROUTINES.

B-1 ANNOUNCE AN ONLINE USER
_ PARAMETERS! _ USAGE NUMBER FROM UOG-IN_ _ __
OUTPUT: R5i: SCOPE NUMBER
ERROR_IFj LL) „NOT_SUBMITTErj
FROM A SCOPE
OR (ID NOT LOGGED IN _
PERMITS A PROGRAM TO INTERACT WITH SCOPE MONITOR. USER
.. . SHOULD NOT PATCH 10. DO NOT CALL B<-1) TWICE IN THE SAME __
RUN

BO C0NVE.RT.S . G21 _CHARACTER.._STRJNGS_PACKED
4 PER WORD INTO SCOPE STRINGS PACKED
IN. DISPLAY .FORMAT.
PAHAMETERSI R52*L0CATI0N OF FIRST
WORD.. OF. G2X TEXT,. .
R53.-L0CATI0N TO RECEIVE FIRST WORD
0F__THE CONVERTED TEXT 3LOC*.
THIS ADDRESS MUST BE IN USER CORE.
R?4-NUMBER OF CHARACTERS TO CONVERT. _
OUTPUT! NONE
ERROR IF! R<53 OUT OF BOUNDS.

BI CONVERjS_SCOPE_ CHARACTER_SJ_RJNGS
PACKED IN DISPLAY FORMAT INTO G21
CHARACTERS PACKED 4 PER_WORD.
PARAMETERS 1 R52«-BASE OF SCOPE STRING,
_R53*-LOCATION.TO RECEIVE FUST
WORD OF THE G21 STRING.
THIS„ADDRESS_MUSTBEIN
USER CORE.
_ _ R54«-LENGTH OF SCOPE STRING __
IN WORDS. " "
OUTPUT) __ NONE
ERRORIF! "R53 OUT OF BOUNDS,

82 cTN^EWTETTWD^PPEI^fO^QE
N AT_ POSITION (X, Y) .
PARAMETERS! R52«-RASE OF G21 TEXT " " ~
.. _ PACKED FOR 4 ᴾER WORD. _
R53-LENGTH OF TEXT IN WORDS,
R54-X
R55.-Y
R56.-PAGE NUMBER.
OUTPUT• NONE
ERROR IF! (I) PAGE ALREADY FULL
(II) STRING TOO LONG. 7"'
DISPLAYS TEXT ON SCOPE F_ACE._ ONE MUST 1*^E_ REQUESTED
AVAILABLE SPACE FOR THF PAGE IN QUESTION IN ADVANCE OF '
CALLING aa.

B3 APPEND A LOGI'C BLOCK OF
(ALREj_DY_C_0NVERXEDJ_DJ SPLAY
MATERIAL TO ' PAGE N. "
PARAMETERSI__ "52^8ASE_0F_BL0CK_
TO BE MOVED.
R53*PAGE NUMBER.
OUTPUT! NONE.
„ ERROR IF! _ (JJ NO STORE IN PAGE.
OR (ID ATTEMPT TO OVERLAY
A DELIMIT.

MOVES A BLOCK OF SCOPE COMMANDS INTO *THr.* H-WDTJLE AND APPENDS IT TO AN EXISTING PAGE. CURRENTLY IT ONLY TESTS FOR DELIMITS AND CYCLESJ OTHER ILLEGAL CONSTRUCTIONS ARE NOT_ TESTED FOR) THUS, THE DISPLAY SHOJLD BE WELL FORMED AND SHOULD INCLUDE A STORE.

__ MOVE PAGE N TO USER "CORE, CONVERTING ~
ALL SCOPE CHARACTERS.
PARAMETERSI R52.-RAGE NUMBER,
R53-L0CATIONIN USER CORE TO
RECEIVE CONVERTED TEXT.
OUTPUT *i* NONE .
ERROR IF| (I) R53 OUT OF BOUNDS,
(II) PAGE HAS NO BLOCKS ,

TEXT IS ENTERED INTO USER MEMORY.. NOTE THAT ONLY G21 CHARACTER3 ARE CONVERTED AND ALL ELSE_ IS IGNORED IN THE CONVERSION PROCESS. N0N-G21 CHARACTERS ARE CONVERTED TO BLANKS AND VECTORS ARE SKIPPED.

B5 MOVE A PAGE TO USER CORE
WITHOUT CONVERSION.
PARAMETERS! R52*PAQE NUMBER .
R53*L0CATT0N" IN USER CORE
TO _WH_ICH_ THE BLOCK^ _
._ .. gILL BE MOVED".
OUTPUT! NONE.
ERROR IF! (Tj R53 OUT OF BOUNDS,
OR (JI) PAGE HAS NO BLOCKS,

EVERYTHING FOLLOWING THE DELIMIT IS MOVED TO USER COR?. BE PREP AREO_Xp__ACCEPT THE_ _FULL._P.A_QE .

B6 READ THE CURSOR.
PARAMETERS! NONE
OUTPUT! R52«-X 1
"" R53*Y

THE POSITION OF THE CURSOR IS OBTAINED FROM THE POSITION WORD IN THE H-MODULE,

B7 SET THE CURSOR.
PARAMETERS! R52»*X

R53-Y ,

OUTPUT: NO*NE.

THE POSITION WORD IS CHANGED SO THAT THE CURSOR IS REPOSITIONED AT (X,Y).

B8 READ THE ANALOG KNOBS.

EAR _M6.IERSj NONE_.

OUTPUTj R52-KN08 1

_ • R53-KNQH 2 _ _

R54.-US8R STATE SWITCHES

BIT PATTERN.

R55..ALT(1) OR NORMAL(O) MODE.

GETS THE J_OSITIONS OP THE ANALOG KNOBS FROM THE POSITION WORD IN THE H-MODULE.

89 NOT" YET"**S**PECIFIED. ~ ~ ~ ~

BIO READ THE STATE SWITCHES.

PARAMETERSt R52<-ALT(1 OR NORMAL(1) MODE. .

OUTPUT! R53«-STATE WORD.

~Bli""~'SET' tHE-STATE"''SWl"TCHEsV

PARAMETERS) R52-DESIRED SETTING._

OF STATE WORD.

OUTPUT: JNONE.

LOADS RS2 INTO THE STATE WORD.

812 SET COMPARE ON CHARACTER '

ON_PAGE_N.

PARAMETERS) R54-CHA**R"A**CT"ER TO

COHPARE_OJ__

R55^PAGE NUMBER.

OUTPUT: NONE. _ ^_

ERROR IF: (I) ILLEGAL PAGE NUMBER

_ (ID NO ROOM LEFT ON PAGE _

OR (III) DELIMIT FOLLOWS STORE. ~ ~

CREA_TES__ A__ COMPARE COMMAND FOR THE SUPPLIED SCOPE CHARACTER.E.G\RETURN' WOULD BE 8R75. AND INSERTS IT IN ~ PAGE N IMMEDIATELY FOLLOWING THE DELIMIT, __TH= PROGRAM SHOULD DEFINE THE COMPARE ROUTINE BEFOREHAND,

813 "SET"COMPARE "ROUTINE,

PARAM__TEJ_S_I__ R_5S<-C0MPARE CHAJ?ACTER. LOCATION FOR VALUE

R56*-ADURESS OF

USER ROUTINE.

OUTPUT: NONE.

ERROR IF: R56 OUT OF BOUNDS.

SETS USER ROUTINE TO BE EXECUTED WHEN A COMPARE INTERRUPT OCCURS. NOTE_ THAT __THE ROUTINE MAY BE EXECUTED AT ANY TIME.

B14 SET MEMORY FULL ROUTINE.
PARAMETERS r R56*-ADDRESS OR USER
RJPUT1N_E_
OUTPUTi NONE.
ERROR_IFJ _ R56 OUT OF BOUNDS.
SETS USER ROUTINE TO BE EXECUTED WHEN A MEMORY FULL INTERRUPT IS GENERATED. NOTE THAT THIS ROUTINE MAY BE EXECUTED AT ANY TIME.

B15 GET N BLOCKS FOR PAGE M
PARAMETERS! R52*PAG_' NUMBER
R53-NUMBER OF BLOCKS
OUTPUTI NONE.
ERROR IFt ILLEGAL PAGE NUMBER.
SAME TASK AS ON MANAGEMENT PAGE.

B16 ENABLE" PAGE N. '
PARAMETERS! R52*PAQE NUMBER.
OUTPUT! NONE.
ERROR IF! ILLEGAL PAGE NUMBER.

B17 DISENABLE PAGE N
PARAMETERS! R52-PAGE NUMBER.
_UT__J_L*_ NONE.
ERROR IF| ILLEGAL PAGE NUMBER.

B18 DELETE PAGE N.
PARAMETERS! R52«-PAGE NUMBER
OUTPUTi NONE.
ERROR IF! ILLEGAL PAGE NUMBER.
SAME AS TASK IN OPTION STATE. PAGE IS RETURNED TO AVAILABLE SPACE AND INFORMATION IS LOST.

B19 APPEND PAGE N TO PAGE M.
PARAMETERSI R52-PAGE NUMBER N.
R.3-PAQE NUMBER _M.
OUTPUT! NONE,
ERROR IF! ILLEGAL PAGE NUMBFR.
SAME TASK AS IN OTTTOKTSTTTE.

~B"20 DISENABLE'TIL INPUT "FROM THIS SCOPE. _ _
PARAMETERSI NONE.
OUTPUT! NONE
DTS~ENABLES ALL PAGES FOR THE GIVEN SCOPE,

B21 REMOVE COMPARE ON CHARACTER ON PAGE N.
PARTMETERS(R5'4«-CHARACTER TO
__il'J_l__OJ_!
— R55*PAGE NUMBER.
OUTPUTt NONE.

ERROR IF: ILLEGAL PAGE NUMRER,
SEARCHES "THEPAGE FW"AN~ITc"CUR^r
ON THE SPECIFIED CHARACTER AND IF FOUND, CONVERTS IT TO A
STORE COMMAND. ' " "

"B22 RESET COMPARE"" ROUTINE;
PARAMETERS: NONE.
OUTPUT: NONE.
RESETS THE STANDARD MONITOR
ROUTINE"FOR THE COMPARE R0UTINE, "~
9?3 RESET MEMORY FULL ROUTINE.
PARAMETERS! NONE.
eujpux*-^---^
RESEIS THE STANDARD MONITOR ROUTINE FOR THE MEMORY FULL
ROUTINE,

824 SET ENABLED CURSOR AND INTERRUPT
ON CHARACTER,
PARALTERS R520.
n*o^T
R*4*CHARACTER FOR COMPARE
R55.-PAGP NUMRER."'"
R56.-COMPARE ROUTINE.
OUTPUT. NONE.
E.RRO«!£_: (I..._ILL EG A L__PA G_E_J_UABM.._
/ T t ik_ n n /iuir- r* t nu n a n ^

(J^a m«T n?ᴸpSuMr4
n B
tuts _m.Tt.P n ti i rt d?S an tun =h *

JM HfcS TMP pVrF POSITIONS IMP PURSO; _tS£ Vf Sltc _

f - tn^fpi m thp⁻ rmapif*tpIT* IS?^r-TTU^T^-

p⁹ uhtpS tSts rnp L ' ' Jsn
TO OBTAIN THE SLOPE NUMBER AND TO USE A CONTROL SWITCH, R25
TO UHTAIN THE SLOPENUMBER,AND TO USB A CONTROL SWITCH, B25
SHOULD He "ALLtU A_ WbLL.

~R?5—nPFTNP TlSFW t'NTF~r"riJPTS

°i>l«-^, R52*-USER'ENTRY POINT
IIIE'' '
«n *A_I =_•
NONE
K ᵀ !F,.. USER'ENTRY POINT DOES

R55, R56 WHEN USING 825. THE USER CONTROL SWITCH ALLOWS
THE USER PROGRAM TO ' DECLARE ITSELF~T;NTE™WTTBTF~AS~"
DESIRED. IF IT IS NOT EQUAL TO ZERO WHEN THE INTERRUPT
OCCURS," THE SCOPEMONITOR DOES NOT PASS C0NTR0C~3UT KFEPS
LOOKING _ONCE A SECOND UNTIL THE VALU=_OF THE SWITCH IS
ZERO/ ASECOND INTERRUPT DURING THIS fIME ^ILL GIVE ""'
MJJLTJP.LE__INJJRRUPT ERROR AND BE IGNORED. BUT THE FIRST
ONE WILL STILL BE PROCESSED CORRECTLY,

BJ___SET CRT FIELD ON PAGE N.
"PARAMETERS! R52-N
R53-BITS FOR
CRT FIELD.
OUTPUT i NONE.
THIS ROUTINE ALLOWS THE USER PROGRAM TO DISPLAY ON MORE
THAN ONE SCOPE. THE T22 TABLE IN THE SCOPE MONITOR HAS
BIT PATTERNS FOR EACH SCOPE INDICATING~THAT THE HUMAN HAS "
ALLO_WED_INT_RACTION WITH PROGRAMS FROM OTHER SCOPES. THE
NORMAL ENTRIES ARE Si, $2, AND S3. RESPECTIVELY, JF SCOPE
2_ ALLOWED INTERACTION WITH PROGRAM -ROM SCOPE 1, ITS
ENTRY WOULD BE CHANGED TO $1 + _2, THE USER PROGRAM
INDICATES ITS DESIRE TO DISPLAY IN BOTH SCOPES USING B26
AND PASSING THE LIST PATTERN SI +S2. THE USSR PROGRAM^CAN ~"
ONLY DISPLAY ON THOSE SCOPES FOR WHICH PERMISSION HAS
— BEEN GIVEN; SINCE THE BIT PATTERN IS'EXTRACTED WITH THE"
ENTRY IN THE T22 TABLE. THE EXTRACTED PATTERN BECOMES THE
CRT FIELD OF THE DELIMIT OF SUBSEQUENTLY PRODUCED DISPLAY
PAG iS__

B27 RETURN TO INTERRUPTED COMPUTATION.
PARAMETERS! NONE. " '
OUTPUT! NONE.
USE IN THE US"ER~7NTERRUPT SERVICE ROUTINE" TO CONTINUE THE
INTERRUPTED COMPUTATION. IF YOU WISH TO CHANGE TO A
DTFFERENT LINE"OF COMPUTATION, USE A GO TO STATEMENT.

B~28⁻CLEAR A PAGE". ~ ~ " ""
PARAMETERS^ R52-PAGE NUMBER.
OUT'PUTI WNE"
ERROR IF! ILLEGAL PAGE NUMBER.
INSERTS A STOP_ c OMMAND AFTER "THE DELIMIT "01\T"P"A S'E~"N"'." NOTE
THAT ATTEMPTING TO CLEAR A PAGE OF ZERO LENGTH WILL 7ER0
A DELIMIT AND"PERHAPS DESTROY I^FORMATION.

THE FOLLOWING ARE NOT YET IMPLEMENTED

B29 MOVE PAGENJTO FILE M.
PARAMETER'S! R52"«-N
R53*M

```
        OUT PUT 1          NONE .
B30     HOVE LOGIC BLOCK BASE N LENGTH
        L TO- FILE M.
        PARAMETERS!        R52-N
                           R53«-L
                           R54-M
        OUTPUT;            NONE
B31     MOVE FILE M TO PAGE M,
        PARAMETERS I .     R52*-M
                           R53.-N
        OUTPUT!......_.........NONE. .............

_B_2_   " 0 V. E„F J L E M_TO LOCATION N.
        PARAMETERSJ        R.2-M
                           R53.-N                _     _
        OUTPUT!            NONE.
B33^    READ IN A 3-DIGIT INTEGER
        AT IX, Y],
        ER_ROR_ IF!        ILLEGAL PAGE NUMBER.
        PARAMETERS!        R52*X
                           R53 . Y.        ._. . . . . . . . . . . . . . . ....

B34     READ_IN A STRING OF CHARACTERS      _
        AT [X,Y],
        P_A_AMFTE_RSI      R52-X
                           R53-Y

"B4(T   GETS SCOPE~MAH NUMBER,
        GI_.EN_Q-?0  M_A   NUMBER,
        PARAMETERS!        R52-G-20 MAN NUMBER.
        OUTPUTS:           R53-SCOPE MAN NUMBER.

B_I_    GET_S_COPE MONITOR SYMBOL N.
        PARAMETERS!        R52-N
        OUTPUTS!           R53«-VALUE OF SCOPE
        ~  "  "  "  '              MONITOR SYMBOL.
        N  -      1      U35,    FETCH A MODULE
                  2      T60,   PAGE ADDRESS TABLE
        _  _  _  3      U29,    RELEASE A MODULE
                  4      U5,     PUSH THE STACK
                  5  _   U6,     POP THE STACK
                  A      U17,    EXIT
                  7     J15.    CONUERSION_TABLES_FROM
                                 G-20 -> SCOPE CHARACTERS
                  8      T31,    SCOPE MONITOR TIME USED TODAY
                  9      Y6,     TRACE
                 10      Y105,   TRACE BREAKPOINTS
                 11      Y72,    TRACE TAPLFS
                 12      T74,_  ISR RETURI_J»OINT_
```

TO OBTAIN THE ADDRESS OF A PAGE

```
 AL   DUM*-"2»"OUT^TRUE'l  B<41,DUM,T8Q,DURTDUMVDUM>1

     OUT«-FALSE!   ADDR«-T80-1+SCOPENUM+3*PAGE|
```

ERROR NUMBERS,

THE SCOPE" MONITOR PASSES AN INTEGER IN THE ACCUMULATOR WHICH IS_THE_J-OCATION IN THE SCOPE MONITOR WHERE _THE__ERROR_ "WAS DETECTED. THF FOLLOWING TABLE RELATES THES-" INTEQERS J0 THEIR MEANINGS _

ERROR NO	.ROUTINE	MEANING
		INTERACTION UNACCEPTABLE.EITHcR (A > REMOTE FROM WHICH JOB WAS_____ SUBMITTED IS NOT A SCOPE_I (E _ NOT IN [5,7). OR(B)MANNUMBER OF USrR LOGGED IN ->=MANNUMBER ON JOB CARD OF PROGRAM,
167003	SEVERAL.	PROGRAM ATTFMPTJNG T3 INTERACT____ WITH A SCOPE FOR WHICH PERMISSION HAS_NOT BEEN GI_
167014		BROUTINE WITH THIS NUMBER DOESN'T EXIST.____________
167145	2	PAGE DOESN'T EXIST
167164	2 _	_NO ROOM LEFT ON PA__E______
167 211	3	NOT ENOUGH SPACE,
167355	_2_	EITHER(A) _____ AGE EXISTS OR(B)NO STORE COMMAND FOUND ON PAGF JOR(C) A_ DELIMIT. IMMEDIATELY J_pLLOJ_S__ THE STORE(PAGE FULL)
167„46_	17 0R_18„	PAGE^NO^NOT IN[i,_4]_.______________.________
171365	SEVERAL	PAGE NO.NOT IN [1,4].
173 37-2	SEVERAL	J_AJSE__D_0ESN'T EJX.I ST.________________________
171402	SEVERAL	LOCATION GIVEN IS NOT IN USER CORE I.E. NOT IN t/I_000,/7300_I.____________
171406	SEVERAL	'LOCATION GIVEN IS NOT IN UPPER CORF I.E. NO_T IN [/160000./177777].________

8. SUBPROGRAM LIBRARY.

WE HAVE "ONLY JUST STARTEDTO SET UP THIS LIBRARY. LISTINGS CAN BE OBTAINED^FROM THE RESPECTIVE AND FILES. USER CR3BAB14J

ALGOL SUBPROGRAMS	FILE 32/PJ
FORMULA ALGOL SUBPROGRAMS	FILE 3]./P|
SPITE SUBPROGRAMS	FILE 33/P*

PROCEDURES IN ALGOL AND FORML THE SCOPE ALGOL' LIBRARY CAN NOW BE USED "AS AN OUTER BLOCK TO ANY ALGOL PROGRAM. YOU NEED AN EXTRA" END,OF COURSE. THESE PROCEDURES WERE WRITTEN BY RUDY KRUTAR,JIM KING,ALAN BOND AND DAVE VAVRA, THE LIBRARY IS CURRENTLY BFING MAINTAINED AND _EXTENDED BY_RUSSELL MOORE,TO WHOM SJJGGEjmONS_AN/J^ QUERIES SHOULD BE DIRECTED.

1, INTEGER PROCEDURE LOC(N)t INTEGER N) GIVES THE ADDRESS WHERE THE VALUE OF AN IDENTIFIER IS STORED. FOR ARRAYSTTOC UtII) WILL GIVE THE 1ST WORD OF THE ARRAY.

2. LOGIC PROCEDURE DECML (NUMBER)i INTEGER NUMBERi"GETS THE DECIMAL G-20 CHARACTERS FOR THE VALUE OF NUMBER AND PACKS THEM IN DECML.

3. BOOLEAN PROCEDURE B(BNUM, 352, B53, P54, B55, J356) VALUE BNUMI INTEGER BNUM. H52, ""853, 954,, B55VT5A| CALLS B-ROUT'lW NUMBER BNUM, QN__EJ__ROR EXIT, 8 IS TRUE, NORMAL EXIT FALSE. HENCE. IF B( ) THEN GO TO EXlTj WILL CALL THE B ROUTINE.

4. AN ALTERNATIVE VERSION OF B, WHICH HAS GLOBAL 300LEAN VARIABLES OUT ,ND""PR"." IF OUT IS TRUE PARAMETERS ARE OUTPUT, IF PR IS TRUE, THE VALUES_OF PARAMETERS AND NATURE OF EXIT ARE PRINTED.

..........5 ,PROCEDURE BA(BNUM,ETC) IS A "PROCEDURE RATHER THAN A FUNCTION AND CALLS ON B.. . . . . _

6"i LOGTc"PR0"GE'DTi^E D"ER (TTYTI TOUE X,Yi fNT'EGFR TTYT COMPUTES A HEADER INSTRUCTION AT X,Y. NOTE THAT X,Y_ MUST LIE J_N CO, 1023 J,

7. LOGIC PROCEDURE VECTOR (X,Y,SG)» VALUE X,Y,SGt INTEGER XTYJ LOG fC~SsTc0"MPUTES~A~VEC! TOR "STRING ELEMENT"WT"TTrTx~*~T7~n Y ^ Y. SG = H USUALLY, SG = 2 FLAGS THE VECTOR SO THAT IT WILL BLINK

OR INTENSIFY ACCORDING TO THE SETTING OF STATE SWITCHES.

8. LOGICPROCEDURE CHARAC (CSG, if) INTEGER C SG.I) PRODUCES A SCOPE CHARACTER IN A WORD IN POSITION I » 1, 2, OR 3, SG IS THE TAG FIFLD, SO SG = 2 GIVES BLINKING AND INTENSIFICATION. C IS THE SCOPE CHARACTER NUMBER AS GIVEN IN THE 3UATSE MANUAL.

9. LOGIC PROCEDURE CHARSTR (C1,""C2," C3, SGI." S^2. S33JI

10. PROCEDURE CHARACTER (X, Y, C)J INTEGER X, Y» LOGIC) PUTS "a~""cTTa"Ra"cTER0N "THE"" SCREEN AT P_TnT X,"'Y." "c IS A SC~OPE~ CHARACTER-STRING WORD AS OBTAINED BY_ USING CHARAC OR CHaRSTR,__IT can also be obtained by

__J>J5L2 _j_fIR_* ⁴L ʷHFRE N IS THE SCOPE CHARACTER NUMBER FROM THE" QUATSE MANUAL.' "CHARACTER HAS ONE CHARACTER ᴾER WORD AND A SEPARATE HEADER FOR EACH CHARACTER, AND IS, THEREFORE, WASTEFUL OF SPACE, G-20 CHARACTERS ARE BEST PUT ON THE SCREEN USING 32.

11, _PR0CEDURE_NUM<X _V_,_N)}__ VALUE N) INTEGER X, Y) REAL Nj_ TAKES A REAL NUMBER N, FINDS DECIMAL CHARACTER FORM, AND PUTS IT ON_THE_SCR6EN aT__X,Y IN -5D.32 FOR_MAT.

12, PROCEDURE LINE (XI, YI, X2, Y2M INTEGER XI, YI. X2, Y?) PUTS A LINE FROM (XI, YD TO (X2, Y2), WITH A SEPARATE HEADER.

.....PROCEDURE GENERATE <T,~Y r"T.~ DT, MORN")")"' VALUE DT. NORM j REAL X, Y, T, DT, MORMi GENERATES A CURVE WITH PARAMETER T WHOSE X', YARE GIVEN 9Y EXPRESSION INVOLVING T WHEN ACTUALLY CALLING GENERATE. THjUS. . . . .

GENERATE (A • SI N (T) ,"' _f' * COS (T), T, DT, NORM)')
_._................WILL PLOT AN ELLIPSE.

. . r ^ 6 . ^ - - I T — —j N E - s_QpjgNfS*, —AND" I T~GJT_CTJLTTE_~ThcS"E~TO"R INTERVALS IN T OF DT. IT ASSUMES A SQUARE SCREEN WHOSE LINEAR SIZE IS "NORM IN RELATION TO THE VALUES OF X, Y.

14, CURVE (X, Y, T, DT, TA, TR)J INTEGER X, Yf REAL T, DT, TA-,"TBr PLOTSFROM "TA TO "TB;

15. INTEGER PROCEDURE SCALEX (X)I REAL Xt

INTEGER PROCEDURE SCALEY (Y)I REAL YI THESE "ALLOW EASY SCALING, GLOBAL VARIABLES XA, XB, YA, YB, SXA, S*B, SYA, SYB INDICATE THAT THE PART OF THE SCREEN USEDWILL 8E FROM SXA TO SXB AND SYA TO SYB, WHERE THESE LIE IN [0, 10231, AND THAT THIS WILL CORRESPOND TO VALUES XA, XB, YA, BY IN THE REST OF THE COMPUTATION.___THUS . . . .

CHARACTER (SCALEX (X), SCALEY (Y),C)J PUTS A CHARACTER ON THE SCREEN AT POINT X, Y IN THE USERS SCALE.

THERE ARE AN EQUIVALENT SFT OF PROCEDURES TAKING REAL ARGUMENTS FOR___P_OSITION_ AND USING SCALE X AND SCALE Y, THESE ARE DESIGNATED BY AN ADDED *i"*ON tHENAME. THUS,CHARACTERS, NUM1, LINEl, CURVE1, ETCi

16. REAn.PAGF(N,**RBuFF)** > THjL -nTi ru^TrT^^pT LENGTH.....1 BLOCK> INTO ^_**m!.**P **! on dc"! VhH** ^ **v** tScw »eln **rortM** WORD ,SO IT IS LIKE **A** NORMAL CARD READ, **YOU** MAY THEN "**>cAD FROM** RBUFF USING AL THE NORMAL FORMATTING POWER OF ALGOL,

pRi^TiON PAGE(N,WBUFF,X,Y)) POTS THE CONTENTS OF' PRINT BUFFER___WJBUFf __AFTER NORMAL ALGOL PRINTING (WHICH CAN BE WITH OR WITHOUT <E> OR <W>) ONTO PAGEN AT X,Y. THUS THE FULL GENERALITY OF ALGOL I/O 'S AVAILABLE FOR COMMUNICATION WITH THE SCOPES.

3,. **sETS**'C'OPENUM(N)) INTEGER Nj S**ETS** SCO^S' MUMBER TO N, SO THAT FUR**T**HEP_ CALLS OF B ROUTINES APPLY TO THIS SCOPE. THEY OF COURSE GIVE AN ERROR IF PERMISSION HAS NOT BEEN SIVtN MANUALLY AT THE SCOPE, _____

.........."19. INTEGER "PROCEDURE SCOPENUMj GIVES THE SCOPE NUMBER CURRENTLY SELECTED. IT__ SHOULD NOT BE C0*F_USED_WITH T_H_E_ SCOPE NUM8ER PASSED UPON INTERRUPT.

20 BUTTIN(ENPT,CNTRLSW.INTNUM,SCOPNUM,PAGE IN,PAGEOUT> J INTEGER * "AR0 UM E N TS. DEFINES BUTTON INTERRUPTS,DISPLAYS 'INTERRUPT S NOW DEFINED' ON PIAGEOUT AND CONTINUES COMPUTING. ON INTERRUPT, IT PASSES CONTROL TO ENPT, WHICH CAN RE A CLOSED PROCEDURE OR A LABEL,_____I_T PUTS*X NTERRUpT NUMBER ON PAGEOUT UPON_INTERRUPT "CNTRLSV'=n" INHIBITS INTERRUPTS." INTNUM IS THE NUMBER OF THP BUTTON INTERRUPTING AND SCOPNUM IS THE NUMBER OF THE SCOPE INTERRUPTING.

21. eOWIN<5NPT,CNTRLSH.S-0PNUM.CtiAR«PAGEIN.PASE0UT,CH)I SETS "COMPARE INTERRUPT oT~CHARACTER cWDN~7a1e~1^IN, SIMILAR t0 BUTTIN. ON INTERRUPT, THE CHARACTER WHIGH CAUSED THE INTERRUPT WILL BE FOUND IN IDENTIFIER WHOSE LOCATION IS CH.

NOTE THAT CONTIGUOUS DECLARATION OF SCALARS IN FORML GIVES "ALLOCATIONS IN SUCCEMTvTIWDST WHEREAS I¥"ATfIOL^TT ~GYVES CONTIGUOUS LOCATIONS BUT IN THE REVERSE ORDER TO THE ORDER OF "dec! ar 7tTon7 :

THUS, LOGIC D3, D2, D3J DUMPS (3, D3) I_PJ__T S_TH _J?_ VJ__S_P D3, D2 AND DI.

MACROS AND ROUTINES IN SPITE

1. MACRO BC XX1,XX2,XX3,XX4.XX5,XX6J CALLS INTERFACE ROUTINE 10. EXPECTS ALL ARGUMENTS TO BE CONSTANTS, I.E., FIXED AS ASSEMBLY TIME.

2. MACRO BV XXI, XX2, XX3, XX4, XX5, XX6, EXPECTS XXI TO 9E CONSTANT AND XX2 ..• XX6 TO BE "VARIABLES, I.E., 3E LOCATIONS WHICH CONTAIN THE DFSIRED ARGUMENTS,

3. 10 THTTNTERTACE~RWT"i"N_.

4. THERE IS A _VARIA_NTON B WHICH PUTS A MESSAGE ON THE G-20 TYPEWRITER ASKING FOR THE H-MODULE TO BE SWITCHED, IF IT ISN'.T.

5, SOME MACROS TO^ EASILY GENERATE SCOP? DISPLAY MATERIAL HEADR, VEC. CWD, STOR.

9. USER SUBSYSTEMS.

AN • INTERACTIVE PROGRAM ON THE G-20 IS INEFFICIENT IN ITS USE OF COMPUTER TIME IN THAT IT OFTEN IS IN A LOOP WAITING FOR THE HUMAN TO TELL IT WHAT TO DO NEXT, ALSO THE PROGRAM MUST WAIT IN THE QUEUE BEFORE IT CAN BE INITIATED. THE PAUSE SYSTEM IS USEFUL FOR GETTING SHORT BURSTS OF USER PROGRAM.

IN IMPLEMENTING A TIME SHARING SWAPPING SYSTEM FOR THE. SCOPE MONITOR, IT WAS FOUND EASY TO ALLOW ANY USER TO WRITE SUBPROGRAMS OF RELOCATABLE REENTRANT ASSEMBLY CODE WHICH ARE ORGANIZED BY THE SCOPE MONITOR. SWAPPED IN AN OUT AS REQUIRED AND AS SPACE PERMITS, _E_LOCATED IN CORE AND LINKED TOGETHER DYNAMICALLY IN A SIMPLE WAY._ THERE IS ALSO AN AUXILIARY MACRO SYSTEM WHICH ALLOWS THE CONVERSION OF ORDINARY ASSEMBLY CODE INTO THE REQUIRED REENTRANT RELOCATABLE MODULAR FORM. IT TURNS OUT THAT ANY MODULE OF ANY. USER CAN CALL AN.Y MODULE OF ANY OTHER IN AS VIOLENTLY A RECURSIVE WAY AS REQUIRED, AND THAT ONLY ONE COPY OF ANY MODULE IS IN CORE EVEN IF CALLED BY SUBSYSTBMS FROM ALL THREE SCOPES AT ONCE.

TO CONVERT CODE TO MODULAR FORM, ONE USES THE MACROS AND ROUTINES ON USFR CR3SAB14. FILE 34. THEN ONE BREAKS THE CODE AS FOLLOWSI

LAYOUT		EXAMPLE	
FILE 34 PACKAGE		USER	CR38AB14; FILE 34/Pj INSERT $
NON RELOCATABLE (GLOBAL) SYMBOL DECLARATIONS		LBL	T9 0|
BEGIN		BEGIN	
RELOCATABLE (LOCAL) SYMBOL DECLARATIONS		LBL	E20I
ENTRY POINT DECLARATIONS		ENPT	1. EI
		ENPT	2, E2
(REENTRANT) CODE	EI	ENT	
		PUSH	**51**
		CLA	0 1*
		STL	2.50
		TRM	E3
		EXIT	
OR NONREENTRANT	E2	ENT	
		TRM	E4
		TRA	1- EI
EXTERNAL IDENTIFICATIONS TO	E3	ISMOO	1. 5. •AB14)
ENTRY POINTS OF OTHER MODULES	E4	ISMOD	2. 7. *ADO3J
E.G. E3 IS ENTRY POINT 1 OF MODULE 5 OF USER AB14			
END		END	
STORE		STORE	6. »AB14|
STORES THE GENERATED MODULE. E.G.. AS MODULE 6 OF USER AB_4			

THE PtTSht MACRO' TJECL~A~R¥S STACK VARIABLES. fc , 9 • . PTTSfl 5" DEC LARES 5 VARIABLES AND PUSHES THE STACK, ONE THEN USES THESE VARIABLES WITH THE POINTER IN REGISTER 50.

THUS CLA ?,50 CLEAR AND ADD SECONO STACK VARIABLE.

ST I 3.50 PUT IN THIRD STACK VARIABLE"". "

PUSH MUST IMMEDIATELY FOLLOW THE ACTUAL ENTRY POINT (TO ALLOW THE Mark t"o~b1T~sTackeo> . * rFentrTnt routine does not exit thhough **ifs"** MARK BUT THROUGH THE STACKED MARK USING POP N. WHERE N IS THE NUMBER OF' STACK VARIABLES IN THE "ROUTINE. TO EXIT 'BACK TO THE S C OPE MONI TOR USE EXIT. THE S T A CK I NG. JNCLUD IN G S T ACKING THE MARK T N W T, 50 (WHICH SHOULD THEREFORE NOT BE USED BUT ALWAYS ALLOWED FOR) AND PUSHING. POPPING, ERROR RECOVERY IS ALL DONE BY- THE SCOPE MONITOR,

STACKED VARIABLES HAVE TO BE USED TO KEEP THE VALUES OF VARIABLES N EE OED " "DU1RTnU RECURS I'VE c AL LS OR "AN Y "T IM E THE'CO D"E "MAY " HAVE TO WAIT. TO SWAP IN ANOTHER MODULE ONE HAS TO WAIT FOR THE DISC. SO SfXCKEin/ARIABLES HAVE TO BE US'ED FOR ANY VALUE'S, SET" BEFORE ANY TRM, WHICH ARE REFERRED TO AGAIN AFTER THE RETURN

THROUGH THE MARK. THIS IS BECAUSE ANOTHER USER MAY ENTER THE SAME CODE DURING THE WAIT". ONE CAN HAVE MODULES OF REGULAR CODE BUT IT CANNOT CALL ITSELF RECURSIVELY. AND CANNOT BE SHARED BY ANY OTHER SYSTEM. NOTE THAT EACH USER HAS HIS OWN NAMES FOR ALL HIS IDENTIFIERS. HAVING CREATED THE SUBSYSTEM. IT CAN BE LOADED FROM THE PROGRAM STATE. MODULES WILL NORMALLY BE MARKED AS DISPENSIBLE AFTER USE. AND ARE LIKELY TO BE SWAPPED OUT IF THE SPACE IS NEEDED FOR SOMETHING ELSE) HOWEVER, THE USER CAN MARK ANY MODULE AS RETAINED' WITH AN I.NTERRUPT ON THE PROGRAM PAGE. HE CAN *RELEASE' ALSO, LOADING A MODULE"AUTOMATICALLY RETAINS IT, OR ONE CAN SIMPLY ASK TO TRANSFER TO A MODULE WHICH WILL LOAD IT IF NECESSARY, EXECUTE IT AND RELINQUISH IT.

THE ADVANTAGE OF SUBSYSTEMS IS, OF COURSE, THEIR EFFICIENCY THEY CAN BE USED ON AN INTERRUPT BASIS WITHOUT SUBMITTING A Q-gQ PROGRAM. A SUBSYSTEM CAN USE 8 ROUTINES TO SET UP DISPLAYS, ETC. IN PRINCIPLE; ASSEMBLY CODE AND EVEN OCTAL CODE GENERATED BY A COMPILER CAN BE CONVERTED TO SUBSYSTEM FORM, SPACE PERMITTING. IN ORDER TO HAVE A DATA AREA TO WORK ON, IT IS SUGGESTED THAT SOME MODULES BE RESERVED AS DATA AREAS WITH THE ENTRY POINTS GOING TQ DATA ACCESSING FUNCTIONS, SUCH MODULES COULD THEN BE LOADED AND RETAINED IN CORE AND THE- CODE MODULES BE PURE PROCEDURES WHICH COULD. SWAP IN AND OUT AND MANIPULATE THIS DATA. THEY ARE NOT ACTUALLY SWAPPED OUT. JUST RELEASED TO AVAILABLE SPACE. AND, WHEN NEXT NEEDED, A NEW COPY SWAPPED IN.

FOR PASSING PARAMETERS INDEPENDENTLY OF PARTICULAR DATA AREAS, REGISTERS 52-58 CAN BE USED. THESE ARE SAVED DURING WAITING FOR THE DISC TO SWAP IN THE NEXT MODULE.

THE TEXT EDITOR IS A SEPARATE SUBSYSTEM DEVELOPED BY MIKE COLEMAN, AND THEN ADAPTED TO WORK WITH THE SCOPE MONITOR.

TO DEBUG A SUBSYSTEM, ONE SHOULD FIRST GET IT WORKING AS COMPLETELY - AS POSSIiBLE BY RUNS IN LOWER CORE WITH LINEPRINTER OUTPUT. THEN ONE CAN RUN IT IN THE H-MODULE BY RUNNING A WAITING PROGRAM IN LOWER CORE, SO THAT YOU CAN ONLY CLOBBER YOURSELF. THE WAITING PROGRAM IS BEST WRITTEN IN UPDATE AND CAN THEN GIVE A DUMP OF THE H-MOOULE AND RELOAD A FRESH COPY OF THE SCOPE MONITOR AT THE TERMINATION OF THE RUN. IN THIS WAY, ONE CAN DEBUG A SYSTEM IN 3 MINUTE BURSTS WHTHQUT ENDANQERING INNOCENT USERS AND HITHOUT BOTHERING THE OPERATORS TO DO DUMPS, WHEN THE USER SUBSYSf_M™!S SUPPOSEDLY DEBUGGED? IT CAN BE RUN ANY TIME WITH NORMAL USER PROGRAMS IN LOWER CORE, BUT If MUST FIRSr'PASS AN ACCEPTANCE TEST, THE ACCEPTANCE TEST PROGRAM CAN BE OBTAINED FROM A. H. BOND.

10,QRASp

'GRASP'ISA GRAPHICAL SYSTEM,AKJN TO 'SKETCHPAD»7"DEVELOPED BY GENE THOMAS ON THE G-21, AND DESCRIBED BY HIM AT' THE ACM CONFERENCE 1967, IT WORKS WITH OUR SCOPES AND IS WRITTEN IN ALGOL 20, SO _U SHOULD 1BE_ EASJLY_ TRAJ_SFER_BLE T0 T_HE_360 . IT JS CURRENTLY BEING MAINTAINED AND EXTENDED BY RON BUSHYAGER,

GRASP (GRAPHIC SERVICE PROGRAM) IS A GENERAL GRAPHIC MODEL. BUILDING SYSTEM. IT ISUSED IN TWO WAYS:

1. IT PROCESSES AN INPUT STREAM OF CARDS IN ASJJMPLE "LANGUAGE,WHOSE FORMAT IS SIMILAR TO A SEQUENCE OF'ALGOL PROCEDURE CALLS. THIS ALLOWS THE USER TO DEFINE AND NAME GRAPHICAL ELEMENTS LIKE POINTS,LINES,ETC,,BUILD NAMED CONFIGURATIONS FROM THESE ELEMENTS AND _DL_'LICATE INSTANCES OF THESE CONFIGURATIONS AT DIFFERENT LOCATIONSAND ORIENTATIONS IN THE [3] MODEL "SPACE. THE STRUCTURE OF TH_E_MODEL NESTED, SO ONE HAS CONFIGURATIONS AT VARIOUS LEVELS. ONE CAN INPUT CARDS PROM THE NORMAL INPJT STREAM OR FROM THE SCOPE FACE AND ONE CAN OUTPUT A [g] DISPLAY DERIVED "FROM THE MODPLTON T1HE^N"EVPRTNTTR OR THE SCO=>E FACE. THE 12} DISPLAY IS COMPLETELY SPECIFIED BY THE USER AS TO ITS SCALE, REGION OF INTERSET WITHIN THE MODEL, VIEWPOINT IN THE MODEL SPACE AND ORTHOGRAPHIC OR STEREOGRAPHIC PROJECTION. THE MODEL CAN ALSO BE CHANGED IN VARIOUS SIMPLE WAYS "- PARTS OF IT CAN BE ROTATE0, MOVED OR DELETED , AND THE MODEL CAN BE SAVED ON AN AUXILIARY AND TT_E.

2, THE USER CAN USE PART OF THE GRASP SYSTEM AS AN OUTER BLOCK TO ~AN A' QOL PROGRAM/ WHICH CONTAINS PROCEDURE CALLS TO GRASP " PROCEDURES, SIMILAR TO THE LANGUAGE, BUT EMBEDDED IN ANY ALGOL CONSTRUCtIONS~. THTS FMRWRAM WO ULTJ~~C~0NSTITUT= AN 'APPTTCATION PROGRAMS AND, IN _THE_ GRASP LANGUAGE, ONE_ CAN INSTRUCT THE__ APPLICATIONPROGRAM TO BE CALLED FROM ITS AND FILE AND APPLIED TO THE MODEL. ;

GRA~SP~ DTTRS NOT HAVE CONSTRAINT SAfTSFACTiON FETN^ESTgUTTT IN. THERE .IS .QUITE A GOOD AND COMPREHENSIVE USER MANUAL AVAILABLE.

11. HOW THE SCOPE MONITOR WORKS.

A. RELATIONSHIP OF THE" SCOPES TO THE 9-21 AND THE MAIN MONITOR

THIS SECTION CAN BE SKIPPED,

.TJ_E_G__21 HAS SEVERAL 8K MEMORY MODULES ON A 8USS, AND, I_N ADDITION, THE H-MODULE, WHICH CAN BE SWITCHED IN AS REQUIRED TO REPLACE THE Q-MODULE. THE CORE LOCATIONS OF THE H-HOOULE ARE /160,000 TO /177,'777, AND A PROGRAM WILLi COMMUNICATE WITH THIS CORE WHEN THE H-MQPULE SWITCHED IN) OTHER^ISE, IT- WILL COMMUNICATE WITH THE G-MQDULE, SWITCHING IS DONE BY SETTING S13 IN I_E_CE REGISTER. IT CAN ONLY BE SWITCHEP IF THE OPERATOR HAS SET THE MODULE SWITCHES. TO SEE IF IT IS SWITCHABL5 WE MUST READ THE STATUS REGISTER SR (REGISTER 5) AND LOOK AT S4. THE NORMAL PE IMAGE PROTECTS THE H-MODULE, AND SO WE CAN RESET THAT TO /7fl OR /13 ACCORDING TO WHFTHER THE CORE IS INVERTED OR NOT. THE CORE IS INVERTED IF THE (ABC) BUTTON HAS BEEN SET BY THE OPERATOR, AND T_HIS CAN BE READ BY LOOKING AT SI OF SR.IF .1 IS SET, THE USE /70. THE . MAIN MONITOR IS CONTINUALLY SERVICING INTERRUPTS FROM TELETYPES. ETC., AND WHEN IT DOES SO IT STACKS THE ACCUMULATOR AND THE NC REGISTER ONLY* THE MAIN MONITOR,INTERRUPT STACK IS 4 DEEP AND CIRCULAR__wHEN IT_RESTORES CONTROL TO THE PROGRAM, IT RESTORES THE ACCUMULATOR AND TRANSFERS TO (NO AND IT RESETS PE AND CE TO A STANDARD PATTERN, NOT TO THE PATTERNS IN OPERATION WHEN THE INTERRUPT OCCURRED. IT RESETS THEM FROM THE PE AND CE IMAGES, WHICH ARE <169*1) AND (133*5), RESPECTIVELY. HENCE* WE MUST EITHER TURN CONTROL OFF OR RESET THESE EVERY TIME AFTER CONTROL HAS BEEN ' ON. THE SCOPEL JNTS^{R}RUPT BUTTONS SET 813 IN IR AND THE MAIN MONITOR SENDS CONTROL>0 THE SCOPE MONITOR, THE SCOPE iNTERRIUPT BUTTONS, AT THE SAME TIME, SET THE INTERRUPT WORDS IN THE H-MOPULE, AND THE SCOPE MONITOR READS THESE,

FIGURE 8 SHOWS CODE TO SWITCH IN AND OUT THE H-MODULE. THE CLOCK INTERRUPT TO THE SCOPE MONITOR CAN BE EASILY PATCHED AND FOR SPECIAL EFFECT. THE SCOPE MONITOR CANNOT USE ANY OF THE USUAL FACILITIES OF THE MAIN MONITOR, LIKE I ROUTTNTS". AS THESE MAY 9E~ IN USE BY THE LOWER CORE PROGRAM. COMMUNICATION WITH THE DISC IS EFFECTED BY USING THE TELETYPE DISC ROUTINE, AND THE SCOPE MONITOR ONLY ENTERS THFM IF THEY ARE FREE AND WAITS OTHERWISE, THE SCOPE F TLES""7R"E~TO"SPICTA LTY^RES
TYPE 25. THE _3L_0CKS _ARE OF LENGTH 160,, HALF TH5_ USUAL-BLOCK LENGTH. DISC SPACE IS HANDLED IN GLOBS ON AN AVAILABLE S'ACE LIST " BY THE SCOPE MONITOR*

```
          TO SWITCH IN THE MW-12

                    ERA   0    , SR)
                    IEZ   0   S4J
                    TRA       LU
                    EXR      777776,CE>
                    ERA   0   /77777,CE1    "
                    UNL   0   S13J
                    OAD   0   o;
                    LDR   0    ,ce;
                    ERA   0    ,sr;
                    IEZ   0
                    OCA   0   /70-/13J
                    OCA     07131
                    L"Dfr"~0   TPET                                          •
                    fRA        LZl
LT                  pfTO"rern              h module nut swi ichabl_
L2                  NOTTMAlTTXTT           HHODT1LE SWTT-C'RET) IN.CONTKOX"IS""OFF"


          TO RESTORE J_J__MJU._USER SETTINGS
          ________L_R______1^9+1*PEJ_____
                    LDR        135*5,CEI
```

FIGURE 8.

B. THE H-MODULE

THE LAYOUT OF THE H-MODULE IS SHOWN IN FIGURE 91 THE ACTUAL PATH OF THE SCANNER I'S AS SHOWN IN THE SMALL FIGURE. IN ORDER NOT TO UPSET THE SCANNER* IT IS DIVERTED MOMENTARILY TO A SMALL! LOOP. LOCATIONS O AND 1 OF THE H-MODULE. ON ANY REARRANGEMENT OF THE DISPLAY AREA•

THE SYSTEM MESSAGES SIT IN THE H-MODULE AND ARE MADE; VISIBLE ON A GIVEITSCOPE 9Y SETTING THE LOWER BITS OF THE DELIMIT, SYSTEM MESSAQES DISPLAY IN ALTERNATE MODE AND ON ALL PAGES.

C. PROCESSING OF INTERRUPTS, WAITING, REENTRANT CODE

THE SCOPE MONITOR IS LAID OUT AS AN INTERRUPT CLASSIFICATION PART AND THEN TABLES OF ENTRY POINTS FOR THE MEANINGS OF INTERRUPTS IN EACH STATE. TO EXECUTE A TASK THE APPROPRIATE ENTRY POINT IS ENTERED IN THE PART OF THE CODE WHICH IS REENTRANT, WHEN THE CONTROL REACHES A POINT WHERE IT HAS TO WAIT FOR THE: DISC OR FOR THE HUMAN TO TYPE IN SOMETHING ON THE SCOPE, IT MERELY SETS UP A REQUEST AND RETURNS TO THE ISR, LEAVING ALL THE LOCAL VARIABLES AND MARKS, FOR THE ROUTINES SO FAR PASSED THROUGH, IN THE STACK (THERE IS ONE STACK FOR EACH SCOPE). WHEN THE OPERATION IS COMPLETE, IT CARRIES ON WHERE IT LEFT OFF.

/160000 USED BY SCANNER

10 • STATE WORD | |

11 INTERRUPTWORD | SCOPE 1 | NORMAL

12 POSITION WORD J \

20 " | SCOPE 2 | CONSOLE

30 " | SCOPE 3 | GROUPS

110 • i ALTERNATE

120 " | CONSOLE

130 m Lj___"__!

ISO INTERRUPT ENTRY POINT

151 ; TRA__J_SR__. . . . .

15? CLOBBER WORD JRQ1

154 USER ENTRY POINT

155 TRA USER INTERFACE J

160 S_YM30L__TA8LE AND SYSTEM VARIABLES IN FI*ED LOCATIONS

180-1700 TABLES AND ALL DATA USED BY THE SCOPE MONITOR

1700-3100 SYSTEM DISPLAY PAGES

_3100_-440_q ISR

4400-10000 ROUTINES FOR CARRYING OUT OPERATIONS

!_00p0-177777_DISPLAY AREA

177776 DELIMIT TO PROTECT

177777 CYCLE TO DISPLAY PAGES

FIGURE 9". " :

THE QOHPLETIO_ J _ _ AM OPERATJON IS EITHER TRIGGERED BY AN INTERRUPT LIKE THE COMPARE INTERRUPT ON THE RETURN CHARACTER, OR. IN_THE CAS£ OF DISO TRANSFERS. THE SCOPE MONITOR KEEPS LOOKING TQ SEE IF if CAN COMP-ETE THE OPERATION, IN THIS CASE TO ENTER THE MAIN MONITOR DISC ROUTINES.

THIS TIME SHARING, INTERRUPT PROCESSING, MECHAMISM WAS _DESIGjNED AND IMPLEMENTED__**9Y**_ *4S3* RY RIGJtTJNOUR_

D. INTERACTION WITH THE USER PROGRAM

_ THE B ROUTINES ARE JUST A PART OF THE SCOPE MONITOR WHICH IS EXECUTED BY THE NEW PROGRAM. FIGURE 10 IS THE INTERFACE ROUTINE. P_NE CALLS A B ROUTINE BY PUTTING THE NUMBER OF THE B ROUTINE IN THE ACCUMULATOR AND THE SUCCESSIVE ARGUMENTS IN REGISTERS 52-56 AND DOING A TRM TO 10. THIS BLOCK OF CODE IS INCLUDED IN THE 8 PROCEDURE IN ALGOL;

USER INTERRUPTS ARE HANDLED DIFFERENTLY FROM INTERNAL INTERRUPTS. THEY ARE CLASSIFIED IN THE ISR.. 3UT CONTROL IS NOT TRANSFERRED TO THE USER PROGRAM UNTIL AFTER ALL THE SWITCHES AND MAIN MONITOR REGISTERS HAVE BEEN RESTORED JJST BEFORE CONTROL WOULD BE TRANSFERRED BACK TQ MAIN MONITOR. ' AT THIS POINT, THE SCOPE MONITOR EXECUTES ANY USER INTERRUPTS BY TRANSFERRING WITH CONTROL ON TQ THE USER ENTRY POINT IN LOWER CORE. ACTUALLY,IT STORES ITS OWN MARK IN THE USER ENTRY POINT AND DOES A TRE TO ENTRY POINT +1. THUS IF; THE FIRST INSTRUCTION TURNS CONTROL OFF,ONE CAN MAINTAIN CONTROL OFF IN AN INTERACTIVE PROGRAM,

```
10

          exr   A.   /77776.CE)       CONTROL 0-F
          STI        L20I             SAVE PARAMETER
          ERA   0     , sr;           READ STATUS REGISTER
          IEZ   JL_  $4 j             IS THE MM-12 SWITCHABLE
          TRA        I2t              NO EXIT
          LDR   0    /20302.CE)       SWITCH TO THE MM-12
          CAL        /1-0152)         GET THE CLOBBER WORD
          IUO        L10*             IS IT INTACT
          TRA        L2J              NO EXIT
          LDR   0     .PE)            RESET MEMORY PROTECT
          CLA        L20I             REFETCH THE PARAMETER
          TRM        /160154I         ENTER THE SCOPE MONITOR
LO        LDR        169+1.pe;        RESTORE MEMORY PROTECT
          LDR        133+5.CE)        RESTORE CE REGISTER
          TRE   1    10)              EXIT
L2        CLS   0    i;               SET EXIT SWITCH TO ERROR CONDITION
          TRA        LO)              EXIT
L10       ALF        1JR01I           CLOBBER WORD
L20       LWD         >               TEMP
          LBL
```

FIGURE 10.

E. THE TRANSIENT VERSION

IN THE TRANSIENT VERSION UNDER DEVELOPEMENT.ONLY THE ISR AND TABLES___WI_LL BE REStDENT,OCCUPYINO ABOUT„I5JU- WORDSj_____TH__J_.T_HE_R ROUTINES AND THE SYSTEM MESSAGES ARE SWAPPED IN AS REQUIRED AND ALL MODULES.WHETHER THEY BE SYSTEM CODE.USER CODE. SYSTEM MESSAGES OR USER DISPLAYS,ARE TREATED EOUIVALENTLY IN THE SAME AVAILABLE SPACE THE SCHEOULIJ_G___IS SUCH THAT_ MODULES ARE KEPT IN CORF AS LONG AS POSSIBLE,TO MINIMISE UNNECESSARY SWAPPING. THUS , A USE OF SEVERAL RELA'ED _FACILITJES_SHOJJLD_ INVOLVE NO SWAPPING. _J__THIS WAY, FOR LIGHT USE THE TRANSIENT VERSION SHOULD RUN AS FAST AS THE RESIDENT VERSION. AND FOR HEAVY USE. EITHER CODE OR DISPLAY AREA, THE TRANSIENT VERSION WILL BE ABLE TO CARRY OUT OPERATIONS IMPOS_SIBLE FOR THE RESIDENT VERSION ,BUT WITH_.ESS EFFICIENCY AND SLOWER RESPONSE.

3CPL Syntax in Backus Normal Form

```
<cap>   ::=  A|B|...Z
<small>  ::=  a|b| .7.2.
<diqit>  ::=  0| 1| . . .9
<cctd»   t:i  _TlT...7~

<nl>   .z-   <cap>|<small>|<digit>
<n2>   ts-   <nl>|<n2><nl>
<name>        <small>|<cap><n2>


<stringconstant>    7:=        _

<ol>   : : r   <octd>|<ol><octd>
<octn»        *8<ol>
-decn>   ,i=  ^iqit>|<decn><digit*
<numfcer>   ; :.   <decn>|<octn>


<3-o9>         frifilfizlf^ 1 1
<4-op>         ±I-II±I_
<5-op>
<6-op>
<7-op>         lshiftjr.hift
<»-op-»
*9-op>

               1
<priir,ary-E>  s : - <strino> I <string-constant* 1 <number> I true! raise 1 (E) I
               eqvj neqv
             valof<b.ock>|<primary-E>(<E-list>)I<primary-E>[<E>1|
             <name>l!<string>i|<stringconstant*|I<number>
<2-E>    :=   <prlmary-E»|<2-op»<2-E»
<3-E>        <2-E>  <2-E><3-op»<3-E>
<4-E>        <3-E>  <3-E><4-op><4-E>
<5-E>        <4-E>  <4_E>*5-op><5-E>
<6-E>        <5-E>  <5-E><6-op><6-E>
<7-E>      r  <6~E>  <6-E><7-op><7-E>
<3-E>      r  <7-E>  <7-E><8-op><8-£>
<9-E>   :  =  <8-E>  <6-E><9-op><9-E>
        =  <9-E>|<9-E>__<E>_<E>

<E11>   ::=  <E>|<E11>_<E>
<E-li_t>          <null>T<Ell>

<nll>    :=  <name>|<nll>_<name>
<D1>     =  <nll>r<Ell>
<D2>     -   <name>(<namelist>)be<block>
<D3»     =  <name _<namelist>j_<E>
-D4>     =  <name> ~vec<constant>
^manifesto   :, ="^n7me> = <constant>
<manifest2>          <manifistl>|<manifestl>j.<manifest2>
manifest*  t      <null> , <manifest2>
<globall>   :iz   <name>_<constant*
<clobal2>   it-   <globall>|<globall>_;<global2>
```

i

```
<qlobal>  ,,=  <null>|<qlobal2>
<D5>       <D1H<D2> | <D3>!<D4>|-manifest* I <global*
<D>       <D5>I<D5>also<D>

<C1>  ::-  «E11>:=<E11>
<C2>       <F>X<^list>i
<C3>       croto<E>
<C3>  ::=  break
<C4>       return
<C5>  ::=  finish
<Cf>>   =  ~~rer~~ resultis<E>
<C1>  ::=  switchon<E> into<block>
<CS>     =  <block>       ~
<C9>  ...  <cl>|<C2>!<C3>|<C4>|<C5>!<C6>l<C7>|<C8>
^C10>  i-.z  if<E>then<C>
<C11>       tist<ETthen<C>else<C>
<C12>  ...  u7l-ss<E^dB<C>
<C13>  ::=  while<E>d^Tc>
<C14>  ::-  HnTII<E>do<C>
<C15>  :;=  f5F^an.e> = *E>to<E*_|g<C>
<C16>       ^C9>repeat      ™
<C17>   --  <C9>re,,eatwhUe<E>
<C18>       <C9>repeatuntil«E>
<CI9>       <C9>~C1C> 1<G11>|<C1?>|<Cl%>|<C14*|<C15>I
            <C1G>|<C17>]<C18>
<L1>  ::"  <name>_
«.L2>  ::=  _____<constant> ;
«L3>  " *=  d-flult!
<L>     -  <L1»|«L2»|«_3»
<C>     =  <C19»|<L>«C>
<Clist>  ::=  <null> |j.<c><Clist>
<Dlist>  ,,:  <null>|:<D><Dli-t>
<body>    r  <D><Dlist><Clist>|<C*«Clist>
<blbck>   >.z  _<body>j_
```