A PROGRAMMED FORMALIZER FOR A FRAGMENT OF ENGLISH

by

Sylvan Cappell

To the
National Science Foundation
(Grant No. GN 107)
A PROGRAMMED FORMALIZER FOR A FRAGMENT OF ENGLISH

by

Sylvan Cappell

Applied Mathematics Division

To the

National Science Foundation
(Grant No. GN 107)

IMPORTANT NOTICE

Approved for public release by the director of the National Institute of Standards and Technology (NIST) on October 9, 2015

U. S. DEPARTMENT OF COMMERCE
NATIONAL BUREAU OF STANDARDS
A PROGRAMMED FORMALIZER FOR A FRAGMENT OF ENGLISH

This paper describes a computer program written to translate English into symbolic logic by an algorithm developed by Walter Sillars.\(^1\) This algorithm, termed "formalizer," has been written for use in a picture language machine and the vocabulary it applies to is directed to that purpose.\(^2\) As input the program accepts a parsed sentence, one which has been analyzed grammatically. The parser used is described in a paper by Donald Cohen.\(^3\) The formalizer program is quite general and, like the parser, can be modified or expanded for other grammars than the one discussed here.

Sillars' paper describes a formalizer written for a grammar for a fragment of English, Grammar 12R. A corrected version of the grammar and formalizer appears in Appendix I. Grammar 12R is a subset of Grammar 12 developed by B. Kirk Rankin, III.\(^4\)

The formalizer program was written in the COMIT programming language, knowledge of which is assumed in this paper. The program appears in Appendix II together with model input and output.

The parsed input describes which rules of Grammar 12R would be used in the generation of the sentence used as input to the parser. The existence of discontinuous rules in a grammar would necessitate the rearrangement of the rule numbers in the output of the parser, in the general case. However, for the special case of Grammar 12R, this is unnecessary.
The formalizer program will accept any number of parser inputs which are separated by a card with the word NEW on it. The program terminates when there are no more input cards to be read in.

The output of the formalizer program is in symbolic logic. A variable is symbolized by a subscripted X. UQ is the symbol used to represent the universal quantifier, EQ the existential quantifier, UN the Sillars U quantifier (Russell and Whitehead's E!), CNJ the "conjunction" and * - represents Church's 'T. All other symbols are the same as Sillars'.

Throughout most of the program, what Sillars calls pseudo-wffs are stored in the workspace. The program uses only shelves 1-6.

The first part of the program, the part before the rule named GRAM, reads in the input cards until it reaches one with NEW on it. The program assumes that consecutive digits form part of the same rule number and that non-digital characters separate any two rule numbers. The program assembles the rule numbers in the order that they are to be used on shelf 1 and places an *B after the last rule number. An $X_1$ (x/.1 in COMIT) is stored on shelf 3. Whenever Sillars' algorithm calls for a new variable the contents of shelf 3 are used and the subscript value of the X in shelf 3 is increased by 1. The program places the initial symbol, TOP, inclosed in brackets in the workspace. Brackets inclose all pseudo-wffs. *B is used internally to represent [ and *C represents ]. In the output, however, they are converted to "(" and ")" respectively.

- 2 -
The program then branches to the rule named EXIT. It picks up the first grammar rule number on shelf 1 and looks it up in the list GRAM. It then branches to the formalizer instructions corresponding to the grammar rule number, erases the rule number and returns to EXIT. Those grammar rule numbers which have no corresponding formalizer instructions, the ones which Sillars calls not applicable, NA, are simply erased and control returns to EXIT.

After the last grammar rule number has been picked up the program picks up the *B on shelf 1 and finds it in the list GRAM. It then branches to STOP and the symbolic logic statement corresponding to the original sentence is given as output. The program then prepares to read in the next input sentence.

The subroutine named FUNC (the name of its first rule) locates the shortest pseudo-wff containing all occurrences of NV and places an *M immediately before it and an *N immediately after it. It terminates one of the subrules of OUT, which then branches back into the main body of the program.

The program can be modified quite readily. For each new rule assign a new number. Place a card under GRAM with the number on the left and a zero in the right half and branch to a point in the program where the actual formalizer rules are to be executed. Control must finally return to EXIT.

If an instruction requires finding the shortest pseudo-wff containing all occurrences of some constituent $\alpha$, substitute NV for
each $\alpha$, indicate a subrule of OUT to be used, and branch to FUNC. (Subrules can, of course, be added to OUT.) If so desired, every NV can then be replaced by $\alpha$.

To substitute an $X_i$ for each NV and to then increase the subscript of the $X$ in shelf 3 by 1, branch to XOG. Control will automatically be returned to EXIT.

The program treats any symbol beginning with PC or SC as if it consisted only of one of those two respective symbols. The processing of a PC rule without a quantifier begins at APC1 and with a quantifier at APC2 and all SC rules at ASC. Each rule number associated with a DEP or DES rule leads in the program to a series of instructions where quantifiers are generated. The rules UNAT, SQUAT, TQUAT and NQUAT substitute the appropriate quantifier and branch to QUADS where CNJ is substituted for $\Box$. UQUAT substitutes a UQ for Q and PLY for $\Box$. More quantifier rules, accomplishing similar functions, can be added to the program.

A rule can be taken out simply by removing the card with the appropriate rule number under GRAM.

In the final analysis this program is to be used in the predicate evaluator in the picture language machine. The output format in which all propositions which form part of other propositions are inclosed in parentheses should facilitate its use for this application. Thus a proposition of the general form $p \land q \lor r$ would be given in output as $((p \land q) \lor r)$. 

- 4 -
In determining the truth or falsehood of a proposition which has several component propositions, the predicate evaluator must selectively choose which of the latter to begin with. Thus in evaluating propositions of the form \( p \lor q \lor r \) it will usually be advisable to begin by analyzing the truth-value of \( r \). If, however, \( r \) is in some sense very complex, it might be preferable to begin with \( p \).
APPENDIX I. The Grammar and Formalizer

Here is the revised grammar for which the program in Appendix II has been written. For the notational conventions, see Sillars. ¹/

\[
\begin{align*}
\text{TOP} & = \text{CL1} \\
\text{TOP} & = \text{CL3} \\
\text{TOP} & = \text{CL5} \\
\text{TOP} & = \text{CL7} \\
\text{TOP} & = \text{CL11} \quad \text{SR} \\
\text{TOP} & = \text{CL13} \\
\text{TOP} & = \text{CL15} \\
\text{TOP} & = \text{CL17} \\
\text{AAP1} & = \text{WPN} + \text{COM1} \\
\text{AAP1} & = \text{WNP} + \text{COM1} \\
\text{AAF} & = \text{COM1} \quad \text{SR} \\
\text{AAP1} & = \text{WOP} + \text{J} \\
\text{AAP1} & = \text{WOP} + \text{J} \\
\text{AAS} & = \text{WN} + \text{COM1} \\
\text{AAS} & = \text{WN} + \text{COM1} \\
\text{AAS} & = \text{COM1} \quad \text{SR} \\
\text{AAS} & = \text{WO} + \text{J} \\
\text{AAS} & = \text{WO} + \text{J} \\
\text{APN} & = \text{ET} + \text{PNBET} \quad \text{SR(2)} \\
\text{AR} & = \text{are} \quad \text{NA} \\
\text{ARNTC} & = \text{aren't}
\end{align*}
\]
CL1 = SUB1X + PRE0
CL3 = SUB6X + PRE1
CL5 = SUB4X + PRE3
CL7 = SUB7W + PRE4
CL11 = SUB10 + PRE11
CL13 = SUB11 + PRE13
CL15 = SUB13 + PRE15
CL17 = SUB16 + PRE17

\{ \begin{align*}
CL_i &= PRE_j(SUB_k) \\
CO &= , \quad \text{NA} \\
COL &= \text{Color} \quad \text{COL} = \text{Smc} \\
COL &= \text{Size} \quad \text{COL} = \text{Smz} \\
COM1 &= \text{LP} \\
COM1 &= \text{RE} \\
COM1 &= \text{COM1} \\
COM2 &= \text{PNP2} \quad \text{SR} \\
COM3 &= \text{COM1} \\
COM3 &= \text{SNP2} \\
\end{align*} \}
Procedure for rewriting \( (QNV) [NN(NV)] \)

where NN is either N or NPLUR:

\( A(NV) \)

DE is any \( DE^i \) or \( DES^i \)

1. If \( DES = \text{the} \), \( Q = U \)

If \( DE = \text{No} \) or any, \( (QNV) = T(\Xi NV) \)

If quantifier is deleted, \( Q = \Xi \). In all other cases,

\[ \begin{align*}
\text{DEP12} &= \text{some} \\
\text{DEP12} &= \text{the} \\
\text{DEP12} &= \text{two} \\
\text{DEP12} &= \text{all} \\
\text{DEP12} &= \text{no} \\
\text{DEP2} &= \text{some} \\
\text{DEP2} &= \text{the} \\
\text{DEP2} &= \text{two} \\
\text{DEP2A} &= \text{any} \\
\text{DEP5S} &= \text{some} \\
\text{DEP5S} &= \text{two} \\
\text{DEP5S} &= \text{no} \\
\text{DES1} &= \text{a} \\
\text{DES1} &= \text{one} \\
\text{DES13} &= \text{the} \\
\text{DES13} &= \text{a} \\
\text{DES13} &= \text{every} \\
\text{DES13} &= \text{one} \\
\text{DES13} &= \text{each} \\
\text{DES16} &= \text{the} \\
\text{DES16} &= \text{a} \\
\text{DES16} &= \text{one}
\end{align*} \]

Then replace all instances of NV by the next (in alphabetic order) individual variable which has not yet been used.

2. If \( DE = \text{all}, \text{each} \) or \( \text{every} \), then

\[ \Box = \Xi \text{, otherwise } \Box = \&. \]

3. Replace every expression of the form \( Vi(Vj) \), where \( Vi \) and \( Vj \) are individual variables, by \( (Vi = Vj) \).
DES16 = each
DES16 = every
DES16 = no
DES2 = the
DES2 = a
DES2 = one
DES7 = a
DES7 = one
DES7 = no

DUM1 = R1 + NPH    DUM1(a, NPH) = R1(a, NPH1, NPH2)
DUM2 = SNASN
DUM2 = PNAPN    SR
DUM2 = PNBET
ES = ET + SNP2    SR(2)
ET = and    NA
F1 = T2 + G1
F2 = T2 + G2    SR(2)

G1 = right    MT + G1 = Mrot; G1 = Rt
G1 = left     MT + G1 = Molf; G1 = Lf
G1 = top      MT + G1 = Mtop; G1 = Tp
G1 = bottom   MT + G1 = Mbot; G1 = Bot
G2 = center   MT + G2 = Mcen; G2 = Cen
G2 = middle   MT + G2 = Mmid; G2 = Mid
I = is    NA

- 9 -
\begin{align*}
IA &= J1 \\
IA &= J2 \\
IA &= J1 + J2 \quad \text{IA(a)} = J1(a) \, \& \, J2 \,(a) \\
\text{ISNTC} &= \text{isn't} \\
J &= J1 \\
J &= J2 \\
J1 &= \text{big} \quad J1 = \text{Bg} \\
J1 &= \text{little} \quad J1 = \text{Lt} \\
J1 &= \text{large} \quad J1 = \text{Lg} \\
J1 &= \text{small} \quad J1 = \text{Sm} \\
J2 &= \text{black} \quad J2 = \text{Bk} \\
J2 &= \text{white} \quad J2 = \text{Wh} \\
\text{JER} &= \text{bigger} \quad \text{JER} = \text{Bgr} \\
\text{JER} &= \text{littler} \quad \text{JER} = \text{Ltr} \\
\text{JER} &= \text{larger} \quad \text{JER} = \text{Lgr} \\
\text{JER} &= \text{smaller} \quad \text{JER} = \text{Smr} \\
LL &= Z1 \\
LL &= Z2 \\
\text{LP} &= LL \\
\text{MO} &= \text{more} \\
\text{MT} &= \text{MO : TN} \\
N &= \text{triangle} \quad N = \text{Tr} \\
N &= \text{square} \quad N = \text{Sq} \\
N &= \text{circle} \quad N = \text{Cir}
\end{align*}
NPH = SNP2 SR
NPH = PNP2 SR
NPLUR = triangles NPLUR = Tr
NPLUR = squares NPLUR = Sq
NPLUR = circles NPLUR = Cir
NT = not NT = \top
P = in P = In
P = near P = Nr
P = below P = Bel
P = above P = Ab
P = touching P = Tch
P1 = to
P1 = on
P1 = at NA
P2 = in
P3 = between P3 = Bet
P4 = of NA
PA1 = IA : AAP1 PA1(a) = IA(a) & AAP1(a)
PA1 = \emptyset : AAP1 \{ SR
PA1 = IA \}
PA2 = IA : AAP1 PA2(a) = IA(a) & AAP1(a)
PA2 = IA SR
\[
\begin{align*}
PC_{12} &= \text{DEP}_{12}:\text{NFLUR} \\
PC_{2} &= \text{DEP}_{2}:\text{NFLUR} \\
PC_{2} &= \emptyset :\text{NFLUR} \\
PC_{2A} &= \text{DEP}_{2A}:\text{NFLUR} \\
PC_{2A} &= \emptyset :\text{NFLUR} \\
PC_{4} &= \text{DEP}_{2} + \text{NFLUR} \\
\{A(PC_{i}) = (QNV) [\text{NFLUR(NV)} \square A(NV)]
\}
PC_{4} &= \text{NFLUR} \\
PC_{412} &= \text{DEP}_{12} + \text{NFLUR} \\
PC_{4A} &= \text{DEP}_{2A} + \text{NFLUR} \\
PC_{4A} &= \text{NFLUR} \\
PC_{4S5} &= \text{DEP}_{5S} + \text{NFLUR} \\
PC_{4S5} &= \text{NFLUR} \\
PC_{4S6} &= PC_{4S5} \quad \text{SR} \\
PN_{APN} &= \text{PNBET} + \text{APN} \quad A(PN_{APN}) = A(\text{PNBET}, \text{APN}) \\
PN_{BET} &= \text{PC}_{2} + \text{PA}_{1} \quad A(\text{PNBET}) = \text{PA}_{1}(\text{PC}_{2}) \& A(\text{PC}_{2}) \\
PN_{BET} &= \text{PC}_{4} \quad \text{SR} \\
PN_{P12} &= PC_{412} \quad \text{SR} \\
PN_{P12} &= PC_{12} + \text{PA}_{1} \quad A(PN_{P12}) = \text{PA}_{1}(PC_{12}) \& A(PC_{12}) \\
PN_{P2} &= PC_{2} + \text{PA}_{1} \quad A(PN_{P2}) = \text{PA}_{1}(PC_{2}) \& A(PC_{2}) \\
PN_{P2} &= PC_{4} \quad \text{SR} \\
PN_{P2A} &= PC_{2A} + \text{PA}_{1} \quad A(PN_{P2A}) = \text{PA}_{1}(PC_{2A}) \& A(PC_{2A}) \\
PN_{P2A} &= PC_{4A} \quad \text{SR} \\
PN_{P6S} &= PC_{4S6} \quad \text{SR}
\end{align*}
\]
\[
PPLB = P3 + DUM2 \quad \text{PPLB}(a) = P3(a, DUM2)
\]
\[
PREO = VPO
\]
\[
PRE1 = VP1
\]
\[
PRE11 = VP11
\]
\[
PRE13 = VP13
\]
\[
PRE15 = VP15
\]
\[
PRE17 = VP17
\]
\[
PRE3 = VP3
\]
\[
PRE4 = VP4
\]
\[
R = R1
\]
\[
R = R2
\]
\[
R = R4
\]
\[
R = R5
\]
\[
R1 = LL + F4 \quad \text{SR}(1)
\]
\[
R2 = JER + TN \quad \text{SR}(1)
\]
\[
R4 = P \quad \text{SR}
\]
\[
R5 = MT + DUM1 \quad R5(a,b) = MT \ DUM1(a,b)
\]
\[
RE = R + NPH \quad \text{RE}(a) = R(a, NPH)
\]
\[
RE = PPLB \quad \text{SR}
\]
\[
SA = IA : AAS \quad \text{SA}(a) = IA(a) \ & \ AAS(a)
\]
\[
SA = IA \quad \text{SR}
\]
\[
SA = \emptyset : AAS \quad \text{SR}
\]
\[
SAME = \text{same} \quad \text{NA}
\]
\[ \begin{align*}
SC_1 &= DES_1 : N \\
SC_{13} &= DES_{13} : N \\
SC_{16} &= DES_{16} : N \\
SC_2 &= DES_2 : N \\
SC_3 &= DES_1 + N \\
SC_{37} &= DES_7 + N \\
SC_4 &= DES_2 + N \\
SC_{413} &= DES_{13} + N \\
SC_{416} &= DES_{16} + N \\
SC_7 &= DES_7 : N \\
SNASN &= SNP_2 + ES \\
SNP_1 &= SC_1 + SA \\
SNP_{13} &= SC_3 \\
SNP_{13} &= SC_{413} \\
SNP_{16} &= SC_{13} + SA \\
SNP_{16} &= SC_{13} + SA \\
SNP_{16} &= SC_{416} + SA \\
SNP_2 &= SC_4 \\
SNP_7 &= SC_{37} \\
SNP_7 &= SC_7 + SA \end{align*} \]

\[ A(SC_i) = (QNV) [N(NV)A(NV)] \]

\[ A(SNASN) = A(SNP_2, ES) \]

\[ A(SNP_1) = SA(SC_1) \& A(SC_1) \]

\[ A(SNP_{13}) = SA(SC_{13}) \& A(SC_{13}) \]

\[ A(SNP_{16}) = SA(SC_{16}) \& A(SC_{16}) \]

\[ A(SNP_2) = SA(SC_4) \]

\[ A(SNP_7) = SA(SC_7) \& A(SC_7) \]
\[
\begin{align*}
\text{SUB1} &= \text{PNP2A} \\
\text{SUB10} &= \text{PNP2} \\
\text{SUB11} &= \text{PNP12} \\
\text{SUB13} &= \text{SNP13} \\
\text{SUB16} &= \text{SNP16} \\
\text{SUB1X} &= \text{SUB1Y} \\
\text{SUB1Y} &= \text{THERE} : \text{SUB1} \\
\text{SUB3} &= \text{SNP1} \\
\text{SUB4X} &= \text{SUB4Y} \\
\text{SUB4Y} &= \text{THERE} : \text{SUB3} \\
\text{SUB6} &= \text{PNP6S} \\
\text{SUB6X} &= \text{SUB6Y} \\
\text{SUB6Y} &= \text{THERE} : \text{SUB6} \\
\text{SUB7} &= \text{SNP7} \\
\text{SUB7W} &= \text{SUB7Y} \\
\text{SUB7Y} &= \text{THERE} : \text{SUB7} \\
\text{T2} &= \text{the} \\
\text{TH} &= \text{that} \\
\text{TH} &= \text{which} \\
\text{THERE} &= \text{there} \\
\text{THSAM} &= \text{T2 + SAME} \\
\text{TN} &= \text{than}
\end{align*}
\]
\[
\begin{align*}
VPO &= \text{ARNTC} : \text{COM1} \\
VP1 &= \text{AR} : \text{COM1} \quad \text{SR(2)} \\
VP11 &= \text{ARNTC} + \text{COM2} \\
VP13 &= \text{AR} + \text{COM2} \quad \text{SR(2)} \\
VP15 &= \text{ISNTC} + \text{COM3} \\
VP17 &= \text{I} + \text{COM3} \quad \text{SR(2)} \\
VP3 &= \text{ISNTC} : \text{COM1} \\
VP4 &= \text{I} : \text{COM1} \quad \text{SR(2)} \\
WA &= \text{TH} + \text{AR} \quad \text{NA} \\
WI &= \text{TH} + \text{I} \quad \text{NA} \\
WN &= \text{WI} + \text{NT} \quad \text{WN} = \text{NT} \\
WN &= \text{WI} \quad \text{WN} = \emptyset \\
WN &= \text{NT} \quad \text{WN} = \text{NT} \\
WNP &= \text{WA} + \text{NT} \quad \text{WNP} = \text{NT} \\
WNP &= \text{WA} \quad \text{WNP} = \emptyset \\
WNP &= \text{NT} \quad \text{WNP} = \text{NT} \\
WO &= \text{WI} + \text{NT} \quad \text{WO} = \text{NT} \\
WO &= \text{WI} \quad \text{WO} = \emptyset \\
WOP &= \text{WA} + \text{NT} \quad \text{WOP} = \text{NT} \\
WOP &= \text{WA} \quad \text{WOP} = \emptyset \\
Z1 &= \text{P1} + \text{P1} \\
Z2 &= \text{P2} + \text{P2} \quad \text{SR(2)}
\end{align*}
\]
APPENDIX II. The Formalizer Program

The COMIT program which realizes the algorithm is listed below. To illustrate the output format of the program, we present the formalization of the sentence:

There aren't any triangles on the right.

The parse of this sentence is:

2(39(351(352(381(-THERE))(339(221(195(160(-TRIANGLES)))))))
(239(394(35(-AREN*T))(65(148(146(427(172(-ON))(119(378(-THE))
(121(-RIGHT))))))))))

(Where the numbers are names of the rules of 12R which were used in generating it.) The output for this sentence from the formalizer program is:

*- + *( + *( + EQ + X / .1 + *) + *( + *( + TR + *( + X / .1 + *) + * CNJ + *( + RT + *( + X / .1 + *) + *) + *) + *) + * + * + * +
### APPENDIX II

#### FORMALIZER

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>STAR</strong></td>
<td>$=\ast B+\ast C+X/\ast 1$</td>
</tr>
<tr>
<td><strong>BEGIN</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>EXIT</strong></td>
<td>$=\ast A+1$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>FINISH</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>BEGINA</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>SUB</strong></td>
<td>$1$</td>
</tr>
<tr>
<td><strong>GRAM</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>SQUAT</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>TOUQAT</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>SQUAT</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>TOUQAT</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>SQUAT</strong></td>
<td>$=$</td>
</tr>
<tr>
<td><strong>TOUQAT</strong></td>
<td>$=$</td>
</tr>
</tbody>
</table>

---

### Notes

- The table indicates various operations and assignments within a formalization process.
- The commands include arithmetic and logical operations.
- The values associated with each command seem to be placeholders or identifiers for a specific context or calculation.

---

18-19
<table>
<thead>
<tr>
<th></th>
<th></th>
<th>= 0</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>7</em>5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>7</em>6</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>7</em>7</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>7</em>8</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>8</em>3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>8</em>9</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>0</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>1</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>2</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>4</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>6</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>7</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>9</em>8</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*0</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*1</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*2</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*4</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*6</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*7</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*8</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>0*9</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*0</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*1</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*2</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*4</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>1*6</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*0</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*1</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*2</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*4</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>2*6</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*0</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*1</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*2</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*4</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*6</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*7</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*8</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>3*9</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4*0</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4*1</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4*2</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4*3</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4*4</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4*5</td>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td><em>1</em>4<em>4</em>6</td>
<td>= 0</td>
<td>R146</td>
</tr>
<tr>
<td><em>1</em>4*7</td>
<td>= 0</td>
<td>R147</td>
</tr>
<tr>
<td><em>1</em>4*8</td>
<td>= 0</td>
<td>R148</td>
</tr>
<tr>
<td><em>1</em>5*2</td>
<td>= 0</td>
<td>R152</td>
</tr>
<tr>
<td><em>1</em>5*3</td>
<td>= 0</td>
<td>R153</td>
</tr>
<tr>
<td><em>1</em>5*4</td>
<td>= 0</td>
<td>R154</td>
</tr>
<tr>
<td><em>1</em>5*6</td>
<td>= 0</td>
<td>R156</td>
</tr>
<tr>
<td><em>1</em>5*8</td>
<td>= 0</td>
<td>R158</td>
</tr>
<tr>
<td><em>1</em>5*9</td>
<td>= 0</td>
<td>R159</td>
</tr>
<tr>
<td><em>1</em>6*0</td>
<td>= 0</td>
<td>R160</td>
</tr>
<tr>
<td><em>1</em>6*1</td>
<td>= 0</td>
<td>R161</td>
</tr>
<tr>
<td><em>1</em>6*2</td>
<td>= 0</td>
<td>R162</td>
</tr>
<tr>
<td><em>1</em>6*4</td>
<td>= 0</td>
<td>R164</td>
</tr>
<tr>
<td><em>1</em>6*6</td>
<td>= 0</td>
<td>R166</td>
</tr>
<tr>
<td><em>1</em>6*7</td>
<td>= 0</td>
<td>R167</td>
</tr>
<tr>
<td><em>1</em>6*8</td>
<td>= 0</td>
<td>R168</td>
</tr>
<tr>
<td><em>1</em>6*9</td>
<td>= 0</td>
<td>R169</td>
</tr>
<tr>
<td><em>1</em>7*0</td>
<td>= 0</td>
<td>R170</td>
</tr>
<tr>
<td><em>1</em>7*5</td>
<td>= 0</td>
<td>R175</td>
</tr>
<tr>
<td><em>1</em>7*7</td>
<td>= 0</td>
<td>R177</td>
</tr>
<tr>
<td><em>1</em>7*8</td>
<td>= 0</td>
<td>R178</td>
</tr>
<tr>
<td><em>1</em>7*9</td>
<td>= 0</td>
<td>R179</td>
</tr>
<tr>
<td><em>1</em>8*0</td>
<td>= 0</td>
<td>R180</td>
</tr>
<tr>
<td><em>1</em>8*1</td>
<td>= 0</td>
<td>R181</td>
</tr>
<tr>
<td><em>1</em>8*2</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>1</em>8*3</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>1</em>8*4</td>
<td>= 0</td>
<td>APC1</td>
</tr>
<tr>
<td><em>1</em>8*6</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>1</em>8*7</td>
<td>= 0</td>
<td>APC1</td>
</tr>
<tr>
<td><em>1</em>9*0</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>1</em>9*1</td>
<td>= 0</td>
<td>APC1</td>
</tr>
<tr>
<td><em>1</em>9*2</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>1</em>9*4</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>1</em>9*5</td>
<td>= 0</td>
<td>APC1</td>
</tr>
<tr>
<td><em>2</em>0*0</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>2</em>0*1</td>
<td>= 0</td>
<td>APC1</td>
</tr>
<tr>
<td><em>2</em>0*2</td>
<td>= 0</td>
<td>R202</td>
</tr>
<tr>
<td><em>2</em>0*5</td>
<td>= 0</td>
<td>R207</td>
</tr>
<tr>
<td><em>2</em>0*6</td>
<td>= 0</td>
<td>APC2</td>
</tr>
<tr>
<td><em>2</em>0*7</td>
<td>= 0</td>
<td>R208</td>
</tr>
<tr>
<td><em>2</em>0*8</td>
<td>= 0</td>
<td>R209</td>
</tr>
<tr>
<td><em>2</em>0*9</td>
<td>= 0</td>
<td>R210</td>
</tr>
<tr>
<td><em>2</em>1*0</td>
<td>= 0</td>
<td>R211</td>
</tr>
<tr>
<td><em>2</em>1*1</td>
<td>= 0</td>
<td>R212</td>
</tr>
<tr>
<td><em>2</em>1*2</td>
<td>= 0</td>
<td>R213</td>
</tr>
<tr>
<td><em>2</em>1*3</td>
<td>= 0</td>
<td>R214</td>
</tr>
<tr>
<td><em>2</em>1*4</td>
<td>= 0</td>
<td>R215</td>
</tr>
<tr>
<td><em>2</em>1*5</td>
<td>= 0</td>
<td>R216</td>
</tr>
<tr>
<td><em>2</em>1*6</td>
<td>= 0</td>
<td>R220</td>
</tr>
<tr>
<td><em>2</em>2*0</td>
<td>= 0</td>
<td>R221</td>
</tr>
<tr>
<td><em>2</em>2*1</td>
<td>= 0</td>
<td>R222</td>
</tr>
<tr>
<td><em>2</em>2*2</td>
<td>= 0</td>
<td>R235</td>
</tr>
<tr>
<td><em>2</em>3*5</td>
<td>= 0</td>
<td>R236</td>
</tr>
<tr>
<td><em>2</em>3*6</td>
<td>= 0</td>
<td>R237</td>
</tr>
<tr>
<td><em>2</em>3*7</td>
<td>= 0</td>
<td>R238</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>9</td>
</tr>
</tbody>
</table>
\*3*7*1 = 0 R371
\*3*7*2 = 0 R372
\*3*7*4 = 0 R374
\*3*7*5 = 0 R375
\*3*7*7 = 0 R377
\*3*8*6 = 0 R386
\*3*8*7 = 0 R387
\*3*8*8 = 0 R388
\*3*8*9 = 0 R389
\*3*9*0 = 0 R390
\*3*9*1 = 0 R391
\*3*9*2 = 0 R392
\*3*9*3 = 0 R393
\*3*9*4 = 0 R394
\*3*9*5 = 0 R395
\*3*9*7 = 0 R397
\*3*9*9 = 0 R399
\*4*0\*1 = 0 R401
\*4*0\*3 = 0 R403
\*4*0\*6 = 0 R406
\*4*0\*7 = 0 R407
\*4*1\*7 = 0 WNNT
\*4*1\*8 = 0 WNO
\*4*1\*9 = 0 WNNT
\*4*2\*0 = 0 WNOPNT
\*4*2\*1 = 0 WNPT
\*4*2\*2 = 0 WNPNT
\*4*2\*3 = 0 WONT
\*4*2\*4 = 0 WOPN
\*4*2\*5 = 0 WOPN
\*4*2\*6 = 0 WOPN
\*4*2\*7 = 0 R427
\*4*2\*8 = 0 R428
\*8 = 0 STOP
$1 = 0

R2 TOP=CL*1 EXIT
R4 TOP=CL*3 EXIT
R6 TOP=CL*5 EXIT
R8 TOP=CL*7 EXIT
R10 TOP=CL*1\*1 EXIT
R12 TOP=CL*1\*3 EXIT
R14 TOP=CL*1\*5 EXIT
R16 TOP=CL*1\*7 EXIT
R25 AAP\*1=WN+COM\*1 EXIT
R26 AAP\*1=COM\*1 EXIT
R27 AAP\*1=WOP+J EXIT
R28 AAS=WN+COM\*1 EXIT
R29 AAS=COM\*1 EXIT
R30 AAS=W0+J EXIT
R33 APN=PNET EXIT
R39 CL\*1=PRE*0+**(+SUB\*1X**) EXIT
R53 CL\*3=PRE*1+**(+SUB\*6X**) EXIT
R59 CL\*5=PRE*3+**(+SUB\*4X**) EXIT
R61 CL\*7=PRE*4+**(+SUB\*7W**) EXIT
R41 CL\*1\*1=PRE*1\*1+**(+SUB\*1\*0**) EXIT
R43  CL*1#3=PRE*1#3+(+SUB*1#3+)
R45  CL*1#5=PRE*1#5+(+SUB*1#3+)
R47  CL*1#7=PRE*1#7+(+SUB*1#6+)
R63  COL=SMC
R64  COL=SMZ
R65  COM*1=LP
R66  COM*1=RF
R67  COM*2=COM*1
R68  COM*2=PNP*2
R69  COM*3=COM*1
R70  COM*3=SNP*2
UNAT  Q=UN
SQUAT  Q=S
QUADS  QUAD=CNJ
TQUAT  Q=T
EQUAT  Q=FQ
UQAT  Q=UQ
*  QUAD=PLY
NOQUAT  *(+Q*#---(+#)
R111  DUM*1#1+(#.+NPH#)=#R*1#2+3#4#5#4#5#6
R112  DUM*2=SNASN
R113  DUM*2=PNAPN
R114  DUM*2=PNBET
R117  ES=SNP*2
R119  F*1=G*1
R120  F*2=G*2
R121  MT+G*1=MORT
*  G*1=RT
R122  MT+G*1=MOLF
*  G*1=LF
R123  MT+G*1=MTOP
*  G*1=TP
R124  MT+G*1=MBOX
*  G*1=BOT
R125  MT+G*2=MCN
*  G*2=CFN
R126  MT+G*2=MMID
*  G*2=MID
R130  IA=J*1
R131  J=J*2
R132  IA*1#1*(#.+#)##C=1#1+J#1#3#4#5#6+CNJ#1+J*2#3#4#5#6#6
R134  J=J*1
R135  J=J*2
R136  J*1=BG
R137  J*1=LT
R138  J*1=LG
R139  J*1=SM
R140  J*2=BK
R141  J*2=WH
R142  JER=BRGR
R143  JER=LTR
R144  JFR=LGR
R145  JFR=SMR
R146  LL=Z*1
R147  LL=Z*2
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R14A</td>
<td>LP=LL</td>
<td></td>
</tr>
<tr>
<td>R152</td>
<td>N=TR</td>
<td></td>
</tr>
<tr>
<td>R153</td>
<td>N=SQ</td>
<td></td>
</tr>
<tr>
<td>R154</td>
<td>N=CIR</td>
<td></td>
</tr>
<tr>
<td>R156</td>
<td>NO=NPLUR</td>
<td></td>
</tr>
<tr>
<td>R158</td>
<td>NPH=SNP*2</td>
<td></td>
</tr>
<tr>
<td>R159</td>
<td>NPH=PNP*2</td>
<td></td>
</tr>
<tr>
<td>R160</td>
<td>NPLUR=TR</td>
<td></td>
</tr>
<tr>
<td>R161</td>
<td>NPLUR=SQ</td>
<td></td>
</tr>
<tr>
<td>R162</td>
<td>NPLUR=CIR</td>
<td></td>
</tr>
<tr>
<td>R164</td>
<td><em>R+NT=</em>+1</td>
<td></td>
</tr>
<tr>
<td>R166</td>
<td>P=IN</td>
<td></td>
</tr>
<tr>
<td>R167</td>
<td>P=NR</td>
<td></td>
</tr>
<tr>
<td>R168</td>
<td>P=REL</td>
<td></td>
</tr>
<tr>
<td>R169</td>
<td>P=AB</td>
<td></td>
</tr>
<tr>
<td>R170</td>
<td>P=TCH</td>
<td></td>
</tr>
<tr>
<td>R175</td>
<td>P*3=BET</td>
<td></td>
</tr>
<tr>
<td>R177</td>
<td>PA<em>1+(+%+%)=<em>B+IA+(+3+</em>)+**C+CNJ+B+AAP</em>1+2+3+4+**C</td>
<td></td>
</tr>
<tr>
<td>R178</td>
<td>PA<em>1=AAP</em>1</td>
<td></td>
</tr>
<tr>
<td>R179</td>
<td>PA*1=IA</td>
<td></td>
</tr>
<tr>
<td>R180</td>
<td>PA<em>2+(+%+%)=<em>B+IA+(+3+</em>)+**C+CNJ+B+AAP</em>1+2+3+4+**C</td>
<td></td>
</tr>
<tr>
<td>R181</td>
<td>PA*2=IA</td>
<td></td>
</tr>
<tr>
<td>APC1</td>
<td>PC=NV</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>BACH A,OUT A</td>
</tr>
<tr>
<td>APC2</td>
<td>PC=NV</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>BACH B,OUT A</td>
</tr>
<tr>
<td>BACH A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BACHB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BACHA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BACHB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBACH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EQUA</td>
<td>X+(+%+%)=2+1+**+3+4</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>X3</td>
</tr>
<tr>
<td>CRACH</td>
<td>$1=1/11</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$1=1+1</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>NV=</td>
<td></td>
</tr>
<tr>
<td>R202</td>
<td>PC<em>4S</em>6=PC</td>
<td></td>
</tr>
<tr>
<td>R207</td>
<td>PC*6S=PC</td>
<td></td>
</tr>
<tr>
<td>R208</td>
<td>$1+(+PNAPN++)=1+2+PNBET++APN+4</td>
<td></td>
</tr>
<tr>
<td>R209</td>
<td>PNBET=NV</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>OUT B</td>
</tr>
<tr>
<td>XOB</td>
<td>NV=PC</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>R210</td>
<td>PNBET=PC</td>
<td></td>
</tr>
<tr>
<td>R211</td>
<td>PNP<em>1</em>2=NV</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>OUT C</td>
</tr>
<tr>
<td>XOC</td>
<td>NV=V</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>R212</td>
<td>PNP<em>1</em>2=PC</td>
<td></td>
</tr>
<tr>
<td>R213</td>
<td>PNP<em>1</em>2=NV</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>OUT B</td>
</tr>
<tr>
<td>R214</td>
<td>PNP*2=NV</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>$</td>
<td>OUT B</td>
</tr>
<tr>
<td>R215</td>
<td>PNP*2=PC</td>
<td></td>
</tr>
<tr>
<td>R216</td>
<td>PNP*2=NV</td>
<td></td>
</tr>
</tbody>
</table>
P220  PNP*2A=NV  //OUT C
P221  PNP*2A=PC
P222  PNP*2A=NV
P223  PNP*6S=NV  //OUT C
P234  PNP*6S=PC*4S*6
P235  PNP*6S=NV  //OUT B
P236  PNP*#(+$#=#)=P*#3+2+3+,+DUM*2+4
P237  PRE*0=VP*0
P238  PRE*1=VP*1
P239  PRE*1*1=VP*1*1
P240  PRE*1*3=VP*1*3
P241  PRE*1*5=VP*1*5
P242  PRE*1*7=VP*1*7
P243  PRE*3=VP*3
P244  PRE*4=VP*4
P294  R=R*1
P295  R=R*2
P296  R=R*4
P297  R=R*5
P300  R*1=IL
P301  R*2=JFP
P302  R*4=D
R303  R*5+(+$#=#)=MT+DUM*1+2+3+4
R304  R*6=COL
R305  RE+(+$#=#)=R+2+3+,+NP*4
R306  REPPLO:
R307  SA+(+$#=#)=B+IA+2+3+4+C+CNJ+H+ASS+2+3+4*C
R310  SA=1A
R311  SA=AAS
ASC  SC=NV  //OUT D,RACH R
XOF  NV=SC  //OUT F
R324  SNP*1=NV  //OUT F
R325  SNP*1=SC
R326  SNP*1=SC
R327  SNP*1*3=SC
R328  SNP*1*3=NV
R329  SNP*1*6=SC
R330  SNP*1*6=NV
R331  SNP*2=SC
R332  SNP*7=SC
R333  SNP*7=NV
R334  SUB*1=PNP*2A
R335  SUB*1*0=PNP*2
R344 SUB*1*1=SNP*1*2
R347 SUB*1*3=SNP*1*3
R350 SUB*1*6=SNP*1*6
R351 SUB*1*X=SUB*1*Y
R352 SUB*1*Y=SUB*1
R356 SUB*3=SNP*1
R363 SUB*4*X=SNP*4*Y
R364 SUB*4*Y=SNP*4
R369 SUB*6=SNP*6
R371 SUB*6*X=SNP*6*Y
R372 SUB*6*Y=SUB*6
R374 SUB*7=SNP*7
R375 SUB*7*X=SNP*7*Y
R377 SUB*7*Y=SUB*7
R386 V=NV
* $ //OUT F
XOF NV=V*1
* $ //OUT G
XOF NV=V*1
* $ //OUT H
XOF NV=V*1
* $ //OUT I
R392 V*4=NV
* $ //OUT 1
R392 V*4=NV
* $ //OUT 1
R394 *P+VP*0=#+-1+COM*1
* VP*0=-+1+COM*1
* R395 VP*1=-COM*1
R397 *B+VP*1*1=#+-1+COM*2
* VP*1*1=#+-1+COM*2
R399 VP*1*3=-COM*2
R401 *R+VP*1*5=#+-1+COM*1
* VP*1*5=#+-1+COM*3
R403 VP*1*7=-COM*3
R406 *R+VP*2=#+-1+COM*3
* VP*3=#+-1+COM*3
R407 VP*4=-COM*1
WNNT WN=NT
WN0 WN=0
WNPNT WNP=NT
WNP0 WNP=0
WONT WO=NT
WO0 WO=0
WOPN WOP=NT
WOP0 WOP=0
R427 Z*1=F*1
R428 Z*2=F*2
ZZ2

FUNC

ZY1

* 

ZY2

* 

ZY1

* 

YY1

* 

YY2

* 

YY2

* 

YY1

* 

**YY1**

* 

**YY2**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**

* 

**YY1**

* 

**YY2**
REFERENCES


