```
||INPUT ||LALR ||LISTS ||CHAIN
||TABLE1
        id
        num
        lnum
        string
        lstring
        char

        .
        ;
        :
        ..
        =>
        ←
        =
        #
        <
        >
        <=
        >=
        ~
        +
        -
        *
        /
        ↑
        .
        @
        !

        INTEGER
        CARDINAL
        CHARACTER
        BOOLEAN
        STRING
        RECORD
        POINTER
        ARRAY
        DESCRIPTOR
        PROCEDURE
        PORT
        SIGNAL
        ERROR
        PROCESS
        PROGRAM
        MONITOR
        RELATIVE
        LONG
        TYPE
        FRAME

        TO
        ORDERED
        BASE
        OF
        PACKED
        RETURNS
        MONITORED
        OVERLAID
        COMPUTED
        MACHINE
        DEPENDENT

        DIRECTORY
        DEFINITIONS
        IMPORTS
        EXPORTS
        SHARES
        LOCKS
        USING
        PUBLIC
        PRIVATE
        ENTRY
        INTERNAL
```

```
        CODE

        ABS
        AND
        MAX
        MIN
        MOD
        NOT
        OR
        LENGTH
        NEW
        START
        FORK
        JOIN
        LOOPHOLE
        SIZE
        FIRST
        LAST
        MEMORY
        REGISTER

        NULL
        IF
        THEN
        ELSE
        WITH
        FROM
        FOR
        INCREASING
        DECREASING
        IN
        THROUGH
        UNTIL
        WHILE
        REPEAT
        FINISHED
        RETURN
        EXIT
        LOOP
        GOTO
        GO
        WAIT
        RESTART
        NOTIFY
        BROADCAST
        STOP
        RESUME
        CONTINUE
        RETRY
        TRANSFER
        STATE
        OPEN
        ENABLE
        ANY
        EXITS

        )
        ]
        }
        END
        ENDLOOP
        ENDCASE

        (
        [
        {
        BEGIN
        DO
        SELECT

        EOF
||TABLE2
        goal
        unit
        directory
        includelist
```

includeitem
definitions
module
classhead
defhead
defbody
locks
interface
imports
exports
modulelist
moduleitem
shares
declist
declaration
attributes
entry
idlist
idlist'
identlist
identlist'
typeexp
typeid
typecons
monitored
dependent
reclist
pairlist
pairitem
typelist
variantpair
variantpart
vcasehead
tagtype
variantlist
variantitem
subreclist
ordered
base
pointertype
pointerprefix
array
indextype
transfermode
arguments
arglist
returnlist
fieldlist
initialization
initvalue
codelist
procaccess
statement
block
blockhead
begin
bindlist
binditem
exits
elsepart
casehead
casestmtlist
casestmtitem
caselabel
casetest
otherpart
forclause
direction
dotest
do
doexit
exitlist
exititem
enables
catchhead
catchlist

```
        catchitem
        catchcase
        lhslist
        statementlist
        statementlist'
        transfer
        optargs
        explist
        orderlist
        keylist
        keyitem
        optexp
        exp
        transferop
        caseexplist
        caseexpitem
        disjunct
        conjunct
        negation
        not
        relation
        optrelation
        relop
        relationtail
        range
        interval
        bounds
        sum
        addop
        product
        multop
        factor
        primary
        desclist
        prefixop
        typeop
        lhs
        qualifier
        memory
||TABLE3
goal            ::=0    . unit .
                  |0    . unit ..
unit            ::=10   directory definitions module
directory       ::=4
                  |9    DIRECTORY includelist ;
includelist     ::=6    includeitem
                  |7    includelist , includeitem
includeitem     ::=11   id : FROM string
                  |222  id : FROM string USING [ idlist ]
definitions     ::=4
                  |0    DEFINITIONS FROM idlist ;
module          ::=12   id : classhead = attributes block
                  |12   id : defhead = attributes defbody
classhead       ::=13   PROGRAM arguments interface
                  |201  MONITOR arguments locks interface
defhead         ::=14   DEFINITIONS shares
defbody         ::=21   begin declist END
locks           ::=202
                  |203  LOCKS primary
                  |204  LOCKS primary USING id : typeexp
interface       ::=0    imports exports shares
imports         ::=4
                  |9    IMPORTS modulelist
exports         ::=4
                  |0    EXPORTS idlist
modulelist      ::=6    moduleitem
                  |7    modulelist , moduleitem
moduleitem      ::=15   id
                  |16   id : id
shares          ::=4
                  |0    SHARES idlist
declist         ::=5
                  |7    declist declaration ;
declaration     ::=22   identlist attributes entry typeexp initialization
                  |23   identlist attributes TYPE = attributes typeexp
attributes      ::=24
```

```
                |25   PUBLIC
                |26   PRIVATE
entry          ::=223
                |224  ENTRY
                |225  INTERNAL
idlist         ::=8    idlist'
idlist'        ::=27   id
                |28   id , idlist'
identlist      ::=8    identlist'
identlist'     ::=27   id :
                |28   id , identlist'
typeexp        ::=1    id
                |0    typeid
                |0    typecons
typeid         ::=29   INTEGER
                |30   CARDINAL
                |31   CHARACTER
                |32   BOOLEAN
                |33   STRING
                |34   id . id
                |35   id id
                |36   id typeid
typecons       ::=37   interval
                |38   id interval
                |39   typeid interval
                |40   { idlist }
                |41   monitored dependent RECORD reclist
                |42   ordered base pointertype
                |43   array indextype OF typeexp
                |44   DESCRIPTOR FOR typeexp
                |45   transfermode arguments
                |212  id RELATIVE typeexp
                |213  typeid RELATIVE typeexp
                |46   LONG typeexp
                |47   FRAME [ id ]
monitored      ::=85
                |205  MONITORED
dependent      ::=48
                |49   MACHINE DEPENDENT
reclist        ::=50   [ pairlist ]
                |50   [ typelist ]
                |51   [ pairlist , variantpair ]
                |52   [ variantpair ]
                |53   [ variantpart ]
pairlist       ::=6    pairitem
                |7    pairlist , pairitem
pairitem       ::=54   identlist attributes typeexp
typelist       ::=55   typecons
                |55   typeid
                |56   id
                |57   typecons , typelist
                |57   typeid , typelist
                |58   id , typelist
variantpair    ::=54   identlist attributes variantpart
variantpart    ::=59   SELECT vcasehead FROM variantlist ENDCASE
vcasehead      ::=60   id : attributes tagtype
                |61   COMPUTED tagtype
                |62   OVERLAID tagtype
tagtype        ::=63   *
                |0    typeexp
variantlist    ::=6    variantitem ,
                |7    variantlist variantitem ,
variantitem    ::=64   idlist => subreclist
subreclist     ::=0    reclist
                |65   NULL
ordered        ::=85
                |67   ORDERED
base           ::=85
                |67   BASE
pointertype    ::=68   pointerprefix
                |0    pointerprefix TO typeexp
pointerprefix  ::=3    POINTER
                |0    POINTER interval
array          ::=66   ARRAY
                |67   PACKED ARRAY
indextype      ::=4
```

```
                |0    typeexp
transfermode   ::=69  PROCEDURE
                |70   PORT
                |71   SIGNAL
                |72   ERROR
                |73   PROCESS
                |74   PROGRAM
arguments      ::=0   arglist returnlist
arglist        ::=4
                |0    fieldlist
returnlist     ::=4
                |0    RETURNS fieldlist
fieldlist      ::=9   [ pairlist ]
                |9    [ typelist ]
initialization ::=75
                |66   ← initvalue
                |67   = initvalue
initvalue      ::=0   exp
                |76   procaccess block
                |77   CODE
                |78   MACHINE CODE BEGIN codelist END
codelist       ::=214 orderlist
                |215  codelist ; orderlist
procaccess     ::=79
statement      ::=80  lhs
                |81   lhs ← exp
                |82   [ explist ] ← exp
                |83   block
                |84   IF exp THEN statement elsepart
                |86   casehead casestmtlist ENDCASE otherpart
                |87   forclause dotest do enables statementlist doexit ENDLOOP
                |90   EXIT
                |216  LOOP
                |91   GOTO id
                |92   GO TO id
                |93   RETURN optargs
                |94   transfer lhs
                |207  WAIT lhs
                |95   ERROR
                |96   STOP
                |97   STOP [ ! catchlist ]
                |98   NULL
                |99   RESUME optargs
                |100  CONTINUE
                |101  RETRY
                |102  lhs ← STATE
block          ::=0   blockhead END
                |89   blockhead exits END
blockhead      ::=17  begin enables declist statementlist
begin          ::=3   BEGIN
                |18   BEGIN OPEN bindlist ;
bindlist       ::=6   binditem
                |7    bindlist , binditem
binditem       ::=19  exp
                |20   id : exp
exits          ::=9   EXITS exitlist
                |9    EXITS exitlist ;
elsepart       ::=4
                |0    ELSE statement
casehead       ::=66  SELECT exp FROM
                |67   WITH binditem SELECT optexp FROM
casestmtlist   ::=6   casestmtitem ;
                |7    casestmtlist casestmtitem ;
casestmtitem   ::=105 caselabel => statement
caselabel      ::=6   casetest
                |7    caselabel , casetest
casetest       ::=106 optrelation
                |107  exp
otherpart      ::=4
                |0    => statement
forclause      ::=4
                |108  FOR id ← exp , exp
                |109  FOR id direction IN range
                |110  THROUGH range
direction      ::=111
                |111  INCREASING
```

```
                |112  DECREASING
dotest          ::=4
                |0    WHILE exp
                |113  UNTIL exp
do              ::=3    DO
                |18   DO OPEN bindlist ;
doexit          ::=114
                |115  REPEAT exitlist
                |115  REPEAT exitlist ;
                |116  REPEAT exitlist ; FINISHED => statement
                |116  REPEAT exitlist ; FINISHED => statement ;
                |117  REPEAT FINISHED => statement
                |117  REPEAT FINISHED => statement ;
exitlist        ::=6    exititem
                |7    exitlist ; exititem
exititem        ::=118  idlist => statement
enables         ::=85
                |119  ENABLE catchitem ;
                |120  ENABLE BEGIN catchlist END ;
                |121  ENABLE BEGIN catchhead END ;
catchhead       ::=6    catchcase ;
                |7    catchhead catchcase ;
catchlist       ::=0    catchitem
                |122  catchhead catchitem
catchitem       ::=123  catchcase
                |5    ANY => statement
catchcase       ::=105  lhslist => statement
lhslist         ::=6    lhs
                |7    lhslist , lhs
statementlist   ::=4
                |0    statement
                |8    statementlist'
                |124  statementlist' statement
statementlist'  ::=6    statement ;
                |7    statementlist' statement ;
transfer        ::=125  SIGNAL
                |126  ERROR
                |218  RETURN WITH ERROR
                |127  START
                |128  RESTART
                |208  JOIN
                |209  NOTIFY
                |210  BROADCAST
                |129  TRANSFER WITH
                |130  RETURN WITH
optargs         ::=4
                |0    [ explist ]
explist         ::=8    orderlist
                |8    keylist
orderlist       ::=6    optexp
                |7    orderlist , optexp
keylist         ::=6    keyitem
                |7    keylist , keyitem
keyitem         ::=140  id : optexp
optexp          ::=4
                |0    exp
exp             ::=141  transferop lhs
                |143  IF exp THEN exp ELSE exp
                |144  casehead caseexplist ENDCASE => exp
                |145  lhs ← exp
                |0    disjunct
transferop      ::=125  SIGNAL
                |126  ERROR
                |146  NEW
                |127  START
                |211  FORK
                |208  JOIN
caseexplist     ::=6    caseexpitem ,
                |7    caseexplist caseexpitem ,
caseexpitem     ::=105  caselabel => exp
disjunct        ::=C0   conjunct
                |147  disjunct OR conjunct
conjunct        ::=C0   negation
                |148  conjunct AND negation
negation        ::=C0   relation
                |149  not relation
```

```
not             ::=0    ~
                  |0    NOT
relation        ::=C0   sum
                  |150  sum optrelation
optrelation     ::=0    relationtail
                  |151  not relationtail
relop           ::=152  =
                  |153  #
                  |154  <
                  |155  <=
                  |156  >
                  |157  >=
relationtail    ::=0    relop sum
                  |158  IN range
range           ::=0    interval
                  |1    id
                  |0    typeid
                  |38   id interval
                  |39   typeid interval
interval        ::=159  [ bounds ]
                  |160  [ bounds )
                  |161  ( bounds ]
                  |162  ( bounds )
bounds          ::=0    exp .. exp
sum             ::=C0   product
                  |142  sum addop product
addop           ::=163  +
                  |164  -
product         ::=C0   factor
                  |142  product multop factor
multop          ::=165  *
                  |166  /
                  |167  MOD
factor          ::=C0   primary
                  |168  - primary
primary         ::=C0   lhs
                  |2    num
                  |226  lnum
                  |169  char
                  |170  string
                  |219  lstring
                  |171  [ explist ]
                  |172  prefixop [ orderlist ]
                  |220  INTEGER [ explist ]
                  |221  CARDINAL [ explist ]
                  |141  typeop [ typeexp ]
                  |173  @ lhs
                  |174  DESCRIPTOR [ desclist ]
desclist                ::=0    exp
                  |175  exp , exp
                  |176  exp , exp , typeexp
prefixop        ::=177  LONG
                  |178  ABS
                  |179  MIN
                  |180  MAX
                  |181  BASE
                  |182  LENGTH
typeop          ::=183  SIZE
                  |184  FIRST
                  |185  LAST
lhs             ::=1    id
                  |0    ( exp )
                  |0    lhs qualifier
                  |186  LOOPHOLE [ exp ]
                  |187  LOOPHOLE [ exp , typeexp ]
                  |188  memory [ exp ]
qualifier       ::=189  [ explist ]
                  |190  [ explist ! catchlist ]
                  |191  . id
                  |192  ↑
memory          ::=193  MEMORY
                  |194  REGISTER
```