

PFC/RR-82-4

DOE/ET-51013-31  
UC20

Nonalgebraic Symbol Manipulators for Use in  
Scientific and Engineering Modeling:  
Introducing the FORSE  
(FORtran Symbol Expander)

by Joel H. Schultz and Jonathan D. Lettvin  
Massachusetts Institute of Technology  
Plasma Fusion Center  
Cambridge, MA 02139

February, 1982

**Nonalgebraic Symbol Manipulators for Use in  
Scientific and Engineering Modeling:**

**Introducing The FORSE  
(FORtran Symbol Expander)**

**by Joel H. Schultz and Jonathan D. Lettvin**

**M.I.T. Plasma Fusion Center Research Report PFC/RR-82-4**

**February, 1982**

**Nonalgebraic Symbol Manipulators for Use in  
Scientific and Engineering Modeling:**

**Introducing The FORSE  
(FORtran Symbol Expander)**

**by Joel H. Schultz and Jonathan D. Lettvin**

**M.I.T. Plasma Fusion Center Research Report**

**February, 1982**

**ABSTRACT**

A system of nonalgebraic symbol manipulators, called The FORSE (FORtran Symbol Expanders) has been developed to document and prepare input-output for Fortran programs. The use of documentation at the level of the individual equation is defended. The operation of The FORSE is described, along with user instructions and a worked example.

Nonalgebraic Symbol Manipulators for Use in  
Scientific and Engineering Modeling:  
Introducing The FORSE  
(FORtran Symbol Expander)  
by Joel H. Schultz and Jonathan D. Lettvin  
M.I.T. Plasma Fusion Center Research Report  
February, 1982

**A. Introduction**

It is generally recognized that the state of the art in the documentation and debugging of scientific and engineering software is inadequate, even when compared with the corresponding problems for computer operating system or business software. While sharing all of the same problems of how to subdivide large programs along a comprehensible, maintainable and extendable structural principal, scientific and engineering programs have an additional feature, which sets them apart from operating systems and business programs: the models are controversial, and a single wrong equation can completely falsify the results of a 50,000 line program. These incorrect answers will not be corrected by the feedback of a test department or an angry user, as in the case of an operating system. If the error in a key allowable parameter is less than a factor of two, it will probably remain undetected until the catastrophic failure of an engineering system or until another unused paper study recedes into oblivion. Stated another way, a fundamental characteristic of scientific and engineering models is that internal self-consistency does not guarantee correctness except in a very narrow sense. There exists, of course, the same mechanism for avoiding avoidable gross error that is usually adopted for important programs in all other engineering disciplines: expert review. This process is usually not applied to engineering software, because the software is incomprehensible to anyone but the author, and frequently to the author as well. Until now, adequate documentation has required both strong management and a high tolerance for boredom.

If there were a standard method for preparing documentation and input-output automatically, requiring no more time or keystrokes than poorly documented code, it would be a major step towards reviewability and physical reasonableness in engineering and scientific software. We have developed a program which prepares documentation and input-output commands called The FORSE (FORtran Symbol Expander). This code addresses symbol expansion at the level of the individual equation, rather than attempting to reveal the

structure of a code through flow diagrams or structure charts. While documentation has traditionally meant the preparation of flowcharts or data flow diagrams, we will argue below that a prose description of the equations, accompanied by a readable presentation of the equations and a table of the variables, is far more useful to the unambiguous documentation of scientific and engineering codes. Similarly, the automated preparation of listings of all the assumed variables and their values by an automated input-output device with an auxiliary program to form publishable tables, grouped by units, is a very useful tool for the preparation of component and parameter inventories, and the use of those inventories to check the physical reasonableness of a model.

This paper describes programs written in the Emacs Extension Language dialect of MacLisp, which does all the work of preparing output and publishable documents for FORTRAN programs. This program system, called The FORSE for "FORtran Symbol Expander", is currently working on the Honeywell Multics computer at M.I.T. and the Macsyma Consortium PDP-10 at M.I.T. (In order to implement The FORSE on a PDP-10, it was necessary for Lettvin to implement the first portable LISP Emacs, along with an extension language that matches the Honeywell Multics product by Greenberg [GR80].)

### A.1 Why an Equation Displayer is Better than a Flow Chart

Systems of automatic documentation, including automated flow-chart generation, have had limited utility for scientific and engineering applications. We have implemented an equation displayer as an alternative method for the following reasons:

- 1 Engineering and scientific programs, even of large systems, tend to have a relatively simple flow chart, which can fit on one or two pages. However, a single equation can be complex, controversial, and require a moderate amount of documentation.
- 2 A system which writes a prose description of the algorithmic method allows a code to be submitted for peer review. In the author's opinion, it is the general absence of peer review that has historically most commonly compromised the validity of numerical methods in design, causing the horror stories about complex, sophisticated codes which never give the right answers or which take more time to use correctly than hand calculations. A related benefit, observed by the author in his own codes, is that the automatic rewriting of equations in publishable form reveals transcription errors in long equations that are much harder to detect in a line of Fortran code.

- 3 Unfortunately, a large and growing fraction of hardware engineers, particularly machinery designers, do design work in order to produce paper studies, rather than machinery itself. In order to complete these studies and prepare ancillary publications rapidly, a prose writing program which writes the methods section of a paper is worth far more than a flow chart. This is the purpose of the FORSE Texifier. The Texifier has already been used to write a 50 page [SC81] and a 200 page [SC82] methods section for two reports by Schultz.
- 4 The lexical analysis of input-output comments also allows the automated preparation of variable lists with complete descriptions of the variables. This is useful for debugging and reading the methods section, as well as frequently being contractually obligated.
- 5 The documentation of a program at the equation level allows all controversies concerning method to be discussed freely. This discussion can be prewritten in non-"input-output" comments or postwritten in the text generating language, as desired.
- 6 Automated generation of input-output statements which include complete descriptions of the variables is a useful debugging tool, allowing direct observation of the values of all the inputs and computed variables in a noninterpreted language, such as Fortran.
- 7 The standardized format of the output allows the use of an automated table generating code, which has been written, in order to group the outputs by units, and rewrite the formats so that all answers are between 1 and 1,000, with the proper prefix automatically added to the units.
- 8 Due to their poorly understood psychology, engineers do not like to use other people's codes, but they love to steal other people's equations. Therefore, a program to display and document individual equations unambiguously is more likely to help other engineers than is a black box library routine.

#### **B. The FORSE: A FORtran Symbol Expander**

The FORSE is a system of LISP programs which operate on a skeleton of Fortran equations and comments and expand the skeleton into more useful forms. The expanded forms include Fortran programs with formatted input and output descriptions of every variable, prose descriptions of the program with the equations in publishable form, alphabetic lists of the variables with descriptions, publishable output tables grouped by dimensions with scientific prefixes and Tektronix graphics programs. The two largest and most important routines in the FORSE are the Forsifier, which parses a Fortran program and writes its input and output

commands, and the Texifier, which accepts information from the parser and writes the methods section of a publishable paper, describing the Fortran program.

## **B.1 The FORSE Forsifier**

The FORSE Forsifier consists of a parser/lexical analyzer and an input-output preparation section. The parser and lexical analyzer prepares a symbol table, identified in the code by the name "forced", which contains a list of all the variables, whether they are assigned or unassigned, a formal description of the meaning of the variables, and the physical units of the variables. The distinction between assigned and unassigned variables and Fortran keywords is made by the parser. The meanings of the variables and their units are determined by the lexical analysis of special "input-output comments", all of which have the format "*c variable name is the variable description (units)*" Comments not of this form are ignored by the Forsifier (but not by the Texifier, to be described below).

### **B.1.1 The Parser**

The parser analyzes the equations and comments of a program and stores the properties of individual variables in a symbol table.

The first act of the Parser whenever either the Forsifier or the Texifier is run is to initialize the symbol table and the working buffers. This is done by the function (forse-entry). (forse-entry) removes all of the interned variables and their properties from previous runs of The FORSE.

The parsing sequence is as follows:

- 1 Fortran line continuations are removed.
- 2 Fortran logic expressions are identified and flagged on the list of lines, named "sym".
- 3 Unary variable functions are identified and flagged.
- 4 The Fortran lines are parsed in order. Comment lines and labelled lines are identified. The individual line is then parsed by the function (forse-parse-line), as described in section B.1.1.1.

#### **B.1.1.1 (forse-parse-line)**

(forse-parse-line) parses a Fortran line as follows:

- (A) A line of Fortran code is appended to the list of Fortran lines.
- (B) Properties of the line of code are identified and appended to the property list of the symbol, representing the line of code. These properties include its assigned variable, the right-hand side of an equation, the label if there is one, the unlabelled form of the line, a predicate to indicate whether the assigned variable is a constant, and a list of Fortran keywords (e.g. subroutine) in the line.

## B.2 The FORSE Texifier

The FORSE Texifier uses the same parser and lexical analyzer as the Forsfier. It rewrites the equations and comments in prose form in the  $T_E X$  text generating language by Knuth [KN79]. This language is a widely accepted typesetting system, which is particularly well suited to the attractive preparation of highly technical text, as illustrated below in the worked example in Appendix B. After the program has been analyzed, as described above, the following text arranging operations are performed:

- 1 Variable names are represented with subscripts, or "Englishified". Variable names with the same names as Greek letters are represented as Greek letters.
- 2 Equations are represented as publishable equations. The equations are numbered, with the equation number at the left-hand margin. Equations are centered. Numerators are centered above denominators. Square roots appears as radicals. Parentheses are sized to fit the height of the quantity within parentheses. Exponents to the base 10 are changed from e-format to " $\times 10^{\text{exponent}}$ ".
- 3 A sentence is inserted before each equation of the form "The *assigned variable description* (*assigned variable units*) is given by:".
- 4 After the first equation in which an unassigned variable appears, the clause "where *variable name* is the *variable description* (*units*)." is inserted.
- 5 After the first equation in which a constant appears, the clause "where *constant name*, the *constant description*, is *constant value*" is inserted.
- 6 All comments which are not of the input-output form appear as prose statements in the text.
- 7 After the prose text, complete tables of the unassigned and assigned variables are prepared in alphabetical order by variable name. The tables include the variable description and units.



## B.2.1 The Texifier

The Texifier creates a prose document and runs after the Fortran Parser. It prepares a prose description of the program, describing each equation, along with overall description of subroutines and functions. This is followed by an alphabetic list of variables, followed by a program listing.

The sequence of operations in the Texifier is as follows:

- 1 The original FORTRAN program listing is detabified (TABS converted to spaces), then copied into a new buffer for "texification".
- 2 Special characters in TEX, such as %, \$, <, and >, which appear in the program listing have a backslash inserted in front of them, which identifies them as special characters in TEX.
- 3 The comments and equations are converted to TEX form by the function (make-text), as described in section B.2.1.1. This function does not complete the job of creating text. The lack of obvious structure about to be described is an accident of The FORSE's historical development.
- 4 (make-text) got as far as converting equations to the TEX format, removing "c " from comments and converting assignment statement comments to the right form. Comments about unassigned variables (inputs) are then added, by adding a clause after the first appearance of an unassigned variable of the form "where *unassigned variable name* is the *unassigned variable description (units)*".
- 5 Comments about constants are added by adding a clause after the first appearance of the constant in an equation of the form "where *constant name = constant value (units)* is the *constant description*".
- 6 A preliminary conversion of individual variables to TEX format includes the the default creation of subscripts and Greek letters. It is assumed that subscripting begins (1) Immediately following a Greek letter, (2) immediately following a string of upper case letters beginning a variable name, or (3) following the first letter of the variable name, when it is lower case.
- 7 The preliminary selection of parameter conversions is displayed in a menu and converted into the final selection by the user, interactively. A carriage return accepts the default selection.
- 8 The text is converted to the TEX versions of names with dimensioned variables being converted first, so that undimensioned variables with the dimensioned variables names can be identified by their original forms and converted next. Function and subroutine names are considered to be arrays, for the purpose of

text creation.

- 9 Some additional conversion of Fortran to TEX forms is made. The "e" exponent in a Fortran number is changed to the TEX form  $\times 10^{exp}$ . Logical variables are changed from their Fortran form to their TEX form (e.g. `.le. => \le`). Parentheses are converted to the TEX form, which makes mathematical parentheses approximately the same height as the expressions they are enclosing. (e.g. `( => \left(`).
- 10 The parameter lists of unassigned and assigned variables are prepared. The appropriate TEX commands for creating and closing off tables are inserted. The table columns are arranged in alphabetic order by variable name. Each line of the table includes three columns: the variable name, the variable description and the variable units.
- 11 The variables in the parameter lists are converted to TEX form, as was done previously for the variables in the equations and comments.
- 12 Further cleanup is done on the file in order to make the TEX file more attractive and easier to convert to a Xerox PRESS file. Equation lines are separated from text lines by blank lines. All post-equation comments other than the first comment after an equation have "`\ noindent where`" replaced by ",and". Whitespace is inserted between the commas of arrays, in order to prevent TEX from interpreting arrays as a single word. Oversize lines are broken up into several lines.
- 13 A TEX version of the program listing itself is prepared. This involves using the TEX "`\ nofill`" command on every line, in order to prevent the program from being interpreted as a paragraph.

#### B.2.1.1 (make-text)

(make-text) writes the text of the description of the Fortran program in the TEX text generating language.

The function operates as follows.

- [A] Equations are converted to TEX form by the function (make-eqs), as described in section B.2.1.1.1.
- [B] Non-input-output comments are converted to prose form by removing the occurrences of "c" in FORTRAN comments.
- [C] All input-output comments are deleted. The information contained in the input-output comments has already been accumulated by the parser.
- [D] Equations are preceded by sentences of the form "The *assigned variable name (units)* is calculated by:".

[E] Equation lines are numbered by inserting `TEX \ eqno` commands at the end of each equation.

#### B.2.1.1.1 (make-eqs)

(make-eqs) converts FORTRAN equation lines to TEX equations.

- [i] FORTRAN equations are identified by the presence of "=" on a line that isn't a comment.
- [ii] Braces are placed around each expression in the FORTRAN equation, obviating the need for changing the ranking of operators from FORTRAN to TEX.
- [iii] FORTRAN operators are converted to TEX operators (e.g. `*` => `SPACE`).
- [iv] FORTRAN unary operators are converted to TEX operators (e.g. `alog` => `\ ln`).
- [v] TEX equation mode is designated by adding `<<` to the beginning of the equation and `>>` to the end.

#### C. Conclusions

The FORSE is a powerful tool in the preparation of documentation for Fortran programs and in writing papers about engineering studies undertaken with the use of Fortran programs. The uses to which this system of programs has been put include:

- Writing the methods sections of reports on studies using Fortran codes.
- Writing all of the input-output for large system sizing codes, such as *Isabelle* (Isabelle accelerator design options) and *toksysc* (a tokamak system sizing and costing code).
- Preparing attractive tables for publication and converting descriptive output to simple graphic output, as in the bundle divertor costing and sizing program, described in [SC81].

Further work envisioned in the near future for LISP-Fortran hybrid programming include the following:

- Development of an automated worked example generator, that would interpret a Fortran program at the same time that it documented it, and write a description of a worked example in TEX.
- Development of more sophisticated engineering program expanders, that write most of the Fortran code to begin with from a condensed description language. The tool we imagine would be a cross between a meta-language and an expert program.

## References

- [GR80] B.S. Greenberg, "Multics Emacs: an Experiment in Computer Interaction", Proc 4th Annual Honeywell Software Conf, Honeywell Information Systems, Inc., Jan 1980
- [KN79] D.E. Knuth, "Tau Epsilon Chi, a system for technical text", American Mathematical Society, Oct 1979
- [SC81] J.H. Schultz, "A Costing and Sizing Code for Highly Irradiated Normal Magnets", M.I.T. Research Report PFC/RR-81-13, 1981
- [SC82] J.H. Schultz and R.J. Thome, "A Costing and Sizing Model of the Isabelle Accelerator", Francis Bitter National Magnet Laboratory Report, Dec 1981

## Appendix A

### Use of The FORSE

In order to use The FORSE on the M.I.T. Multics system, you must be given access to two files, "forse.4" and "tex.7" by the author. We will assume that the file will be transferred to the working directory of the user, so that no further pathname need be specified when the files are loaded. After logging in, the user will enter Emacs by typing

**MUL** > emacs *filename*

where boldface indicates the computer's prompt to the user and italics indicates a generalized form, in this case "*filename*". Once in the file to be forsified (have input-output statements generated) or texified (documented), the user types

csc-x loadfile forse.4

If the file is to be texified, the user then types

exc-x loadfile tex.7

In order to forsify the file, the user types control-Z Z. In order to texify the file, the user types control-ZT. Both the forsifier and the texifier have a single interruption, in which the user is queried by the program. In the case of the forsifier, he is queried about undescribed variables and about which variables he wishes to be output. In the case of the texifier, he is queried about the default transformations of the variable names from Fortran to a subscripted TEX form. Otherwise, that's all there is to it.

In MC, the user logs in and enters the LISP environment by typing L control-H. The rest of the sequence in the LISP environment should be

Alloc? n CR

```
(load"pfc;emacs")
;loading EMACRO
;loading DEBUG 159
;loading GRINDEF 462
;loading FORMAT([M])DSK:FORMAT;FORMAT 806
;loading YESNOP 44
;loading QUERIO 51
;loading EXTSTR 96
;loading TTY 17
;loading DEFSTRUCT 237
;loading CHAR 6
;loading ESTR 3
;loading DEFMAX 98
;loading DPRINT 1
;loading GCDEM N 14
;loading BREAKLEVEL
68955
```

```
(load"jhs;fors")
```

```
The FORSE is being loaded
```

```
;loading BACKQ 53
;loading SHARPM 80
```

```
T
```

```
(e)
```

Notice that "CR" for carriage-return does not appear in this example, because carriage-returns are not necessary. The above example is only typical of the computer's response, of course, since the version numbers of the automatically loaded files are subject to change. The last command "(e)" places the user in the editor.

Inside the editor, which is an Emacs Extension Language nearly identical in its side-effects with the original implementation by Greenberg on Honeywell Multics [GR80], forsification is done by typing control-XZ, while texification is done by typing control-XT. At present, there is not yet a full screen redisplay in this

editor, so the menu for changing the defaults for TEX subscripting of variables cannot be implemented. This means that the user must accept the default subscripting done by the FORSE.

## Appendix B: Sample Run

Below is a sample of the work of the FORSE textifier. The program listing from which the prose text and parameter lists were prepared is also prepared by The FORSE, and is shown after the parameter list. Other papers, written by the FORSE [SC81], [SC82] are included in the references.

Thsyst size sizes and costs lower hybrid wave plasma heating systems. To determine the optimal launching frequency of a grid, this code takes into account the shift of  $n_z$ , due to scattering at the plasma edge. the launching grid (m)

The permittivity of free space ( $F/m$ ) is given by:

$$\epsilon_0 = \frac{1}{(36.\pi \times 10^9)} \quad (1)$$

The effective ion mass ( $kg$ ) is given by:

$$m_i = a_{mueff} m_p \quad (2)$$

where  $m_p$ , the mass of a proton, is  $1.67 \times 10^{-27}$  ( $kg$ ), and  $a_{mueff}$  is the effective atomic mass of the fuel (e.g. 2.5 for 50-50 DT).

The cyclotron frequency of the electrons ( $radians/s$ ) is given by:

$$\Omega_{ce} = \frac{eB_t}{m_e} \quad (3)$$

where  $m_e$ , the mass of an electron, is  $9.11 \times 10^{-31}$  ( $kg$ ), and  $e$ , the charge on an electron, is  $1.6 \times 10^{-19}$  ( $C$ ), and  $B_t$  is the field on axis ( $T$ ).

The central electron plasma frequency ( $radians/s$ ) is given by:

$$\omega_{pe} = \sqrt{\left(\frac{n_{eo}}{\epsilon_0}\right)\left(\frac{e^2}{m_e}\right)} \quad (4)$$

where  $n_{eo}$  is the central electron density ( $m^{-3}$ ).

The ion plasma frequency ( $radians/s$ ) is given by:

$$\omega_{pi} = \frac{\omega_{pe}}{\sqrt{amu_{eff}1839.}} \quad (5)$$

The ion cyclotron frequency (*radians/s*) is given by:

$$\Omega_{ci} = \frac{\Omega_{ce}}{(a_{mucff}1839.)} \quad (6)$$

The cold plasma lower hybrid frequency (*radians/s*) is given by:

$$\omega_h = \sqrt{\omega_{pi}^2 \left(1 + \frac{\omega_{pe}^2}{\Omega_{ce}^2}\right)} \quad (7)$$

The lower hybrid frequency for absorption by ion Landau damping near the the plasma center is determined iteratively, using an expression with a thermal correction term to the equation for the lower hybrid frequency of a cold plasma. The thermal correction term is a function of the parallel wave number, which is itself a function of the launching frequency. The two equations for launching frequency and parallel wave number are iterated several times and appear to converge in all cases. The following is the first guess at the launching frequency (*radians/s*)

The launching frequency (*radians/s*) is given by:

$$\omega = 1.5\omega_h \quad (8)$$

The following is the first guess at the parallel index of refraction ()

The thermal correction factor () is given by:

$$a_T = 3 \left(\frac{\omega_{pi}}{\omega}\right)^2 \left(\frac{kT_{io}}{m_i c^2}\right) \quad (9)$$

where  $c = 3. \times 10^8$  is the speed of light (*m/s*), and  $k$ , Boltzmanns constant, is  $1.6 \times 10^{-19}$  (*J/eV*), and  $T_{io}$  is the central ion temperature (*eV*).

$$a_T = a_T + 0.75 \left(\frac{\omega_{pi}\omega}{(\Omega_{ce}\Omega_{ci})}\right)^2 \frac{kT_{eo}}{m_i c^2} \quad (10)$$

where  $T_{eo}$  is the central electron temperature (*eV*).

The launching frequency (*radians/s*) is given by:



$$\omega = \sqrt{\frac{\omega_{pi}^2 \left( 1 + n_z \sqrt{\left( \frac{m_i}{m_e} \right) a_T} \right)}{\left( 1 + \frac{\omega_{pi}^2}{(\Omega_{ce} \Omega_{ci})} \right)}} \quad (11)$$

where  $n_z$ , the parallel index of refraction at the center of the plasma, is 5. ()

The free-space wave length at the launching frequency ( $m$ ) is given by:

$$\lambda = \frac{2\pi c}{\omega} \quad (12)$$

The height of the wave guide, long dimension ( $m$ ) is given by:

$$h_{wg} = 0.5 s_{f_{wgcut}} \lambda \quad (13)$$

where  $s_{f_{wgcut}}$ , the safety factor of the wave guide dimensions vs. cutoff, is 1.1 ().

The width of the wave guide, short dimension ( $m$ ) is given by:

$$w_{wg} = .5 h_{wg} \quad (14)$$

The parallel wave number at the plasma edge ( $m^{-1}$ ) is given by:

$$k_{partim} = \frac{\pi}{w_{wg}} \quad (15)$$

Since edge turbulence is expected to isotropize the scattering angle, and because all values of  $\alpha < 0$  are returned to the wave guide, .7 is assumed as the spectrum averaged value of  $\alpha$ . The ratio of central to edge electron density is typically 10 in Alcator A. Lower values (3-5) have been reported in Alcator C.

The electron density at the edge ( $m^{-3}$ ) is given by:

$$n_{edge} = \frac{n_{co}}{n_{rat}} \quad (16)$$

where  $n_{rat}$ , the ratio of electron density center-to-edge, is 5. ()

The plasma frequency at the edge (*radians/s*) is given by:

$$\omega_{pco} = \sqrt{\left(\frac{n_{edge}}{\epsilon_0}\right)\left(\frac{e^2}{m_e}\right)} \quad (17)$$

according to Schuss [SC80].

The parallel wave number at the center of the plasma () is given by:

$$k_{par} = k_{parlim} \left( 1 + \left( \frac{\omega_{pco}}{\omega} \right) \alpha \epsilon \left( 1 - \frac{1}{q_{lim}} \right) \right) \quad (18)$$

where  $\alpha$ , the pectrum averaged scattering angle, between -1 and 1, is 0.7 (), and  $q_{lim}$  is the safety factor at the limiter (), and  $\epsilon$  is the plasma inverse aspect ratio ().

The parallel index of refraction at the center of the plasma () is given by:

$$n_z = \frac{k_{par} c}{\omega} \quad (19)$$

The parallel index of refraction at the plasma edge () is given by:

$$n_{z,edge} = \frac{n_z k_{parlim}}{k_{par}} \quad (20)$$

The following test prevents an error if  $n_{maxwh} < 0$ .

$$test_1 = 1 - \frac{\omega_2}{(\Omega_{ce} \Omega_{ci})} \quad (21)$$

For  $n_z > n_{maxwh}$ , mode conversion to the whistler wave cannot take place at any density, and will be accessible to the plasma center

The maximum parallel index for mode conversion at  $\omega$  () is given by:

$$n_{maxwh} = \sqrt{\frac{1}{\left(1 - \frac{\omega_2}{(\Omega_{ce} \Omega_{ci})}\right)}} \quad (22)$$

where  $n_{maxwh}$ , the maximum parallel index for mode conversion at  $\omega$ , is 0. ().

For  $\omega > \omega_{max}$ ,  $n_{maxwh}$  is infinite and no part of the wave is accessible to the plasma center at any value of  $n_z$ .

The maximum allowed value of the launching frequency (*radians/s*) is given by:

$$\omega_{max} = \sqrt{\Omega_{ce}\Omega_{ci}} \quad (23)$$

The peak of the reflected  $n_z$  power spectrum, due to backscattering () is given by:

$$n_{back} = \frac{c}{\omega} \left( \frac{\pi}{w_{wg}} \right) \quad (24)$$

The lower hybrid wave length ( $m$ ) is given by:

$$\lambda_{lh} = \frac{2. \pi c}{\omega_{lh}} \quad (25)$$

The optimum improvement in plasma-waveguide coupling is effected if the main and auxiliary waveguide vanes project 1/2 a wave length from the vessel wall.

The height of the vanes from the vacuum wall ( $m$ ) is given by:

$$h_{vane} = .5\lambda \quad (26)$$

The area of the wave guide ( $m^2$ ) is given by:

$$a_{wg} = w_{wg}h_{wg} \quad (27)$$

Since EIA WG WR-510, -430 and -340 all have wall thicknesses of .2 cm, the functional dependency of thickness with frequency is ignored.

The thickness of the vertical coolant passage between guides ( $m$ ) is given by:

$$t_{vcool} = .1w_{wg} \quad (28)$$

The thickness of the horizontal coolant passage between grills ( $m$ ) is given by:

$$t_{hcool} = .2h_{wg} \quad (29)$$

The overall wave guide grill packing factor () is given by:

$$wg_{pack} = \frac{a_{wg}}{(2t_{septum} + t_{vcool} + w_{wg})(t_{hcool} + 2t_{septum} + h_{wg})} \quad (30)$$

where  $t_{septum}$ , the thickness of the interwaveguide septum, is  $2 \times 10^{-3}$  (m).

The total number of wave guides in the heating system () is given by:

$$n_{wg} = \frac{P_{heat}}{(\eta_{launch} P_{Dmax} a_{wg})} \quad (31)$$

where  $P_{Dmax}$  is the maximum permitted power density in the grid ( $W/m^2$ ), and  $\eta_{launch}$  is the efficiency of heat deposition from grid into the plasma (), and  $P_{heat}$  is the total total auxiliary heating power into the plasma (W).

The number of wave guides in a port () is given by:

$$n_{port} = \frac{n_{wg}}{n_{ports}} \quad (32)$$

where  $n_{ports}$  is the number of heating ports ().

The area required by one port ( $m^2$ ) is given by:

$$A_{port} = \frac{n_{wg} a_{wg}}{(n_{ports} W_{gpack})} \quad (33)$$

The number of wave guides in a grid () is given by:

$$n_{grid} = \frac{L_{tormax}}{(w_{wg} + 2t_{septum} + t_{vcool})} \quad (34)$$

where  $L_{tormax}$  is the maximum space in the toroidal direction for the launching grid ().

The number of stacked grids in a port () is given by:

$$n_{stack} = \frac{n_{port}}{n_{grid}} \quad (35)$$

The cutoff wavelength for the TE10 mode ( $m$ ) is given by:

$$\lambda_{cTE10} = 2h_{wg} \quad (36)$$

The waveguide wavelength of the TE10 mode ( $m$ ) is given by:

$$\lambda_{wg} = \frac{\lambda}{\sqrt{1 - \frac{\lambda^2}{\lambda_{cTE10}^2}}} \quad (37)$$

The field for multipactor breakdown in "unseeded" air. ( $V/m$ ) is given by:

$$E_{mult} = 1.06 \times 10^{-12} \omega^2 w_{wg} \quad (38)$$

This is used as a basis for design, but is less valid than the following equation. Ohkubo [OH77] reports that the dielectric breakdown of 1 GHz waves in a resonator, seeded with electrons, is independent of pressure from  $10^{-6}$  torr to  $4^{-3}$  torr. Thomas' handbook [TH72] gives a formula for the power-carrying capacity of waveguides. The maximum possible power transmission through a waveguide ( $W$ ), then, is given by:

$$P_{bd} = 6.63 \times 10^{-4} \frac{E_{rfmax}^2 w_{wg} h_{wg} \lambda}{\lambda_{wg}} \quad (39)$$

where  $E_{rfmax}$ , the maximum permissible rf electric field, is  $2.2 \times 10^5$  ( $V/m$ ), from Ohkubo [OH77].

The safety margin in power transmission ( $S_{Mprf}$ ) is given by:

$$S_{Mprf} = \frac{P_{bd}}{(P_{Dmax} h_{wg} w_{wg})} \quad (40)$$

The cutoff frequency of the TE<sub>10</sub> mode ( $Hz$ ) is given by:

$$f_{cTE10} = \frac{2\pi c}{\lambda_{cTE10}} \quad (41)$$

The cutoff wavelength for the TE<sub>01</sub> mode ( $m$ ) is given by:

$$\lambda_{cTE01} = 2w_{wg} \quad (42)$$

The cutoff frequency for the TE<sub>01</sub> mode ( $Hz$ ) is given by:

$$f_{cTE01} = \frac{2\pi c}{\lambda_{cTE01}} \quad (43)$$

The cutoff wavelength for the TE<sub>11</sub> mode ( $m$ ) is given by:

$$\lambda_{cTE11} = \frac{2h_{wg}}{\sqrt{1 + \left(\frac{h_{wg}}{w_{wg}}\right)^2}} \quad (44)$$

The cutoff frequency for the TE<sub>11</sub> mode ( $Hz$ ) is given by:

$$f_{cTE11} = \frac{c}{\lambda_{cTE11}} \quad (45)$$

The attenuation of copper waveguide is taken from Thomas [TH72].

The attenuation of the TE01 mode in copper guide ( $db/m$ ) is given by:

$$\alpha_{TE10} = \frac{7.35 \times 10^{-5} \left( \frac{h_{wg}}{w_{wg}} s f_{wgcut}^{1.5} + \frac{1}{\sqrt{s f_{wgcut}}} \right)}{h_{wg}^{1.5} \sqrt{s f_{wgcut}^2 - 1}} \quad (46)$$

The total attenuation of the TE01 mode in the waveguides ( $db$ ) is given by:

$$N_{pwg} = \alpha_{TE10} L_{wg} \quad (47)$$

where  $L_{wg}$  is the total length of a waveguide run ( $m$ ). Zero degree cut sapphire windows have been shown by Johnson [JO64], to have a loss of only 1.7 W/kW at 35 kW/cm<sup>2</sup> at 8 GHz.

The circulators protecting the klystrons in the Alcator C lower hybrid heating experiments provide over 35 db, with an insertion loss of .23 db at 4.6 GHz [RI80].

The "attenuation" of launching into the plasma ( $db$ ) is given by:

$$\alpha_{launch} = -4.35 \ln \eta_{launch} \quad (48)$$

The total attenuation of the rf system ( $db$ ) is given by:

$$N_{ptot} = N_{pwg} + \alpha_{circ} + \alpha_{wind} + \alpha_{launch} \quad (49)$$

where  $\alpha_{wind}$ , the attenuation in the waveguide window, is .01 ( $db$ ), and  $\alpha_{circ}$ , the attenuation in the circulator, is .23 ( $db$ ).

The output rf power required from the tubes ( $W$ ) is given by:

$$P_{tubes} = P_{heat} 10^{\left(\frac{N_{ptot}}{10}\right)} \quad (50)$$

The rf power needed to feed one waveguide ( $W$ ) is given by:

$$P_{tube} = \frac{P_{tubes}}{n_{wg}} \quad (51)$$

The Alcator C klystrons converted dc to rf power with 35 % efficiency. Varian has reported high-power klystrons with 60 % efficiency. In 1964, Skowron built a 400 kw CW crossed-field amplifier with a gain of 9 db at 3,000 MHz and an overall efficiency of 72 %.

The dc anode power required by the microwave tube ( $W$ ) is given by:

$$P_{anode} = \frac{P_{tubes}}{\eta_{cfa}} \quad (52)$$

where  $\eta_{cfa}$ , the efficiency of a crossed-field amplifier, is .72 (). to a regulator tube ( $W$ )

The unregulated power supplied by a high-voltage rectifier ( $W$ ) is given by:

$$P_{rect} = \frac{P_{anode}}{\eta_{reg}} \quad (53)$$

where  $\eta_{reg}$ , the efficiency of a high-voltage series regulator, is .85 ().

The power drawn from the line by the rf system ( $W$ ) is given by:

$$P_{line} = \frac{P_{rect}}{\eta_{rect}} \quad (54)$$

where  $\eta_{rect}$ , the efficiency of a high-voltage diode rectifier and transformer, is .95 (). The following microwave system costs are scaled from the Alcator C lower hybrid system costs. [R180]

The cost of the circulator (\$) is given by:

$$C_{circ} = .015P_{tubes} \quad (55)$$

The cost of the power supplies (\$) is given by:

$$C_{ps} = .0868P_{rect} \quad (56)$$

The cost of waveguides. (\$) is given by:

$$C_{wg} = 42.L_{wg}n_{wg} \quad (57)$$

The plumbing cost includes waveguides, directional couplers, waveguide harmonic filters, low-level logic and low-level TWT amplifiers.

The cost of microwave plumbing and electronics (\$) is given by:

$$C_{plumb} = 25 \times 10^3 n_{wg} \quad (58)$$

The cost of the waveguide windows (\$) is given by:

$$C_{wind} = \frac{.094 P_{heat}}{\eta_{launch}} \quad (59)$$

The cost of rf tubes, either klystrons or crossed-field amplifiers. (\$) is given by:

$$C_{tube} = .283 P_{tubes} \quad (60)$$

The total lower hybrid heating system cost (\$) is given by:

$$C_{system} = C_{tube} + C_{wind} + C_{plumb} + C_{ps} + C_{circ} + C_{wg} \quad (61)$$



## References

- [JO64] F.O. Johnson, "Waveguide Windows for MultiKilowatt Applications, " Proc. of Internatl. Conf. on Microwave Circuit Theory and Information Theory, Tokyo, Japan, Sept. 7-11, 1964
- [OH77] Kunizo Ohkubo and Kiyokata Matsuura, "Study of RF Voltage Breakdown for LHRH in 0.8-0.9 GHz", IPPJ-T-27, March 1977
- [R180] K. Rice, private communication
- [SC80] J. Schuss et al, "Lower Hybrid Heating in the Alcator A Tokamak, " M.I.T. Plasma Fusion Center Report PFC/RR-80-6
- [TH72] H.E. Thomas, Handbook of Microwave Techniques and Equipment, Prentice-Hall, 1972

## INPUT VARIABLES

Parameter	Description	Units
$B_t$	field on axis	(T)
$L_{tormax}$	maximum space in the toroidal direction for the launching grid	()
$L_{wg}$	total length of a waveguide run	(m)
$P_{Dmax}$	maximum permitted power density in the grid	(W/m <sup>2</sup> )
$P_{heat}$	total total auxiliary heating power into the plasma	(W)
$T_{eo}$	central electron temperature	(eV)
$T_{io}$	central ion temperature	(eV)
$amu_{eff}$	effective atomic mass of the fuel	()
$\epsilon$	plasma inverse aspect ratio	()
$\eta_{launch}$	efficiency of heat deposition from grid into the plasma	()
$n_{eo}$	central electron density	(m <sup>-3</sup> )
$n_{ports}$	number of heating ports	()
$q_{im}$	safety factor at the limiter	()

## ASSIGNED VARIABLES

Parameter	Description	Units
$A_{port}$	area required by one port	( $m^2$ )
$C_{circ}$	cost of the circulator	(\$)
$C_{plumb}$	cost of microwave plumbing and electronics	(\$)
$C_{ps}$	cost of the power supplies	(\$)
$C_{system}$	total lower hybrid heating system cost	(\$)
$C_{tube}$	cost of rf tubes, either klystrons or crossed-field amplifiers.	(\$)
$C_{wg}$	cost of waveguides.	(\$)
$C_{wind}$	cost of the waveguide windows	(\$)
$E_{mult}$	field for multipactor breakdown in "unseeded" air.	( $V/m$ )
$E_{rfmax}$	maximum permissible rf electric field	( $V/m$ )
$K_{cfa}$	power gain of a crossed-field amplifier	(db)
$N_{pilot}$	total attenuation of the rf system	(db)
$N_{pwg}$	total attenuation of the TE01 mode in the waveguides	(db)
$\Omega_{ce}$	cyclotron frequency of the electrons	( $radians/s$ )
$\Omega_{ci}$	ion cyclotron frequency	( $radians/s$ )
$P_{anode}$	dc anode power required by the microwave tube	( $W$ )
$P_{bd}$	maximum possible power transmission through a waveguide	( $W$ )
$P_{line}$	power drawn from the line by the rf system	( $W$ )
$P_{rect}$	unregulated power supplied by a high-voltage rectifier	( $W$ )
$P_{tube}$	rf power needed to feed one waveguide	( $W$ )
$P_{tubes}$	output rf power required from the tubes	( $W$ )
$SM_{prf}$	safety margin in power transmission	()
$\alpha_T$	thermal correction factor	()
$\alpha$	spectrum averaged scattering angle, between -1 and 1	()
$\alpha_{TE10}$	attenuation of the TE01 mode in copper guide	(db/m)
$\alpha_{circ}$	attenuation in the circulator	(db)
$\alpha_{launch}$	"attenuation" of launching into the plasma	(db)
$\alpha_{wind}$	attenuation in the waveguide window	(db)
$\alpha_{wg}$	area of the wave guide	( $m^2$ )
$c$	speed of light in a vacuum	( $m/s$ )
$e$	charge on an electron	( $C$ )
$\epsilon_0$	permittivity of free space	( $F/m$ )
$\eta_{cfa}$	efficiency of a crossed-field amplifier	()
$\eta_{kly}$	efficiency of a klystron	()
$\eta_{rect}$	efficiency of a high-voltage diode rectifier and transformer	()
$\eta_{reg}$	efficiency of a high-voltage series regulator	()

ASSIGNED VARIABLES - Continued

Parameter	Description	Units
$f_{cTE01}$	cutoff frequency for the TE01 mode	(Hz)
$f_{cTE10}$	cutoff frequency of the TE10 mode	(Hz)
$f_{cTE11}$	cutoff frequency for the TE11 mode	(Hz)
$h_{vane}$	height of the vanes from the vacuum wall.	(m)
$h_{wg}$	height of the wave guide, long dimension	(m)
$k$	is Boltzmanns constant	(J/eV)
$k_{par}$	parallel wave number at the center of the plasma	()
$k_{parlim}$	parallel wave number at the plasma edge	( $m^{-1}$ )
$\lambda$	free-space wave length at the launching frequency	(m)
$\lambda_{cTE01}$	cutoff wavelength for the TE01 mode	(m)
$\lambda_{cTE10}$	cutoff wavelength for the TE10 mode	(m)
$\lambda_{cTE11}$	cutoff wavelength for the TE11 mode	(m)
$\lambda_h$	lower hybrid wave length	(m)
$\lambda_{wg}$	waveguide wavelength of the TE10 mode	(m)
$m_e$	mass of an electron	(kg)
$m_i$	effective ion mass	(kg)
$m_p$	mass of a proton	(kg)
$n_{back}$	peak of the reflected $n_z$ power spectrum, due to backscattering	()
$n_{edge}$	electron density at the edge	( $m^{-3}$ )
$n_{grid}$	number of wave guides in a grid	()
$n_{maxwh}$	maximum parallel index for mode conversion at $\omega$	()
$n_{port}$	number of wave guides in a port	()
$n_{rat}$	ratio of electron density center-to-edge	()
$n_{stack}$	number of stacked grids in a port	()
$n_{wg}$	total number of wave guides in the heating system	()
$n_z$	parallel index of refraction at the center of the plasma	()
$n_{zedge}$	parallel index of refraction at the plasma edge	()
$\omega$	launching frequency	(radians/s)
$\omega_h$	cold plasma lower hybrid frequency	(radians/s)
$\omega_{max}$	maximum allowed value of the launching frequency	(radians/s)
$\omega_{pe}$	central electron plasma frequency	(radians/s)
$\omega_{peo}$	plasma frequency at the edge	(radians/s)
$\omega_{pi}$	ion plasma frequency	(radians/s)
$s_{f_{wgcut}}$	safety factor of the wave guide dimensions vs. cutoff	()
$l_{hcool}$	thickness of the horizontal coolant passage between grills	(m)

### ASSIGNED VARIABLES - Continued

Parameter	Description	Units
$t_{septum}$	thickness of the interwaveguide septum	(m)
$t_{vcool}$	thickness of the vertical coolant passage between guides	(m)
$wg_{pack}$	overall wave guide grill packing factor	()
$w_{wg}$	width of the wave guide, short dimension	(m)

Program Listing - lhsyst\_size.fortran

```

c lhsyst size sizes and costs lower hybrid wave plasma heating systems.
c To determine the optimal launching frequency of a grid, this code
c takes into account the shift of nz, due to scattering at the plasma edge.
c Lwg is the total length of a waveguide run (m)
c Ltormax is the maximum space in the toroidal direction for the launching grid(m)
c the launching grid (m)
c Bt is the field on axis (T)
c Tio is the central ion temperature (eV)
c Teo is the central electron temperature (eV)
c neo is the central electron density (m**-3)
c nports is the number of heating ports ( )
c Pheat is the total total auxiliary heating power into the plasma (W)
c etalaunch is the efficiency of heat deposition from grid into the plasma ( )
c PDmax is the maximum permitted power density in the grid (W/m**2)
c amueff is the effective atomic mass of the fuel (e.g. 2.5 for 50-50 DT)
c qlim is the safety factor at the limiter ( )
c epsilon is the plasma inverse aspect ratio ( )
c c is the speed of light in a vacuum (m/s)
  c=3.e8
c me is the mass of an electron (kg)
  me = 9.11e-31
c mp is the mass of a proton (kg)
  mp=1.67e-27
c k is Boltzmanns constant (J/eV)
  k = 1.6e-19
c epsilono is the permittivity of free space (F/m)
  epsilono = 1./(36.*pi*1.e9)
c e is the charge on an electron (C)
  e = 1.6e-19
c mi is the effective ion mass (kg)
  mi=amueff*mp
c Omegace is the cyclotron frequency of the electrons (radians/s)
  Omegace = e*Bt/me
c omegape is the central electron plasma frequency (radians/s)
  omegape =sqrt( (neo/epsilono)*(e**2/me))
c omegapi is the ion plasma frequency (radians/s)
  omegapi=omegape/sqrt(amueff*1839.)
c Omegaci is the ion cyclotron frequency (radians/s)
  Omegaci=Omegace/(amueff*1839.)
* c omegalh is the cold plasma lower hybrid frequency (radians/s)
  omegalh=sqrt(omegapi**2*(1+omegape**2/Omegace**2))
c The lower hybrid frequency for absorption by ion Landau damping near the
c the plasma center is determined iteratively, using an expression with
c a thermal correction term to the equation for the lower hybrid frequency
c of a cold plasma. The thermal correction term is a function of the parallel
c wave number, which is itself a function of the launching frequency. The
c two equations for launching frequency and parallel wave number are iterated
c several times and appear to converge in all cases.
c
c The following is the first guess at the launching frequency (radians/s)
  omega=1.5*omegalh
c The following is the first guess at the parallel index of refraction ( )
  nz=5.
  do 373 nomega=1,5
c aT is a thermal correction factor ( )
  aT=3*(omegapi/omega)**2*(k*Tio/(mi*c**2))
  aT=aT+0.75*(omegapi*omega/(Omegace*Omegaci))**2*(k*Teo/(mi*c**2))
c omega is the launching frequency (radians/s)
  omega=sqrt(omegapi**2*(1+nz*sqrt((mi/me)*aT))/(1+omegapi**2/(Omegace*Omegaci)))
c lambda is the free-space wave length at the launching frequency (m)
  lambda=2*pi*c/omega
c sfwgcut is the safety factor of the wave guide dimensions vs. cutoff ( )

```

```

    sfgcut=1.1
c hwg is the height of the wave guide, long dimension (m)
  hwg=0.5*sfgcut*lambda
c wwg is the width of the wave guide, short dimension (m)
  wwg=.5*hwg
c kparlim is the parallel wave number at the plasma edge (m**-1)
  kparlim=pi/wwg
c alpha is a spectrum averaged scattering angle, between -1 and 1 ( )
c Since edge turbulence is expected to isotropize the scattering angle, and
c because all values of alpha < 0 are returned to the wave guide, .7 is
c assumed as the spectrum averaged value of alpha.
  alpha=0.7
c The ratio of central to edge electron density is typically 10 in Alcator A.
c Lower values (3-5) have been reported in Alcator C.
c nrat is the ratio of electron density center-to-edge ( )
  nrat=5.
c nedge is the electron density at the edge (m**-3)
  nedge=neo/nrat
c omegapeo is the plasma frequency at the edge (radians/s)
  omegapeo=sqrt((nedge/epsilono)*(e**2/me))
c kpar is the parallel wave number at the center of the plasma ( )
c [SC80] J. Schuss et al, "Lower Hybrid Heating in the Alcator A Tokamak,"
c M.I.T. Plasma Fusion Center Report PFC/RR-80-6
  kpar=kparlim*(1+(omegapeo/omega)*alpha*epsilon*(1-1/qlim))
c nz is the parallel index of refraction at the center of the plasma ( )
  nz=kpar*c/omega
c nzedge is the parallel index of refraction at the plasma edge ( )
  nzedge=nz*kparlim/kpar
373 continue
c The following test prevents an error if nmaxwh < 0.
  nmaxwh=0.
  test1=1-omega**2/(Omegace*Omegaci)
  if (test1 .lt. 0)go to 75
c nmaxwh is the maximum parallel index for mode conversion at omega ( )
c For nz > nmaxwh, mode conversion to the whistler wave cannot take place
c at any density, and will be accessible to the plasma center
  nmaxwh=sqrt(1/(1-omega**2/(Omegace*Omegaci)))
75 continue
c omegamax is the maximum allowed value of the launching frequency (radians/s)
c For omega > omegamax, nmaxwh is infinite and no part of the wave
c is accessible to the plasma center at any value of nz
  omegamax=sqrt(Omegace*Omegaci)
c nback is the peak of the reflected nz power spectrum, due to backscattering ( )
  nback=(c/omega)*(pi/wwg)
c lambdaalh is the lower hybrid wave length (m)
  lambdaalh=2.*pi*c/omegalh
c The optimum improvement in plasma-waveguide coupling is effected if the
c main and auxiliary waveguide vanes project 1/2 a wave length from the vessel
c wall.
c hvane is the height of the vanes from the vacuum wall. (m)
  hvane=.5*lambda
c awg is the area of the wave guide (m**2)
  awg=wwg*hwg
c Since EIA WG WR-510, -430 and -340 all have wall thicknesses of .2 cm,
c the functional dependency of thickness with frequency is ignored.
c tseptum is the thickness of the interwaveguide septum (m)
  tseptum=2.e-3
c tvcool is the thickness of the vertical coolant passage between guides (m)
  tvcool=.1*wwg
c thcool is the thickness of the horizontal coolant passage between grills (m)
  thcool=.2*hwg
c wgpack is the overall wave guide grill packing factor ( )
  wgpack=awg/((2.*tseptum+tvcool+wwg)*(thcool+2.*tseptum+hwg))

```

c nwg is the total number of wave guides in the heating system (  
 $nwg = P_{heat} / (e_{tlaunch} * P_{Dmax} * awg)$   
 c nport is the number of wave guides in a port (  
 $nport = nwg / nports$   
 c Aport is the area required by one port ( $m^{**2}$ )  
 $Aport = nwg * awg / (nports * wgpack)$   
 c ngrid is the number of wave guides in a grid (  
 $ngrid = Ltormax / (wgg + 2 * tseptum + tvcool)$   
 c nstack is the number of stacked grids in a port (  
 $nstack = nport / ngrid$   
 c lambdacTE10 is the cutoff wavelength for the TE10 mode (m)  
 $lambdacTE10 = 2 * hwg$   
 c lambdawg is the waveguide wavelength of the TE10 mode (m)  
 $lambdawg = lambda / \sqrt{1 - (lambda / lambdacTE10)^{**2}}$   
 c Emult is the field for multipactor breakdown in "unseeded" air. (V/m)  
 c This is used as a basis for design, but is less valid  
 c than the following equation.  
 $Emult = 1.06e-12 * omega^{**2} * wwg$   
 c Ohkubo [OH77] reports that the dielectric breakdown of 1 GHz waves in a  
 c resonator, seeded with electrons, is independent of pressure from  
 c  $10^{**6}$  torr to  $4^{**3}$  torr.  
 c [OH77] Kunizo Ohkubo and Kiyokata Matsuura, "Study of RF Voltage Breakdown  
 c for LHRH in 0.8-0.9 GHz", IPPJ-T-27, March 1977  
 c Erfmax is the maximum permissible rf electric field (V/m)  
 $Erfmax = 2.2e5$   
 c Thomas handbook [TH72] gives a formula for the power-carrying capacity  
 c of waveguides.  
 c [TH72] H.E. Thomas, Handbook of Microwave Techniques and Equipment,  
 c Prentice-Hall, 1972  
 c Pbd is the maximum possible power transmission through a waveguide (W)  
 $Pbd = 6.63e-4 * Erfmax^{**2} * wwg * hwg * lambda / lambdawg$   
 c SMprf is the safety margin in power transmission (  
 $SMprf = Pbd / (PDmax * hwg * wwg)$   
 c fcTE10 is the cutoff frequency of the TE10 mode (Hz)  
 $fcTE10 = 2 * pi * c / lambdacTE10$   
 c lambdacTE01 is the cutoff wavelength for the TE01 mode (m)  
 $lambdacTE01 = 2 * wwg$   
 c fcTE01 is the cutoff frequency for the TE01 mode (Hz)  
 $fcTE01 = 2 * pi * c / lambdacTE01$   
 c lambdacTE11 is the cutoff wavelength for the TE11 mode (m)  
 $lambdacTE11 = 2 * hwg / \sqrt{1 + (hwg / wwg)^{**2}}$   
 c fcTE11 is the cutoff frequency for the TE11 mode (Hz)  
 $fcTE11 = c / lambdacTE11$   
 c The attenuation of copper waveguide is taken from Thomas [TH72].  
 c alphaTE10 is the attenuation of the TE01 mode in copper guide (db/m)  
 $alphaTE10 = (7.35e-5 / hwg^{**1.5}) * ((hwg / wwg) * sfwgc^{**1.5} + 1 / \sqrt{sfwgc^{**2} - 1}) / \sqrt{sfwgc^{**2} - 1}$   
 c Npwg is the total attenuation of the TE01 mode in the waveguides (db)  
 $Npwg = alphaTE10 * Lwg$   
 c Zero degree cut sapphire windows have been shown by Johnson [J064], to  
 c have a loss of only 1.7 W/kW at 35 kW/cm $^{**2}$  at \* GHz.  
 c [J064] F.O. Johnson, "Waveguide Windows for Multi-Kilowatt Applications,"  
 c Proc. of Internatl. Conf. on Microwave Circuit Theory and Information  
 c Theory, Tokyo, Japan, Sept. 7-11, 1964  
 c alphawind is the attenuation in the waveguide window (db)  
 $alphawind = .01$   
 c The circulators protecting the klystrons in the Alcator C lower hybrid  
 c heating experiments provide over 35 db, with an insertion loss of .23 db  
 c at 4.6 GHz.  
 c alphacirc is the attenuation in the circulator (db)  
 $alphacirc = .23$   
 c alphalaunch is the "attenuation" of launching into the plasma (db)  
 $alphalaunch = -4.35 * alog(e_{tlaunch})$   
 c Nptot is the total attenuation of the rf system (db)



```

Nptot=Npwg+alphacirc+alphawind+alphalaunch
c Ptubes is the output rf power required from the tubes (W)
  Ptubes=Pheat*10**(Nptot/10.)
c Ptube is the rf power needed to feed one waveguide (W)
  Ptube=Ptubes/nwg
c The Alcatraz C klystrons converted dc to rf power with 35 % efficiency.
c Varian has reported high-power klystrons with 60 % efficiency.
c etakly is the efficiency of a klystron ( )
  etakly=.5
c In 1964, Skowron built a 400 kw CW crossed-field amplifier with a gain of
c 9 db at 3,000 MHz and an overall efficiency of 72 %.
c etacfa is the efficiency of a crossed-field amplifier ( )
  etacfa=.72
c Kcfa is the power gain of a crossed-field amplifier (db)
  Kcfa=9.
c Panode is the dc anode power required by the microwave tube (W)
  Panode=Ptubes/etacfa
c etareg is the efficiency of a high-voltage series regulator ( )
  etareg=.85
c Prect is the unregulated power supplied by a high-voltage rectifier (W)
c to a regulator tube (W)
  Prect=Panode/etareg
c etarect is the efficiency of a high-voltage diode rectifier and transformer ( )
  etarect=.95
c Pline is the power drawn from the line by the rf system (W)
  Pline=Prect/etarect
c The following microwave system costs are scaled from the Alcatraz C lower
c hybrid system costs. [RI80]
c [RI80] K. Rice, private communication
c Ccirc is the cost of the circulator ($)
  Ccirc=.015*Ptubes
c Cps is the cost of the power supplies ($)
  Cps=.0868*Prect
c Cwg is the cost of waveguides. ($)
  Cwg=42.*Lwg*nwg
c The plumbing cost includes waveguides, directional couplers, waveguide
c harmonic filters, low-level logic and low-level TWT amplifiers.
c Cplumb is the cost of microwave plumbing and electronics ($)
  Cplumb=25e3*nwg
c Cwind is the cost of the waveguide windows ($)
  Cwind=.094*Pheat/etalaunch
c Ctube is the cost of rf tubes, either klystrons or crossed-field amplifiers. ($)
  Ctube=.283*Ptubes
c Csystem is the total lower hybrid heating system cost ($)
  Csystem=Ctube+Cwind+Cplumb+Cps+Ccirc+Cwg
end

```