

User Commands

EQNTOTT(1)

NAME

eqntott - generate truth table from Boolean equations

SYNOPSIS

```
eqntott [ -l ] [ -f ] [ -s ] [ -r ] [ -R ] [ -.key ] [ cc  
options ] [ files ]
```

DESCRIPTION

Eqntott generates a truth table suitable for PLA programming from a set of Boolean equations which define the PLA outputs in terms of its inputs. When neither `-f` nor `-s` is specified, input and output variables must be mutually exclusive. If the `-s` option is given, an output variable may be used in an expression defining another output variable: the expression for the first output is substituted for the the name of that output when it is encountered. The `-f` option allows outputs to be defined in terms of their previous values in a synchronous system (e.g. an FSM): the same name appearing as both an input and an output may be thought of as referring to two distinct variables, or the same variable at two distinct times. (The `-f` and `-s` options are mutually exclusive.)

If the `-r` option is specified, eqntott will attempt to reduce the size of the truth table by merging minterms. The `-R` option (implies `-r`) forces eqntott to produce a truth table with no redundant minterms. The truth table generated does not represent a minimal covering of the truth functions, but does preserve some ``don't care'' information for some other program to use.

If the `-l` option is specified, eqntott will output a truth table which includes the name of the pla and its inputs and outputs as specified in PLA(5).

The form that the output takes is controlled by the string key, described below. Input is taken from files (standard input default) and run through the C macro preprocessor of `cc(1)`, to permit comments, file inclusion, macros, and conditional processing. The `cc` options `-D`, `-I`, and `-U` are recognized and passed on to the preprocessor.

Equation Syntax:

```
name = expression;
```

Associates a truth function defined by expression with the output name, both of which are defined below. If an output name is assigned more than one expression, the effect is identical to a single assignment to the output of the logical disjunction of all the original expressions.

```
NAME = name ;
```

Defines the name of the pla to be ``name''. If not specified, the name of the pla is the name of the input file with any postfixes removed.

INORDER = name [name]... ;

Defines the order in which inputs appear in the truth table. If not specified, the order is that in which the inputs appear in the source.

OUTORDER = name [name]... ;

Defines the order in which outputs appear in the truth table. If not specified, the order is that in which the outputs appear in the source.

#### Expression Syntax:

name

A name is used to specify an input or output. The name must begin with a letter or underscore; subsequent characters may be letters, digits, underscores, asterisks, periods, square brackets, or angle brackets.

ZERO (or 0)

Builtin input that always has the value zero (false).

ONE (or 1)

Builtin input that always has the value one (true).

?

Builtin input that always has the value ``don't care''.

( expression )

Parenthesis may be used to change the order of evaluation.

! expression

Gives the complement of expression.

expression & expression

Gives the logical conjunction of the two expressions. The & operator associates left to right, and has the same precedence as !.

expression | expression

Gives the logical disjunction of the two expressions. The | operator also associates left to right, and has a lower precedence than &.

#### Output Format

The output format may be controlled to a small extent using the character string key. The string is scanned left to right, and at each character code, a piece of output is generated corresponding to the character encountered. If -.key is not specified, the string ``iopte'' is used, or ``iopfte'' with the -f option.

code output generated

```
e .e
f .f output-number input-number
  (one line for each feedback path, numbers refer to Or-
  and And-plane truth table column numbers)
h a human readable version of the truth table (q.v.)
i .i number-of-inputs
I .I input-name
  (one line for each input, in order)
l a truth table with the name of the pla, its inputs and
  its outputs
p .p number-of-product-terms
n .n number-of-product-terms
o .o number-of-outputs
O .O output-name
  (one line for each output, in order)
S PLA connectivity summary
t PLA personality matrix (q.v.)
v eqntott version information
```

The truth table (personality matrix) consists of a line for each minterm, beginning with that minterm and followed by the values of the various outputs. The minterm is composed of a single character (0, 1, or -) for each input in the conventional fashion. The output values are represented by one of the three characters (0, 1, or x). Some white space is added for readability's sake.

In the human readable format, each line of output represents one term in the sum-of-products expression for an output. The line begins with the name of the output, which is enclosed in parentheses for the value ``don't care''. Then follow the names of the inputs in the product; complemented inputs are preceded by a !.

#### SEE ALSO

cc(1).

#### DIAGNOSTICS

Syntax errors are written to the standard error output and should be self-explanatory.

#### BUGS

-l should be the default, but some pla tools can't handle the full format. Eqntott likes its options separately; i.e. -f -l works but -fl doesn't.

#### AUTHOR

Bob Cmelik.  
-l option added by Jeff Deutsch.