

# QuineHP v1.0

## Logic functions minimizer

by Leonardo Laguna Ruiz

August 2005

This program minimizes Boolean logic functions expressed in the form:

$$f(A,B,C,...,N) = \sum m(m_1, m_2, \dots, m_n) + \sum d(d_1, d_2, \dots, d_n)$$

where  $m_1, m_2, \dots, m_n$  are the minterms and  $d_1, d_2, \dots, d_n$  are "don't care" terms.

### Instalation:

QuineHP v1.0 doesn't need any special installation, just download the file to the calculator.

### Usage:

#### The input:

Example 1:

If you want to minimize

$$f(A,B,C) = \sum m(1,3,4,5)$$

you need to push in the RPL stack the next strings

```
2: ""  
1: "1 3 4 5"
```

where

```
2: ""           is the list of "don't care" terms (empty in this case)  
1: "1 3 4 5"    is the minterm list (numbers separated with spaces)
```

Example 2:

To minimize

$$f(A,B,C,D) = \sum m(1,5,7,8,10) + \sum d(0,2,4)$$

```
2: "0 2 4"
```

1: "1 5 7 8 10"

### The output:

QuineHP v1.0 pushes a string with all the step-by-step solution. This string is best viewed with "lupa" (by Mario de Lama) available from [www.hpcalc.org](http://www.hpcalc.org). For the example 1 you will see as output:

```
"
  n:1
+001:-    1
+100:-    4
  n:2
+011:-    3
+101:-    5
|-----|
  n:1
0-1:*     1 3
-01:*     1 5
10-:*     4 5
|-----|

          1 4 3 5
P0  0-1:  1 0 1 0
P1  -01:  1 0 0 1
P2  10-:  0 1 0 1
```

Minimized form:

```
P0 + P2
A'C + AB'
"
```

This string has three parts: reduction of the minterms list, reduction of the prime implicants table and solution.

In the reduction of the list part, the following takes place:

```
|-----|   separates the lists
n:x          means the next terms have x "ones"
```

the +,- and \* meant for each term that:

+	the term is a minterm
-	the term was successfully combined with other
*	the term is a prime implicant

In the reduction of the table part, you will see how the table is reduced eliminating the essential prime implicants and applying properties to solve a cover problem. Sometimes the problem can't be solved in this way, so QuineHP v1.0 starts an algorithm to find the best solutions.

The solution part of the string can be shown in different forms that depend on the number of solutions.

- A main single solution:

"Minimized form:

$P_0 + P_2$   
 $A'C + AB'$ "

The best solution is the sum of the prime implicants  $P_0$  and  $P_2$ . Expressed as a sum of products is  $A'C+AB'$ , where  $A'$  means "not A" and the variable "A" represents the most significant bit of the minterm.

- Multiple solutions

"Minimized forms:

$P_0 + P_3 + P_4$   
 $A'B' + BC' + AC$

$P_1 + P_2 + P_5$   
 $A'C' + B'C + AB$ "

In this case we've got two solutions and we can choose between  $P_0+P_3+P_5$  or  $P_1+P_2+P_5$ .

"Minimized forms:

$P_0 + P_4$   
 $AB'C'D' + A'CD'$

+

$P_1 + P_6 + P_7$   
 $A'B'D' + BC'D + ACD$

$P_2 + P_5 + P_8$   
 $A'C'D' + B'CD + ABD$ "

In this case we've got two solutions that share some terms, so the first solution is  $P_0+P_4+P_1+P_6+P_7$  and the second is  $P_0+P_4+P_2+P_5+P_8$

**Notes:**

- You don't need to explicitly define the number of variables used, QuineHP determines it automatically based in the greater term, but you can't have more than 16 variables.
- Do not repeat any term because a crash can occur. This will be fixed in the next version.

**Contact:**

Any suggestions, bug report or question can be sent to:

Leonardo Laguna Ruiz  
modlfo@yahoo.com