

RUNNING ESPRESSO

ESPRESSO is a DOS program that must be run from the DOS prompt. ESPRESSO must be given an input file describing the logic system to be minimized, and it produces an output file with the minimized results. The input and output files described below are given in the Berkeley format. To run ESPRESSO, perform the following steps.

1. Go to the MS DOS prompt.
2. Change directories to C:\ESPRESSO
3. At the DOS prompt in the ESPRESSO directory, type “edit *filename*”, where *filename* is the name of the input file you have created. Create your input file (as illustrated below), save, and exit back to the ESPRESSO directory.
4. At the DOS prompt in the ESPRESSO directory, type “espresso *filename*” to run ESPRESSO. ESPRESSO will automatically create an output file called *filename.out* containing the minimized or reduced results.
5. To view the output file, type “edit *filename.out*” at the DOS prompt.

ESPRESSO input files

All logic data input to ESPRESSO must be in **canonical minterm code form**. For example, in

$$F(A, B, C, D) = \sum m(4,5,6,8,9,10,13) + \phi(0,7,15)$$

the minterms 4,5,6,... would appear as 0100, 0101, 0110... in the ESPRESSO input file. Minterms are entered into ESPRESSO when the function output is asserted (i.e., logic 1) or when the output is a don't care (a dash). Typically, minterms are only input into ESPRESSO when the function output is either a 1 or a don't care. The following is an annotated example of an ESPRESSO input file. This input file is derived from the minterm code logic equation above. To create such an input file, use the DOS “edit *filename*” command, where *filename* is your chosen name for the input file. Note that a “--” indicates a comment that must not be present in the input file. Missing minterms are assumed to be logic 0.

```
.i 4          -- .i n tells ESPRESSO n logic inputs are present
.o 1          -- .o m tells ESPRESSO m functions are present
.ilb A B C D  -- All input variable names are provided on the “.ilb” line.
.ob F         -- The function name(s) are provided on the “.ob” line.
.p 10        -- .p N tells ESPRESSO that N minterms follow.
0100 1
0101 1
0110 1       -- Listing of minterms derived from the equation above. Note
1000 1       the “1” or “-” in the rightmost column, signifying an
1001 1       asserted output or a don't care, respectively.
1010 1
1101 1
0000 -
0111 -
1111 -
.e           -- .e End-of-file marker. Must be the last line of the input file.
```

ESPRESSO output files

Running ESPRESSO (by typing “ESPRESSO *filename*” at the DOS prompt) will cause an output file, called *filename.out*, to be automatically created. If ESPRESSO is run with the above input file, the following output file results. The file has been annotated for reference by using the comment symbol “--”.

```
.i 4          -- These first four line are copied by ESPRESSO from the input
.o 1          file to the output file.
.ilb A B C D
.ob F
.p 3          -- Number of product terms required in the minimized output
1-01 1       -- Product terms produced by ESPRESSO. A “-“ in an output
10-0 1       p-term means the corresponding binary variable is not
01-- 1       present in that p-term.
.e           -- End-of-file marker
```

This output file will yield the Boolean expression $F(A,B,C,D) = \overline{A}CD + \overline{A}\overline{B}D + \overline{A}B$. If multiple outputs are present, then a given p-term would be included for an output only if a “1” existed in the output column on the given p-term row.

Minimizing Multiple Output Systems with ESPRESSO

For multiple output functions, the first four lines of the input file must be altered to meet the requirements of the multiple output system. To illustrate, consider the BCD-to-XS3 conversion truth table. The input file called fig619a is given in the ESPRESSO Berkeley format as

```
.i 4
.o 4
.ilb A B C D
.ob A' B' C' D'
.p 16
0000 0011
0001 0100
0010 0101
0011 0110
0100 0111
0101 1000
0110 1001
0111 1010
1000 1011
1001 1100
1010 ----
1011 ----
1100 ----
1101 ----
1110 ----
1111 ----
.e
```

Here, the “-“ symbol is used to indicate don’t care outputs for the six non-essential minterms.

The resulting output file, named by EXPRESSO called fig619a.out, is given by

```
.i 4
.o 4
.ilb A B C D
.ob A' B' C' D'
.p 8
-100 0100
-010 0101
-0-1 0100
--11 0010
-110 1001
-1-1 1000
--00 0011
1--- 1000
.e
```

This output file is given in the Berkeley format suitable for directly programming a PLA. Notice that there are three shared PIs. If a PAL is to be programmed from this table all three shared PIs will have to be represented by three additional p-terms for a total of 11 p-terms.

Both of the above files are provided in the EXPRESSO directory on this CD-ROM. The output file results yield the following expressions:

$$A' = B\overline{C}\overline{D} + BD + A$$

$$B' = \overline{B}\overline{C}\overline{D} + \overline{B}\overline{C}D + \overline{B}CD$$

$$C' = CD + \overline{C}\overline{D}$$

$$D' = \overline{B}\overline{C}\overline{D} + \overline{B}C\overline{D} + \overline{B}CD$$

where all three p-terms for D' are covered by p-terms in A' , B' and C' as shared PIs. Note that EXPRESSO misses the fact that $D' = \overline{D}$ which is obvious from an inspection of the input file (truth table). These expressions represent a total gate/input tally of 11/28 excluding possible inverters, 10/25 if $D' = \overline{D}$.

POS Output Results by Using EXPRESSO

POS output results can be obtained by the following procedure:

- [1] Complement all 1's and 0's in the input file if originally given in minterm code form. If maxterm code expressions are provided, then treat them as minterms to construct the input file—do not complement the 1's and 0's in this case.
- [2] Run EXPRESSO as indicated above.
- [3] Complement all 1's and 0's in the output file to obtain the POS result.