CUPL™ (2.0)
The Universal Compiler For Programmable Logic QUICK REFERENCE GUIDE

CUPL and CSIM
INPUT and OUTPUT FILES

THE TEMPLATE FILE
Supplied with the CUPL files is a file named TMPL.PLD. This is a blank template file which you should use to build your logic source file. Its structure is as follows:

```
PARTNO <FOR_THIS_FUNCTION>;  
NAME <YOUR_FILE_NAME>;  
DATE <DATE_OF_LAST_CHANGE>;  
REV <CURRENT_REV_NO>;  
DESIGNER <YOUR_NAME>;  
COMPANY <YOUR_COMPANY>;  
ASSEMBLY <WHERE_PLD_USED>;  
LOCATION <ON_PC_BOARD>;  
```

ẫ***************  
/ * <FILL IN TITLE BLOCK WITH> */  
/ * <EXPLANATION OF LOGIC FUNCTION> */  
/ ***************  
/ * ALLOWABLE TARGET DEVICES: */  
/ * <LIST MENU OF DEVICES> */  
/ *  
/ ** INPUTS */  
PIN <NUMBER> = <NAME>;  
/ **<DESCRIPTION>**/  
/ ** OUTPUTS */  
PIN <NUMBER> = <NAME>;  
/ **<DESCRIPTION>**/  
/ ** DECLARATIONS AND INTERMEDIATE VARIABLE DEFINITIONS */  
/ <INTERMEDIATE_VARIABLE = EXPRESSION>;  
/ <FIELD_BIT.FIELD_VAR = [VARIABLE LIST]>;  
/ ** LOGIC EQUATIONS */  
/ <OUTPUT = EXPRESSION>;  

RUNNING CUPL
In order to compile a specific logic source file (YOUR_FILE_NAME.PLD) for a specific target device, type the following:

CUPL [FLAGS] TARGET_DEVICE CODE YOUR_FILE_NAME

For example:

CUPL -J -A P16L8 RAMCNTRL

which would compile the logic source file RAMCNTRL.PLD for a generic PAL16L8 Target Device, while producing a JEDEC File (YOUR_FILE_NAME.JED) and an absolute format file (YOUR_FILE_NAME.ABS) which must be present if the simulator (CSIM) is to be run later.

This list of Target Device types and their corresponding device codes can be found in an Appendix of the manual.

A list of CUPL option flags is presented below:

- J → Produce YOUR_FILE_NAME.JED, the JEDEC format downloadable file.
- A → Produce YOUR_FILE_NAME.ABS for later use by CSIM.
- S → Automatically run CSIM after running CUPL.
- L → Produce YOUR_FILE_NAME.LST with line numbers and error messages.
- I → Produce YOUR_FILE_NAME.HL downloadable HL format file for IFL.
- H → Produce YOUR_FILE_NAME.HEX "ASCII Hex Space" format file.
- F → Produce YOUR_FILE_NAME.DOC with fuse map file.
- X → Produce YOUR_FILE_NAME.DOC with fully expanded product-terms.
- G → Program security fuse.
- R → Disable global product-term merging. (FPLA devices)
- M0 → Perform no logic minimization.
- M1 → Perform local logic minimization (default).
- M2 → Perform logic minimization until equations fit in target device.
- M3 → Perform full logic minimization.
- D → Deactivate unused OR-terms (Increases speed in FPLAs).
- U → Use specified library for compilation.

* PAL is a Registered Trademark of Monolithic Memories, Inc.
CUPL SYNTAX

• LOGICAL OPERATORS
  & = LOGICAL AND
  # = LOGICAL OR
  $ = LOGICAL XOR
  ! = LOGICAL NEGATION

• FREE FORM COMMENT STRUCTURE:
  /# = START COMMENT
  #/ = END COMMENT

• VARIABLE EXTENSIONS
  VAR.D = EXP; /# D of D FLIP-FLOP */
  VAR.J = EXP; /# J of J-K FLIP-FLOP */
  VAR.K = EXP; /# K of J-K FLIP-FLOP */
  VAR.R = EXP; /# R of R-S FLIP-FLOP */
  VAR.S = EXP; /# S of R-S FLIP-FLOP */
  VAR.AR = EXP; /# ASYNCHRONOUS FF RESET */
  VAR.AP = EXP; /# ASYNCHRONOUS FF RESET */
  VAR.AP = EXP; /# ASYNCHRONOUS FF PRESET */
  VAR.SR = EXP; /# SYNCHRONOUS FF RESET */
  VAR.SP = EXP; /# SYNCHRONOUS FF PRESET */
  VAR.DE = EXP; /# /# THREE-STATE ENABLE */

• NODE DECLARATIONS:
  NODE VARIABLE.NAME; /# SINGLE NODE AS WHEN */
  NODE [VARIABLE.LIST]; /# USED FOR ‘COMPLEMENT_..ARRAY’ */
  /# /# IN IFL DEVICES */
  /# /# /# /

• THE DISTRIBUTIVE PROPERTY: (From BOOLEAN Algebra)
  A Б (B Б C) is replaced by A&B # A&C
  (where & operations are performed before # operations)
  deMorgan’s Theorem: (From BOOLEAN Algebra)
  !(A&B) is replaced by !A & !B
  !(A&B) is replaced by !A # B

NOTE: This symbolology tends to create large numbers of Product-Terms.

PALASM-T0-CUPL LANGUAGE TRANSLATOR

This program will convert logic source files in the PALASM format to the CUPL source file format.
To convert one or more PALASM source files to CUPL format type:

```
PTOC FILE_NAME1 FILE_NAME2 ...(RET)
```

For example:

```
PTOC BUS_CNTL ASM (RET)
```

which would produce the following CUPL format files:

```
BUS_CNTL.PLD
```

and, if the original PALASM file had the “Function Table” information,

```
BUS_CNTL_SI
```

THE CUPL PREPROCESSOR

This program operates on the CUPL source file before compiler operations actually begin. Capabilities include:

String Subsystem:

```
$DEFINE ARG1 ARG2
```

where ARG1 is replaced with ARG2 until

```
$UNDEF ARG1 is encountered.
```

File Inclusion:

```
$INCLUDE FILENAME
```

where the referenced file becomes part of the specification at Compile time.

Conditional Compilation:

Portions of the source specification may be compiled or not depending on whether or not the Argument (ARG) has been defined using the $DEFINE command. The formats are:

```
$IFDEF ARG...
$STATEMENTS...
$IFDEF ARG...
```

or, if not defined:

```
$IFDEF ARG...
$STATEMENTS...
$IFDEF ARG...
```

CSIM: THE SIMULATOR

CSIM is a Stimulus/Response Function Table oriented simulator which will compare each expected response with that which logic in the associated .PLD file would produce given the specified stimulus.

The simulator input file (YOUR_FILE.NAME.SI) must contain the same header information as the associated logic source file (YOUR_FILE.NAME.PLD).

Also, the #A option flag to produce an absolute file (YOUR_FILE.NAME.ABS) and also with the –A option flag if you would like CSIM to append the function table Test-Vector information to your .JED file in order to produce a .JED file with both fuse and testing information.

The general format for the .SI file is:

```
"HEADER INFORMATION"
```

ORDER:

```
VAR1, VAR2, ..., VARN
```

VECTORS:

```
STIMULUS_PATTERN 1 RESPONSE_PATTERN 1
STIMULUS_PATTERN 2 RESPONSE_PATTERN 2
...
...
STIMULUS_PATTERN N RESPONSE_PATTERN N
```

Within the Vector Table. inputs are defined with ‘1’ (+SV), ‘O’ (GND) while outputs are defined with ‘H’ (+SV), ‘L’ (GND), and ‘Z’ (High Impedance). Don’t cares are represented by X. A * in the response field causes the simulator to determine the output according to the logic definition contained in the .ABS file which was derived from your .PLD file.

DIRECTIVES

(Place on any row of vector table)

```
$MSG "YOUR_MESSAGE"
$REPEAT N
$TRADE N
$EXIT
```

To run CSIM, type:

```
CSIM FLAGS [TARGET..CODE YOUR..FILE..NAME]
```

CSIM FLAGS

```
-J Produces a .JED file with test vectors.
-L Produces a .SO file (simulator output).
-V Displays simulator output vectors.
-U Uses specified library for compilation.
```

*PALASM is a Registered Trademark of Monolithic Memories, Inc.