

# Digital System Design Project 2 – Exact Boolean Minimization

Due: 23:59, Oct. 9, 2025

The sum of products (SOP) is a standard form for 2-level Boolean function representation. The corresponding literals are ANDed in product terms and then ORed together as the output. The SOP representation can be realized in either AND-OR gates or NAND-NAND gates. To reduce the circuit cost, it is important to minimize the numbers of product terms and literals. Dr. Maurice Karnaugh proposed a pictorial form of a truth table, known as the Karnaugh map, and combined adjacent squares recursively for prime implicant (PI) generations. However, the Karnaugh map is difficult to scale up when the number of Boolean variables is greater than 6. Dr. Quine and Dr. McCluskey developed a computer-based tabular form to identify all PIs efficiently. This is the well-known Quine–McCluskey algorithm. Since the minimum SOP representation is the subset of PIs, Dr. Petrick proposed a technique to determine all minimum SOP solutions from the PI candidates. In this project, your C/C++ program reads a single-output Boolean function (with don't cares) in PLA format and outputs its minimum SOP representation in PLA format using the Quine-McCluskey algorithm followed by the Petrick Algorithm. Report the number of product terms and literals.

Please submit your report and program according to the following rules:

- 1- The font size of your report is 12 in PDF format.
- 2- The filename of your report is your student ID (e.g., B12345678.pdf).
- 3- Generate your own 3 PLA files **with don't cares** (4 to 6 variables).
- 4- Post the content of input and output PLA files.
- 5- Upload a tarball (e.g., B12345678.tgz) of your program source code, your PLA files, and a text ReadMe file, which illustrates how your program is to be compiled in a Linux environment (e.g., WSL).

PLA Example: input.pla

```
.i 4
.o 1
.ilb a b c d
.ob f
.p 7
00-0 1
0-11 1
1-01 1
0101 1
1111 -
100- 1
-01- 1
.e
```

SYNOPSIS

```
%> PROGRAM PLA_IN_FILE PLA_OUT_FILE
```

Run-time Example:

```
%> mini input.pla output.pla
Total number of terms: 4
Total number of literals: 8
```

```
%> cat output.pla
```

```
.i 4
.o 1
.ilb a b c d
.ob f
.p 4
-1-1 1
-0-0 1
1--1 1
--11 1
.e
```

## Reference

- [1] [https://en.wikipedia.org/wiki/Quine%E2%80%93McCluskey\\_algorithm](https://en.wikipedia.org/wiki/Quine%E2%80%93McCluskey_algorithm)
- [2] [https://en.wikipedia.org/wiki/Petrick%27s\\_method](https://en.wikipedia.org/wiki/Petrick%27s_method)