Topic Support Guide

Cambridge International AS & A Level Computer Science

9608

For examination from 2017

Topic 3.3.3 Karnaugh Maps

A DIVISION OF
CAMBRIDGE ASSESSMENT

Contents

Introduction

How to use this guide

The aim of this guide is to facilitate your teaching of Cambridge International AS and A Level Computer Science, syllabus topic 3.3.3 Karnaugh Maps This is an advanced theory topic that is part of topic 3.3 Hardware, and follows on from topics on logic gates and circuit design, and Boolean algebra. The guidance and activities in this resource are designed to help teachers devise programmes of study which provide teaching time devoted to theory work as well as activities that consolidate learning.

Section 1 lists some key terms used in this topic and their definitions. Section 2 introduces the theory of Karnaugh maps and a link to worked examples. Section 3 provides links to online resources on Karnaugh maps. Section 4 gives ideas for class and homework activities.

Learning objectives

Using this document should help you guide learners in the following syllabus learning objectives:

- show understanding of Karnaugh Maps
- show understanding of the benefits of using Karnaugh Maps
- solve logic problems using Karnaugh Maps

Prior knowledge

Before you begin teaching this topic you should:

- have a good working knowledge of truth tables, or have studied topic 3.3.1 Logic gates and circuit design
- have a good working knowledge of Boolean algebra, or have studied topic 3.3.2 Boolean algebra
- know how to generate a Karnaugh map for a given logical expression with two, three or four variables
- know how to produce an optimal simplified expression from a given Karnaugh map
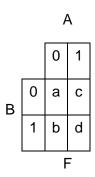
1. Key terms

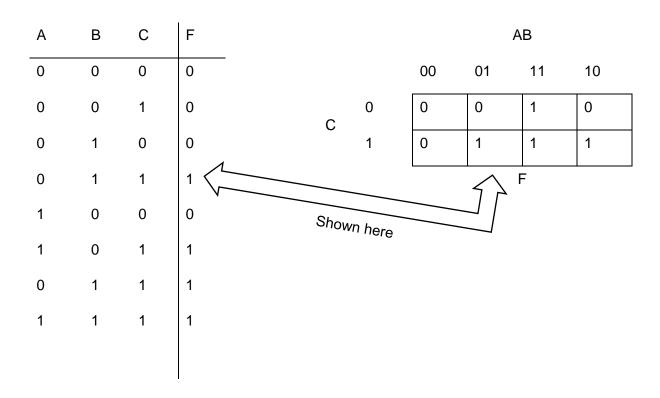| Word/phrase	Meaning
Boolean algebra	Algebra in which variables may have one of two values (True or False, 1 or 0)
Boolean operators	System of operators used for logical operations. The following symbols are used in this guide: • NOT: $\overline{A}$, $\overline{B}$, $\overline{C}$ • AND: ∎ • OR: +
Karnaugh maps	A grid layout of the inputs and outputs of a logic function with the input values on the outside of the grid, and the output values in the squares of the grid
truth table	A table listing all possible combinations of inputs and the corresponding outputs for a given logic function

2. Theory and worked examples

2.1. What are Karnaugh maps?

Karnaugh maps are similar to truth tables, in that they show all possible combinations of input variables and output for a given function. However, in Karnaugh maps, the variable values form the coordinates of the map (row and column headings) and the function output forms the squares in the map. The relation between a truth table and Karnaugh map is shown below, for a generalised two-variable function where A, B are the variables and the function output for each combination is determined by F:

A	B	F
0	0	a
0	1	b
1	0	c
1	1	d

A

	0	1
0	a	c
1	b	d

B

F

An example for a specific three-variable expression:

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
0	1	1	1
1	1	1	1

AB

C	00	01	11	10
0	0	0	1	0
1	0	1	1	1

F

Shown here

For the AB variables, note that adjacent values are arranged so that they only change by one bit at a time.

The benefit of Karnaugh maps is that their layout enables output 1s to be grouped together. A set of simplification rules for grouping (see Section 3 Online resources) can be followed to eliminate unnecessary variables, see:

https://www.wisc-online.com/learn/career-clusters/manufacturing/dig5103/karnaugh-maps

for a clear explanation, and derive the simplest logical expression of the function. Knowing this simple expression means that the number of logic devices required to implement a particular function in a logic circuit can be minimised. Similarly, in software applications, knowing the simplest expression minimises the number of instructions/ comparisons needed.

2.2. Worked examples

Use the following link to work through examples of two- and three-variable Karnaugh maps and the simplification rules:

http://www.ee.surrey.ac.uk/Projects/Labview/minimisation/karnaugh.html

3. Online resources

There are numerous resources available that introduce and work through examples of Karnaugh maps. The following are a selection of these you should consider using.

The content of websites is dynamic and constantly changing. Schools are strongly advised to check each site for content and accessibility prior to using it with learners. Cambridge International Examinations is not responsible for the accuracy or content of information contained in these sites. The inclusion of a link to an external website should not be understood to be an endorsement of that website or the site's owners (or their products/services).

3.1. Websites

Accessible slides, giving a step-by-step introduction to three-variable Karnaugh maps that clearly shows how variables are eliminated and algebraic expressions simplified.

https://www.wisc-online.com/learn/career-clusters/manufacturing/dig5103/karnaugh-maps

Karnaugh maps from Bucknell University in the US. A good introduction to three and four-variable Karnaugh maps including some interactive worked examples and self-assessment tests.

http://www.facstaff.bucknell.edu/mastascu/elessonshtml/Logic/Logic3.html

Accessible lecture notes from the University of Surrey that work through examples of two and three-variable Karnaugh maps, alongside the arithmetical simplifications. Shows how the simplification rules are applied. Includes practise problems and answers for learners.

http://www.ee.surrey.ac.uk/Projects/Labview/minimisation/karnaugh.html

Karnaugh maps class notes from the University of Toronto. Introduction to two, three and four-variable Karnaugh maps and simplification rules aimed at learners and including practise problems.

http://www.eecg.toronto.edu/~ahouse/mirror/engi3861/kmaps.pdf

3.2. Videos

A good tutorial on how to apply the simplification rules to Karnaugh maps to generate optimal solutions. Explains clearly how 1s can be grouped together within a Karnaugh map.

https://www.youtube.com/watch?v=PA0kBrpHLM4

Three-variable Karnaugh map. A step-by-step guide to constructing a Karnaugh map from a three-variable truth table, and deriving an optimal logical expression.

https://www.youtube.com/watch?v=UdpV-ksadkQ

Four-variable Karnaugh map. A step-by-step guide to constructing a Karnaugh map from a four-variable truth table and deriving an optimal logical expression.

https://www.youtube.com/watch?v=FOf00W8WSBg

4. Class and homework activities

To cover the learning objectives in this topic, learners will need to practise the following:

- derive a truth table for a given logical expression as the first step in creating a Karnaugh map
- generate a Karnaugh map grid for two, three and four variables
- apply a series of rules to a Karnaugh map to identify blocks of 1s
- derive an optimal expression from a completed Karnaugh map by working out which variables can be eliminated.

Questions 3–9 of the following link can be used to practise these steps:

http://www.allaboutcircuits.com/worksheets/karnaugh-mapping/

4.1. Homework questions

The following can be used to consolidate students' learning.

Minimise the following problems using the Karnaugh maps method.

1. $(\overline{A}.\overline{B}.\overline{C}) + (\overline{A}.\overline{B}.C) + (A.\overline{B}.\overline{C}) + (A.\overline{B}.C)$
2. $(\overline{A}.\overline{B}.C) + (\overline{A}.B.\overline{C}) + (A.B.\overline{C}) + (A.\overline{B}.\overline{C})$

4.2. Homework solutions

1. $(\overline{A}.\overline{B}.\overline{C}) + (\overline{A}.\overline{B}.C) + (A.\overline{B}.\overline{C}) + (A.\overline{B}.C)$

<div align="center">AB</div>

	00	01	11	10
C 0	1			1
1	1			1

The edges of the map connect up, therefore there is a single group of 1s, which contains the 0 and 1 values of variables A and C. These variables can be eliminated and the minimised result is $\overline{B}$.

2. $(\overline{A}.\overline{B}.C) + (\overline{A}.B.\overline{C}) + (A.B.\overline{C}) + (A.\overline{B}.\overline{C})$

<div align="center">AB</div>

	00	01	11	10
C 0		1	1	1
1	1			

By using the rules of simplification and ringing of adjacent cells in order to make as many variables redundant, the minimised result obtained is $(\overline{A}.\overline{B}.C) + (B.\overline{C}) + (A.\overline{C})$.