University of Macau
Faculty of Science and Technology
Department of Computer and Information Science
CISB121 Digital Systems
Syllabus
Part A – Course Outline

Compulsory course for Computer Science

Catalog description:
(2-2) 3 credits. This course introduces the basic principles and concepts of modern digital systems. This includes the study of combinational and sequential systems using standard modules such as shifters, adders, registers, and counters etc. The advanced techniques for designing, analyzing and implementing the digital circuits are introduced with an emphasis on practical design techniques and circuit implementation. In which, design by using a hardware description language (HDL), such as VHDL to write a behavioral model of the circuit's functionality, will be introduced as well. Major topics include number systems, Boolean algebra, logic components, combinational and sequential logic analysis and design, and digital subsystems. The laboratory provides more insight into the design and implementation of digital systems using the hardware components as well as programmable implementation technologies.

Course type:
- Theoretical with substantial laboratory/practice content

Prerequisites:
- None

Textbook(s) and other required material:
- M. Morris Mano& Michael D. Ciletti, ***Digital Design*****,** 5th Edition, Pearson Education, 2013.

References:
- Ronald J. Tocci, Neal S. Widmer and Gregory L. Moss, ***Digital Systems Principles and Applications***, 11th Edition, Pearson Education, 2011.

Major prerequisites by topic:
- None

Course objectives:
1. Learn fundamental principles and concepts of modern digital systems. [a, b]
2. Learn binary systems in representing digital information, and simple Boolean expressions using the theorems and postulates of Boolean algebra. [a, b, c]
3. Study the *Karnaugh map* for simplifying gate circuits, use standard combinational functions to design and analyze combinational circuits. [a, b, c]
4. Learn various sequential circuits' components such as flip-flops, registers, shift register and counters, use them to build sequential circuits. [a, b, c]
5. Design and simulate more complex digital systems by applying HDL descriptions. [a, b, c, j]

Topics covered:

- **Binary systems (2 hrs)** − various binary systems suitable for representing information in digital systems, binary number system and binary codes, addition and subtraction of signed binary numbers and decimal numbers in BCD.
- **Logic functions (2 hrs)** − basic postulates of Boolean algebra and the correlation between Boolean expressions and their corresponding logic diagrams, logic operations for two variables, the most useful logic gates used in the design of digital systems.
- **Gate-level minimization (3 hrs)** − map method for simplifying Boolean expressions and the digital circuits constructed with AND-OR, NAND, or NOR gates, two-level gate circuits and their method of implementation, HDL with simple gate-level modeling examples.
- **Combinational logic (8 hrs)** − the formal procedures for the analysis and design of combinational circuits, some basic components used in the design of digital systems, such as adders and code converters, frequently used digital logic functions, and their use in the design of combinational circuits, HDL examples in the gate-level, dataflow, and behavioral modeling to show the alternative ways available for describing combinational circuits in HDL.
- **Synchronous sequential logic (7 hrs)** − the formal procedures for the analysis and design of clocked (synchronous) sequential circuits, the gate structure of several types of flip-flops together with a discussion on the difference between level and edge triggering, design examples are presented with emphasis on sequential circuits that use D-type flip-flops, and behavioral modeling in HDL for sequential circuits.
- **Registers and counters (4 hrs)** − various sequential circuits components such as registers, shift registers, and counters, basic digital components for building more complex digital systems, and HDL descriptions of shift registers and counter are presented.
- **Memory and programmable logic (2 hrs)** − random access memory (RAM) and programmable logic devices, memory decoding and error correction schemas, combinational and sequential programmable devices such as ROM, PLAs, CPLDs, and FPGAs.

Class/laboratory schedule:

Timetabled work in hours per week			**No of teaching weeks**	**Total hours**	**Total credits**	**No/Duration of exam papers**
Lecture	**Tutorial**	**Practice**				
2	1.3	0.7	14	56	3	1 / 2 hours

Student study effort required:

Class contact:	
Lecture	26 hours
Tutorial	18 hours
Laboratory	10 hours
Mid-term exam	2 hours
Other study effort	
Self-study	20 hours
e-Quizzes (homework)	10 hours
Total student study effort	96 hours

Student assessment:

Final assessment will be determined on the basis of:

Assignments and e-Quizzes	20%
Laboratory and class exercises	20%
Mid-term exam	20%
Final exam	40%

Course assessment:

The assessment of course objectives will be determined on the basis of:

- Homework. e-quizzes, labs and exams. Course evaluation

Course outline:

Weeks	Topic	Course work
1	**Digital Systems and Binary Numbers** Present the various binary systems suitable for representing information in digital systems. The binary number system is explained and binary codes are illustrated. Addition and subtraction of signed binary numbers and decimal numbers in BCD will be introduced.	(TBA)
2	**Boolean Algebra and Logic Gates** Introduce the basic postulates of Boolean algebra and shows the correlation between Boolean expressions and their corresponding logic diagrams. All possible logic operations for two variables are investigated, and the most useful logic gates used in the design of digital systems are determined. The characteristics of integrated circuit gates are briefly mentioned.	
3-4	**Gate-Level Minimization** Cover the map method for simplifying Boolean expressions. The map method is also used to simplify digital circuits constructed with AND-OR, NAND, or NOR gates. All other possible two-level gate circuits are considered and their method of implementation is explained. HDL will be introduced with simple gate-level modeling examples.	
4-7	**Combinational Logic** Outline the formal procedures for the analysis and design of combinational circuits. Some basic components used in the design of digital systems, such as adders and code converters are introduced. Frequently used digital logic functions are explained, and their use in the design of combinational circuits is illustrated. HDL examples are given in the gate-level, dataflow, and behavioral modeling to show the alternative ways available for describing combinational circuits in HDL.	
7-10	**Synchronous Sequential Logic** Outline the formal procedures for the analysis and design of clocked (synchronous) sequential circuits. The gate structure of several types of flip-flops is presented together with a discussion on the difference between level and edge triggering. A number of design examples are presented with emphasis on sequential circuits that use D-type flip-flops. Behavioral modeling in HDL for sequential circuits is explained.	
11-12	**Registers and Counters** Deal with various sequential circuits components such as registers, shift registers, and counters. These digital components are the basic building blocks from which more complex digital systems are constructed. HDL descriptions of shift registers and counter are presented.	
13	**Memory and Programmable Logic** Deal with random access memory (RAM) and programmable logic devices. Memory decoding and error correction schemas are briefly discussed. Combinational and sequential programmable devices are presented such as ROM, PLAs, CPLDs, and FPGAs.	
14	**Review**	

Contribution of course to meet the professional component:
This course prepares students with fundamental knowledge to database design and management.

Relationship to CS program objectives and outcomes:
This course primarily contributes to the Computer Science program outcomes that develop student abilities to:
(a) An ability to apply knowledge of computing and mathematics appropriate to the programme outcomes and to the discipline;
(b) An ability to apply knowledge of a computing specialisation, and domain knowledge appropriate for the computing specialisation to the abstraction and conceptualisation of computing models;
(c) An ability to analyse a problem, and identify and define the computing requirements appropriate to its solution;
(j) An ability to use current techniques, skills, and tools necessary for computing practice with an understanding of the limitations.

Relationship to program criteria:

Criterion	DS	PF	AL	AR	OS	NC	PL	HC	GV	IS	IM	SP	SE	CN
Scale: 1 (highest) to 4 (lowest)		3		2			3							1

Discrete Structures (DS), Programming Fundamentals (PF), Algorithms and Complexity (AL), Architecture and Organization (AR), Operating Systems (OS), Net-Centric Computing (NC), Programming Languages (PL), Human-Computer Interaction (HC), Graphics and Visual Computing (GV), Intelligent Systems (IS), Information Management (IM), Social and Professional Issues (SP), Software Engineering (SE), Computational Science (CN).

Course content distribution:

Percentage content for			
Mathematics	Science and engineering subjects	Complementary electives	**Total**
0%	100%	0%	100%

Persons who prepared this description:
Dr. Sam Chao, Prof. Zhi Guo, Dr. Fai Wong, Mr. Miguel Gomes da Costa Junior

Last update:
Mr. Miguel Gomes da Costa Junior

Part B – General Course Information and Policies

2nd Semester

Instructor: Mr. Miguel Gomes da Costa Junior — Office: E11-4088
Office hour: Check on my timetable or by appointment — Phone: 8822 4378
Email: mcosta@umac.mo

Time/Venue: TBA

Grading distribution:

Percentage Grade	Final Grade	Percentage Grade	Final Grade
100 - 93	A	92 – 88	A–
87 - 83	B+	82 – 78	B
77 - 73	B–	72 – 68	C+
67 - 63	C	62 – 58	C–
57 - 53	D+	52 – 50	D
below 50	F		

Comment:
The objectives of the lectures are to explain and to supplement the text material. Students are responsible for the assigned material whether or not it is covered in the lecture. Students who wish to succeed in this course should read the textbook (chapter related) prior to the lecture, should work all in-class exercises and e-quizzes (homework) and should made use of the material provided at UMMoodle such as examples and extra material. You are encouraged to look at other sources (such as other textbooks, websites, etc.) to complement the lectures and text.

Homework policy:
The completion and correction of the in-class exercises and e-quizzes are a powerful learning experience.

- In-class exercises will be used to review lectures.
- E-quizzes are electronic homework exercises with individual submission.

Lab experiments:
The lab experiments and assignments provide students with meaningful hands-on experience.

- Each lab experiment is in a group up to 3 students. Assignments may be individual.
- Each experiment should be demonstrated during the lab. A lab report should be submitted one week after the lab session.

Exams:
One 90-minutes-mid-term exam and one 2-hour final exam will be held during the semester. Both the mid-term and final exams are closed book examinations.

Note:

- The lecture session is an important part of this course and attendance is compulsory. At most 20% absence without leave is allowed.
- Check UMMoodle (ummoodle.umac.mo) regularly for announcements, e-quizzes and lectures. Report any mistake on your grades within one week after posting.
- No make-up exam will be given except for CLEAR justification such as medical proof.
- Cheating and plagiarism are absolutely prohibited by the university.