# Part IA — Digital Electronics

Lectures by xxx
Latex by Z.Yan

Michaelmas 2016

# Contents

# 1 Combinational Logic

## 1.1 Boolean Algebra

$a + b = b + a$ communication

$(a + b) + c = a + (b + c)$ association

$(a.b).c = a.(b.c)$

$a.(b + c + ...) = (a.b) + (a.c) + ...$ Distribution

$a + (b.c....) = (a + b).(a + c)....$

$a + (a.c) = a$ Absorption

$a.(a + c) = a$

$\overline{a + b} = \bar{a}.\bar{b}$ DeMorgan's theorem

$\overline{a.b} = \bar{a} + \bar{b}$

## 1.2 Logic Minimisation

3 important Techniques:

- Algebraic manipulation
- Karnaugh (K) mapping (a visual approach)
- Tabular approaches e.g. Quine-McCluskey Method

**Definition** (Disjunctive normal form (DNF))**.** A boolean function expressed as the disjunction (ORing) of its minterms.

**Definition** (Sum of product form (SOP))**.** A boolean function expressed as the ORing of ANDED variables. (not necessarily minterms)

**Definition** (Conjunctive normal form (CNF))**.** A boolean function expressed as the conjunction (ANDing) of its maxterms.

**Definition** (Sum of product form (SOP))**.** A boolean function expressed as the ANDing of ORED variables. (not necessarily maxterms)

**Definition** (Maxterms)**.** The disjunction of all the variables either in complement or uncomplemented form.

**Definition** (Minterms)**.** The conjunction of all the variables either in complement or uncomplemented form.

**Definition** (Cover)**.** A term is said to cover a minterm if that minterm is part of that term

**Definition** (Prime implicant)**.** A term that cannot be further combined

**Definition** (Essential implicant)**.** A implicant that covers a minterm that no other prime implicant covers

**Definition** (Covering set)**.** A minimum set of prime implicants required to cover all minterms

Q-M method techniques:

-

# 2 Sequential Logic

## 2.1 Memory Elements

**Definition** (Combinational logic)**.** The output depends only on the condition of the latest inputs.

**Definition** (Sequential logic)**.** The output depends not only on the latest inputs, but also on the condition of earlier inputs. They contain memory elements.

- A memory stores data - usually one bit per element.
- A snapshot of the memory is called the state.
- A one bit memory is often called a bistable. i.e. It has 2 stable internal states.
- Flip-Flops and latches are particular implementations of bistable.

**Definition** (RS Latch)**.** An RS Latch is a memory element with 2 inputs: Reset(R) and Set(S) and 2 outputs Q and $\bar{Q}$.

| $SR$ | $Q'Q'$ | comment |
|---|---|---|
| 00 | $QQ$ | hold |
| 01 | 01 | reset |
| 10 | 10 | set |
| 11 | 00 | illegal |

**Definition** (State transition table)**.** An alternative way of viewing its operation.
It can also be expressed in the form of state diagram.

| current state | input | next state |
|---|---|---|

**Definition** (State Diagram)**.** It shows the input conditions required to transition between states.

**Definition** (Asynchronous operation)**.** The output changes occur directly in the response to changes in the inputs.

**Definition** (Synchronous operation)**.** The output of a sequential circuit is constrained to change only ay a time specified by a global enabling signal. This signal is generally known as the system clock.

The functions of clocks:

- It is a square wave signal at a particular frequency.
- It imposes order on the state changes.
- Allows lots of states to appear to update simultaneously.

**Definition** (Transparent D Latch)**.** Its output state is only permitted to change when a valid enable signal (which can be the system clock) is present.

| D EN | $Q'Q'$ | comment |
|---|---|---|
| x0 | $QQ'$ | RS hold |
| 01 | 01 | RS reset |
| 11 | 10 | RS set |

The complement function ensures that R and S can never be 1 at the same time.

**Definition** (Master-Slave D Flip-Flop)**.** The outputs changes only on the either rising (positive going) or falling edges (negative going) of the clock (i.e. enable) signal.

**Definition** (J-K Flip-Flop)**.** Differ from the D Flip-flop by the illegal state replaced by a toggle state.

| $JK$ | $Q\bar{Q}'$ | comment |
|---|---|---|
| 00 | $QQ$ | hold |
| 01 | 01 | reset |
| 10 | 10 | set |
| 11 | $\bar{Q}Q$ | toggle |

**Definition** (T Flip-Flop)**.** This is essentially a J-K Flip-Flop with its J and K inputs connected together and renamed as the T input.

| $T$ | $QQ'$ | comment |
|---|---|---|
| 0 | $QQ$ | hold |
| 1 | $\bar{Q}Q$ | toggle |

**Definition** (Setup time)**.** Is the minimum duration that the date must remain stable on the FF input before the clock edge.

**Definition** (Hold time)**.** Is the minimum duration that the data must remain stable one the FF input after the clock edge.

**Definition** (Propagation delay)**.** Is the time between the clock edge changes and the output changes.

**Definition** (Counter)**.**

- A clocked sequential circuit that goes through a predetermined sequence of states.
- A commonly used counter is an n-bit binary counter. This has n FFs and $2^n$ states which are passed through in the order $0, 1, 2...2^n - 1, 0, 1...$
- Uses include:
  - Counting
  - Producing delays of a particular duration
  - Sequencers for control logic in a processor
  - Divide by m counter (a divider), as used in a digital watch.

**Definition** (Ripple counter)**.**

- Can be made by cascading together negative edge triggered T-type FFs operating in toggle mode, i.e. T=1
- It is not clocked using the same clock. This is not a synchronous design.
- Output do not change at the same time, i.e. Synchronously. So hard to know when output is actually valid.
- Propagation delay builds up from stage to stage, limiting maximum clock speed before miscounting occurs.
- The counter output signals double the repetition period of the previous one. (known as dividers)
- To count not a power of 2:
  - Use FFs having a Reset/Clear input
  - Use an AND gate to detect the count of 10 and use its output to reset the FFs.

**Definition** (modified state transition table)**.** This table has additional columns that define the required FF inputs. (excitations as it is known)

| current state | input | next state | FF inputs |
|---|---|---|---|

**Definition** (Characteristic table)**.**

| FF inputs | next state |
|---|---|

**Definition** (Excitation table)**.**

| current state next state | FF inputs |
|---|---|

**Note.** This is vitally important in considering the do not care state when using the J-K FFs to build the sequential logic

**Definition** (Synchronous counter)**.**

- Write down the state transition table

- Determine the FF excitation (easy for D-Types)
- Determine the combinational logic necessary to generate the required FF excitation from the current states.

  **Warning.** Remember to take into account any unused counts since these can be used as do not care states when determining the combinational logic circuits.

**Definition** (Shift register)**.**

- A shift register can be implemented using a chain of D-type FFs
- Has series input $D_{in}$ and parallel output $Q_0, Q_1$ and $Q_2$
- Data moves on position to the right on the application of each clock edge
- preset and clear inputs on the FFs can be utilised to provide a parallel date input feature.
- Date can then be clocked out through $Q_2$ in a serial fashion, i.e. We now have a parallel in, serial out arrangement
- Serial in, parallel out shift register arrangement can be used as the basis for a serial data link

**Definition** (Serial data link)**.** – One data bit at a time is sent across the serial data link

- Less wires are required than for a parallel data link

## 2.2 Synchronous State Machines

**Definition** (Finite state machine (FSM))**.** A deterministic machine (circuit) that produces outputs which depend on its internal state and external inputs

**Definition** (States)**.** The set of internal memorised values, shown as circles on the state diagram

**Definition** (Inputs)**.** External stimuli, labelled as arcs on the state diagram

**Definition** (outputs)**.** Results from the FSM

**Definition** (Moore machines)**.** Outputs from Moore machines come directly from clocked FFs so:

- They have guaranteed timing characteristics
- They are glitch free

**Definition** (Mealy machine)**.** Outputs from Mealy machine depends upon the timing of the inputs

**Note.** Any Mealy machine can be converted to a Moore machine and vice versa, though their timing properties will be different.

**Definition** (Self start)**.** The FSM can eventually enter a known state from any of the unused states.

- If not, add additional logic to do this i.e. include unused states in the state transition table along with a valid next state
- Alternatively use asynchronous Clear and Preset FF inputs to set a known (used) sate at power up

**Definition** (State assignment)**. Sequential state assignment** Simply assign the states in an increasing natural binary count using minimum FFs $\log_2 m$

**slide state assignment** By inspection we can see that we can use any of the FF outputs as the wanted output.

**Shift register assignment** The FFs are connected together to form a shift register. In addition, the output from the final shift register in the chain is connected to the input of the first FF: consequently the data continuously cycles through the register

**Definition** (One hot state encoding)**.** This is a shift register design style where only one FF at a time holds a 1

- Have 1 FF per state
- Result in simple fast state machines
- Outputs are generated by ORing together appropriate FF outputs

**Note.** The one hot is easier to design but it results in more hardware compared with the sequential state assignment design

**Definition** (Redundant state)**.** Sometimes when designing state machine it is possible that unnecessary states may be introduced.

**Warning** (Elimination algorithm by table)**.** This is a non-completed algorithm. It is possible to do some further simplification even after the algorithm finished!

**Definition** (Generic array logic (GAL))**.** They are similar in concept to PALs, but have the option to make use of a D-types FF in the OR plane (one following each OR gate). In addition, the outputs from the D-types are also made available to the AND plane (in addition to the usual inputs)

- Consequently it becomes to possible to build programmable sequential logic circuits

**Definition** (Field Programmable gate array (FPGA))**.**

- FPGA are the latest type of programmable logic
- Are a sea of programmable wiring and function blocks controlled by bits downloaded from memory
- Function units contain a 4-input 1-output look-up table with an optional D-FF in the output