08-Development_Methodology_for_Digital_Circuits

August 22, 2019

1 8 - Development Methodology for Digital Circuits

1.1 Development Model - V Diagram

1.2 Scrumm Planning Poker

https://en.wikipedia.org/wiki/Planning_poker

Weight Tasks only with specfic values, mostly close to Fibbonacci numnber.

Value	Description
coffee	Omit from evaluation
0	No effort
$\frac{1}{2}$	-
1	-
2	-
3	-
5	-
8	-
13	-
20	-
40	-
100	-
?	unsure
∞	cannot be completed

1.2.1 Specification Phase

System Specification Solely created by Costomer

Specification Documents

- General Spezification
- Funktional Specification
- Performance specification

1.2.2 Draft Phase

- Split per functions
- Architecture Document
  - Small Description
  - List In- Outputs and Pin definitions
  - Split digital circuit into functional blocs
  - Estamation of size of digital circuit
  - Estimation consumation
  - List critical functions
  - Test strategy, Validation strategy, definition Test-I/O

1.3 Synchron vs Asynchrone Systems

1.3.1 Gate Delay Considerations

The gate delay time must be taken into account when designing the circuit.

Due to the cycle time, there is often a delay between a change at the input and the corresponding adaptation at the output. This depends on the technology used and can vary greatly from one gate to another within the same technology. The delay also depends on the gate load.

Due to these delays, the output of a logic block may be in a different state than the input for a short moment. Such a time delay corresponds to the situation where two almost simultaneous transitions between two identical successive states occur at the block output. Such time disturbances are due to the delay times. The variation of an input signal can reach the output in two or more parallel paths, each with different delays.

1.3.2 Gate Delay Independence

Gate delay times are unreliable system parameters. It is therefore recommended that systems be designed so that they are not affected by gate delay times.

Synchronous logic is used for this purpose. A synchronous logic system is a system in which the clock signals of all toggle circuits are connected to the same command signal.

1.3.3 Sequential Logic and Gate Delay Rules

Rule 1 Any sequential logic must be synchronous. Therefore, if possible, use a single external clock generator that is only active on one edge. The most important elements of these sequential systems are the D or E flip-flop.

Rule 2 It is forbidden to use the logic element cycle time to generate pulses (different cycle times of the original signal and the delay chain).

Rule 3 It is preferable to route the clock signals directly to the flip-flop rather than controlling them with a logical element to avoid any time interference at the inputs. It is also necessary that the signals at the inputs synchronized by the clock keep the setup and hold times with respect to the clock.

Remarks: If the number of gates or consumption is to be kept as low as possible, compliance with the third rule may prove unfavourable. Asynchronous frequency dividers require much

less elements than synchronous frequency dividers. If they are used together with logic elements that work at a higher frequency, the divider outputs must be re-synchronized with the higher frequency signal. Some input or output blocks require asynchronous operation to respond quickly to external conditions. In this case, the asynchronous part of the system must be kept to a minimum. It is important to intensively analyze the system's timing behavior for the asynchronous system parts and to design function blocks without gate delays.

Rule 4 Each circuit must be set to a known state when it is fed into the circuit and at the start of simulation. The asynchronous inputs set and reset of the flipflops must be used. These inputs must not be used to operate the circuit, but only to ensure testability. Therefore, no signals with time interferences should be routed to the asynchronous inputs of the flip-flops. The set and reset signals must not originate from purely combinatorial decodings.

Rule 5 The shortest period of a clock generator of a synchronous computer is calculated as follows:

$$T_{min} \leq T_{ClkQ_{max}} + TQD_{max} + T_{skew} - T_{setup_{max}}$$

- $TClkQ$ - is the delay time between the clock edge and the flip-flop output Q
- TQD_{max} - is the delay of the longest chain of gates between an output Q of a sequential logic and an input D of a sequential logic responsive to the same edge of the same clock
- $Tskew$ - is the clock shift with respect to the clock inputs of the sequential logic
- $Tsetup$ - is the setup time of the sequential logic

Remarks: In order to increase the functional speed of a circuit, synchronization registers can be inserted into the large gate chains.

Rule 6 The input and output signals of a system must be synchronized with the aid of D-flip-flops. By means of the figure below, the time disturbances and metastable states that occur when a signal passes from an asynchronous to a synchronous system can be filtered.

Rule 7 The disappearance of the internal initialization signal of the circuit must be synchronous to the clock, but its appearance is asynchronous.

Rule 8 Do not overload the gate outputs too much. For large signals, estimate the fan-out of the gate that generates it and the fan-in of the gate that must control it.

1.4 Verification Phase

1.4.1 Validation

Do we create **THE** correct product?

1.4.2 Verification

Do we create a correct product?

1.4.3 Techniques

- Simulation of VHDL code
- Examination of the design documentation by competent auditors
- Mutual proofreading of the VHDL code by the various team members

Identificationnumber of the Specification request	Description	Testbench & Simulation Setup	Time of validation	Validation status	Attachment
`DSxx-YYXXX` `DSxx` Number in Specification document `YY` Chapter number `XXX` Number of Functional Specification	Keywords of Functional Specification to be validated	Identification of used Testbanch and Simulation Steps	Time when the validation took place	Validation Method (Analyses, visual verification, Print, Log file, automatic validation	Validation Status (OK

1.5 Integration Phase

1.5.1 Results

The following elements needs to be part of the document: * A detailed design report with a detailed description of the design activities, the coding and verification, the validations of the partial circuits as well as the integration and final validation. The focus must be on the conscientiousness of the validations performed and on the elements that will allow another team to continue the project at a later stage. * User manual. This is the document that is delivered to the customer (end user). It must describe the installation, settings and use of the material. This document is actually written in the course of the project, as it adopts the model developed during the general design phase (which is based on the functional description of the product created during the specification phase).

1.5.2 Execution Techniques

- Scheduling problems: The delays accumulated in the previous phases will inevitably become noticeable as the deadline approaches. Participants therefore often panic instead of focusing on quality and organisation. There is only one solution: PREVIEW, PREVIEW, PREVIEW. Whether the deadline is met is already decided during the specification phase. The developers and especially the project managers have to be very careful, especially in the initial phase (the two weeks that are missing at the end are often lost at the beginning). By conscientious project monitoring, catastrophes can also be avoided by redefining objectives in good time.
- Managing the work of the group: Many reflections of the previous phases can and/or must be carried out together in the group. However, this phase is characterised by a lot of individual work. The work can only be done if it is managed in parallel. But the simple addition of individual tasks does not automatically lead to their sum. This requires organization. There-

fore, a structure must be set up for the distribution of tasks among team members and the time required for synchronization. The larger the group, the more precise the organization needs to be and the longer the synchronization takes. This time must be explicitly taken into account during planning. For a group of 5 to 6 people working full-time on a project, a daily briefing (coffee break) and a structured weekly meeting seem to be a realistic solution.

See also Scrum Development