Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

Characterizing WDT subsystem of a Wi-Fi controller in an Automobile based on MIPS32 CPU platform across PVT

Sukrutha A Jain[1], Avinash Bharadwaj[2], Chandrashekar Murthy B N[3]

[1]Department of Electronics & Communication Engineering,
SJCE , JSS Science and Technology,
Mysore.
E-mail: sukruthaajain@gmail.com

[2]Staff Engineer,
WSG Validation Team,
Microchip Technology,
Bangalore. Inc.
E-mail: avinash.bharadwaj@microchip.com

[3]Department of Electronics & Communication Engineering,
SJCE, JSS Science and Technology,
Mysore.
E-mail: chandrashekar@sjce.ac.in

Abstract - Currently due to increase in stress in present lifestyle, there is a need for people to choose automated cars for transportation purpose. As a result the controllers incorporated in the automobiles must be critically characterized to ensure safety, security and confront of the users.

The whole SoC consists of various Sub-modules among which WDT is one of the crucial and most common block found in all silicones. Yet validation of this module has to be done so as to guard the system from going into an hang-over due to some malfunction in hardware devices in the real-time environment or some software faults.

This paper emphasizes the characterization of the watch dog timer in a Wi-Fi controller based on MIPS32 CPU platform across PVT. Hence the watchdog timer module is critically characterized across PVT to stress test the system and the obtained data logs are analyzed to form a comprehensive report.

Keywords: SoC, watchdog timer, MIPS32, validation, characterization

1. INTRODUCTION

An automated car which when in auto mode after setting the destination will drive the user safely. This module ensure a fail-safe mode using watchdog timer (WDT) and various timer modules. It reduces the risk of accidents and delivers the passengers to their location safely. It not only takes care of the user but also the pedestrians but also the automobile. The controller acts as an Electronic Control Unit (ECU) which controls the engine management system, head lights, locking system, infotainment systems et.all which is also used to communicate with the other automobiles on the road.

Initially the chip is verified after the design level which is called verification. Then the design is fabricated into a Silicon which is referred as SoC (System on Chip). Post-Silicon validation is a critical characterization process in which the modules of the system are evaluated after the SoC is manufactured. Post-Silicon validation is a process in which the fabricated SoC is characterized for adherence to all specified functional and electrical features; or deviations in a lab setup. This is done by assembling the fabricated and packed SoC (referred to as DUT hereon) on a test-bench with associated other components which are required to characterize the system depending up on the purpose for which the SoC was designed for.

The main goal of this paper is to characterize all use cases of the device in order to ensure that the SoC functions as expected and meets the customer needs and specifications. System level validation is done on the devise where

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

the actual software program is fed into the DUT through an IDE and then run on the headwear. Validation is performed for each and every blocks of the sub-module and then the whole sub-module is evaluated to confirm the combined functionality of the sub-module. Then the related sub-modules are made to communicate and interact with each other. Therefore the whole SoC is validated from internal pins to the peripherals for various types of input combinations.

Initially, the specifications related limitation of the device is understood by referring to the Design, Objective, Specification, Programmers Guide, User Reference Manual and Data Sheets of the DUT(device under test). Familiarities with the schematics of the test board are also prerequisites. Then a validation plan is prepared keeping in view all the characteristics key features of the sub-module. It should cover all the logical and variable test cases of the module covering each and every feature as mentioned in document. So that it provides assurance that the DUT meets the pre-determined specifications. A validation procedure is prepared to promise means of executing those test cases. The test suite is built to implement those listed cases as per the prepared procedure. These test suits are run on a few samples of DUT to check it's adherence to specifications, reliability, accuracy and interoperability of the DUT in real time systems. Collected data is analysed and summarized in the form of reports which are then used to update datasheets and other customer visible literature.

Major advantages of validation process is to provide documented evidence that DUT adheres to it's design specifications and thereby eliminate the probability of product recall from the market. It helps to consolidate the silicon parts usage depending on specified range of operation.

The WDT can be a sub-module with in the Chip or it can be a hardware timer out-side the module. It can be either internal or external WDT but the functionality of it to reset the CPU if the main program fails to reset the WDT does not change. The period register is fed with a value that is maximum time delay or elapse time required for the long sub-program to be executed. If the time elapses the WDT gives out an interrupt which is given to the system as an NMI(Non-maskable Interrupt). During the execution of the main program watchdog timer is serviced periodically i.e the WDT timer/counter is reset. If the CPU becomes busy or hangs up executing some erroneous software program then watchdog timer helps the system to get back to it's normal routine by resetting the system.

2.LITERATURE REVIEW

The paper "Design of Watchdog Timer for Real Time Applications" explains a part of the engineering and plan of a better-quality windowed guard dog clock and its execution in FPGA. The guard dog clock turns totally autonomous of the processor and licenses changing the clock limits as indicated by the application. A few blame recognition procedures are incorporated with the guard dog for the primary discovery of whimsical programming modes. It has the capacity to distinguish the disappointment type and then permits the product adequate time for storing required data, before starting a reset[1].

The paper "High-Voltage Watchdog Timers Enhance Automotive System Safety", explains a vehicle-operating mode that allows a fail-safe mode to be entered and permits the vehicle to be driven home slowly. Limp-home mode can be implemented so that a faulty car can get its passengers home safely, while limiting damage to the car's engine and other components. Limp home includes both the engine-management system and the ancillary safety equipment, such as lights and windshield wipers, which are generally controlled by the body control unit (BCU) [2].

"TPS3813-Q1 Processor Supervisory Circuits With Window-Watchdog" explains the TPS3813-Q1 supervisory circuits provide circuit initialization and timing supervision, which are primarily for DSPs and processor-based systems. The paper also describes the pin configuration of the chip and functionality in detail [3].

The paper "A Design and Implementation of Watchdog Based on Observer Pattern and Finite State Machine", proposes and explains a method with Observer Pattern and Finite State Machine for watchdog design. The simulation indicates that it will greatly improve the reliability and maintainability of the program because sophisticated protection strategies can be implemented easily without having too much unmanageable control logic [4].

Paper "Connectivity verification of Zynq UltraScale+ MPSoC With TTC and WDT Interrupts", presents the basic Connectivity verification of TTC and WDT blocks with performing write and read operations (connectivity checks) to the appropriate registers by using the advanced extensible interface (AXI) and Advanced Peripheral Bus (APB) in the whole architecture with different interrupts[5].

Article, "An Improved Watchdog Timer to Enhance Imaging System Reliability In The Presence Of Soft Errors" explains Satellite/Ariel imaging frameworks is situated at highest elevations. It also examines the impact of transient blames on chip basis imaging frameworks [6].

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

"FPGA Implementation of Multiple Hardware Watchdog Timers for Enhancing Real-Time Systems Security" paper manages to verify the ongoing frameworks by giving them extra external guard dog clocks. It proposes the fundamental idea of the numerous equipment guard dog clocks framework which depicts the proposed design of the framework giving 256 equipment guard dog clocks[7].

The paper "A Design and Implementation of Watchdog Based on Observer Pattern and Finite State Machine", presents, a strategy with Viewer Pattern and Finite State Machine for guard dog execution is proposed[8].

"Application of Software Watchdog as a Dependability Software Service for Automotive Safety Relevant Systems", confronts the difficulties coming about because of the expanding thickness of the user programming segments and higher reliability prerequisites of things to come wellbeing frameworks in the car hardware, a constancy programming administration to screen singular application programming parts in run time is required so as to improve the general framework trust worthiness[9].

"Improved watchdog timer for control the IBM PC based autonomous computer systems" describes the watchdog timer as personal and embedded computers that utilizes the keyboard communication port. The device generates the special scan codes periodically which awaits corresponding answer (scan codes for toggling the LED indicators on the keyboard) from the monitored system. When it dose not receive the answer, it resets the whole system [10].

In "FPGA-based concurrent watchdog for real-time control systems" it deals with the FPGA implementation of the customized watchdog processor for a particular embedded system. In this case, the circuit does not provide a timer for the several processes but provides more complex fully customized functionality. It provides a reasonableness check on some variables[11].

"Embedded Systems with Increased Reliability Using the Multiple Watchdog Timers Approach", describes the prototype CPLD implementation of a system which includes only one watchdog timer and communication interface. It also explains the enhanced and optimized architecture of the multiple hardware watchdog timer system [12].

3.METHODOLOGY

A watchdog timer is a free runnning timer which consists of a hardware timing device which is repeatedly serviced by a software program routine. It watches the device and acts as a fail safe guard. It safeguards the system by acting as a barrier against dead lock or lockup or execution of endless loop.

The main program consists of a WDT reset routine which is executed periodically at regular intervals repeatedly, if not the CPU will be reset by the guard dog timer which is why it's named as watch dog timer. Figure 1 explains the functionality of the watch dog timer. The WDT can be a sub-module with in the Chip or it can be a hardware timer out-side the module. It can be either internal or external WDT but the functionality of it to reset the CPU if the main program fails to reset the WDT does not change.

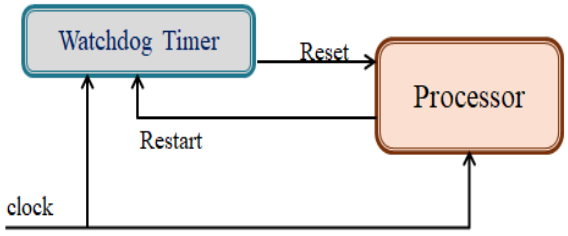

Figure 1: A typical Watchdog timer

Generally watch dog timer will always be in enabled state, as a precautionary measure to avoid the device the device from going into a hang-over state or a software malfunction. The required time lapse can be configured in the period register. Various WDT time-out periods can be configured as per the user needs in order to delay the rate of incrementing of the counter. This in turn increases the WDT time-out or in other words more time can be allocated to critical sub-routines of CPU. WDT is one of the wake source to wake the device from either idle or sleep modes.

Key features of WDT module include the software and hardware controllability of the module. It has time-out period which can be configured by the user. WDT period registers can be configured by the user as per their needs. The counter counts at the rate of input clock and pre-scalar value. The comparator compares both the

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

counter and period register values periodically, when they both match then the WDT reset event is enabled in order to reset the system. The WDT is a hard ware free-running timer which uses the low-power RC oscillator as the base clock. As it's a free-running clock it does not need any external components to maintain the module's functionality. Therefore, the WDT will continue it's operation even if the system's primary clock source i.e., the crystal oscillator which remains non-functional under normal operation like in Sleep mode.

It has two separate counters, Run mode counter and Sleep mode counters. It has an added advantage as Run mode counter only works in run mode and retains the value till the system exits from sleep mode. The Sleep mode counter remains active only in sleep or idle modes. Block diagram of WDT is as shown in figure2 below.

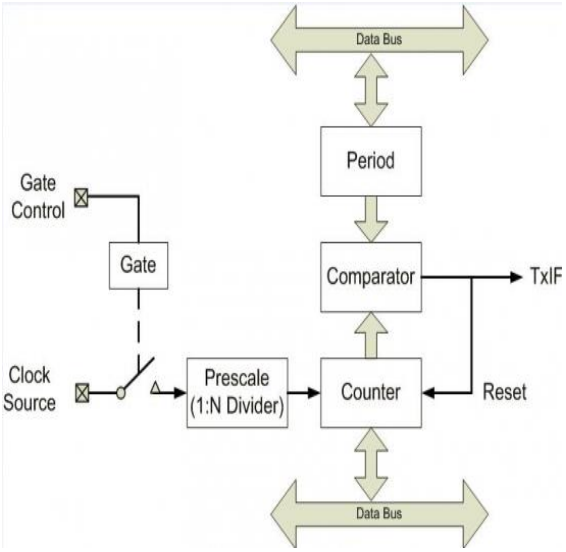

Figure 2: Block diagram of WDT

A. **Clock Source:** The WDT is a free-running timer with a configurable post-scaler. The counter is clocked with an external reference clock until the counter value exceeds the period register value. If enabled, the WDT will never stop it's operation operate even if the main processor clock (e.g., the crystal oscillator) fails. The clocks are selected based on the select line of the MUX. The different clocks that can be provided are SOSC (Secondary Oscillator), LPRC or pb clock.

B. **Window mode:** The WDT can have two modes of operation, Windowed and non-Windowed. In Windowed mode, software can clear the WDT counter only when the counter is in its final window range before a period match occurs. There are four window size options (25%, 37.5%, 50% and 75% of the total WDT period). The window size is determined by the configuration fuses. If the software tries to reset out-side the window range then it causes device reset instead of counter reset. In non-Windowed configuration mode, software can clear the WDT counter any time before the period match occurs. This mode can only be activated in run-mode.

C. **WDT Period Selection:** The internal LPRC Oscillator is used as the clock source, which nominally has a frequency of 32 kHz. Therefore a nominal time-out period of 1 millisecond is provided by it if post-scalars are not configured.

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

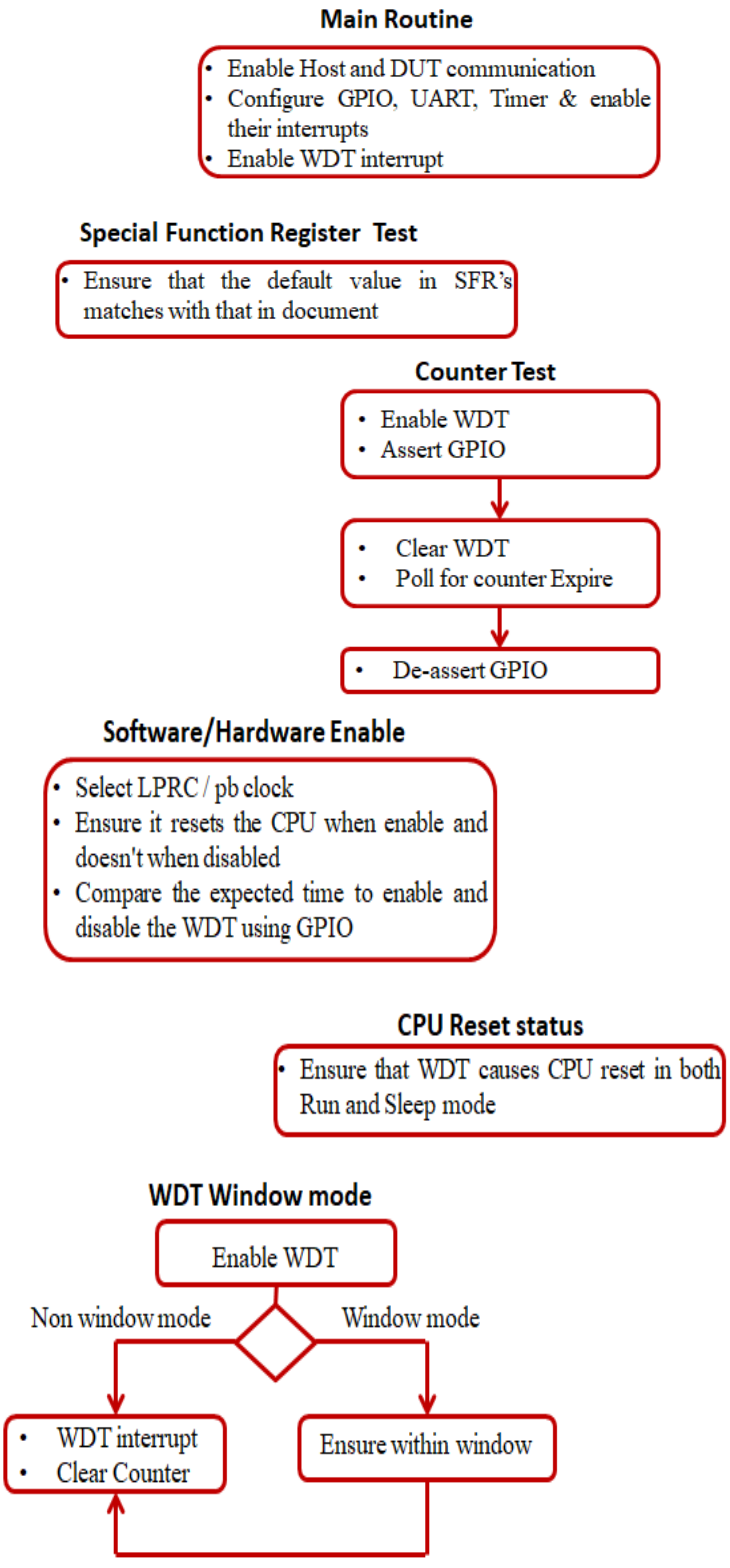

Figure 3: Flow chart of characterization plan of WDT

A general purpose pin (GPIO) is used to check functionality in windowed and gate mode of operation. The time out period is read by using the asserted and deserted bits of GPIO using the oscilloscope and then comparing the both the measured and the expected time out periods. Characterization of period and Special Function

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

Registers (SFR)by software write using CPU. Windowed functionality is validated by ensuring that an attempt to clear the WDT out-side the window range will cause a CPU and the software can clear the WDT any time before the period match occurs. The checks can be summarized as in figure 3.

Special function registers (SFRs) act as an interface between the core and peripheral hardware functionality of the DUT. They are commonly mapped into the processor's address space. So this test is done to validate whether all the bit fields in the WDT related registers are set to the default values as per the document.

Counter test is performed in order to ensure that the internal counter of WDT is ticking and working as expected. Software enable/ disable provides flexibility to the user to choose when to have WDT functionality enabled or disabled. Hardware enable/ disable ensures that the module can even be controlled by using fuse bits or Reset button. CPU Reset test ensures that WDT causes CPU reset in all the modes i.e run mode, idle mode and sleep mode. Window mode tests validates the functionality of the module in different modes and the WDT reset event takes place only within the specified window range and if attempted to clear it outside the window range, then it leads to devise reset.

Table 1.1: Summarized table of passed and failed tests

Test	Description	Status	Explanation
Special Function Register Test	This test validates whether all the bit fields in the WDT related registers are set to the default values as per the document	Pass	Therefore it Satisfies the documented default values of registers
Counter Test	This test verifies weather the counter is working as expected, by verifying the time required to reset the CPU as counter value cannot be read directly.	Fail	Some test cases have failed due to mismatch between measured and expected time periods
Software/Hardware Enable	Verify whether the Watchdog timer enable and Watchdog timer window mode enable methods: • Can be enabled by fuses • Cannot be disabled by SW • Can be disabled by HW	Pass	Therefore it Satisfies the functionality of the block that it can be enabled by software switch and fuse bit, but cannot be disabled by software.
WDT Window Mode	Verifies that WDT reset event takes place only within the specified window range and Any attempts to clear the WDT when the window is not active would cause a device Reset.	Fail	This test case is violating CPU reset functionality across VT at different window modes randomly

The test suit is run on several parts selected at random across process skews, a specified range of operating voltage and temperature (PVT) to check reliability, accuracy, compatibility of the chip with real time operating systems. These collected data are analysed and summarized as in table 1.1. There was potential for inaccuracies in counter countdown due to low distribution part specific fall out of on-chip low power RC oscillator with a predominantly low temperature trend.

An on-chip precision oscillator which is known to have high sensitivity to load has exhibited out of bounds variation in frequency with a low distribution part specific trend at random temperatures. This might have potentially caused the system Watchdog timer to malfunction.

The graphs are plotted by observing the logs obtained during the data collection process. Some test cases have failed due to violation of CPU reset functionality across VT at different window modes randomly. A few other cases have failed due to mismatch between measured and expected time periods.

The graphs are plotted in order to analyze the pass/fail results among many iterations across VT. The graphs are as shown in figure4 – 8.

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

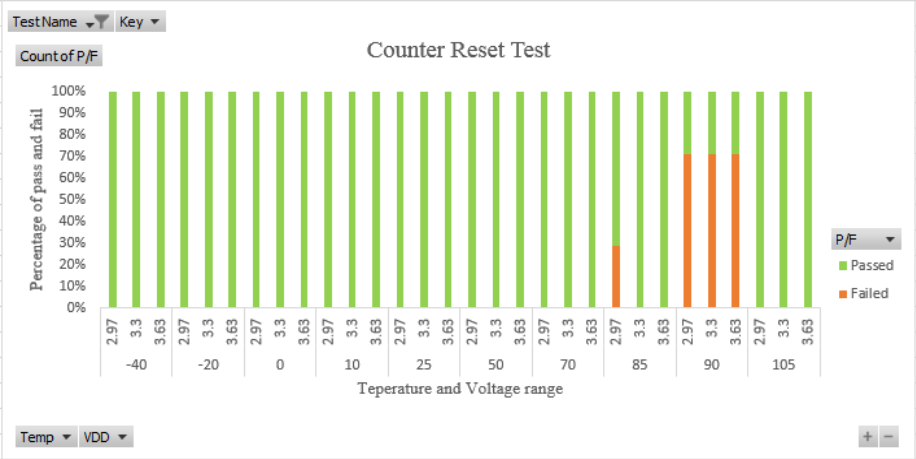

Figure 4: Analysis of Counter Reset Test.

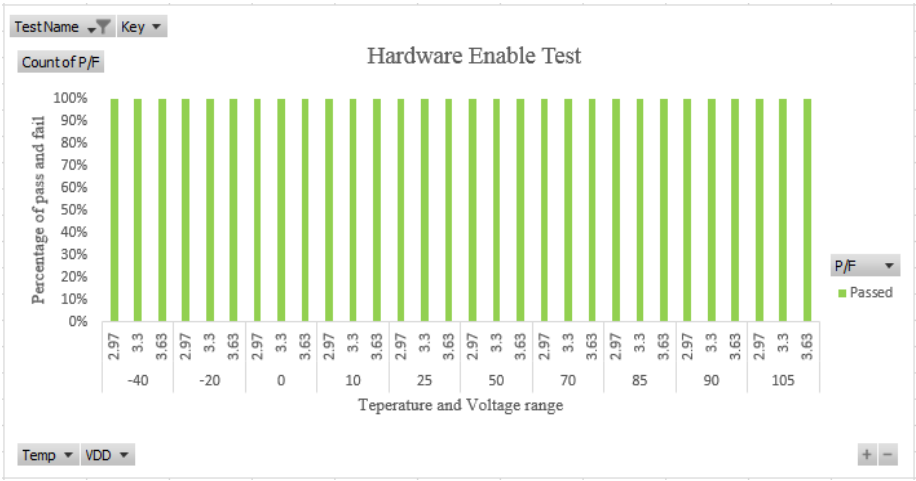

Figure 5: Analysis Hardware Enable Test.

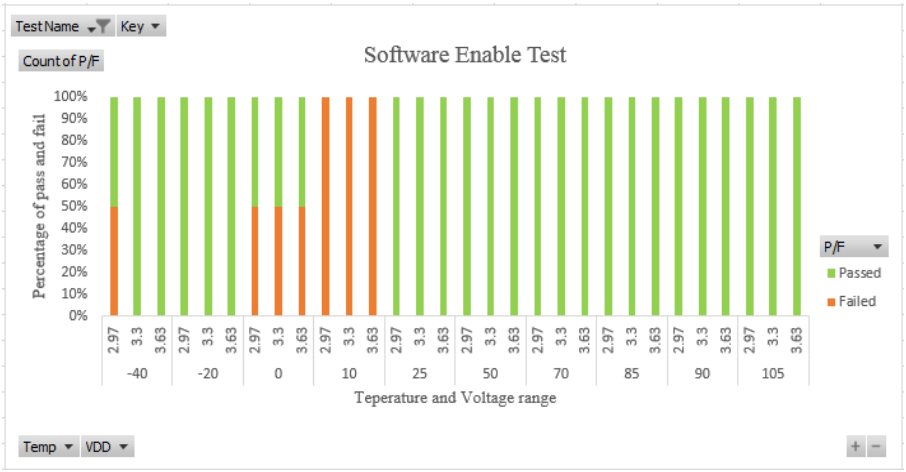

Figure 6: Analysis of Software Enable Test.

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

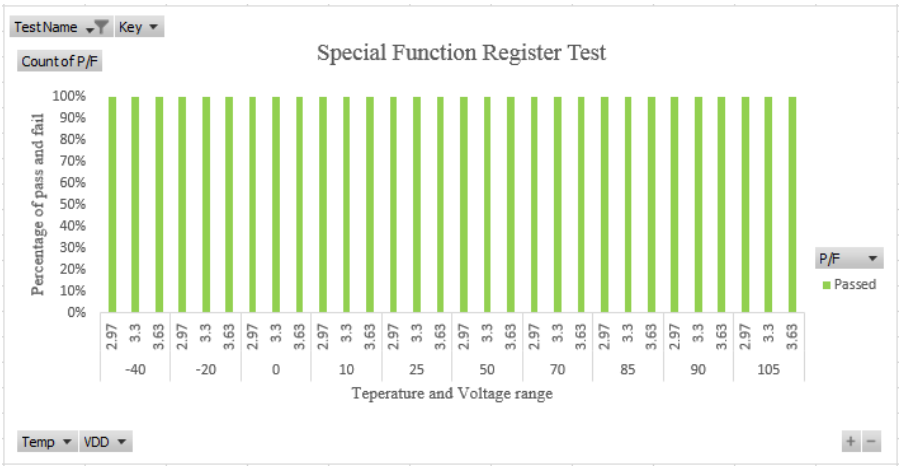

Figure 7: Analysis of Special Function Register Test.

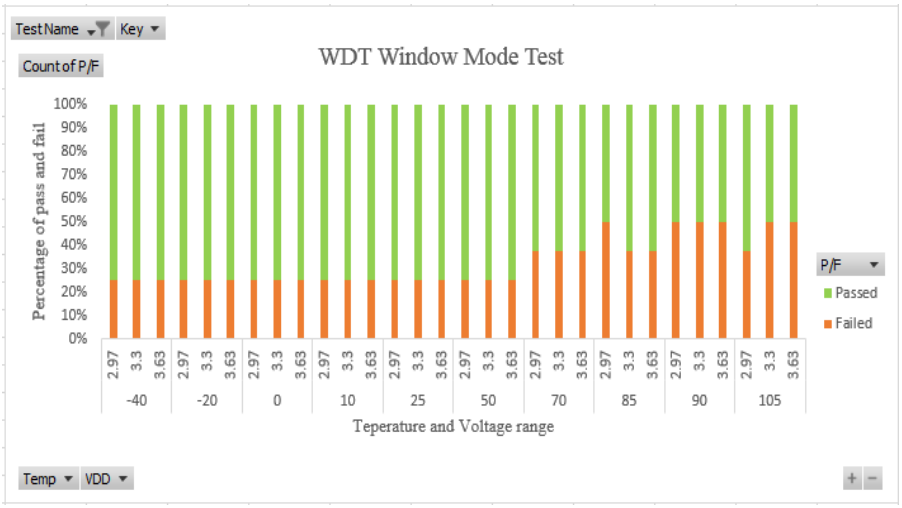

Figure 8: Analysis of WDT Window Mode Test.

The overall summarized graph with all test cases in order to analyze the whole module is as shown in figure 9.

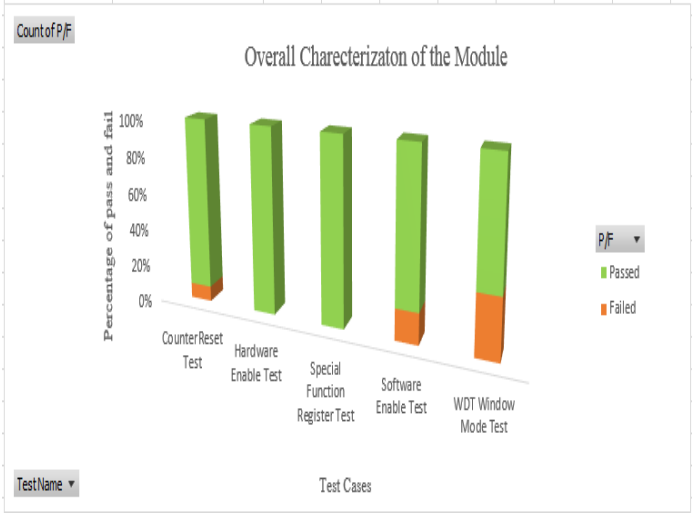

Figure 9: Analysis of Overall Characterization of the Module.

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

4. CONCLUSION

This paper presents the method to characterize the Watchdog timer subsystem of a Wi-Fi module based on a MIPS32 platform across VT. This module can be incorporated in an automobile to increase the safety and reliability of the system. It demonstrates the method of validating and characterizing the functionality a subsystem using the Test suit. The Watchdog timer is characterized to ensure the features and reliability of the devise in the real time environment. Therefore WDT is critically characterized in order to make sure that it satisfies all it's functional and electrical behaviours. The data logs are studied in detail and comprehend the test cases in to a detailed encapsulated report.

References

[1] S.Shilpa, Ch.Umashankar, S V S Prasad, "Design of Watchdog Timer for Real Time Applications", IJITEE, Volume-8, Issue- 9S2, July 2019.
[2] Robert Regensburger, "High-Voltage Watchdog Timers Enhance Automotive System Safety", Automotive Product Definitions, Maxim Integrated Products, Germany Jan 05, 2012.
[3] "TPS3813-Q1 Processor Supervisory Circuits With Window-Watchdog", Texas Instruments, October 2019.
[4] Xinyun Liu, Shihang Chen, Wenqiang Song, "A Design and Implementation of Watchdog Based on Observer Pattern and Finite State Machine", IEEE International Conference on Reliability, Maintainability and Safety (ICRMS), pp-407 – 411, 20014.
[5] A.RAJANI, Assistant Professor, JNTUK P.Vijitha, "Connectivity verification of Zynq UltraScale+ MPSoC With TTC and WDT Interrupts", International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017.
[6] A.M. El-Attar and G. Fahmy, "An improved watchdog timer to enhance imaging system reliability in the presence of soft errors," in Signal Processing and Information Technology, 2007 IEEE International Symposium on. IEEE, Dec. 2007, pp. 1100–1104.
[7] M. Pohronska and T. Kraj covic, "FPGA implementation of multiple hardware watchdog timers for enhancing real-time systems security," in EUROCON-International Conference on Computer as a Tool (EUROCON),2011 IEEE. IEEE, Apr. 2011, pp. 1–4.
[8] Xinyun Liu, Wenqiang Song, Shihang Chen, "A Design and Implementation of Watchdog Based on Observer Pattern and Finite State Machine", International Conference on Reliability, Maintainability and Safety (ICRMS), 2014.
[9] Xi Chen, Juejing Feng et.all, "Application of Software Watchdog as a Dependability Software Service for Automotive Safety Relevant Systems", 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 2007.
[10] Kochan, R.Kopylchak, Korkishko, "Improved watchdog timer for control the IBM PC based autonomous computer systems", in Modern Problems of Radio Engineering, Telecommunications and Computer Science, 2002.', pp. 181-182.
[11] Giaconia.G, Di.Stefano et.al "FPGA-based concurrent watchdog for real-time control systems", 2003, Electronics Letters 39(10), pp. 769-770.
[12] Pohronska.M and Krajcovic, "Embedded Systems with Increased Reliability Using the Multiple Watchdog Timers Approach," 2010 International Conference of Applied Electronics Proceedings, J. Pinker, ed., University of West Bohemia in Pilsen, 2010, pp. 273-276
[13] Microchip Technology Internal Documents.

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020

Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2020)
Vol.02/ No.04
Pages: 187-196
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2020.4.001

Authors Biography

SUKRUTHA A JAIN is perusing M.Tech., at SJCE in Automotive Electronics. Working as an intern at Microchip Technology Inc., as Post-Silicon Validation Engineer in Wireless Silicon Group.

AVINASH BHARADWAJ: Post-Si validation and board design engineer with around 17 years of hands-on experience in embedded systems industry, of which 15 years have been in semiconductor industry. Experienced in leading and directly managing teams responsible for validation/reference board design and post-Si validation. As technical lead for post-Si validation, responsible for coordinating all validation tasks from First Si till product qualification, including functional and electrical validation/debug, SI testing and AC characterization and power measurement.

CHANDRASHEKAR MURTHY B N: Working as an Assistant Professor in the Department of Electronics and Communication Engineering. He graduated and obtained his M.Tech., from AMC engineering college, his area of specialization is Digital Electronics and communication.

ISSN: 2582-337X
Submitted: 22.10.2020
Accepted: 17.11.2020
Published: 24.11.2020