Firmware-Driven Bluetooth ATE Testing Optimization

Manibharath N, Pragya Nahar
Test Engineering
Infineon Technologies, San Jose
Manibharath.N@infineon.com, Pragya.Nahar@infineon.com

Abstract—Over the past two decades, Bluetooth technology IEEE 802.15.x has seen significant improvements, with each version enhancing power efficiency without sacrificing range. However, achieving top yields while keeping test costs low remains a challenge in the ATE (Automatic Test Equipment) manufacturing environment. This paper offers a detailed study of Bluetooth testing methods in ATE production. We compare the effectiveness of two approaches: pattern-based and firmware-based, demonstrating a 10% increase in yield and a 58% reduction in test time through firmware-based testing. Our results aid in optimizing Bluetooth testing in ATE production, supporting cost-effective solutions for dependable and efficient connectivity.

Keywords—ATE, Bluetooth, GFSK, RF, Yields, Test time

I. Introduction

Bluetooth technology performance test specifications, as outlined by the Bluetooth Special Interest Group (SIG), form the foundation for Automated Test Equipment (ATE) testing. Key Bluetooth test parameters include Tx Power (transmit power), Adjacent Channel Power (ACP), Differential Error Vector Magnitude (DEVM), and modulation characteristics such as Delf1 and Delf2. These parameters are related to the performance metrics of Bluetooth transmitters. Tx Power indicates the strength of the transmitted signal, while ACP measures the power leakage into neighboring channels. DEVM assesses the signal quality of differentially encoded Bluetooth Enhanced Data Rate (EDR) packets. The Delf1 and Delf2 parameters pertain to frequency deviation modulation characteristics.

Additionally, Receiver Packet Error Rate (RX PER), RSSI (Received Signal Strength Indicator), Rx ADC (Analog-to-Digital Converter) SNR (Signal-to-Noise Ratio), and Gain tests are essential metrics for assessing Bluetooth receiver performance. By analyzing these parameters, engineers can ensure the reliable and efficient operation of Bluetooth devices across various applications. In the ATE environment, both pattern-based and firmware-based testing methods can be utilized to evaluate these parameters.

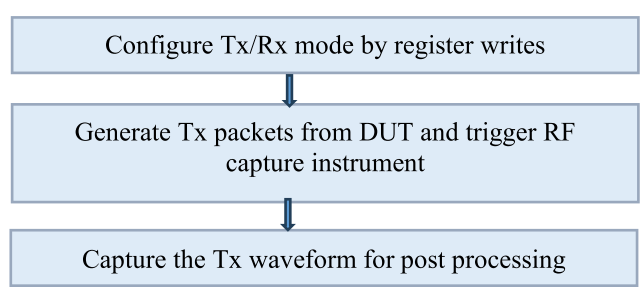
This paper examines the trade-offs between pattern-based and firmware-based approaches for ATE testing. It shows how challenges were overcome to achieve top-tier yields and testing efficiency by switching to a firmware-based Bluetooth testing method. By comparing the two approaches and explaining the optimization process, this study offers valuable insights for engineers aiming to improve the efficiency and reliability of Bluetooth testing in production settings.

II. Discussion

In the traditional pattern-based method, the JTAG / SWD interface is used to write device configuration registers, setting the Bluetooth mode. Calibration algorithms, such as Analog PMU(Parametric measurement unit) output, TSSI (Transmitter Signal Strength Indicator), and LOIQ (In-phase and Quadrature-phase Imbalance), are implemented and run to estimate calibration codes. Bluetooth Tx patterns are then executed, producing Tx-related output modulated waveforms. In the pattern-based method, the ATE capture is triggered within the pattern, ensuring that the captured data stays synchronized with the device's transmission. This approach allows for precise measurement and analysis of Bluetooth signal characteristics.

In a pattern-based approach, the device is set to the desired Tx mode (Classic BT, Low energy BLE, LE2, or BLELR) by JTAG writes to the 100s of registers. Different power mode configurations are also set via JTAG register writes. The pattern then runs on the Bluetooth pins, producing PRBS (Pseudo-Random Binary Sequence) packets in the selected Tx mode. In Rx mode, the device is configured for reception using JTAG register writes. The Rx ADC data is captured into internal RAM and later processed to determine parameters such as RXPER, RSSI, SNR, and GAIN. This approach allows for accurate measurement and analysis of Bluetooth signal characteristics in both Tx and Rx modes, ensuring the reliable performance of Bluetooth devices in various applications.

Figure 1 below shows the flow of the pattern-based method for Bluetooth testing on ATE.

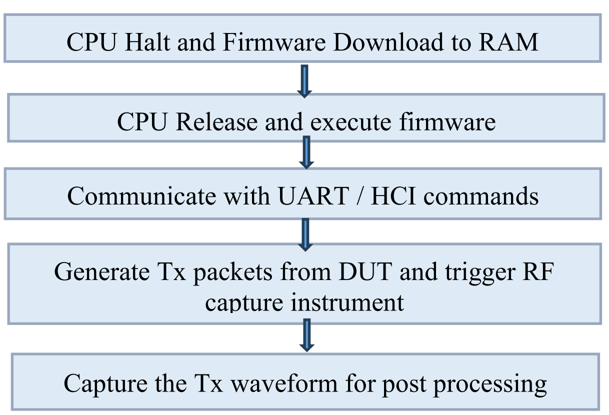


In a firmware-based approach, on-chip CPUs are used. The main steps of firmware download, like any other functional code, include stopping the CPU and loading the application firmware into RAM. After the firmware is loaded, the CPU's program counter is set to the firmware's starting address. The CPU is then released, allowing it to execute the firmware. Once released, the firmware performs all calibrations immediately, and calibration codes are updated in registers. Then, SRAM registers such as the boot status register and boot check register are read to confirm that the firmware executed successfully. These registers and specific bits indicate that RF initialization and calibrations are complete, and that the firmware has reached a stable state for receiving HCI commands or using RF registers for testing. Therefore, polling is necessary to wait for BT to be ready before reading these registers.

This process is specific to the chip design and communication protocol chip supports; it's essential to define the protocols and CPU usage before silicon, so that the ATE hardware and software can be designed accordingly.

To communicate with the device, HCI commands are defined using the UART (Universal Asynchronous Receiver-Transmitter) protocol and sent from the ATE to the device. The device responds to the HCI commands, enabling configuration of the device in the required Tx or Rx mode. Once the device starts executing the downloaded firmware, the HCI reset command is sent to the device, and a confirmation response is received from it. The device is now ready to be configured in different modes using standard HCI commands. This eliminates the need to write specific registers manually.

Figure 2 below shows the flow of the firmware-based approach for Bluetooth testing on ATE.



The firmware-based approach offers several benefits, including faster calibration execution, reduced test time, and improved overall efficiency in Bluetooth ATE testing. By leveraging the power of the device's firmware burned into ROM, engineers can optimize the testing process and ensure the reliable performance of Bluetooth devices in various applications.

III. ATE captured waveforms in firmware-based and register-based infrastructure.

In a pattern-based approach, the Bluetooth Transmission output and the ATE RF Receiver instrument are synchronized, as the ATE instrument is triggered to capture simultaneously when the device begins transmitting. This capture, performed by the ATE output, is synchronized with the device output and is called Synchronous capture. Demodulation of this Synchronous captured waveform is straightforward using the Agilent VSA software method of the ATE, without the need to extract the frame because the start of the capture coincides with the beginning of the frame. The whole frame is extracted from the capture waveform along with buffer samples after the end of the frame, using the required capture length.

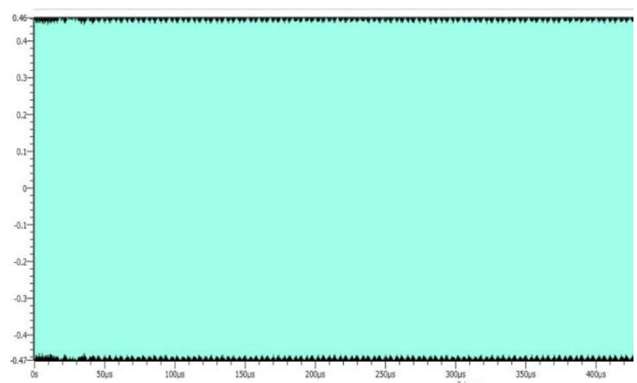


Fig. 3 shows the time domain plot of a single frame of the Synchronous captured Tx mode. The X-axis represents time in milliseconds, and the Y-axis shows amplitude in volts.

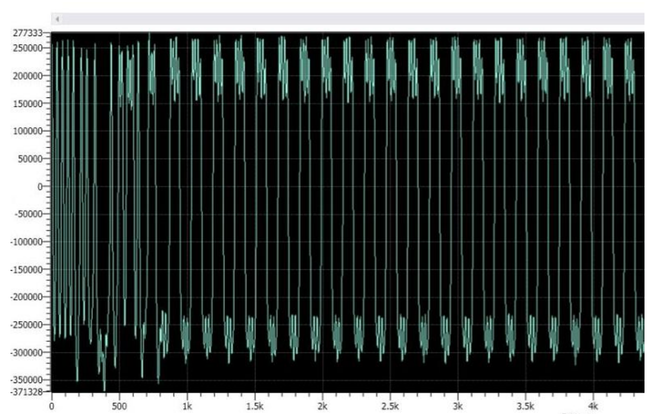


Fig. 4 shows the GFSK demodulated waveform from Fig. 3. The X-axis represents the sample number, and the Y-axis shows the frequency deviation in Hz.

In the firmware-based approach, after sending the HCI command to configure the device to Bluetooth Tx mode, the ATE captures the device output for the number of samples specified in the code once the device starts transmitting. This capture occurs asynchronously, without any synchronization with the device output. Demodulating the asynchronous capture requires extracting a complete frame, which includes the Bluetooth preamble, header, payload data, and CRC (Cyclic Redundancy Check). Extracting the frame from the asynchronous data involves identifying the start of the frame. The start is detected by calculating the moving average of 200 samples. When the moving average exceeds the threshold value—ranging from a very low value, indicating no data, to a higher value—the frame's start is identified.

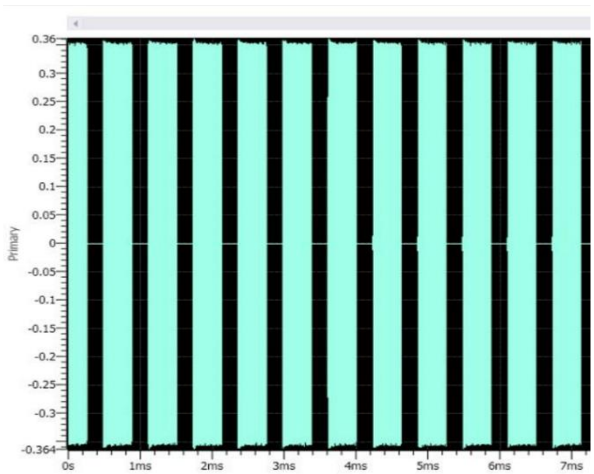


Fig. 5 shows the time domain plot of the asynchronous captured Tx mode. The X-axis represents time in milliseconds, and the Y-axis shows amplitude in volts.

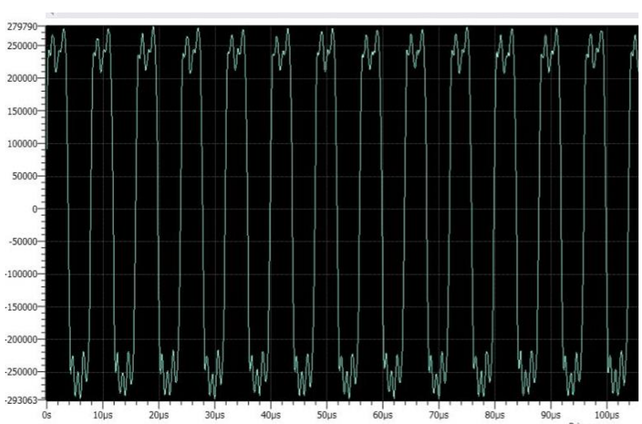


Fig. 6 displays the GFSK demodulated waveform after extracting the frame from the waveform in Fig. 5. The X-axis indicates the sample number, and the Y-axis shows the frequency deviation in Hz.

IV. TRADE-OFFS

A pattern-based approach requires writing device configuration settings for each Bluetooth mode and power, which the Design or Validation team must provide. This needs ongoing discussion and tracking to ensure the device settings are current. As a result, the Test Engineering team spends consistent time verifying if device configurations are updated, affecting the cycle time for test program deployment.

A firmware-based approach requires downloading the latest firmware revision from the repository whenever it is updated by the firmware team. The Test Engineering team doesn't need to check with the Design or Validation teams to review the device configuration, as the firmware will be updated with the changes. This helps reduce the cycle time of the test program setup by about three weeks and streamlines the process without much review or repeated discussions with the design or validation teams.

A pattern-based approach requires coding for all calibrations of Bluetooth devices, such as LOIQ Cal, correcting the amplitude and phase of the I and Q components of the local oscillator to improve DEVM results, TSSI Cal, power amplifier setting correction to ensure consistent power across devices, Bluetooth PMU Cal, Bluetooth Power Management Unit calibration, and Temp Sense Cal, temperature sense calibration. These calibrations will increase additional test time and extend the test program coding and debugging cycle. In contrast, a firmware-based approach doesn't require these calibrations to be implemented in test code, as the firmware runs these algorithms on power-up and stores the calibration values. The test engineering team only needs to read these calibration values, which helps save significant test time, as shown in the Table below, and reduces the cycle time for program deployment. ATE can also use a trimmed version of customer firmware to minimize test time. These requirements should be communicated to the firmware team for better planning.

Test	**Test Time Per Dut (sec)**		
	Pattern Based	**Firmware Based**	***Savings (%)***
BT Setup	1.4	1.1	21
BT Tx Tests	7.5	2.6	65
Total	8.9	3.7	58

The pattern-based approach achieves synchronous capture by ATE, so the frame extraction step is not required. The firmware-based approach achieves asynchronous capture, which involves frame extraction during post-processing. This does not affect test time because it is handled in the background by the tester.

The firmware-based approach uses UART HCI commands for each Bluetooth test parameter, which are generic commands applicable to all next-generation devices. This significantly reduces the test program setup time by 4-6 weeks compared to the pattern-based approach, where device configuration settings are specific to each device and need to be tuned and re-established by the Test engineering team. The firmware-based approach also simplifies debugging Bluetooth tests, as calibrations, device configurations, and modes are all embedded into the firmware.

The yield failures in the pattern-based approach are seen as 10% higher compared to the firmware-based approach for the same test coverage, which was root-caused as a register settings-related failure. It is very challenging to find the correct settings that fit across PVT (Process Voltage and Temperature) corners. To overcome process variations and improve yield, firmware-based testing is preferred. Yield failures in the pattern-based approach are found to be false failures, as those rejects pass on the bench setup. This may be due to calibration routines for TSSI and LOIQ, which are implemented on ATE and do not provide accurate results, or because register settings do not match the bench settings. The firmware-based approach overcomes these issues seen in the pattern-based approach and can recover those yields.

Yield Failures (%)		
Pattern Based	**Firmware Based**	**Delta**
11%	1%	10%

Through a thorough analysis of the strengths and weaknesses of each method, this work aims to guide the development of more effective and efficient Bluetooth testing strategies.

REFERENCES

[1] Schaub, K., and J. Kelly, Production Testing of RF and System-on-a-Chip Devices for Wireless Communications

[2] Bluetooth® Test Suite Spec, Revision: RF.TS. p33 edition 2.

[3] Bluetooth Core Specification, Revision: v5.3, Core Specification Working Group.