Validation of New Technologies: WiGig and iSCT (Intel® Smart Connect Technology); and study of framework for seamless connectivity

Major Project Report

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology

in

Electronics and Communication Engineering

(Communication Engineering)

By

Shah Niyati Kamalbhai

(12MECC27)

Electronics and Communication Engineering Branch
Electrical Engineering Department
Institute of Technology
Nirma University
Ahmedabad-382 481

May,2014

Validation of New Technologies: WiGig and iSCT (Intel® Smart Connect Technology); and study of framework for seamless connectivity

Major Project Report

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology

in

Electronics and Communication Engineering

(Communication Engineering)

By

Shah Niyati Kamalbhai
(12MECC27)

Under the guidance of:

Mrs. Vandana Bhat
Intel Technology India Pvt. Ltd

Prof. Sachin Gajjar
Faculty, EC, Nirma University

Electronics and Communication Engineering Branch
Electrical Engineering Department
Institute of Technology
Nirma University
Ahmedabad-382 481

May,2014

Declaration

This is to certify that

i) The thesis comprises my original work towards the degree of Master of Technology in Communication Engineering at Nirma University and has not been submitted elsewhere for a degree.

ii) Due acknowledgement has been made in the text to all other material used.

- Shah Niyati K
12MECC27

Certificate

This is to certify that the Major Project entitled **"Validation of New Technologies: WiGig and iSCT (Intel® Smart Connect Technology); and study of Framework for Seamless Connectivity"** submitted by **Shah Niyati K (12MECC27)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in Communication Engineering of Nirma University, Ahmedabad is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this Major Project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Date: Place: Ahmedabad

Internal Guide

(Prof. Sachin Gajjar)
Faculty, EC

Program Coordinator

(Dr. D.K.Kothari)
Professor, EC

HOD, EE

(Dr. P.N. Tekwani)
Professor, EE

Director

(Dr. K.Kotecha)
Director, IT, NU

This to certify that Ms. Shah Niyati Kamalbhai (12MECC27), a student of M.Tech EC (Communication), Institute of Technology, Nirma University was working in this organization since 12/6/2013 and carried out her thesis work titled Validation of New Technologies: WiGig and iSCT (Intel Smart Connect Technology); and study of framework for seamless connectivity. She was working in PCCG System Test and Validation under supervision of Ms. Vandana Bhat (mentor) and Ms. Uma Maheswari (manager). She has successfully completed the assigned work and is allowed to submit her dissertation report. We wish her all the success in future.

Acknowledgements

It gives me immense pleasure to present this report on my project **"Validation of new technologies: WiGig and iSCT(Intel® Smart Connect Technology); and study of framework for seamless connectivity"**.

I would like to begin by thanking **Dr. K.Kotecha**, Director of Institute of Technology, Nirma University, **Dr. P.N.Tekwani**, Head of Electrical Engineering Department, and **Dr. D. K. Kothari**, Coordinator of M.Tech Communication Engineering program for allowing me to undertake this thesis work and get industry exposure as well. Next, I would like to thank **Mr. Mukesh Kothari**, Managerial Head of PSTV (PCCG System Test and Validation) and my manager **Mrs. Uma Maheswari** for giving me an opportunity to work on this project, and for constantly reviewing my work and helping me improve.

I am indebted to **Mrs. Vandana Bhat**, my external guide and **Prof. Sachin Gajjar**, my institute guide, for their valuable guidance that proved to be the key to success in overcoming challenges encountered during the course work. I thank my team members and seniors of Nirma University working at Intel Technology India Pvt. Ltd. for their direct or indirect help. Working at Intel, India had been an enriching experience for me. I am extremely overjoyed to express my gratitude towards my friends for their delightful company and for keeping my spirits high throughout.

Last but not the least; I thank my family and the almighty GOD for bestowing their blessings on me.

Shah Niyati K

12MECC27

Abstract

Validation is much beyond testing. It includes "enabling", "standardizing", "integrating" and "delivering". A feature may work successfully when stand-alone, but when it is integrated at system level, co-existence, compatibility and interoperability with other features is a challenge. The scope of work is to validate WiGig and Intel® Smart Connect Technology (iSCT) on an upcoming Intel platform. High speed wireless data transfer and faster wireless connectivity is the need of the hour. The existing band-2.4GHz and 5 GHz are heavily congested and this calls for exploring other bands for wireless communication. WiGig technology, based on the IEEE 802.11ad standard, uses 60 GHz band, supporting data rate upto 7 Gbps, and can co-exist with WiFi and Bluetooth radios. The tri-band solution is emerging as one of the promising wireless technology, improving user experience. Intel Smart Connect Technology is designed to update programs by periodically waking your PC from Sleep/Standby mode for a brief period of time, improving content availability latency from several minutes to few seconds and providing Always Updated user experience. Also, it allows user to remotely access the system even when it is sleeping or is in stand-by mode providing Always reachable user experience even in low power mode. Objective is to enable all the features of WiGig and iSCT on upcoming Intel platform and validate it successfully for timely release. Devices available today typically have several ways or mechanisms of connecting to a remote device (over Bluetooth, Wi-Fi, etc). There is however no synergy or consistency between all these different technologies in terms of how they work, how they connect, etc. If all these wireless pairing experiences are streamlined, end user can have a feel of seamless connectivity. A novel framework- CCF(Common Connectivity framework) by Intel, described in this report, opens up new possibilities for orchestrating wirelessly connected experiences.

Project may be seen as consisting of two parts: Part 1 focuses on enabling WiGig

and iSCT on upcoming Intel platform and validating it at a platform level so that it can be given out to external customers (OEM/ODMs) and bug escapes can be avoided. Early system integration and platform validation helps OEM/ODMs to hit TTM (Time to Market) with higher quality and reduce post launch support costs. Work mainly includes creation of test plan for validation, execution of test cases and debugging of issues faced, to ensure that quality release criteria is met.Part 2 focuses on understanding Intel CCF and deriving its validation scope at platform level.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Full Form
ACPI	Advanced Configuration and Power Interface
AOAC	Always On Always Connected
BIOS	Basic Input/ Output System
BKC	Best Known Configuration
BKM	Best Known Method
CCF	Common Connectivity Framework
CRB	Customer Reference Board
CU	Connection Utility
DP	Display Port
HDCP	High-bandwidth Digital Content Protection
HDMI	High-Definition Multimedia Interface
HID	Human Interface Device
HW	Hardware
IA	Intel Architecture
IF	Intermediate Frequency
In-Proc	In Process
iSCT	Intel® Smart Connect Technology
KPI	Key Performance Indicator
LAN	Local Area Network
LOS	Line Of Sight

LTE	Long Term Evolution
MAC	Media Access Control
NAT	Network Address Translation
NB	Notebook
ND	Net Detect
NIC	Network Interface Controller
ODM	Original Design Manufacturer
OEM	Original Equipment Manufacturer
OFDM	Orthogonal Frequency Division Multiplexing
OS	Operating System
OSPM	OS directed Power Management
OTA	Over The Air
Out-Of-Proc	Out Of Process
PACS	Power Automation and Control System
PAL	Protocol Adaptation Layer
PCCG	PC Client Group
PCH	Platform Controller Hub
PCIe	Peripheral Component Interconnect Express
PHY	Physical Layer
PMF	Platform Mega Feature
POR	Plan Of Record
QRC	Quality Release Criteria
RF	Radio Frequency
RVP	Reference Validation Platform
RW	Remote Wake
SC	Single Carrier
SDK	Software Developer Kit

SIP	Standard Instrumented Platform
SUT	System Under Test
SW	Software
VoIP	Voice over Internet Protocol
WBE	WiGig Bus Extension
WDE	WiGig Display Extension
WiDi	Wireless Display
WiGig	Wireless Gigabit
WLAN	Wireless LAN
WSE	WiGig Serial Extension

Chapter 1

Introduction

1.1 Platform

Platform[1] encompasses all required ingredients, features, capabilities, initiatives and technologies. It is more than a sum of ingredients.The 5 major ingredients in a platform:

- **Hardware**: such as processors, chipsets, communications, memory, boards and systems.
- **Software**: such as Operating Systems(OSs), applications, firmware and compilers.
- **Technologies**: such as Intel® Active Management Technology (AMT), Intel® Smart Connect Technology (iSCT), etc.
- **Standards and Initiatives**: such as WiFi, WiMax, WiGig and so on.
- **Services**: such as digital media distribution, system management services, etc.

1.2 Validation

Literally, validation means testing or establishing a valid proof; or testing the truth of something. When it comes to platform, validation means testing the platform as a whole and supplying it to the ODM/OEMs with an assurance that the HW-SW stack is reliable and stable enough to be used by them.

1.3 Platform Validation-process flow and challenges

Post-silicon validation is the last step in the development of a semiconductor integrated circuit. Dependency between HW/SW ingredients introduces more complexity than ever to achieve overall system quality assurance. Platform Validation[1] is much beyond testing. It includes "Enabling", "Standardizing" , "Integrating" and "Delivering". It is a series of systematic testing activities/processes that aims to ensure/test the Quality, Compatibility and Reliability for platforms.

The work reported in this thesis majorly consists of studying new technologies, and validating them at platform level. Moreover, it includes enabling the implementation of a new technology on an actual platform. Figure below describes the process flow and the challenges involved in validation.

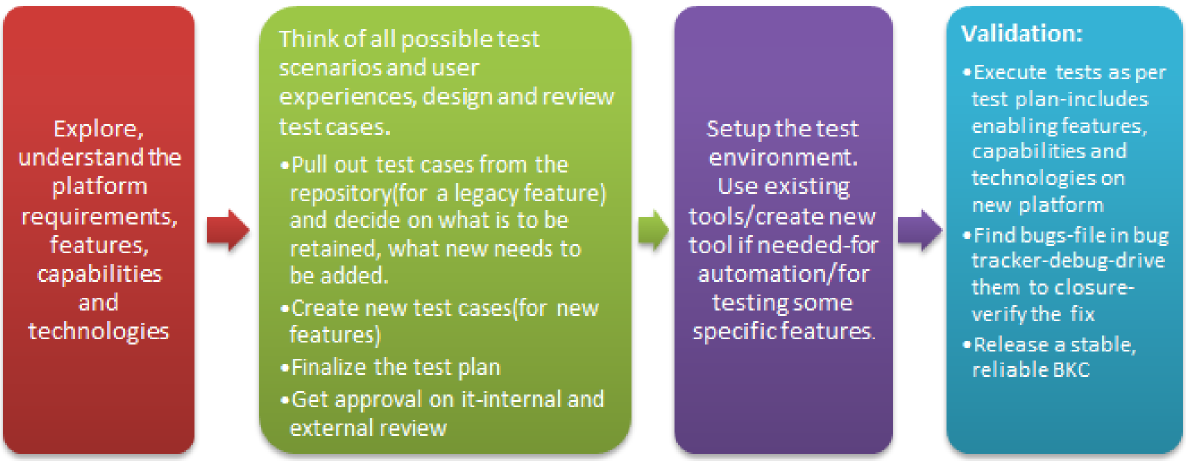

Figure 1.1: Validation Process Flow

1.4 Next generation Intel Platform

The internal code name of upcoming Intel platform is **Broadwell**[4]. It is expected to be released in late 2014. Broadwell is an upcoming micro-architecture- the next

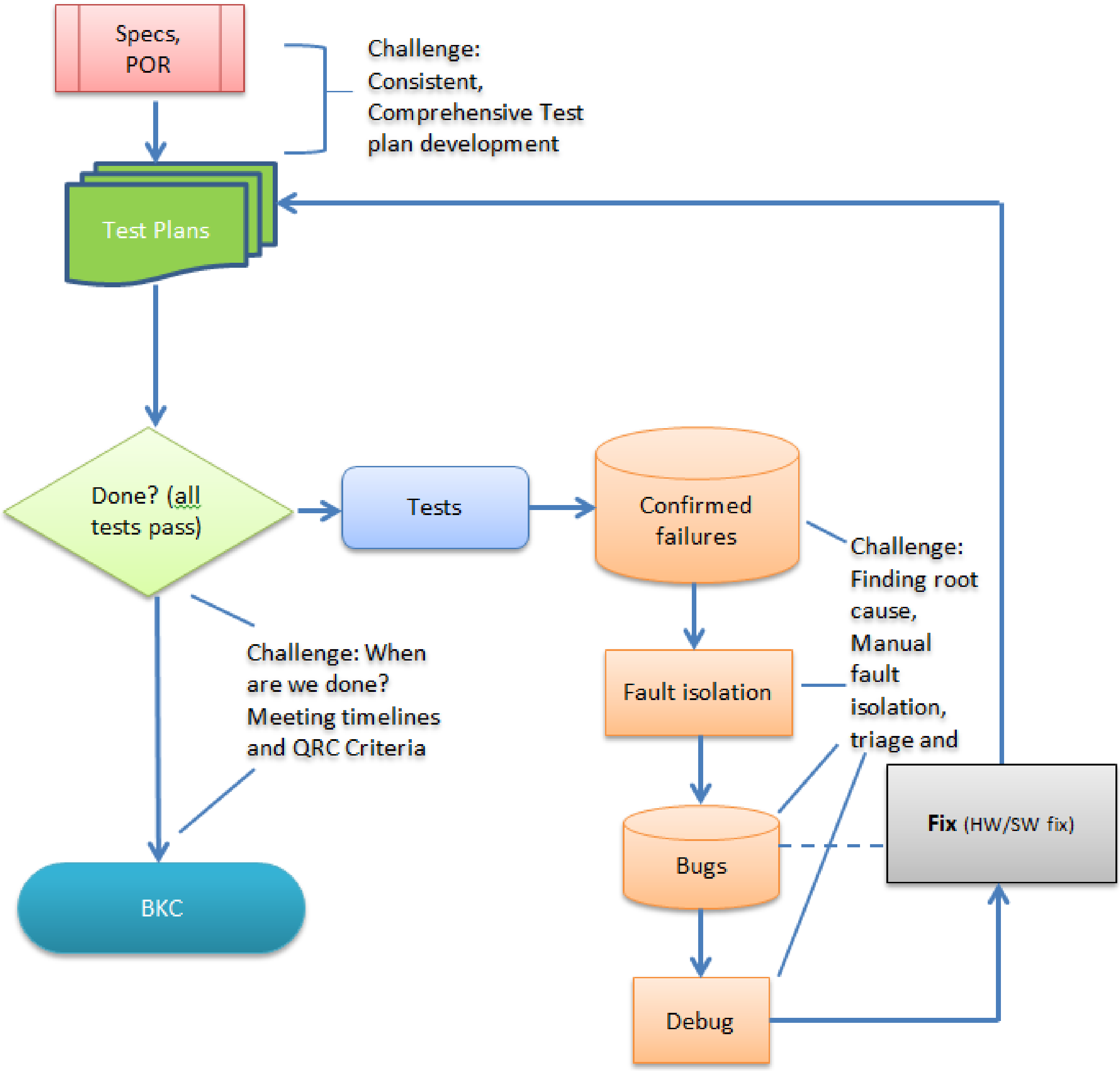

Figure 1.2: Validation-Challenges involved

step in semiconductor fabrication, with feature size reduced to 14 nm. It is expected to power fanless systems with performance significantly above that an Atom processor can deliver. It will come with features like significant power reduction without compromising on performance, twin integrated cameras that would make 3D gesture recognition an integral part of the hardware platform, support new technologies like WiGig and many more enhancements.

Testing is done on Desktop, Laptop, Notebook, Ultrabook, Portable All-in-One mother boards. The boards are RVP (Reference validation Platform-platform SoC) or CRB (Customer Reference Board- board designed for distribution to customers as a vehicle for demonstrating a chip's performance or suitability for a specific application. It is constructed using the same techniques that a customer is apt to use). Project work is done on majorly on Broadwell RVP and CRB and Haswell (Predecessor of Broadwell) RVP and CRB. Figure 1.3 [5] shows how RVP/CRB looks like. Note that the figure given below is of an older platform and is meant for giving an idea of how RVP/CRB looks like. Actual boards of new platform are quite different from this.

Figure 1.3: Example of RVP/CRB Board-Intel® Core™ i5-3610ME processor (PGA) and mobile Intel® QM77 Express Chipset development kit

1.5 WiGig

IEEE 802.11ad is an enhancement to the 802.11 standard that enables multi-gigabit wireless communications in the 60 GHz band[7]. The WiGig MAC and PHY Specification was contributed to the IEEE 802.11ad standardization process, and was confirmed in May 2010 as the basis for the 802.11ad draft standard. 802.11ad was approved by IEEE in late 2012. In 2013, the WiGig Alliance, the organization which developed the WiGig MAC and PHY Specification, unified with Wi-Fi Alliance, consolidating all technology and certification development within Wi-Fi Alliance to deliver closely-harmonized connectivity and application-layer solutions. Now, WiGig and IEEE 802.11ad are often used interchangeably. The WiGig specification allows devices to communicate without wires at multi-gigabit speeds (upto 7 Gbps). It is a new platform mega feature and Intel wishes to incorporate it in the upcoming platforms. The main usage of WiGig is wireless docking. Coexistence of WiFi, Bluetooth and WiGig will free the user from the burden of wires and annoyances of networking. Exploring this new feature, enabling and validating it on an upcoming Intel platform is one of the objectives of this project.

1.6 iSCT

Intel® Smart Connect technology (iSCT) was previously called Always On Always Connected (AOAC). It is a feature in which the platform and its network interface cards provide an Always Updated (Fresh Data) and Always Reachable; anytime access (Remote Wake) usage while providing longer battery life. Smart Connect[16] is a feature that periodically wakes the system from the Windows sleep state to refresh email or social networking applications. When the system is equipped with specific wireless devices, it can detect the presence of known networks while asleep, waking only when connectivity is available (this feature is called Net Detect). It eliminates the latency involved in establishing network connection and updating the applications (eg. email). Moreover, the periodic wake interval to perform the updates is user configurable (15/30/60 min). The wake interval may be set as per the use case. In night mode, the wake interval may be set to a longer duration to save power. This improves battery life.

Another feature of iSCT is Remote Wake[15] that provides Always Reachable user experience. Existing applications allow accessing the system remotely but they require the system to be fully on. With iSCT, system can be woken up from sleep/standup and accessed remotely. This saves significant power. Exploring this technology, enabling and validating each of its features on an upcoming Intel platform is one of the objectives of this project. Also, measuring system power under various scenarios with iSCT enabled and iSCT disabled system, to ensure that the said user experiences("Always updated" with less power consumption) are delivered is a part of validation work carried out.

1.7 Framework for Seamless Connectivity

Seamless or easy connectivity between various devices is becoming increasingly important. **Intel® Common Connectivity Framework (CCF)**[25] empowers this. It abstracts the underlying mechanism used for communication and streamlines the process. CCF SDK is designed to enable developers to create application solutions that seamlessly communicate with other CCF enabled devices around them. CCF applications are form factor and platform agnostic (Tablet/Laptop/Mobile; Windows/Android/iOS). Moreover, CCF3.0 comes with cloud support that enables peer to peer transfer across the world behind different firewalls. When users are in proximity to each other CCF aspires to make it as easy for them to interact with each other as it is to converse face to face. It is a promising solution for seamless connectivity. Studying the framework, understanding the underlying technologies and exploring the SDK is one of the objectives of this project.

1.8 Motivation

It is said that *"Do the work you love; or love the work you do!"*
It is always good to get work matching to the interests and skillset. Moreover, working on new technologies is in personal as well as organizational benefits. Being a student of Communication Engineering, the project is coupled to wireless communication. Low power consumption, low latency, better responsiveness, high speed data transfer, faster connectivity, seamless communication between the devices, etc are highly desirable features, everyone looks for. They are infact, need of the hour in today's world. The technologies worked upon, as a part of this project, provide these features in one or the other way.

Today, the 2.4 and 5 GHz wireless bands are heavily congested. They also lack the capacity to deliver high data rates required for emerging business and consumer applications. IEEE 802.11ad standard-WiGig addresses this problem.

Reducing platform power consumption and dissipation has always been one of the key challenges. Also, in this fast world, where things change with the wink of an eye, users wish to have fresh data available to them instantly as soon as they power on their system. Another desirable feature is accessing the system remotely even when it is in sleep state. iSCT can be thought of addressing these demands without increasing significant power consumption. It is inspired from today's mobile phones, where the phones are sleeping when left idle, still are always on and always connected to the network. Mobiles are in generally off state, but whenever a message/notification comes, it immediately pops up and alerts the user about the new data available. They provide instant on and always updated user experiences. It is always good to leverage good things. So the end user experience may be seen as leveraged, but the way of implementation is different. Remote Wake feature of iSCT allows user to access their media, files and resources of their system (which may be sleeping) remotely from anywhere on the Internet. It is fascinating to learn the underlying technologies that make such user experiences possible.

Seamless connectivity and cloud computing-are the new breakthroughs in wireless and computing world. Intel has its own novel framework for seamless communication, called as Common Connectivity Framework (CCF). Cloud support is also integrated in this. CCF SDK handles data transport regardless of whether that data is to be moved using the cloud, Bluetooth, or a local subnet socket. And if one of the devices moves such that the selected transport no longer works reliably, then CCF handles the transfer to other underlying transport that is provisioned for the connection. Study of this framework in itself provides pronounced learning experience.

1.9 Scope of work

The scope of work is best described by Validation process flow described earlier(in Section 1.3). Work done includes all the steps of validation process, for iSCT and WiGig. Apart from that, study of framework for seamless connectivity-Intel® CCF and finding the validation scope at platform level was done.

1.10 Objectives

Intel® Smart Connect technology and WiGig:

- Understand the technologies and their implementation at platform level, thoroughly.
- Enable them(make them work successfully) on all the product lines(laptop, notebook, ultrabook, etc) of upcoming Intel platform(Broadwell).
- Develop test plan for validation ensuring sufficient coverage so that maximum bugs can be found at an early stage only, file impactful sightings and drive

them to closure so that there are no bug escapes during release and not only QRC is met but also post launch support costs are minimal.

- Write test contents for the test plan.
- Develop debug skills that helps in quick issue resolution.
- Do competitive analysis, think of all possible use cases and ensure to be a deliverer of end to end User experience rather than simply being an enabler of new features and technologies.
- Engage with customers-get their test plan and do gap analysis, provide recommendations, provide them Intel test plan and address their opens.
- Do customer escape analysis by looking into bugs found by customers (if any), post release and take corrective actions to avoid such escapes in future.

Intel® CCF:

- Understand the framework and its underlying technologies.
- Explore CCF SDK.
- Find out the scope of CCF Validation and SDK inclusion in BKC.

1.11 Project flow with timeline

June-July: Ramp-up- understanding platform validation process; acquiring knowledge of all the ingredients that make up a platform; setting and powering up RVP board and hands-on execution; define area of work.

Aug-Sept: Ramp up on iSCT and WiGig- read the software architecture specifications, product design guide, market requirement documents, understand the technical specifications; Prepare iSCT and WiGig test plan for various product lines of

Broadwell and get it reviewed with platform architect, software architect and technology experts with relevant domain knowledge-incorporate their feedback in the test plan.

Oct-Nov: Write test content/test case procedures with expected results and upload it on web server. Work on enabling all the features of iSCT on various product lines of Broadwell platform-debug and resolve all the issues faced-file sightings wherever required and drive them to closure. Also start competitive analysis/ study of WiGig enabled products available in market or planned for launch-find information on WiGig related roadmap of various companies by browsing the details over internet; Study framework for seamless connectivity-CCF, understand the concept of cloud and protocols involved in CCF.

Dec-Jan: Document all the relevant information based on study of WiGig enabled products available in market or planned for launch and provide recommendations if any; Explore CCF SDK and try to find out the scope of integrating the SDK and validating it at platform level. Enable WiGig triband solution and perform sanity checks-debug and resolve all the issues faced-file sightings wherever required and drive them to closure. Simultaneously, finish one time execution(dry-run) of all iSCT test cases("possible to execute test cases") and based on hands-on experience, modify the procedure and expected result wherever required so that all the test cases can be deployed for regular execution(to be executed as per the priority and frequency of execution defined). Debug the issues faced during this and resolve them, file sightings if required.

Feb-March: Explore iSCT and WiGig specific debug tools and document BKMs to use them for debug and triage. Understand the technique to measure system power using SIP Boards; define various scenarios to be considered for measuring system power that can validate the user experience guaranteed by iSCT and execute power measurement accordingly-debug and resolve the issues faced; document the final results. Simultaneously, look into iSCT and WiGig related issues/ bugs reported by execution team-reproduce the issue, do necessary debug and triage to root cause it, report it to relevant team by sighting it in appropriate database and drive it to closure; Validate the fix and close the sighting.

April: Validate the interfaces(PALs) enabled in Intel WiGig solution- debug and resolve all the issues faced-file sightings wherever required and drive them to closure. Actual Dock solution was available only by this time. Hence, setting up the test environment with actual dock, executing test cases that were applicable and documenting the BKMs was done in this month. Simultaneously, debugging, filling relevant sightings for iSCT and WiGig related issues/ bugs reported by execution team were taken care of.

1.12 Thesis Organization

The rest of the thesis is organized as:

Chapter 2, Literature survey, gives overview of and system power states that helps in better understanding of iSCT. Also, it describes WiGig architecture, various PALs(Protocol Adaptation Layers) and Usage models.

Chapter 3 is on WiGig. It discusses about the currently available WiGig based solution in the market, roadmap of various companies in this field-derived as a part of competitive analysis study and Intel's triband solution to survive in the race. Also. it gives a glimpse of the efforts put to enable WiGig on upcoming Intel platform and the type of tests that are done as a part of platform validation.

Chapter 4 is on iSCT. It gives an overview of iSCT and its features, associated power states and the process flow. It tells about the type of tests and primary debug that should be done for fault isolation. Technique of measuring system power and results of power consumption under various scenarios, that validate the advantage brought by iSCT interms of power saving are shown.

Chapter 5 introduces Intel® Common Connectivity Framework and its salient features.It gives an idea on CCF architecture. Also, examples of currently available applications that use CCF are given.

Chapter 6 presents Conclusion, salient contribution and Future Scope of the project.

Chapter 2

Literature survey

2.1 Overview of Platform Validation

Validation is an important phase in the design cycle of a product. It is the process of verifying the functionality or working of a system based on design specifications. Ingredients of various sub domains are taken up, integrated for functionality, compatibility and interoperability and validated at platform level. Sub domains may include:

i. BIOS
ii. Operating System
iii. Graphics
iv. PCH (Platform Controller Hub) and EC(Embedded Controller)
v. ME (Manageability Engine)
vi. Storage
vii. Power (ACPI) and Performance
viii. Responsiveness
ix. Communications (WiFi, LAN, Bluetooth, etc)
x. Touch and Sensors
xi. Perceptual Computing

xii. System Memory
xiii. Thermal Management
xiv. Security
xv. System WHQL(Windows Hardware Quality Labs) Testing and Certification[23]

As a part of Platform Validation, tests focus on: system stress; power cycling; platform features and capabilities and their interactions; all possible user experiences; and testing for Integration, Compatibility and Interoperability. It is not an ingredient focus testing. As test cases focus on platform, many new bugs are found even when all the ingredients are healthy. Example of bugs can be: WiGig driver installation fails if WiFi/Bluetooth drivers are installed; System hard hangs after 50-60 S3 cycles; display freeze observed while playing 3D game using WiDi display, etc.

Key for defining platform validation is to understand all dependencies(includes both hardware and software dependencies), all possible test scenarios, defining them, writing suitable test cases, executing them and validating them as a complete system. On finding bugs, it is required to do some level of debugging on the issue and try to root cause it by doing proper isolation. After this, the issue needs to be reported to the appropriate ingredient team for providing the fix. The fix needs to be tested and verified before closing the bug. The final result is a **BKC-Best Known Configuration**[2]. BKC is a report of components needed to construct a proven stable platform.

BKC Release Process is carried out on a weekly/biweekly/milestone basis for internal as well as external customers. Internal customers means groups within Intel; external customers mean OEMs/ODMs like Dell, Lenovo, Toshiba, Asus, Apple, etc. Every week, the stable ingredient drops are picked and built together. Then, using this stack, a series of test cases of different priorities (depending on the status of ingredient in platform lifecycle), as per test plan, are executed to establish the latest stable stack. The released report may contain grading of the health of the ingredient components, a dashboard indicating pass/fail/blocked percentage of test cases and the details of critical bugs filed. Milestone BKC goes to external customers. It includes completely validated ingredient releases and provides a stable stack.

Thus, platform validation ensures that the final product meets the overall quality and reliability requirements. BKC Process Flow[2] is shown in the figure 2.1.

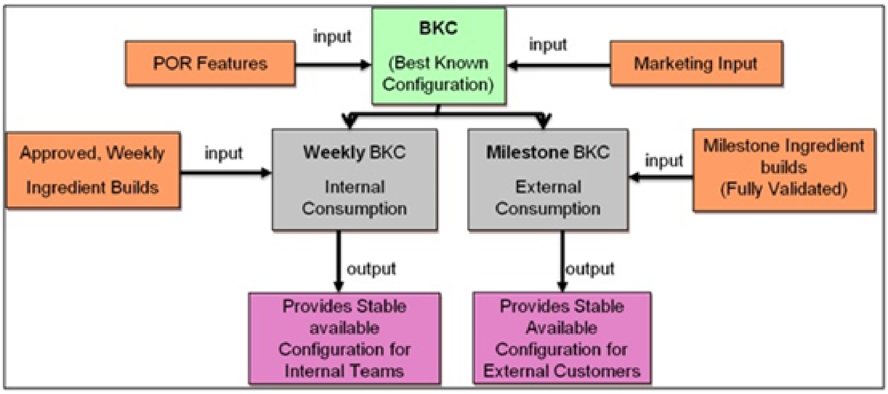

Figure 2.1: BKC Process flow

Most of the bugs need to be fixed by the milestone release timeline. For the initial milestones, wherein potential bugs exist and have fix pending, need to be documented and communicated clearly to the customers. Possibly, some tentative workaround for the bug may also be provided. Bug escapes should be ideally zero. However, some bug escapes are always reported. After every milestone release, post mortems are done to analyse what went well, what did not work out, where were the gaps, where is the scope for improvement, what can be done better, etc. This is a part of process improvement. Also, customer escape analysis is done, reason behind the escape is found and corrective steps are taken like adding some extra coverage if the bug escape reason is identified to be coverage gap. Such measures help in constant quality improvement.

2.2 System Power States

To understand iSCT, it is desirable to understand system states like S0 and S3. Overview on ACPI and system power states given below helps here.

In computing, the Advanced Configuration and Power Interface (ACPI) specification[3] provides an open standard for device configuration and power management by the operating system. ACPI 5.0 is the latest version. It describes the structures and mechanisms necessary to design operating system-directed power management and make advanced configuration architectures possible. ACPI applies to all classes of computers. ACPI Specification is needed to design and build an ACPI-compatible system. ACPI brings power management under the control of the operating system (OSPM). Once an OSPM-compatible operating system activates ACPI, it takes over and has exclusive control of all aspects of power management and device configuration. The OSPM implementation must expose an ACPI-compatible environment to device drivers, which exposes certain system, device and processor states.

The overall power consumption of the system is referred to as System Power States. Trade-off exists between the wake-up latency and power consumption. Lower the power consumption, higher will be the wake-up latency. There are a total of six different power states ranging from S0 (the system is completely powered ON and fully operational) to S5 (the system is completely powered OFF) and the States (S1, S2, S3 and S4) are referred to as sleeping states, in which the system appears OFF because of low power consumption and retains enough of the hardware context to return to the working state without a system reboot. The different Power States are:

Table I: Global States

Global States	**System States**	**Operations**
G0	S0	”System on” is the power-on state for the normal working state of the computer, which means that the Operating System (OS) and whatever other applications in use are running. In this state, the power consumption is at its highest. Sub states are: S0i1, S0i2 and S0i3.
G1(Sleeping)	S1 (Power on Suspend)	System turns off some unnecessary peripheral devices like the monitor or USB or the hard disk to save power. Hardware and processor context, and power to the CPU(s) and RAM is maintained. Hence, there is only a limited amount of power saving.
	S2 (Cache is Flushed)	Processor loses power. Processor context and contents of the cache are lost.
	S3 (Suspend to RAM; Sleep)	Processor and Hardware context, cache contents, and chipset context are lost. System memory is retained. All the applications and the opened documents are stored in the main memory; so, the main memory needs to consume power to maintain these stored data. Sub-states-S3 hot and S3 cold.
	S4 (Suspend to Disk; Hibernate)	System appears to be off. Least power is consumed compared to all other sleep states. The system is almost at an OFF state, expect for a trickle power. It saves the contents of memory to a hibernation file, preserving the state of the operating system, applications, and open documents.

G2 (Soft Off)	S5 (System off)	System is shutdown, and power consumption is at its minimum. Only the devices required to power-on the system consume power like power button, keyboard, etc. A full reboot is required. No previous content is retained.
G3 (Mechanical Off)	-	The state is same as G2/S5 only thing is the power supply chord is removed. Hence it is completely shut off and consumes no power.

2.3 WiGig Architecture and Application Specification (PALs)

High speed wireless data transfer is the need of the hour. The existing band-2.4 GHz and 5 GHz are heavily congested and this calls for exploring other bands for wireless communication. WiGig technology, based on the IEEE 802.11ad standard, uses 60 GHz band, supporting data rate upto 7 Gbps, and can co-exist with WiFi and Bluetooth radios. The tri-band solution is emerging as one of the promising wireless technology, improving user experience.

The WiGig specification defines Physical (PHY) and Medium Access Control (MAC) layers and is based on IEEE 802.11. This enables native support for IP networking over 60 GHz. It also makes it simpler and less expensive to produce devices that can communicate over both WiGig and existing Wi-Fi using tri-band radios (2.4 GHz, 5 GHz and 60 GHz). Figure 2.2 shows WiGig architecture with respect to lower layers[6].

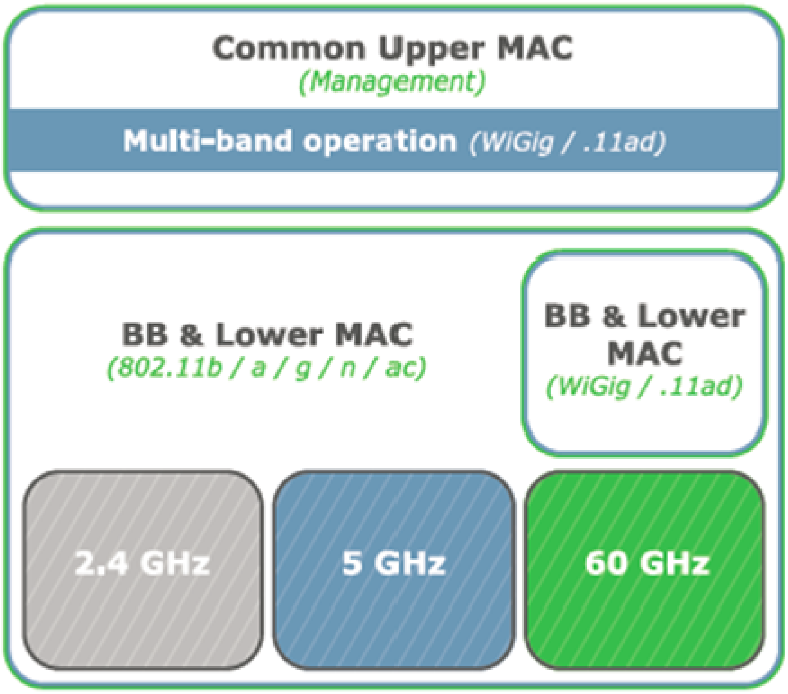

Figure 2.2: Tri-band communication supported by WiGig architecture

2.3.1 Physical layer (PHY)

60 GHz band is unlicensed and widely available for use in most of the countries. Within the 60 GHz band, there is variation in the spectrum available in different countries, as shown in Figure 2.3[8]. In general, the 60 GHz band has much more spectrum available than the 2.4 GHz and 5 GHz bands typically 7 GHz of spectrum, compared with 83.5 MHz in the 2.4 GHz band. Entire band is divided into multiple channels that are much wider, almost 50 times wider than the channels available in 802.11n. They enable 60 GHz devices to support applications that require extremely fast communication with low power consumption, such as uncompressed video transmission and wireless docking.[7]

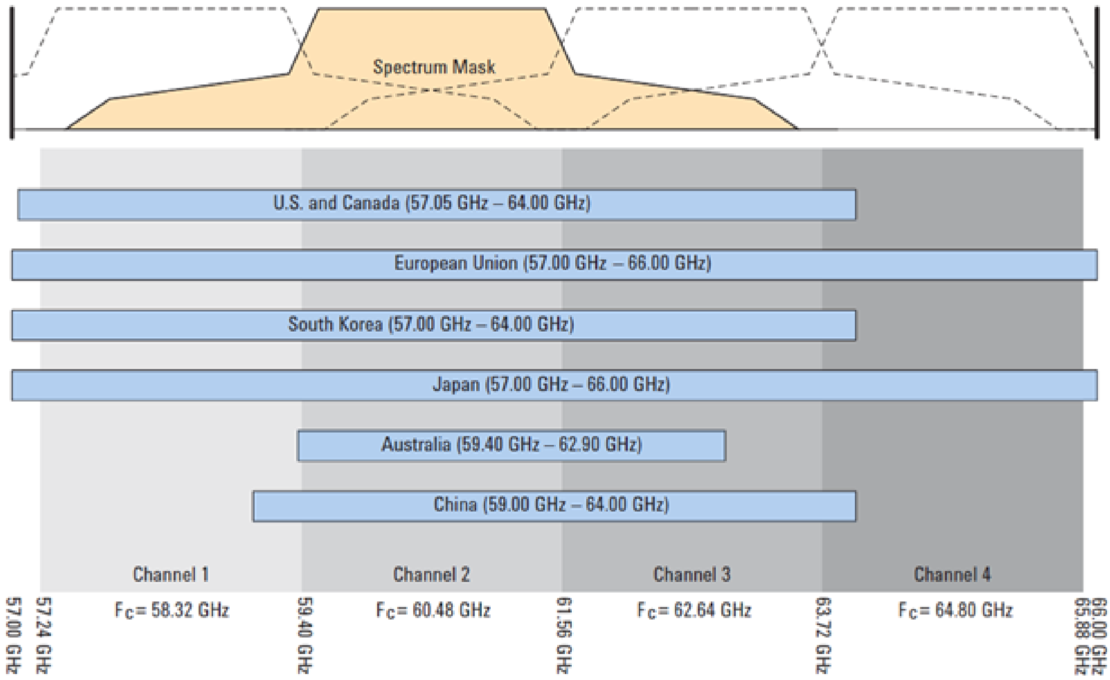

Figure 2.3: 60 GHz Band Channel Plan and Frequency Allocations by Region

Modulation Coding Scheme (MCS)[7]

The specification supports two types of modulation and coding schemes, which provide different benefits:

- Orthogonal Frequency Division Multiplexing (OFDM)
- Single Carrier (SC)

Orthogonal frequency-division multiplexing (OFDM):

One of the main forms of modulation used for WiGig is OFDM. This is a key element of the overall modulation and RF signal format, providing the capability for high data rates while supplying good resilience against multiple paths. It supports communication over longer distances with greater delay spreads, providing more flexibility in handling obstacles and reflected signals. Furthermore, OFDM allows the greatest transmission speeds of up to 7 Gbps.

Single Carrier (SC):

Single Carrier typically results in lower power consumption, so it is often a better fit for small, low- power handheld devices. BPSK, QPSK or 16-QAM on a suppressed carrier located on the channel centre frequency is used. SC supports transmission speeds up to 4.6 Gbps. The two types of schemes share common elements such as preamble and channel coding. This reduces implementation complexity for manufacturers of WiGig devices.

Benefits of using 60 GHz band [9]

The 60 GHz band is an excellent choice for high-speed Internet, data, and voice communications offering the following key benefits:

- Unlicensed band in most of the countries saves significant time and money in obtaining a license from a telecoms regulator.
- Highly secure operation resulting from short transmission distances due to oxygen absorption and narrow antenna beam width.
- Virtually interference-free operation resulting from short transmission distances due to oxygen absorption, narrow antenna beam width, and limited use of 60 GHz spectrum.
- High level of frequency re-use enabled communication needs of multiple customers within a small geographic region can be satisfied.
- Fibre optic data transmission speeds possible wirelessly 7 GHz of continuous bandwidth available compared to ¡0.3 GHz at the other unlicensed bands.
- Mature techniques; now benefiting from advances in component design
- Carrier-class communication links enabled 60 GHz links can be engineered to deliver five nines of availability if desired.

Range and directionality depends on the antenna used. Directivity of antenna plays an important role here. Using adaptive beam forming techniques, highly directional beam with longer range can be obtained.

Currently, 60 GHz is not a free band in India. Hence, testing cannot be done over the air. Testing is done using a coaxial cable that connects two systems-one of which emulates a dock and the other is configured as a notebook system(temporarily-till Intel procures license). This testing can help to serve as reference for validating the design. Vendors can use the validation results, customize their tests accordingly, avail the license and deliver a solution to the users.

2.3.2 MAC Layer

The MAC layer of the WiGig/802.11ad specification]textsuperscript[7] includes new features that support advanced usage models, facilitate integration with Wi-Fi networks, reduce power consumption, and provide strong security. It addresses aspects of channel access, synchronization, association, and authentication required for the 60 GHz operation.

Network architecture:

It is defined in a way that enables two devices to communicate directly with each other, allowing new uses such as rapidly synchronizing two devices and transmitting audio-visual data to a projector or TV at higher rates. It supports the existing 802.11 architecture as well.

Seamless multi-band operation:

MAC Layer is shared with current 802.11 standards. As a result, a communication session can be rapidly and seamlessly transferred between a 60 GHz channel and any lower-frequency Wi-Fi channel, including channels in the 2.4 GHz or 5 GHz band. This innovation enables seamless fallback to 2.4 GHz or 5 GHz Wi-Fi if 60 GHz connectivity is not available. Multi-band operation provides a greatly improved user experience.

Power management:

WiGig CERTIFIED devices takes advantage of a new scheduled access mode to reduce power consumption. Two devices communicating with each other via a directional link may schedule the periods during which they communicate; in between those periods, they can sleep to save power. This type of advanced time sharing capability allows devices to more precisely tailor their power management to their actual traffic workload, and it is especially important for cell phones and other handheld battery-powered devices.

Advanced security:

The IEEE 802.11ad specification builds on the strong security mechanisms defined in IEEE 802.11. WiGig CERTIFIED devices use Galois/ Counter Mode, a highly-efficient mode of encryption that is designed to support higher communication speeds. Encryption is based on the government-grade Advanced Encryption Standard (AES), and can be implemented in hardware for performance and efficiency.

Medium access control sub-layer deals with how to control access to the shared medium. Several Protocol Adaptation Layers (PALs) are developed to support specific system interfaces including data buses for PC peripherals and display interfac12es for HDTVs, monitors and projectors. PALs allow wireless implementations of these standard interfaces that run directly on the MAC and PHY, as shown in figure below. The initial PALs are audio-visual (A/V), which defines support for HDMI and DisplayPort, and input-output (I/O), which defines support for SD, USB and PCIe. PALs help in maximizing performance and reducing power consumption by enabling highly efficient implementations that can be built in hardware directly rather than layering on other protocols.

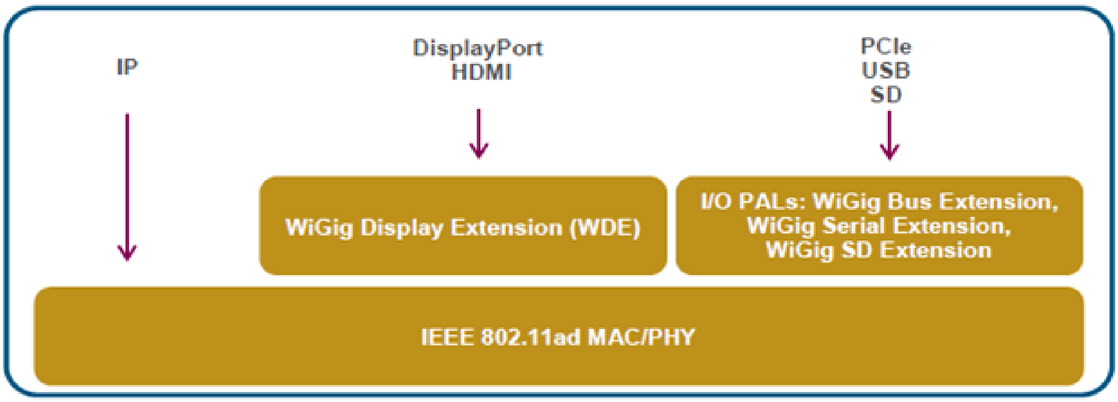

Figure 2.4: WiGig-Protocol Adaptation Layers

2.3.3 Protocol Adaptation Layers(PALs)

Currently defined **PALs**[7] are:

WiGig Display Extension (WDE): Audio/Visual-Remote HDMI, DP, DP++
WDE allows wireless transmission of audio-visual data, e.g. transmitting movies from a computer or digital camera to a TV set or projector. It supports HDCP (High-bandwidth Digital Content Protection) content transmission as well as transmission of compressed/uncompressed video. Low latency and visually lossless compression are the attractive features provided by WDE.

WiGig Bus Extension (WBE): Remote PCIe
PCIe is typically used within computers to connect the CPU and memory to I/O controllers that support storage, network cards and other interfaces. It is also used to connect to media and visual processors to enhance picture quality or offload processing from the CPU. Implementation of this PAL enables multi- gigabit wireless synchronization between devices and connection to storage and other high speed peripherals.

Wireless Serial Extension (WSE): Remote USB
USB is typically used to connect external peripherals and other devices to a host; the USB PAL enables multi-gigabit wireless connectivity between USB devices, and facilitates the development of products such as USB docking stations. Mainly, it is for native USB 3.0 support through which high throughput can be achieved.

WiGig SD Extension: Secure Digital IO
SD memory is widely adopted in mobile devices to store various files, such as documents, photos, and AV contents. The WiGig SD Extension is designed to directly access an SD memory equipped in a remote device from a host device, for instance a smart phone from a notebook PC. The WiGig SD Extension is suitable for resource-limited, battery-operated devices because of its simple implementation that is optimized to memory access and its ability to achieve multi-gigabit file transfer speeds with significant power savings.

2.4 Usage Models

The use of WiGig is pronounced to be beneficial in various ways. Various usage models defined are shown in the Figure 2.5[6].

2.4.1 Wireless Docking

One of the main uses of WiGig is Wireless Docking (WiDock). To understand the usage model for wireless docking, it is good to understand what docking means.
A docking station/dock provides a simplified way of "plugging-in" an electronic device such as a laptop to common peripherals. Such docks become useful while using laptops have lesser ports to connect peripherals than required. A dock allows laptop computers to become a substitute for a desktop computer, without sacrificing their mobile computing functionality. A HP Dock is shown in the fig below. Rear

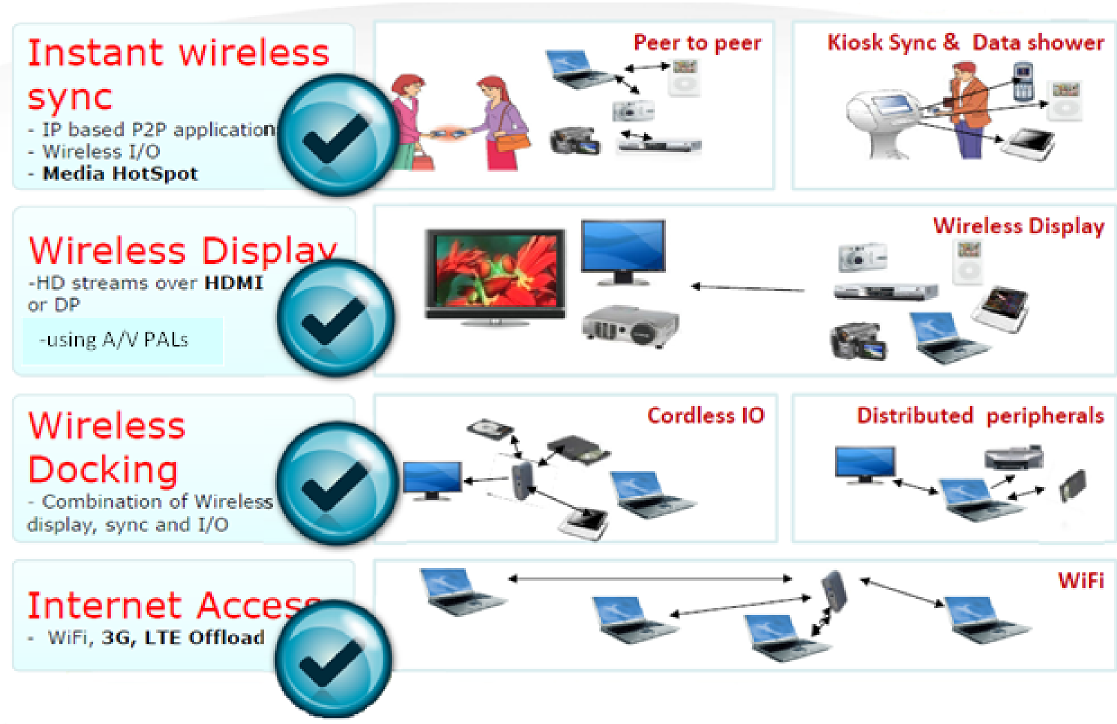

Figure 2.5: WiGig-Usage Models

view shows that multiple ports/connectors are available on the dock. When laptop is docked, all these ports/connectors become accessible for the laptop and appear as if connected to laptop directly. Dock conveniently charges the docked laptop if charger is connected to the dock.

Currently, all available docks are wired, meaning the system (laptop/notebook) needs to be physically connected to the dock to get docked to it(Fig. 2.6 and 2.7). With WiGig, docking can be made wirelessly, just by bringing the system and dock in proximity (LOS of each other), without having any physical connection between dock and system (laptop/notebook). More on WiGig is described in next chapter.

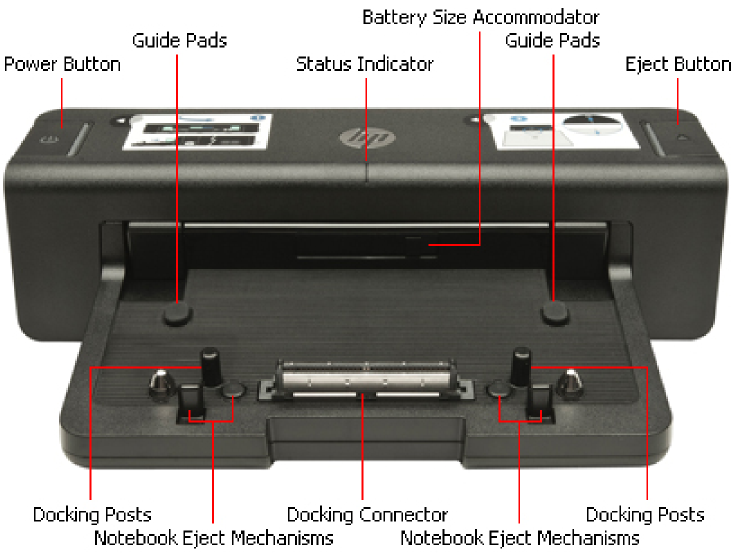

Figure 2.6: A typical HP Dock-Front view

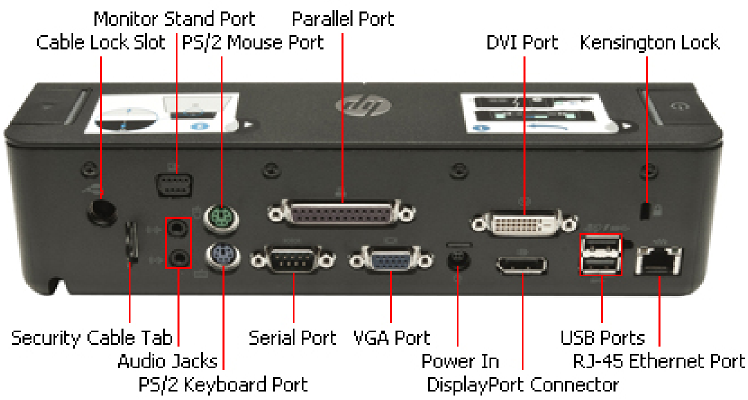

Figure 2.7: A typical HP Dock-Rear view

(HP 90W Docking Station)
(Image Courtesy:: http://www.misco.co.uk/product/Q567201/HP-90-Watt-Docking-Station)

Chapter 3

WiGig

3.1 Overview

WiGig is a new wireless technology that can make a difference in wireless networking. It is based on IEEE 802.11ad standard. The Wireless Gigabit (WiGig) allows multi-gigabit speed wireless communications operating over the 60 GHz frequency band. Data rate is related to bandwidth. The 60 GHz band has more spectrum(7-8 GHz) available than the 2.4 GHz and 5 GHz bands, allowing for wider channels(2 GHz) to support faster data rates of up to 7 Gbps using low power modulation scheme.[7]

As discussed in Chapter 2, main usage of WiGig is wireless docking. WiGig is different from WiFi and Intel® WiDi(Wireless Display)[13]. WiFi is more of networking while WiGig is more of cable replacement. WiDock differs from usage models, such as Miracast®[14] or Intel® Wireless Display[13], in which the user is located further away from the screen (such as on the couch or in a conference room), and would be focused on content consumption (watch a movie, share his screen with others, gaming) rather than productivity or content creation. WiGig uses a docking station on a desk connected to hard drives and computer screens that can connect to mobiles and tablets over the frequency. With all devices connecting wirelessly into one basestation, a user can sync all their devices together by placing them next to one another, rather than plugging in HDMI cables or an equivalent. WiGig would remove the need of connectors, resulting into further shrink in the size of ultrabooks.

WiGig is “directional”; in-room technology. The range is limited because:

- Oxygen readily absorbs 60GHz signals and neutralizes them within a short distance of their source.
- LOS required-cannot penetrate through walls.

That means WiGig will be limited to a maximum of roughly 30 feet (about 9.1 meters), and beyond 15 feet (4.6 meters) or so there will be significant decrease in performance. This is not a hard-limit and range may vary across manufacturers depending on several factors like antenna used, surrounding environment, etc. Limited range is in fact a blessing in disguise. Advantages of limited range:

- No interference-allows having multiple setups side-by side.
- Secured transmission

However, one fact that cannot be overlooked is that 60 GHz is not free band across the globe. It is a licensed band in India and therefore, currently, testing is done in IF mode, using co-axial cable to connect two systems, one of which is configured as notebook and the other emulates dock.

3.2 Beam Management

One of the features of WiGig microwave Wi-Fi is the aspect of antenna beam management. Height of an antenna is inversely proportional to the signal frequency. This means very small antennas are required for WiGig. Phased arrays can be a feasible proposition. The beam-forming is accomplished using a bi-directional training sequence that is appended to each transmission. This enables the system to shape the transmit and/or the receive beams to achieve the optimum link properties. This enables the system to overcome any movement of the transmitter, receiver, or objects between them that might alter the path characteristics[10]

.

The 60 GHz signal cannot penetrate walls similarly to the lower Wi-Fi bands even with adequate antenna gain. However the WiGig signal can propagate off reflections from walls, ceilings, floors and objects using beamforming built into the WiGig system. Beamforming employs directional antennas to reduce interfernce and have better directionality-focused and concentrated beam. When roaming away from the main room the protocol will switch to make use of the other lower bands at a much lower rate, but which propagate through walls[10]. This way, using Adaptive Beamforming, robust multi-gigabit communications at distances greater than 10 m can be achieved. Proper beamforming is absolutely essential requirement for WiGig based solutions.Support for beamforming is defined within the PHY and MAC layers of the WiGig specification[7]. During beamforming proces, transmitter and receiver fine tune their antenna settings to have better directional communication.

Manufactures use their own method for beam-forming. Smart antennas may be used for adaptive beam forming. Direction of Arrival (DoA) algorithm may be used to deploy the same. Multiple directional beams formed using antenna array can provide good directionality with better spatial coverage.

Intel Solution

- Single hardware with tri-band wireless solution-WiFi, Bluetooth and WiGig.
- Intel solution uses its own patented beam-forming technique. The roots of the concept lie in the adaptive phased array system.
- Currently, the main focus is on Wireless Docking usage that helps user to get rid of carrying multiple wires everywhere and provides “cable-free” experience.
- Limited PALs support is planned in the current HW solution(details being

confidential, cannot be disclosed). With that, the dock may support atleast 2 displays (DP/HDMI), multiple USB 2.0/3.0, audio, GbE (LAN) and optionally eSATA.

Figure 3.1: Wireless Docking

3.3 WiGig Integration and Validation

This section gives brief details about the work done as a part of WiGig-integration and validation. By integration, it means integrating WiGig as a component of a system.

- Understanding the technology-referring the white papers and resources available on web, understanding Intel's way of implementation.
- Enabling the technology/feature on various platforms and on various configurations (various Platform+OS+SPI Image combination; eg Broadwell Ultrabook platform, Windows8.1 OS and 5 MB Corporate SPI Image).
- Identifying scope of coverage at platform level and ensuring no duplication or overlaps with any other team's coverage(like ingredient team, testing for certification team, etc)

- Developing Test plan-identifying features and capabilities and their relations, developing test content for integrated features, configuration, compatibility and interoperability, optimizing the number of test cases to improve efficiency and at the same time ensuring enough coverage to avoid any bug escapes.
- Writing test contents as per the test plan (Have written 63 test cases).
- Thinking of possible use case scenarios and adding them in the coverage.
- Setting up proper test environment and executing various tests.
- Finding bugs, filling impactful sightings, doing primary isolation, debug, triage and finding the root cause of the sighted issue, working on tracking the sighting and driving it to closure, verifying the fix and closing the sighting when the issue is resolved.(Have filled 13 issues till now)
- Figuring out if the Test case can be automated and providing a logical flow chart for developing automation code. (2 P1 test cases are automated)
- Documenting BKMs(Best Known Method) for execution, primary isolation process, use of debug tool for log collection (Intel internal tools are available for debug purpose) and debugging, etc.- Prepared BKM for installing driver on notebook and SUT emulating the behavior of dock(in the initial time when actual dock was not available), BKM for WiGig driver installation on notebook side and FW upgrade on Dock side(actual dock), BKM to execute all basic acceptance tests in minimum time and BKM to use debug tools-for generating logs and traces, for generating a dump of events through event monitor tool to understand the point of failure.
- Competitive analysis-look for WiGig based products/plans/roadmap of various other companies and analyse the features of the proposed/available solution.

Apart from these, WiGig being new technology and key user experience provider, the test plan was shared with various OEMs/ODMs willing to start adopting Intel solution and integrate it in their product at an early stage. Their opens/queries regarding the test plan were answered (I was a part of meeting in which the opens were addressed). Their suggestions were incorporated in our test plan as well. As a result of this activity, 9 new test cases were added.

3.4 WiGig-setup and testing

WiGig testing requires atleast one dock and one notebook. Initially, RVP Dock solution being in development phase, was not available for testing. Hence SUT was used that emulated the behavior of an actual dock. Now, actual RVP Dock solution is available and the same is used for testing. This Dock has limited interfaces enabled. External devices like HDMI display, USB drive, etc can be connected to the dock.Block diagrams of test setup for both the scenarios-test with SUT acting as dock emulator and test with actual dock, are shown in the figure. Instead of RF testing, IF mode is used i.e Dock and Notebook are connected with a coaxial cable. The KPIs will vary with this mode of testing as actual KPIs are defined for over the air usage. However, focus is to test functionality, co-existence and user experience. Tests involving measuring KPIs are dropped from execution in India. In the setup where one SUT emulates Dock, both the SUTs(Notebook and Dock) must be preloaded with the latest OS and drivers as per the latest BKC stack. For the actual dock, it needs to be connected to a host notebook with PCIe extender/express card or similar device for Dock side FW upgrade and debugging-collecting logs and traces.

Example of tests: Test plan includes tests with priorities assigned as P1, P2 and P3. P1 means highest priority test-that needs to be run as soon as new driver is available or stack is changed. It includes functional and basic acceptance test. The new driver version may be rejected if P1 tests fail.The table below gives examples of some basic WiGig test cases included in the test plan.

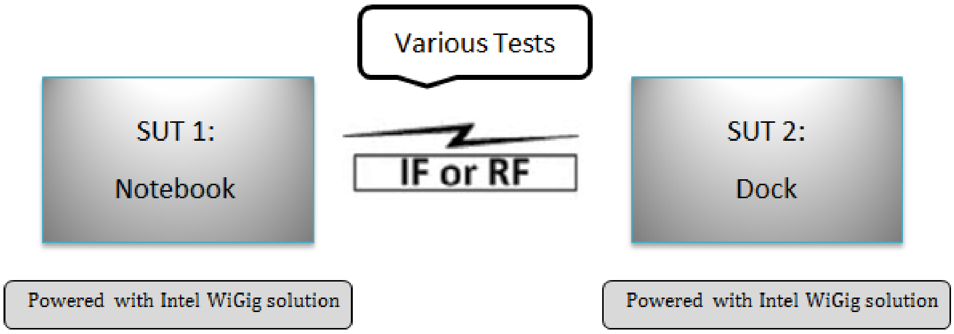

Figure 3.2: WiGig Initial test setup-SUT acting as dock emulator

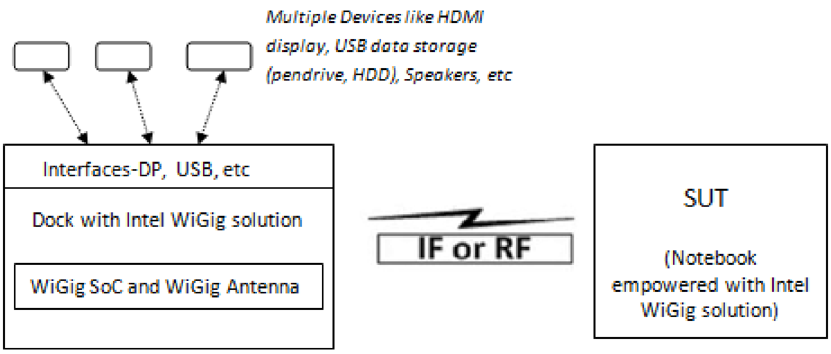

Figure 3.3: WiGig test setup-with actual dock solution

Table I: Example of WiGig tests

Test	**Description**
SW/Driver Install	Install the WiGig new software/driver on both the sides-dock and NB. Verify that installation is successful-Name of the driver should appear under "Programs and Features list found in Control Panel and driver should be enumerated in Windows Device Manager.

Basic Connection	Open WiGig Connection Utility (CU)/ UI on Dock and NB, press RF ON/ enable WiGig on both the sides. Verify that the dock name is present in the scan list at NB side. Connect to the dock and verify the Connected status appearing in the CU on both, Dock and NB.
	Verify that connection is established between Dock and NB by using ping command.
RF Kill	Disable WiGig from the CU (soft kill-RF off from CU) and enable it back to check if it is functioning properly. Similarly, introduce hard RF Kill by using on-board RF Kill switch and test.
Connect/ disconnect	Pair/unpair Dock and NB. Connect and disconnect the dock from the NB/NB from the Dock and verify that anything is not broken.
Basic UDP/TCP traffic test	While connected, configure static IP and perform bidirectional iPerf traffic (using iPerf tool and commands). Check bandwidth and transfer rate for TCP as well as UDP traffic. Verify for Dock and NB configured as data sender as well as receiver.
Auto-connect	Verify that NB gets connected to Dock automatically when Dock is in ON state and NB is coming in range of the dock. This needs that Dock and NB were paired previously and auto-connect option is enabled at NB side.
All dock ports functionality check	Verify basic functionality of all the ports available at Dock side. Dock the NB and check if the devices connected to the dock are getting detected at the NB side and working properly. This includes USB, DP, HDMI, etc.

WiFi/ Bluetooth/ WiGig co-running scenarios	With NB connected to the Dock, turn on WiFi and Bluetooth, browse through the web content, download media from the web, and initiate a search for Bluetooth devices available to connect to, pair with one of them and transfer a file over Bluetooth. All the radios should function properly simultaneously.

Apart from these, the test plan consists tests based on:

- Various interoperability,
- Compatibility and co-existence testing,
- Docking-undocking,
- Multiple(/single) dock-single(/multiple) NB combination scenarios*,
- System stress,
- Power Cycling tests scenario,
- Use cases-based on PALs implemented,
- Interference, range, power, performance metrics and other KPI verification tests*,
- Link Stability tests*,
- Normal user level functional tests with NB WiDocked e.g. web browsing, instant messaging, video streaming, video chat, printing, file transfer, playing games, etc
- User experience cases, etc.

*(*tests not executed in India)*

Preparing a test plan requires thorough understanding of the technology, its dependency and intersection with other technologies/features and ability to think from an end-user perspective. A feature matrix may be formed prior to preparing test plan. As a part of project work, these pre-requisites were met and a complete test plan was prepared from the scratch (since this being new feature, no legacy test plan existed).

Example of an issue found during testing that turned out to be a board design flaw

As WiGig is a new technology and it was to be tested on an upcoming platform for the first time, many issues were faced and challenges were involved in enabling it within the timeline.

With triband (WiFi, Bluetooth and WiGig) solution HW module of Intel, one of the critical issue seen was that system was not allowing WiGig driver installation. It was throwing error message Installation blocked. On checking device manager, it was found that WiFi, Bluetooth was getting enumerated in the list but WiGig was not. Same issue was seen on all the boards. On trying to reproduce the issue on other platform that had entirely different board architecture, it was found that issue was not getting reproduced. With this, it was concluded to be platform specific issue. Event viewer logs were generated, testing was done with debug environment(Intel internal tool that helps to generate dump of events and contains some register values that can help in debug), all the logs were analyzed. No significant information was found from these logs.

Nevertheless, this helped to suspect that card was not getting detected but then question was if that was the case, then WiFi and Bluetooth also should not be working. However, this logic was ruled out after discussion with WiGig team and understanding the high level architecture of the tri-band HW module. It was concluded that WiGig module was not getting identified. Board architecture was analyzed. With the help of board team, it was found that a voltage/current divider network was present near the power supply pin of tri-band HW. This voltage/current divider network was preventing sufficient voltage to reach the voltage supply pin (Vcc) of tri-band HW. As a result, the available power was not sufficient for WiGig module to power on. Resistor unstuffing (removing the resistor and creating an open circuit) was done to remove the voltage/current division. On doing this, WiGig was getting detected and driver also got installed. Thus, after exhaustive debug it was found to be board design flaw.

3.4.1 WiGig Debug

When any issue is encountered, or a test case fails or some unusual result is obtained, primary isolation needs to be done. It is required to debug the issue and try to root cause it before filling a bug.

Steps for primary isolation and debug:

1. Ensure that the setup is proper and prework needed for executing the test is followed. Confirm that the issue doesnt fall under Known Issues list captured in Release notes. This means, it is needed to relook at the Release notes and User Guide to be assured enough on the issue, that it exists.
2. Intel has its own testing framework for WiGig that contains tools to collect logs. Apart from NB SUT and Dock SUT/ actual dock, one SUT acts as Controller. All the SUTs need to be connected to the same network. With this, logs can be generated that can have description of all the events. Analysis of these logs may help in finding the point of failure. Event Monitor is also there that gives timeline of the events. Apart from logs, monitors and traces can also give useful information. It is good to attach logs while sighting an issue.
3. Confirm if issue is board specific or seen across all the boards. If it is found to be board specific, match the FAB, SKU version and reworks done on passing and failing board. If everything is found to be same, check HW-SW details on each and compare them. Something must be missing on the failing board. Such primary isolation helps to find that"something" and fix the issue.
4. If issue is seen on all the boards, verify if the issue is seen across all OS variants as well $\rightarrow$ helps to find if the issue is OS specific. Next, verify if issue is seen across more than one platform Eg. Check if the issue reported for Broadwell ULT is

seen on Broadwell ULX as well; if it is seen on both of Broadwell platforms, check if it is seen on Haswell systems also → helps to find if the issue is CPU/PCH architecture specific or if it is platform specific.

3.5 Current Market Scenario and Competitive analysis [1]

Everyday brings something new in the computing world. Staying ahead of competitors with the dizzying array of technologies and trends is what everyone wants. Intel is not an exception in this. When it comes to cutting through the hype of the latest innovation trend or the next great marketing technology, products that have the potential to create experiences that move people, survive in the race. Focus should be on creating compelling use case experiences rightly and providing value to the customers.

The Wireless Gigabit Alliance (WiGig) was a trade association that developed and promoted the adoption of multi-gigabit speed wireless communications technology operating over the unlicensed 60 GHz frequency band. The alliance was subsumed by the Wi-Fi Alliance in March of 2013. Many companies like AMD, Nokia, Qualcomm, Wilocity, Dell, CISCO, Broadcom, Samsung, Panasonic, Intel, etc comprise the board of directors of this association. Many of them have launched or showcased their WiGig based product. Presented below is a bird view of the WiGig solution based market scenario:

- Wilocity[11] seems to have gained leadership in providing WiGig based solutions by providing 1st WiGig supported chipset.
- Qualcomm and Wilocity announced at CES (Consumer Electronics Show), Jan 2013 the first Tri-band reference design that combines Wireless-N (802.11n),

[1]This analysis was carried out in Nov-Dec, 2013

802.11ac Wi-Fi, and 802.11ad (also known as WiGig) in one single networking product.

- Wilocity also announced at International CES that Dell has become the first system manufacturer to include the wireless chipset in its new Latitude 6430u Ultrabook. With that, Dell became first one to launch tri band WiFi products earlier this year with the industry leading Dell 6430u Ultrabook and D5000 Wireless Dock and it continues to build on that momentum with the new product offerings. Wilocity announced in Sep, 2013 that its WiGig solution will be integrated into several new Dell Premium and High Performance Ultrabooks and Workstations. Dells latest generation Latitude E7240 and E7440 Ultrabook, and the Precision M4800 and M6800 Workstations, feature the tri-band WiFi chipset from Wilocity that integrates the multi-gigabit performance of the 60 GHz band with 2.4 GHz and 5 GHz band whole home Wi-Fi. Wilocitys tri-band solution enables consumers to wirelessly connect from their laptop or workstation to compatible mouse, monitor, keyboard and conference-room equipment peripherals with the available Dell D5000 Wireless Dock for a wire-free desktop.

- Cisco Systems and Wilocity revealed in Nov, 2013 that they have started working together to develop 60 GHz enterprise networking products that could support massive transmission speeds of 5 Gbps.

- Wilocity is expected to announce additional 60 GHz enabled products soon, as the company continues to broaden its portfolio with a wide variety of computing platforms.

- Panasonic has developed a prototype system, in which WiGig is embedded in a tablet that can wirelessly transmit data like photos or videos to displays mounted in the passenger seats of a car. Panasonic demonstrated the use of its WiGig SD Memory card for short range high speed data transfers. However, the range of this solution is just 1-3m. Though range is limited, Panasonic has justified the usages as: transmission of HD video between a car's entertainment system unit tucked away into the dashboard to the screens behind the headrests for passengers in the back seats, without wire and few more usages like this.

- Broadcom's development roadmap is split into two separate paths: the WiGig development (802.11ad); and 802.11ah, an ultra-low-power version of Wi-Fi with immediate benefits for the so-called "Internet of things".
- In Sep, 2013, Wilocity and DisplayLink announced the industry's first demonstration of WiGig-enabled 4K graphics and video. The demonstration featured how Ultra HD video could be streamed from a WiGig enabled notebook and a WiGig docking station to a 4K resolution monitor the dock integrated the latest 4K capable chipset from DisplayLink connected to a 4K resolution monitor, which is more than four times higher resolution than 1080p full HD resolution.
- At its annual Intel Developers Forum in Sep, 2013, Intel demonstrated multi-gigabit wireless docking technology (WiGig) using an ultrabook.

The market trends show that everyone is in the race to deliver a better solution as early as possible. With this scenario, it becomes necessary to be a deliverer of end to end UX(User eXperience) rather than simply being an enabler of new feature. As a part of project work, impactful sightings are to be filled and sufficient coverage is to be assured to drive up the product quality and eliminate any bug escapes.

3.5.1 Dell Wireless Dock D5000

About D5000 Wireless Dock[12]:

Price: $270 (for stand-alone purchase)

Based on: IEEE 802.11 ad

Frequency: 60 GHz

Maximum data rate: 4.6 Gbps

Figure 3.4: Dell Wireless Dock D5000

Pros:

- The Dell Wireless Dock D5000 is the first laptop dock to use the promising WiGig technology, and it provides convenience and extra ports to Latitude 6430u (and other Dell ultrabooks that support this) owners.
- Requires absolutely no physical connection between the notebook and dock.
- Can drive two displays with resolution up to 1920 by 1200 pixels each.
- Three USB 3.0 ports, audio, and gigabit Ethernet.
- Supports HDMI.

Cons:

- Supports only WBE PAL.
- Dock is compatible with certain specific Dell Ultrabooks only.
- Limited Range: Supports beam forming for max 10m (in room).
- Can't daisy chain displays.
- Does not support HDCP.
- To maintain a strong connection, Ultrabook should be kept facing the dock.
- Data transfer rate reduces significantly as compared to wired connection. Eg. File transfer from a docked hard disk v/s from a hard disk connected to the

USB Port of ultrabook directly show significant differences in file transfer rate.

- No WDE support WBE is only used for DP also, this means, multiple format conversions - so more CPU utilization and power consumption.

Over the next 5- 6 years, Wireless Gigabit (WiGig 802.11ad) is expected to become highly pervasive across its ecosystem, thereby penetrating the market to a huge extent. The industry has started realizing the potential of the 802.11ad WiGig market, as there's much more to it than just another technological innovation.

3.6 Summary

This chapter gives an overview of WiGig technology and work done throughout the year as a part of Wigig validation at system/ platform level. To summarize the deliverables from my side, working as domian owner for WiGig were: Developed test plan and wrote 63 test cases, wrote 9 more test cases lately-based on OEM/ODM recommendation, addressed few of the opens/ queries of various OEMs/ ODMs on WiGig-system level validation, did market survey based on information available on google on available WiGig based products and provided a document on the conclusions derived, to the marketing, planning and ingredient teams, provided logical flow chart to automate basic driver acceptance test, documented various BKMs(total-4 BKM) to falicitate testing and debug, filled 13 sightings that were root caused as WiGig issue and found bugs during WiGig validation that were root caused to be board issue and platform collective bugs-meaning issue seen with specific BKC or combination of drivers only. Apart from these, I provided training session on enabling WiGig and testing it at a platform level for various product lines, to 2 Intel internal teams (one team was in India-so it was more of a demo, other team was in Malaysia, so it was more of virtual training session). Knowledge sharing session and demo on Wigig was provided by me to the team I worked in.

Chapter 4

iSCT

4.1 Overview

"The online world never sleeps. Your connection to it shouldn't either." [18]

iSCT is an acronym for Intel® Smart Connect Technology. It was previously called as "Always On, Always Connected (AOAC)". It can be thought of as a combination of three features:

- Periodic Wake (Always updated)
- Remote Wake (Always Reachable/Anytime Access)
- Net Detect (a variant of Periodic Wake)

Periodic Wake[16] automatically updates applications such as e-mail and social networks, periodically, when computer is asleep. With this technology, one does not need to wait for applications to update on waking up the system. It can be thought of as being inspired from mobile technology where display is off but whenever a message or email comes, it will flash that on the screen, notify the user and again go back to sleep. This feature offers "always updated" user experience with improved battery life. With this, fresh data is always available to the user and the latency involved in updating messages/emails or applications on resuming the system from sleep, is removed. With periodic wake, system wakes up from S3 to S0_iSCT state periodically. S0_iSCT is different from full S0. It may be defined as the mode of iSCT operation whereby the system is awake but in a low power state. In this state, it is generally true that audio, graphics and system fans are disabled, creating a mode of operation mimicking a cellphone-like standby/connected state.This results in reduced power consumption and saving of battery life. However, a trade-off exists between update frequency and battery life[19]. The more frequently system is made to enter S0_iSCT to process updates, higher will be the power consumption and lesser will be the battery life.

When iSCT is only timer based (periodic wake) and if there is no network connectivity, power is wasted in waking up the system periodically. This drawback is overcome by Net Detect[16]. It is a low power solution for detecting network connectivity when system is in sleep state. Net Detect is armed when no network is found during periodic wake. With Net Detect enabled, WiFi NIC passively scans for network and triggers the wake event only on availability of the network. This will save power.

A typical use case scenario of Net Detect can be: A user is working on his laptop at home and has enabled iSCT(Periodic wake interval set as 15 min). He then needs to leave for office. So he closes the laptop lid, puts it in bag and leaves. System enters S3. After periodic wake interval (15 min), system wakes but finds no network. Net Detect is armed. Suppose it takes an hour to reach office. So, system won't wake up during this period. Corporate network is available as soon as he reaches office. On network availability, wake event is triggered. So system wakes and performs all the updates immediately. Till the time user reaches his desk, the system is updated. All mails are updated and fresh data is available.

Remote Wake[15] feature offers "Always Reachable" and "Anytime Access" user experience. It is implemented as a cloud service and enables the user to access content or services of a system remotely from another system. It is different from the solutions available in the market that allow remote access to a system that is in S0 state (fully powered on). With this iSCT feature, the system is accessible remotely even when it is sleeping, i.e, system is in S3 state. This saves power significantly. While iSCT draws more power than uninterrupted sleep, the impact is minimal. As the battery runs down and reaches the minimum required battery level (called as battery threshold), updates will stop; i.e iSCT agent will stop running and turn off. Same will be the case when the system temperature rises and reaches certain predefined threshold. This is necessary, especially when laptop is in an unventilated bag or briefcase, for example.

iSCT State diagram and Process flow[16] is explained in the Figure 4.1 on next page.

Table I: iSCT-System states and their implication

State	**Implication**
S0	Fully powered ON. iSCT plays no role when SUT is in this state.
S3	Standard Sleep/Stand by mode. With periodic Wake enabled, SUT will be woken up from this state periodically.
S0_ iSCT	SUT enters this state when it is woken up periodically by iSCT Agent from S3. CPU is configured to run on low power. Audio-Video stay off to save power.
S0_ RW	Similar to S0. Difference is that CPU runs in full frequency and audio-Video may be turned ON.

Primary requirements for iSCT[17]:

i. The feature is available on Ultrabook, All-in-one, and standard PCs powered by the 4th generation Intel Core Processor family.

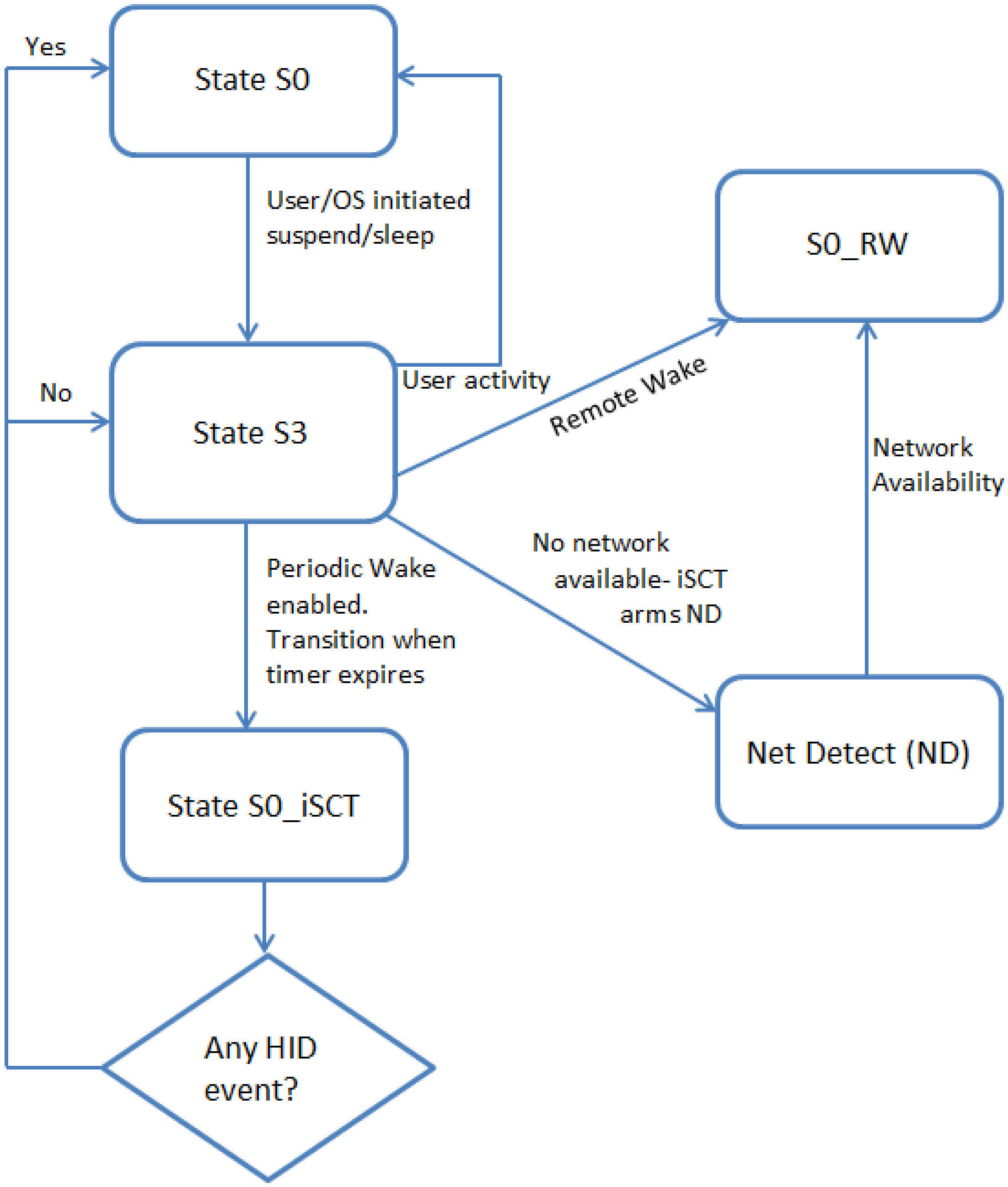

Figure 4.1: iSCT State diagram and process flow; Assumption: Neither battery threshold, nor thermal threshold is reached

ii. Intel® Smart Connect Technology requires features that must be built into the BIOS of the computer system. For this reason, the technology must be included in the computer system at the time of manufacture.

iii. If any computer system included Intel® Smart Connect Technology, but the feature seems to be unavailable, it is to be ensured that the capability is enabled in system BIOS. System manufacturer can help with the instructions on accessing the BIOS and changing the settings.

4.2 iSCT-Integration and Validation

This section gives details of the work done as a part of integrating iSCT at system level and validating it. The scope of work included the following:

- Understanding the technology and the way of implementation,
- Enabling the technology/feature on various platforms and on various configurations (various Platform+OS+SPI Image combination; eg Broadwell platform, Windows8.1 OS and 5 MB Corporate SPI Image),
- Developing Test plan-identifying features and capabilities and their relations, analyze legacy test plan to confirm on the test cases that can be leveraged(leveraged 41 test cases), developing test content for integrated features, configuration, compatibility and interoperability, optimizing the number of test cases to improve efficiency(removed 18 test cases as a part of optimization-ensured indirect coverage in system test plan or direct coverage in ingredient test plan) and at the same time ensuring enough coverage to avoid any bug escapes,
- Writing Test cases-have written 23 new test cases,
- Thinking of possible use case scenarios and adding them in the coverage,
- Working on scope of coverage at platform level,

- Setting up proper test environment and executing various functional, compatibility and interoperability tests,
- Finding bugs, filling impactful sightings, doing primary isolation, debug, triage and finding the root cause of the sighted issue, working on tracking the sighting and driving it to closure, verifying the fix and closing the sighting when the issue is resolved-filled 19 sightings-16 were root caused to iSCT, 2 were identified as Si bug and 1 was falling under platform collective category-seen with specific WiFi driver and iSCT driver combination only.
- Figuring out if the Test case can be automated and providing a logical flow chart for developing automation code-provided logical flow chart to automate basic periodic wake functionality test.
- Documenting BKMs(Best Known Method) for execution, primary isolation process, use of any new debug tool(Intel internal tools are available for debug purpose) and debugging, etc.- Prepared BKM to test periodic wake, remote wake and Net Detect features, BKM to modify registry settings to customize wake interval, on/off net detect feature and BKM to generate iSCT logs and analyze them to find cause of some common failures.
- Carrying out power measurement on end system to verify if the feature actually meets all the KPIs and delivers desired user experience(described in detail in later part of this chapter).
- Got test plan of various ODMs/OEMs and did a detailed study-"Gap Analysis" of their system level iSCT test plan. No significant coverage gap was found. However, comparison revealed few differences in approach of executing few tests, and some corner case testing scenarios. Based on the gaps found, certain scenarios were added in system level test plan and few scenarios were pushed to ingredient team for testing. The intention behind this exercise was to know from our customer perspective, what are the expectations, how far

are we closer to their way of implementation and testing, see if their is any scope of improvement in coverage, content or methodology, avoid any customer escapes, etc.

- Apart from these, customer escape analysis was done for the issues reported out till Beta phase(i.e after Beta release) and based on the escapes, corrective actions were taken as follow-up.

4.3 Testing iSCT and primary debug

Example of Test cases included in iSCT test plan:

1. Functional test cases of all three individual features-periodic wake, remote wake and net-detect with LAN and WLAN; on AC power and DC power.
2. Testing interoperability among each of them.
3. Testing compatibility of all these individual features with other platform features like Wake on LAN, iRST(Intel® Rapid Start Technology-feature where platform consumes power like in Hibernate(S4) but resumes like from Sleep(S3)-less power consumption with better responsiveness), wake from power states like DeepSx(Deep Sleep) and such other power management states and platform level scenarios.
4. Testing transitions from S0_iSCT to full S0 by HID events like key press, mouse-click, power button press, opening of lid.

iSCT testing needs a highly stable network connectivity. Remote Wake needs SUT to be connected to a proxy free network. Hence care should be taken that network connectivity is not lost anytime during testing.

Testing of Remote Wake needs special applications to be installed-one on SUT and other on the machine from which Wake signal is to be sent. Cloud services are used by Remote Wake applications.

When any test fails, first thing to be done is, collecting iSCT logs and analyzing them to figure out the reason for failure. Sometimes, just by looking at the logs, it can be found out that what went wrong, while at other times, exhaustive debug is needed for isolation and finding the root cause. Few primary steps of debug can be-verifying network connectivity, BIOS Settings, checking that the mode of operation is correctly chosen-daytime/nighttime, verifying correctness of system date and time, doing proper traige with (n-1) versions of BIOS, WiFi and iSCT driver versions one after the other for isolation and verifying if it is any regression issue, checking logs, modifying few registry settings and collecting logs using test logging tools[20], etc.

Example of a failure that turned to be PCH architectural design flaw:
Remote wake not occurring from DeepS3
Expected result was that system should wake up from Deep S3 on sending a wake signal. However, actual result was not so. Inspite of sending wake signals multiple times, SUT didn't wake from DeepS3. All other steps of this test case were passing except this final step. Debug done:

- Collect logs and try to analyze them.
- Check if RW is happening from normal S3.
- Change registry settings to enable higher level of logging and analyze the detailed logs generated after doing so.
- Execute same test on different systems (with the same HW-SW stack) to confirm that issue is not board specific.
- Execute same test with different configuration-make changes in the SW stack one by one and verify if it is regression issue.
  - Use n-1* iSCT driver version-if issue is not seen-it means new driver has some issue

- Use n-1* BIOS and check
- Use n-1* WiFi driver and check
- Use n-1* chipset driver and check

**n-1 refers to the version released just before the current version that is in use.*
Above tests will help to know if issue is caused by any of these: iSCT driver, BIOS, WiFi or chipset.

If none of them are causing the issue, try testing on other CPU/PCH.
None of the above steps helped. Issue was seen consistently. Hardware level debug was done. Some pins on the board were probed. It was found that the PCH was not sending any wake signal. Try was made to fix the issue using EC (Embedded Controller) software, but in vain. It was identified that PCH architecture itself was not supporting the feature. As a result, issue was closed with the reason “Will not fix” and the issue was identified as PCH architectural flaw. It was made sure that this was highlighted as a “DCN”-design change notification requirement for PCH.

4.4 Power Measurement

Low power consumption leading to lasting battery life is one of the most important and desirable feature. However, with multiple applications running, with increasing processor transistor count to provide maximum features within one device and increasing clock frequencies for providing higher computational speed, it becomes challenging to achieve low power budget. Everyone wishes for best user experience at little cost of power. iSCT is one of the feature that aims to meet this requirement.

iSCT is Intel technology that provides Always updated (Fresh data on the go) and Always Reachable (Remote wake) usages while providing longer battery life. The stringent power budget requirements may be met by iSCT as a standalone application but when it comes to platform level, wherein it is integrated with a fully loaded system(meaning a system having multiple add in cards connected and multiple software integrated), it becomes necessary to verify if the technology meets its low power targets. It is only after assuring that the desired performance is achieved at system level, the technology can be sold to customers.

The goal is to measure system power of an iSCT enabled system and validate the concept and usage of iSCT. This means it is to be validated that always updated and always reachable user experience is provided by iSCT at reduced system power. If system power is found to be exceeding the required KPI, next level of analysis can be done like measuring power consumption of key components, analyzing the result, finding the component that is contributing towards higher power consumption and identifying a solution. The results may be shared with respective ingredient teams (key components that are consuming higher power) so that they can work on resolving the issue. Various scenarios for which system power measurement was to be done to achieve the goal were identified. Also, iSCT comes with 2 features-periodic wake and net detect, that deliver similar end user experience. However, net detect is a form of smart periodic wake wherein, power is saved further during unavailability of any network connection. So, next I identified the scenarios for validating the advantage of having Net Detect feature.

4.4.1 Methodology

RVP Boards that are used for validation have multiple on-board debug hooks and probe points for fault isolation. As a result, if system power is measured on RVP Board, it will not meet the targeted KPI. SIP Boards dont have these. Hence, SIP Boards are used for power measurement. SIP means Standard Instrumented Platform. These are special boards different from RVP or CRB Boards, used specially for power measurement, analysis and experiments for power optimization. They may be considered as one of the implementation of design derived from reference platform (RVP board design). They provide a method to take consistent power measurements with reduced chances for wiring errors and damage to the board. To measure platform power, a power measurement layer having sense input resistors (milli ohm) for all the power rails is added in the board. These SIP resistors are tapped to measure the power of individual rails. NI DAQ is connected to this for data acquisition and post processing. The concept is shown in figure 4.2:

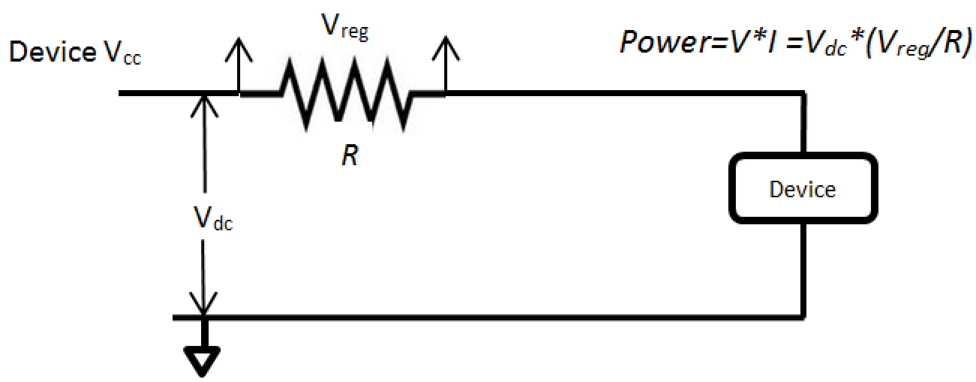

Figure 4.2: Power measuring using SIP resistors-concept

Total System Power is the summation of power consumed by individual components of the platform. One important point to be considered while measuring system power is ensuring that all debug equipment are disconnected from the board and no software debugging tool should be running on the system. This includes serial debugger (connecting this will impact PCH power), JTAG (Joint Test Action Group) debugger, ITP (In-Target Probe) and all Intel internal debug HW and SW tools.

Setup for power measurement and analysis

The setup for power measurement and analysis is shown in figure 4.3. SIP Board

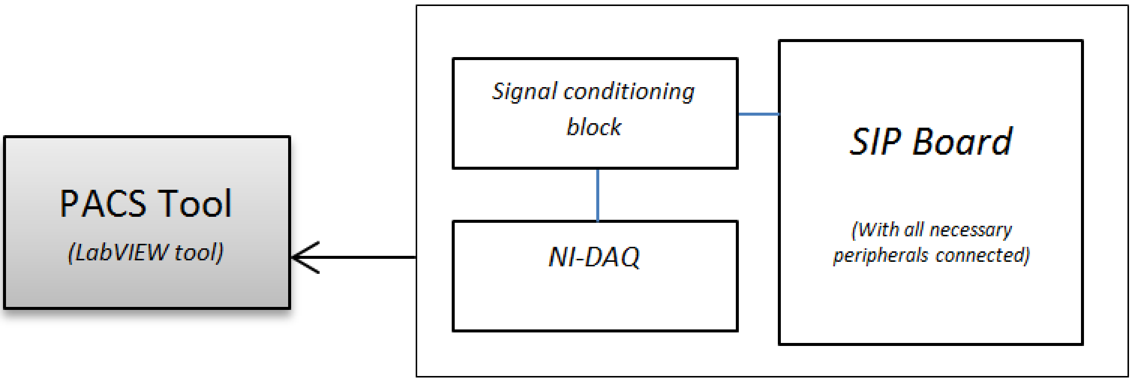

Figure 4.3: Setup for measuring system power- block diagram

is connected to a signal conditional board, which is connected to NI-DAQ. Proper signal conditioning is necessary-this can be achieved by loading proper calibration file. The data acquired by NI-DAQ is fed to PACS[21] - Power Automation and Control System-a software tool running on another computer. PACS is a LabVIEW based tool designed by Intel that makes collection and post processing of power data easier. It can be configured to run for specified duration. The result is a set of .csv files. Out of the many files generated, summary.csv file is used here for power data analysis.

4.4.2 Results

Scenarios:

System power was measured for the below scenarios for half hour(30 min) each:

i. System in sleep(S3)- iSCT disabled
ii. System in sleep(S3)- iSCT enabled-periodic wake interval configured as 10 min
iii. System in S0-display on but no user activity(display idle); web browser open in foreground
iv. System in S0-little user activity like mouse movement, opening paint and web

browser open in background.

The results are shown in the graphs below:

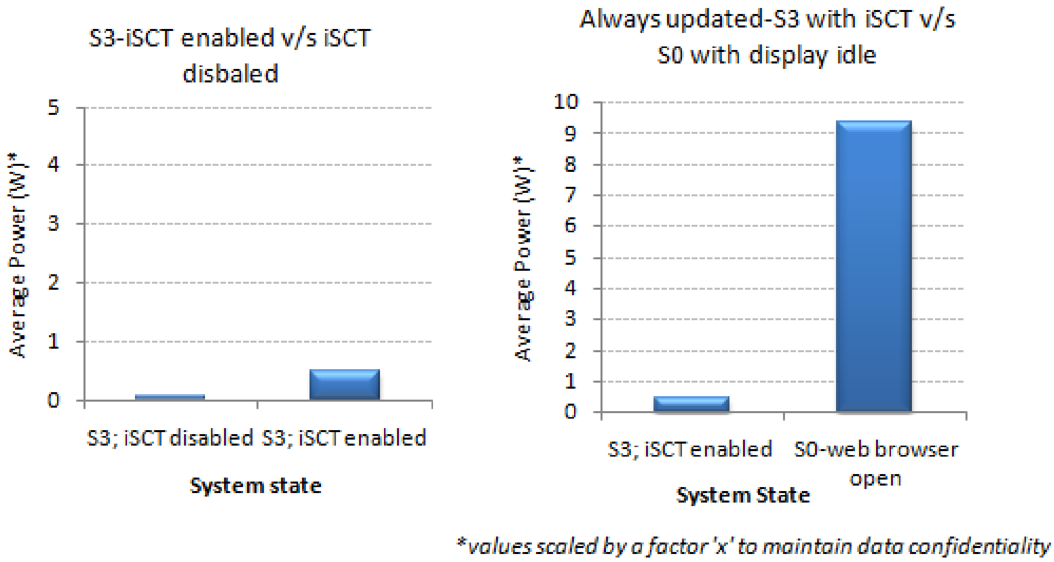

Figure 4.4: Comparison of power consumption under various scenarios

When system is put to sleep (S3), power consumed with iSCT enabled is slightly higher than power consumed with iSCT disabled (approx. 6x more). However, as shown in 2nd graph, iSCT is capable of providing Always Updated user experience with significant power saving, resulting in longer battery life. Lets say user is not doing any activity but wishes to have fresh data available all the time and hence he keeps system in S0 with display on and web browser or web application running in foreground so that it stays updated. The power consumed in such a case is almost 18x more than power consumed with iSCT enabled system in S3 that provides fresh data on resume.

The graph below shows the power consumed in all the four scenarios mentioned earlier.

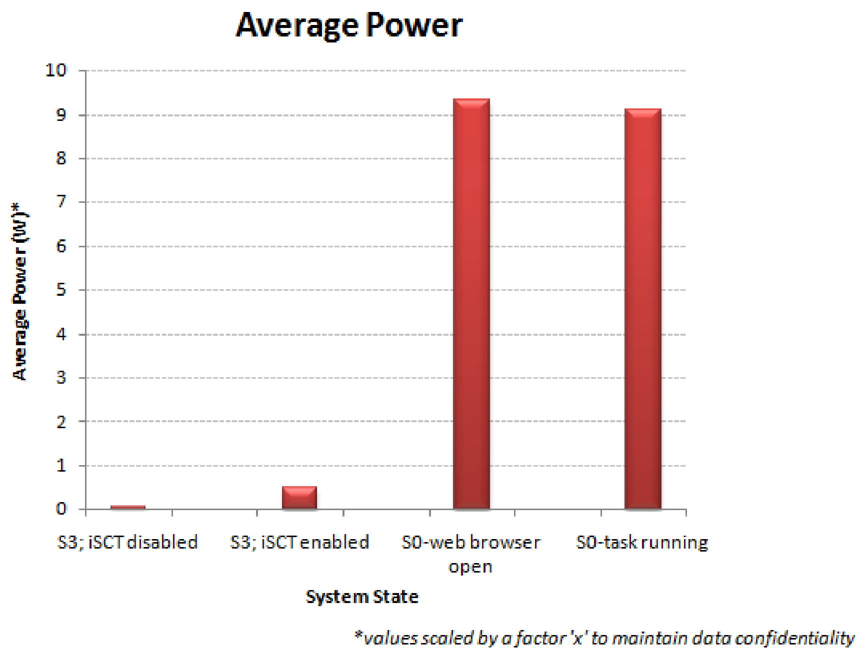

Figure 4.5: Result of power measurement under 4 scenarios

The intention behind covering (iii) and (iv) scenario is to show that web browser open in foreground and web browser open in background also makes a difference in power consumption. Scenario (iv) shows power consumption of a system in general working in S0 state.

Thus, power saving by iSCT was validated. Next goal was to validate the advantage of having Net Detect feature. As stated earlier, Net Detect is like smart periodic wake. It is useless to wake up the system to S0 periodically in the absence of network connectivity. So, when system wakes to S0 and finds that no network is available, it arms Net Detect. As a result, system will periodically scan for network availability and will wake to S0 only on finding network connectivity.

System power was measured for different durations-0.5 hr(30 min), 1, 2, 4, 8, 12 and 15 hours for the following scenarios with ND enabled and ND disabled(periodic wake without network connectivity):

i. PW interval=15 min
ii. PW interval=30 min
iii. PW interval=60 min

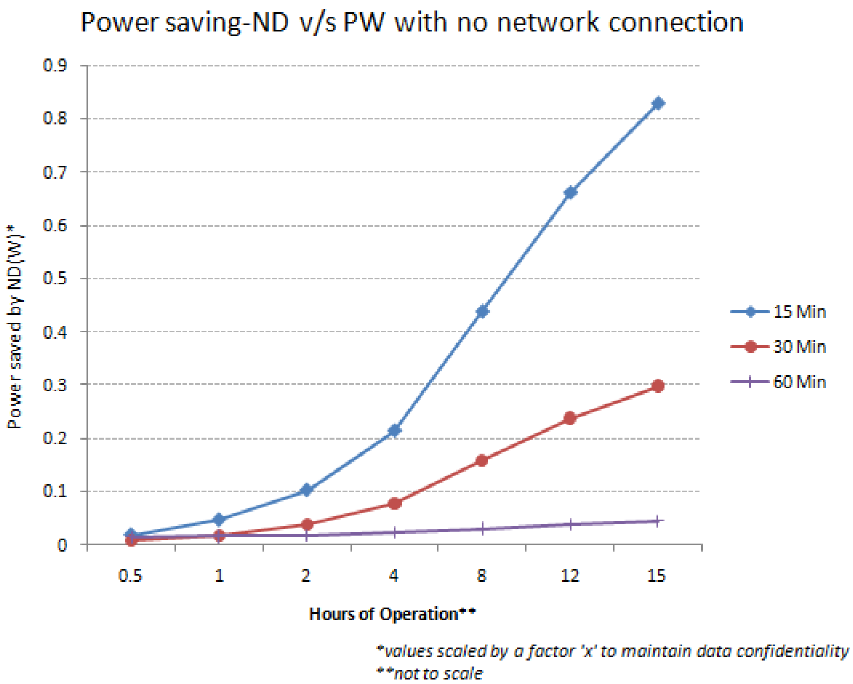

Figure 4.6: Result of power measurement to verify power saving offered by ND feature

Results of these tests are shown in the graph.

Power saved by ND = System power with ND disabled (PW enabled without network connectivity) - System power with ND enabled

Network connectivity being made available only at the end of test duration to ensure that "Always Updated" experience is offered.

In the absence of network, if PW interval is less, power saving is more. This is true because with longer PW interval, system is not waking up that frequently and hence ND offers little power saving over PW without network connection. Next conclusion is, if system stays in S3 with iSCT enabled, to update data periodically, for longer duration, power saving is more. This is also obvious because number of wake events will be more in case of longer test duration. Thus, ND offers more power saving and improved battery life, hence, is an important feature of iSCT.

4.5 Summary

This chapter describes the work carried out as a part of iSCT validation for upcoming Intel platform. Section 4.1 describes the features and the user experience brought by iSCT. It gives an idea of functioning of the feature and the process flow at system level. Next section lists the work done, as a part of role of iCST domain owner. Section 4.3 tells the type of test cases, primary debug and illustrates a failure encountered while testing that was concluded to be Si limitation. To sum up, test plan was prepared in which, 41 test cases were leveraged from the iSCT test plan for previous platform, 18 test cases were removed as a part of optimization and 23 new test cases were added based on the new features and changes in implementation for upcoming platform. 19 sightings were filled in all. Section 4.4 describes the technique deployed for carrying out power measurement and the results of the same. It shows how iSCT is capable of providing "Always Updated" user experience along with saving in power consumption resulting into improved battery life. Also, it proves the advantage of having Net Detect feature on top of periodic wake for better power saving during network unavailability.

Chapter 5

Intel® Common Connectivity Framework

Intel® CCF[25] is a versatile middleware which allows to develop compelling applications (i.e. around gaming, education, collaboration, photo and video sharing, document/photo/video syncing, anti-theft/security, etc.) that can work across platforms (i.e. IA(Intel Architecture)-based Smart Phones (Android), Tablets (Windows 8/Android), and Ultrabooks (Windows 7/8); ARM-based Smart Phones (iOS/Android) and Tablets (iOS/Windows 8/Android), etc.) and across all transport layers (i.e. Bluetooth, WiFi, WiFi Direct, and 3G/4G LTE).

Nowadays many believe that we live in the post-PC era as the amount of various mobile devices is increasing, whereas sharing of content or data between them is getting harder. In reality, that sharing is a key for the always on/always connected era and enabling it is crucially important. Devices available today typically have several ways or mechanisms of connecting to a remote device (over Bluetooth, Wi-Fi Direct or through a common Wi-Fi Access Point). There is however no synergy or consistency between all these different technologies in terms of how they work, how they connect, how users advertise capabilities, etc. The Bluetooth pairing experience is different from the Wi-Fi Direct pairing experience which is likely to be different from the WiGig pairing experience. To have seamless connectivity, it is needed to streamline all of them and make them consistent. Moreover, presently, it is pretty hard do enable interconnection between a mix of devices that may include Apple media tablet, Google Android-based smartphone, Microsoft Windows notebook, and a set-top-box or three connected to TV. While some things can be done, they require messy combination of wires, drivers, memory cards, and proprietary or third-party cloud services.

Intel has defined a framework for seamless connectivity-called **"Common Connectivity Framework (CCF)"** and provides an SDK (Software Development Kit)[24] consisting a cohesive set of APIs that can be used by application developers to have seamless connection between devices in range or over the cloud. With this, a company delivering multiplayer games, collaboration software or social network applications should not have to worry about what network technology to use or how the network connection is established. CCF also enables cross-device connections. The CCF is a type of middleware, which runs on top of the operating system and provides a simple and secure channel between devices for use by other applications. The basic idea of the CCF is to make sharing between devices touch-and-tap easy and unlock new usage scenarios across devices, such as[25]:

- Using smartphone as a controller for a game running on PC;
- Scrolling through web pages on PC using phone or tablets screen as a big touchpad;
- Using phones high-fidelity camera as a webcam for a PC.

Combining the PCs big screen, computing horsepower, and large storage capacity with a mobile devices sensors, touchscreen, and camera opens up all kinds of possibilities.

5.1 Salient features of Intel® CCF

CCF helps to have seamless connectivity:

- Cross device: Ultrabooks, Laptops, Tablets, Smart phones, Convertibles, All-in-ones, Desktop PCs
- Cross OS: Windows 7, Windows 8, Windows Metro UI, Android 2.3, Android 4.0+, iOS 5+
- Multi radios: Wi-Fi, Wi-Fi Direct, Bluetooth, Wired LAN, 3G, LTE
- Multi-Transport: Direct device to device, over the cloud, within same network

CCF allows users on Windows, Android and/or iOS devices to find, connect, share and play (or work) both in proximity as well as over a cloud transport. The CCF SDK plugs into a mobile app and handles all the heavy lifting for advertisement, discovery and connection, so that developers can focus on delivering a great experience instead of the heavy lifting on the communications software.

5.2 More about Intel® CCF

Figure 5.1 shows high level architecture of CCF.

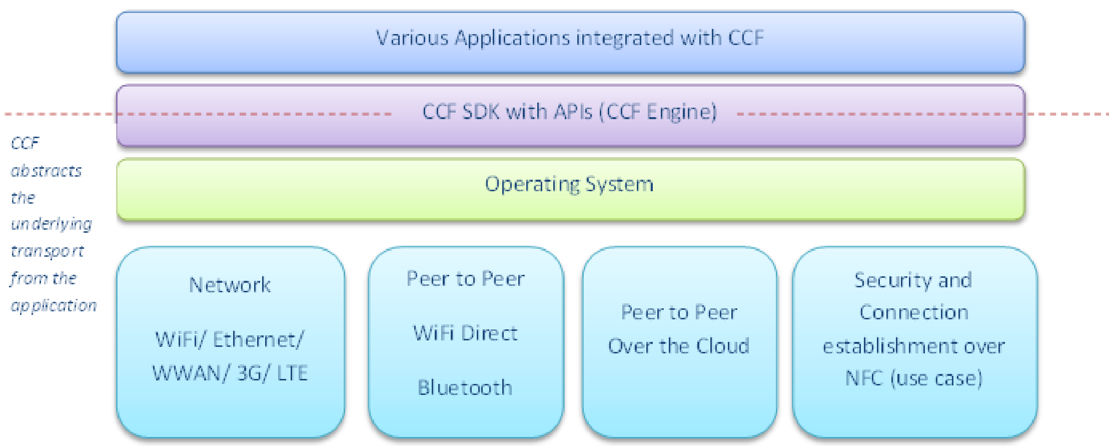

Figure 5.1: CCF Architecture

- CCF communication model is peer to peer.

- CCF SDK acts as a middleware between applications and communication interfaces.

- CCF may run in either of the two modes: "In-proc model[22] or "Out-of-proc model[22].

  - In-proc (In process) means CCF runs as a part of application. In-proc component is implemented as dll and runs in the same process space as its client application.
  - Out-of-proc(Out-Of-Process) means CCF runs as a separate service. Out-of-proc component is implemented as an exe and runs in its own process space.
  - Not all OSs support both the modes.

- CCF provides services like identity management, discovery management, transport management, establishing a secured connection, etc so that developers can focus on creating experiences and not on the connection overhead and management[24].

- Peer to peer transfer over the cloud is supported-

  - This is facilitated by using (ABC)[1]server. NATs and Firewalls cause difficulties for peer to peer transfer. They can block VoIP traffic completely. Servers based on ABC protocol help to overcome these limitations.

- Offers robust connectivity by facilitating "promotion, "failover and "reconnection".

  - Promotion-switching over from a slower to a faster interface

[1]ABC-the protocol used is confidential; Underlying technologies and transport mechanism being confidential are not included in this report. However, documentation is available with the purchase of SDK from Intel.

  - Failover-switching to next possible interface when current interface is no longer available(is disconnected/down)
  - Reconnection-resuming back to the same interface automatically

For example: consider the case where CCF has connected two devices using Bluetooth. Now consider the case where the devices go out of Bluetooth range. The CCF framework knows of other ways to connect the two devices and immediately begins to look for another path for the application data. This process is called "failover" and is one of several examples of CCF actively using different transports to maintain a connection between devices. If CCF finds another path, the application does not necessarily need to know, or deal with, the failover case unless the application has significant latency. If CCF cannot find another path, but the devices are later brought back into range, the CCF framework will reconnect and data will resume flowing over the CCF stream. This process is called "reconnection". Had the case been such that data flow over Bluetooth is getting saturated and meanwhile, it is possible to connect the devices through WiFi due to availability of WiFi network, CCF would switch the data flow from Bluetooth to WiFi. This is called "promotion".

Example of apps available today that support CCF:

i. Speakerfy[26]- to play music across all type of devices in sync. It is a social sound app that allows people in proximity to listen to the same song at same time on multiple devices. The app is available online on Google play store.

ii. McAfee Security Innovations[27]- comes with 4 different security features:

  - Smart Perimeter - Links all the mobile devices of the user together and alerts the user when the devices are separated by more than 30ft. It activates an alarm to let the user quickly identify and recover the device.
  - Safe QR Reader - Ensure the QR codes are safe for browsing

- Data Vault - helps to protect private photos, videos and documents from prying eyes, locally on the device.
- Hidden Device Admin Detector - Detect and remove applications that are hidden from the list of device administration applications-helps in detecting malwares and removing them.
  It is available online on Google play store.

5.3 Scope of work

The framework, terminologies involved, underlying technologies and protocols used, were studied thoroughly, and the knowledge was shared with the team through knowledge sharing/training sessions. Also, exploring currently available apps and hands-on SDK and sample apps integrated with it was done. Initial validation scope and coverage at platform level was identified and presented infront of platform validation architect and CCF software architects. However, many challenges were seen in including the SDK in BKC reports. CCF being SDK, had different customers than actual BKC customers. App Developers, ISVs, enthusiasts, etc were more interested in this SDK rather than direct ODMs/OEMs. Moreover, CCF SDK release milestones were not aligned with the BKC release milestones. Based on feedbacks of marketing and strategic planning groups, it was decided not to include this SDK as a part of BKC release report. It may be included post PV(Platform Validation) milestone stage, wherein a fully stable BKC with its targeted quality is released to customers along with Si(CPU). So, validation scope and coverage that may be provided at this stage was done. Hence, the current scope of work was limited to study of framework only. However, the study in itself was enriching experience.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

Validation is much more than just testing. Platform validation focuses on platform features and capabilities and their interactions. It includes system integration and health assessment of each individual component also. Thus, validation is important. As a part of project work, WiGig (which is infact IEEE802.11ad), a new platform mega-feature and Intel® Smart Connect technology, a solution that provides "Always Updated" and "Always Reachable" user experiences, are validated successfully for upcoming Intel platform. WiGig is wireless technology that allows data transfer at multi gigabit rate. The triband solution (WiFi+Bluetooth+Wigig co-running) will be as pervasive as traditional WiFi is, currently. Challenges are involved in enabling the feature for the first time on a platform. Preparation of test plan that ensures minimum bug escapes to avoid any post launch report of issues is another challenge. Thus, the test plan was prepared diligently to confirm on adequate coverage. Intel® Smart Connect Technology provides three features-Periodic wake, Net Detect and Remote Wake. Standalone as well as co-existence of the features is tested along with interoperability and use case scenarios testing. iSCT is a matured technology while WiGig is in development phase and hence not all of its features are enabled at ingredient level. Thus, WiGig validation is limited to the features that are enabled. Immediate validation of the features as and when they are developed helps in quality improvement from an early stage and helps to hit the market timely.

Intel® CCF provides an out of the box experience to end users by offering nearly seamless connectivity. With so many communication interfaces around, achieving seamless connectivity across the devices and the means of communication between them is a challenge. Intel provides a solution to this in the form of CCF, which helps to provide a robust, intelligent, cross-platform, transport-independent channel for app-to-app communications. CCF comes with cloud support to achieve peer to peer communication even when the users are behind the firewall. Intended users can buy CCF SDK and integrate their application with it to avail the benefits offered by the framework. As a part of project, the framework and its underlying technologies were studied. Validation of the SDK was found to be out of scope at platform and system level at this stage. It may be included lately. Validation scope and coverage to be provided for that was defined.

6.2 Contribution

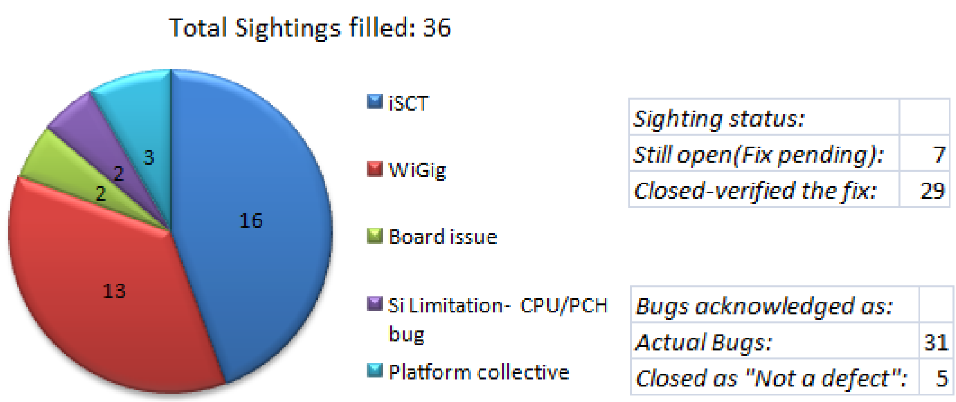

Figure 6.1: Total bugs sighted

Finding bugs and ensuring that it is fixed is the main job of validation engineer.

Bugs can be found only when correct tests are executed. This requires a concrete test plan creation. iSCT and WiGig test plan was created. Test content was written and the execution of test cases yielded into multiple issues. Debug and triage was done to root cause the failure(identify the ingredient causing the bug). Sightings were filled in relevant database to track the progress on solving the bug. These sightings were closed on validating the fix. Process involved working with multiple ingredient teams. Figure 6.1 shows summary of the bugs sighted by me.

Other unique contribution includes test plan creation, providing multiple knowledge sharing sessions(focused on WiGig and CCF) to various teams and documenting several BKMs that would make execution and debug easy. Also, defining scope and coverage for including CCF in BKC stack, post PV (platform validation) milestone was significant.

6.3 Future Scope

As stated earlier, WiGig solution being in evolution stage, it is required to validate all the features of fully enabled WiGig solution in future. System level power measurement carried out for iSCT may be extended to component level power measurement; based on the results, a detail analysis may be done on components consuming higher power and recommendations may be given for any further optimization in power consumption. Also, preparing test plan based on the coverage and scope identified for CCF validation, writing test content and execution of those test cases can be done.

References

[1] Whitepaper on "Platform Validation Overview", *Intel Corporation*, Aug, 2010

[2] "BKC Process and expectations", page: 1-18, *Intel Corporation*, Aug, 2010

[3] "Advanced Configuration and Power Interface Specification Revision 5.0", (Chapter 1,2,3), page: 1-54, Dec, 2011

[4] "Broadwell Platform Architecture Document Revision 1.0", *Intel Corporation*, Jan, 2013

[5] "Intel Core i5-3610ME Processor (PGA) and Mobile Intel QM77 Express Chipset Development Kit" [Online] Available on: `http://www.intel.com/content/www/us/en/intelligent-systems/evaluation-platforms/core-i5-3610me-qm77-development-kit.html`.[Last accessed 02-12-13]

[6] "Defining the Future of Multi-Gigabit Wireless Communications, WiGig White Paper", *Wireless Gigabit Alliance*, July 2010

[7] "WiGig and the future of seamless connectivity", *WiFi Alliance*, Sept 2013

[8] Application Note, "Wireless LAN at 60 GHz - IEEE 802.11ad Explained", *Agilent Technologies*

[9] Mark Stevans, Grant Grafton, "The Benefits of 60 GHz Unlicensed Wireless Communications", White Paper, *Sub10Systems*

[10] "IEEE 802.11ad Microwave Wi-Fi / WiGig Tutorial" [Online]Available on: `http://www.radio-electronics.com/info/wireless/wi-fi/ieee-802-11ad-microwave.php`. [Last accessed 28-11-13]

[11] "Wilocity-Products, Applications and Usage Overview", [Online] Available on: `www.wilocity.com/products`. [Last accessed 04-12-13]

[12] "Dell Wireless Dock - WiGig-spec-sheet", [Online]Available on: `http://i.dell.com/sites/doccontent/consumer/merchandizing/en/Documents/Dell-Wireless-Dock-WiGig-spec-sheet.pdf`.[Last accessed 03-01-14]

[13] "Intel WiDi and Intel Pro Wireless Display", [Online] Available on: https://www-ssl.intel.com/content/www/us/en/architecture-and-technology/intel-wireless-display.html. [Last accessed 04-02-14]

[14] "WiFi Certified Miracast ", [Online] Available on: http://www.wi-fi.org/discover-wi-fi/wi-fi-certified-miracast. [Last accessed 27-12-13]

[15] "Intel Anytime Access Product Requirement and Design Guide", *Intel Corporation*, Aug, 2012

[16] "Intel Smart Connect Technology System Architecture Specification", *Intel Corporation*, June, 2013

[17] "Intel Smart Connect Technology", Intel Support [Online] Available on: http://www.intel.com/p/en_US/support/highlights/sftwr-prod/smrt_cnct. [Last accessed 03-03-14]

[18] "Intel Smart Connect Technology", [Online] Available on: http://www.intel.com/content/www/us/en/architecture-and-technology/smart-connect-technology.html.[Last accessed 12-03-14]

[19] "Considerations for Deploying Intel Smart Connect Technology", Rev 1.0, August, 2012, [Online] Available on: http://download.intel.com/support/services/smartconnect/sb/considerationsfordeployingintelsmartconnecttechnol.pdf. [Last accessed 12-03-14]

[20] "Intel vPro Technology Use Case Reference Design-Intel Smart Connect Technology for Enterprise", Revision 1.0, December, 2012, [Online] Available on: http://downloadmirror.intel.com/22300/eng/Intel(R)%20Smart%20Connect%20Technology%20UCRD%201.0.pdf. [Last accessed 19-03-14]

[21] "PACS tool Guide, Revision 2.1", *Intel Corporation*, Feb, 2013

[22] "In-Process and Out-of-Process Components", Visual Basic Concepts, Microsoft Developer Network, [Online] Available on: http://msdn.microsoft.com/en-us/library/aa242100(v=vs.60).aspx. [Last accessed 12-12-13]

[23] "WHQL Release Signature", Signing Drivers for Public Release, [Online] Available on:http://msdn.microsoft.com/en-us/library/windows/hardware/ff553976(v=vs.85).aspx. [Last accessed 08-11-13]

[24] "Intel CCF 3.0 Beta1 SDK Documentation Preview", *Intel Corporation*, Oct, 2013

[25] "Intel CCF", *Intel Corporation*, June, 2013

[26] "Speakerfy", [Online] Available on: `https://play.google.com/store/apps/details?id=com.optinillc.speakerfy`. [Last accessed 26-12-13]

[27] "McAfee Security Innovations", [Online] Available on: `https://play.google.com/store/apps/details?id=com.mcafee.mmi`.[Last accessed 28-12-13]