TESSOLVE
A Hero Electronix Venture

Volume – 8 | Issue – 02
July 2020

FIRST BIN

A Newsletter for the Semiconductor Engineering Community

Contents

Editorial Team

ADVISORY COMMITTEE
Srinivas Chinamilli
Rajakumar D

EDITORIAL SUPPORT
Anuradha Noone
Srinivasa Rao Peram

TECHNICAL COMMITTEE
Vidyut Yagnik
Srinivasprasad B V
Prashanth Kudva
Sudarshan Sarma HS
Srinivasan C

OPERATIONS SUPPORT
Thirumalesh Babu Murthy

MARKETING
Tanusree Mathad

Printed and Published on behalf of
Tessolve Semiconductor Pvt. Ltd.
Plot # 31, Electronic City,
Phase 2, Bangalore 560 100,
Karnataka, India.
Tel: +91 80 4181 2626

Your kind enquiries / feedback solicited at,
www.tessolve.com
sales@tessolve.com / news@tessolve.com

Dear Customer,

I hope you and your family are staying safe and healthy!

It has been an interesting first half of the year. Covid19 crisis has brought us unprecedented times and like many businesses, we faced several challenges that came with it. Tessolve acted quickly to ensure employee health and safety and at the same time ensure business continuity for our customers.

Tessolve team has taken corrective actions to retarget our supply chains to ensure none of the hardware projects and shipments are affected. We have also rapidly enabled remote work for all the projects including those requiring access to our Test & Validation labs in Bangalore, Singapore, and Malaysia.

I am proud to say that we enabled 100% business continuity for our customers and we are continuing to take corrective measures as the situation evolves. Several of you have commended us that you have not seen any dip in quality of delivery inspite of the crisis. I thank our employees for their dedication and hard work in ensuring continuity in project engagements for all our customers.

We are continuing to make investments to enhance our engineering offerings and provide the better value you. We completed a successful acquisition and integration of T&VS based out of the U.K and India. T&VS is well known for Design Verification expertise and they add significant strength in that area for Tessolve. T&VS brings to us an excellent leadership team with over 400 talented and dedicated engineers. We welcome the founders Mike Bartley, Karthik Nagappan, and their entire team to the Tessolve family.

We are continuing to upgrade our Test and Reliability Qualification labs. We have invested in new testers namely Advantest Smartscale and ETS364 testers to better address increasing project volumes. We have augmented Reliability Qualification labs with CSAM capability as well. As you are aware, Tessolve is the only company in India with comprehensive Package and Silicon qualification labs (HTOL, HAST, ESD/Latchup, etc). Please feel free to contact us with your product qualification needs.

Our latest engineering offerings include mmWave and High-performance computing including PAM4, HBM solutions. We have successfully designed and productized several 7-10nm SOCS as well as high power PIMIC and Analog/RF products.

We look forward to continuing to partner with you to provide valuable solutions for your engineering requirements.

Let me also congratulate all the following special accomplishments in presenting papers and competition by our team members:

Jagadish Kumar Chandrasekaran, Srinivasan C and Gowri Shankar Ilankumaran on their Tutorial presentation "Fast & Furious High Speed I/O (NRZ MHz to PAM4 GHz) and its ATE Challenges" at ITC India 2020.

Kiritkumar Panchal on his article Published " Electric Vehicle Battery Management system and Chargers" on TelematicsWire Magazine.

Best Regards,
Srinivas Chinamilli
Co-Founder & President

Tessolve Showcase

1. VALIDATION OF INTERCONNECTS FOR 56GBPS & 112GBPS DATA TRANSMISSION

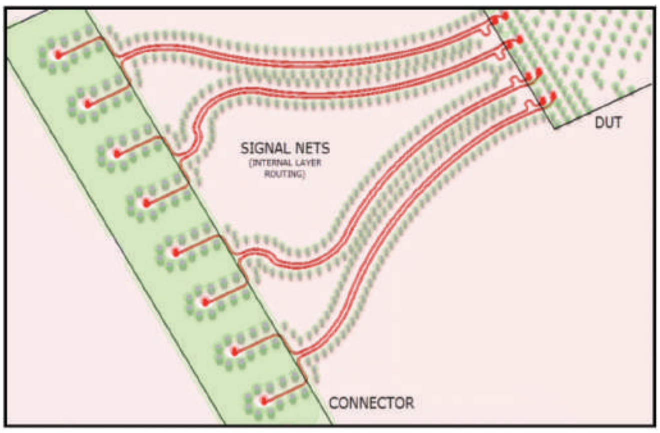
Author: Rajesh Badala Jagadeesh – Sr.Manager SI, Venkatesh Ramashastry – Lead SI

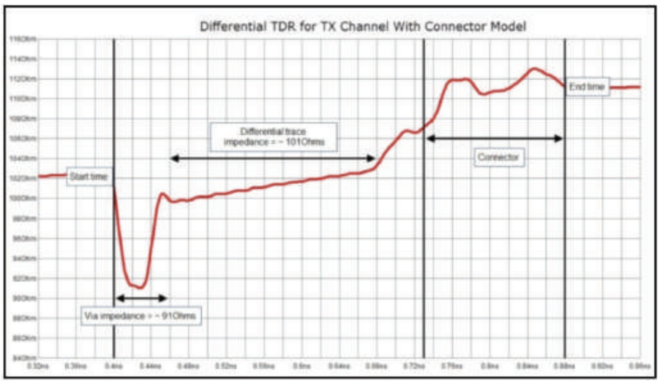
Introduction

This is a case study validating a board interconnect design for transmission of 56Gbps and 112Gbps data rate PAM-4 signals. With the regular NRZ/PAM2 signaling scheme the serial data transmission rate is generally termed in bit rate – Gbps and the approach is binary i.e. 1-bit is transmitted per symbol. For higher speed requirements the SerDes employ PAM4 modulation with 2 bit transmission per symbol. The data transfer rate terms to Baud rate – Gbaud with PAM4 technique. A 28Gbaud transfer is equal to 56Gbps and a 56Gbaud transfer is equal to a 112Gbps data transfer, but the signal core frequencies of 14GHz and 28GHz respectively remain unchanged. PAM4 SERDES data links governed by IEEE 802.3bs standards find application in communication standards like 400GbE/800GbE Ethernet routers, switches etc. serving higher data rate demands for telecom service, data centers and similar market segment.

Interconnect Design & Simulations

Development of interconnects for high-speed data links need design validation thoroughly supported with high accuracy simulations and careful analysis of simulated data. The PCB layout with the best impedance control design practices is essential. In the design considered here, blind via construction is used for layer transition avoiding stub formation in the signal path. Ground via stitching ensuring return path and shielding for signals are in place. Signals traverse several levels of interconnects -the connectors, the pcb, the IC Package-BGA and the die. The below figures represent the layout and the TDR plot.

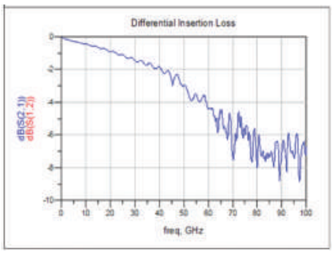


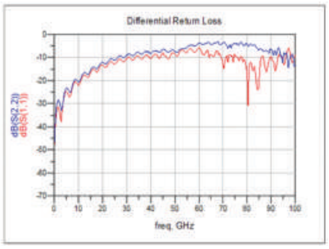


The integrity of the transferred signal is quantified accounting the impact of discontinuities and attenuation by each of the interconnect levels. Every stage is modeled into s-parameters with 3D EM field simulations. A TDR plot generated highlights the impedance deviation and aid in fine-tuning the geometries.

Frequency domain plots for the bandwidth of 100GHz

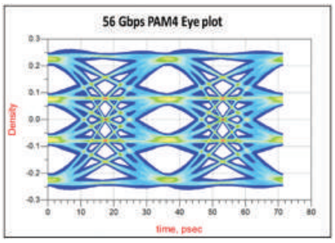
The full channel IL- insertion loss and RL-return loss patterns are examined and verified with interface specific electrical requirement standard. Any attenuation due to material properties and discontinuities are checked and optimized

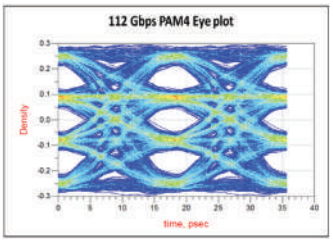




Eye Diagram validation for 56Gbps and 112Gbps data rate

With a complex PRBS pattern run for sufficient duration the PAM 4 eye diagram is generated for 56Gbps and 112Gbps data rates. The eye may initially look stressed but opens up with optimized interconnects and with suitable equalization employed.





		56 Gbps	112 Gbps
Parameter	Spec (mV)	Achieved (mV)	Achieved (mV)
Eye Height 1	100	104	101.6
Eye Height 2	100	105	103
Eye Height 3	100	105	102.8

		56 Gbps	112 Gbps
Parameter	Spec (UI)	Achieved (UI)	Achieved (UI)
Eye Height 1	0.4	0.49	0.42
Eye Height 2	0.4	0.51	0.41
Eye Height 3	0.4	0.5	0.42

Conclusion

With sharper signal rise times, higher data rates, denser circuits, and stringent interface specifications, developing interconnects retaining the electrical integrity gets tougher and iterative. Design guided through simulations is inevitable. Validation gives great confidence to engineers and helps avoid design re-spins, minimizes development times keeping pace with the progressing technology.

2. HIGH SPEED PROCESSING CARD WITH INTEL ARRIA 10 SOC

Author: Akshay Hublikar – Hardware Lead

Tessolve High speed Processing Card is based on the Arria 10 MultiProcessor SoC device from Intel.

The Arria10 SoC devices support Dual Core ARM Cortex-A9 core up to 1.5 GHz speed/core. The Dual ARM Cortex-A9 core with FPGA fabric allows greater flexibility for the system designers and helps lower the system cost and power consumption. This improved logic integration with a rich feature set of embedded peripherals, hardened floating point variable precision DSP blocks, embedded high speed transceivers, hard memory controllers and protocol intellectual property controllers which is ideal for cost-sensitive high-end applications.

The Card uses multiple Buck converters from Intel and Vishay for powering the SOC.

The device also includes on chip memory, external memory interfaces, and a set of peripheral connectivity interfaces.

The Card comes equipped with 2Gb of DDR4-2400 RAM dedicated to the ARM cores, and 4Gb of DDR4-2400 RAM dedicated to the programmable logic. In addition, SDCARD and QSPI NOR Flash is available for booting the high speed card Programmable Logic, Firmware Processing System and local, permanent data storage.

Measuring 95 x 85mm, the Card includes two high speed board to board connectors allowing the board to be targeted to specific applications by fitting it with any compatible front end I/O interfaces.

The high speed card is designed to meet IP65 requirements and uses IP65 connectors for power and Ethernet interface as a result this board can be mainly used for designing different high speed applications such as surveillance cameras, radar modules which need water and dust protection in the outdoor conditions. The board is useful in building various applications such as Test and measurement equipment, Diagnostic medical imaging equipment, Wireless infrastructure equipment, Mobile backhaul, remote radio head designs, Compute and storage equipment, Broadcast and distribution equipment.

Board Specifications of High speed Processing Card:

- Intel's Arria10 SoC/FPGA (Part: 10AS048H3F34I2SG)
- 2GB DDR4 SDRAM (32bit) with ECC for HPS
- 4GB DDR4 RAM (64bit) from FPGA
- Micro SD Connector for HPS booting
- Optional SPI Boot FLASH
- JTAG Header for Debug
- UART to USB debug port
- 1G Ethernet on HPS side
- Integrated DCDC convertors and Simplified HW Power Supply Sequencer
- Power Supply Required : 12V /4 Amps
- Form Factor : 95mm x 85mm
- Diagnostics test suite for HPS side interfaces.
- Linux RTOS 4.9.78 kernel Ported with interface drivers/BSPs
- FPGA High-Speed Transceivers (up to 17.4Gbps) x 18

Advantages of using this card

Balance of high-performance and low-power capabilities of Intel Arria 10 FPGA. With a comprehensive set of power reduction features, Intel Arria 10 FPGAs and SoCs reduce power by up to 40% compared to prior-generation midrange devices. The Intel Arria 10 FPGA core performance of over 500 MHz enables 5X oversampling to provide users 100 MHz of RF bandwidth. By offering higher numbers of variable-precision digital signal processing (DSP) blocks and dual-core ARM Cortex-A9 processor options, along with support for Common Public Radio Interface (CPRI), small form factor pluggable (SFP), and JESD protocols, Intel Arria 10 FPGAs and SoCs are low-risk design alternatives to application-specific integrated circuits. They have the added benefit of adaptability and re-programmability to accommodate emerging and evolving protocols. Using Intel Altera's DMSoC development environment, the user application code can be partitioned between the Arria's available cores, and the programmable logic.

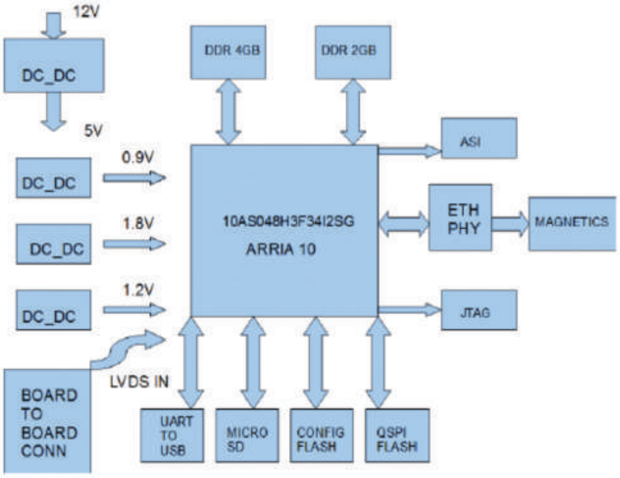
Algorithms compiled to run in hardware in the programmable logic are said to result in ten to one hundred times the speed over software execution.

Challenges Faced

The board involves high speed interfaces such as DDR4 (2400MHz) and Ethernet (1G) which needs proper design and layout considerations. During interface bring up DDR4 was seen hanging during regression testing and after carefully tuning the SI parameters using Intels tool, the DDR was fixed and verified for continuous running.

The Ethernet was working at 100Mbps instead of 1Gbps and the debug results showed us that the traces were getting disconnected in the internal layers due to fabrication issues. To fix this the board was redesigned to run the traces on the top layer and the Ethernet started working at 1Gbps speed.

Block Diagram



Board Snaps

FrontSide

Back Side

The main purpose of designing this board is for evaluating automotive radar chips for a specific customer who was looking to demonstrate their chip capabilities. The customer had designed a radar chip useful for human and vehicle movement detection using FMCW technique. The radar chip operates at 76-81GHz frequency. Automotive radar consists of a transmitter and a receiver. The transmitter sends out radio waves that hit an object and bounce back to the receiver, determining the objects' distance, speed and direction. High speed card receives digital data from the RF board via analog to digital converter. The received data is processed by the high speed card and sent over Ethernet channel to a computer and the data is visualized in the form of 3D point cloud to detect different objects. The product has an enclosure built to ensure thermal stability of the boards to work in outdoor conditions even at higher temperatures of up to 40 degrees. The system has two connectors at backside for power and Ethernet entry. The product is fixed on the front side of the car to detect different objects and the information is further passed to engine control unit for implementing different features for car safety such as Adaptive cruise control, Emergency braking, Cross traffic alert, Lane change assist, Driverless car etc.

3. 3U VPX BOARD WITH ZYNQ ULTRASCALE+ SOC

Author: Harsha Kumar Bhandiwad – Sr. Manager, Test Engineering

Tessolve developed 3U VPX card

Tessolve - System Design Engineering Group has now come up with a new advanced 3U-VPX-ZUS1 module which is a 3U Open VPX module based on the Zynq UltraScale+ MultiProcessor SoC device from Xilinx. Card is a 3U VPX generic platform board with High end SOC wherein XMC/FMC modules can be plugged and can be used for LIDAR / RADAR / SONAR Applications, Image processing, Target detection, EW / ISR, Advanced multi-axes motors controller and multi-gigabit Ethernet communications. End application might be developed for Military, Communication devices. Etc.

Above picture depicts the end use of the 3U VPX card in conduction cooled chassis.

The Zynq UltraScale+ is an SOC integrated with FPGA fabric, consists of a

- **Quad-core ARM Cortex-A53 based Application Processing Unit (APU),**
- **Dual-core ARM Cortex-R5 based Real-Time Processing Unit (RPU)**
- **ARM MALI-400 based Graphic Processing Unit (GPU)**
- **UltraScale+ Programmable Logic (PL) in a single device.**

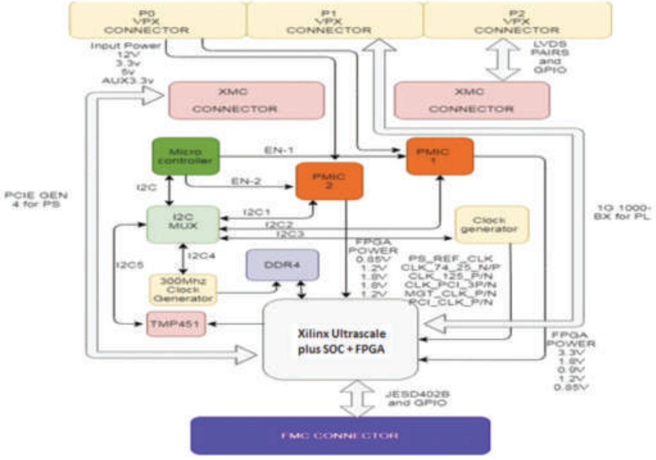
It also includes on-chip memory, external memory interfaces, and a rich set of peripheral connectivity interfaces.

The 3U-VPX-ZUS1 module uses two advanced PMIC devices from Infineon Technology – which are controlled using PMBus and I2C. The device also includes on chip memory, external memory interfaces, and a set of peripheral connectivity interfaces.

The 3U-VPX-ZUS1 module comes equipped with

- **2Gb of DDR4-2400 RAM dedicated to the ARM cores**
- **2Gb of DDR4-2400 RAM dedicated to the programmable logic.**
- **SDCARD for booting option**
- **2x 512Mb of QSPI NOR Flash**
- **Firmware Processing System and local, permanent data storage.**
- **On-board FMC site compliant to the Vita 57.1 HPC std. with 40 Diff Pairs and 8 MGT**
- **Measuring 100 x 160mm, module includes a FPGA mezzanine card (FMC) connector allowing the board to be targeted to specific applications by fitting it with any compatible front end I/O interface cards as shown in the above image.**

3U-VPX-ZUS1 Module includes XMC carrier connectors which support Single size XMC (High Speed PCI Mezzanine Card) Module, End-Point Processor XMC Module, Front Panel I/O interface aperture and complies with XMC, refer the below block diagram.



Board Specifications - 3U VPX Interfaces

- **VITA 46.0/46.4/65.0 VPX/OpenVPX Specifications compliant**
- **X4 or X8 PCIe lane connected to Zynq Ultrascale+ Programming Logic / any other user high speed SERDES bus can be connected**
- **1000BASE-SGMII links on VPX Control Plane**
- **JTAG OpenVPX VITA 65.0 Profiles**
- **Board Management Controller (BMC) using TI MSP430™ family**

VITA 57 FMC Slot

- Compliant to the High Pin Count (HPC) VITA 57.1 specification
- 8x high-performance MGT @ up to 16.3 Gb/s to/from Zynq Programmable Logic
- 40 LVDS (LA [0:33], HA[0:5]) to/from FPGA Zynq Ultrascale+ Programmable Logic
- 2x clocks to Zynq Ultrascale+ Programmable Logic and 2x clks. for GTH Transceiver
- VADJ = 1V8 (default). 2V5 not supported by Zynq Ultrascale+ HP Banks

XMC SLOT

- XMC carrier supports Single size XMC Module.
- XMC compliant with CMC has 4 diff pair to support PCIe X4 Gen2 connected Zynq Utrascale+ Processing System and Supports Front I/O
- Support End-Point Processor XMC Module and XMC VPWR of +5V OR +12V

Using Xilinx's SDSoC development environment, the user application code can be partitioned between the Zynq's available cores, and the programmable logic. Algorithms compiled to run in hardware in the programmable logic are said to result in ten to one hundred times the speed over software execution.

Software Features

- Petalinux tool used for Linux development
- All PS related drivers are developed using Xilinx design & can be enabled independently or based on the HDF file generated from Vivado.
- AXI based drivers are developed for PL based interfaces.
- Have developed 'Pass through' drivers using MMap.

Takeaways

- The board when plugged inside the chassis can act as a Single Board Computer in the VPX system which improvises Size, Weight, Power and Cost (SWaP-C).
- Through Make in India initiative M/s Tessolve has developed this first SBC board using Xilinx Zynq Ultra Scale Plus SOC ship, this is first of its kind where we have all technical support from INDIA only.
- Users can contact us for this COTS module for their development activity for which the lead time would be minimal and support would be to the maximum extent including deployment, Integration and Testing.

Tessolve Engineering Challenge Contest

1. TESTBENCH GENERATION FROM WGL

**Author: Nivethitha Murugan – Design Engineer2,
Jovin Basil Roy J – Director – VLSI Design**

ABSTRACT

The Waveform Generation Language (WGL) is a data description language supported by Test Systems Strategies Inc and is also an intermediate file format used by the semiconductor industry for converting digital test patterns from a logic simulator to tester hardware. The vector representation in WGL is cycle-based. Only the cycle-based simulators can simulate WGL files. The event-based simulators such as any RTL/Netlist HDL simulators (Questa, VCS, Xcilium) need HDL files like testbench with vectors to simulate. More often the WGL at ATE gets modified for fine tuning the test. This modified WGL needs to be simulated again by the design team. This indeed demands a hdl based testbench which can simulate the WGL to reproduce the expected behaviors of the design in the HDL simulator environment.

Keywords: WGL,HDL,testbenchA

Introduction

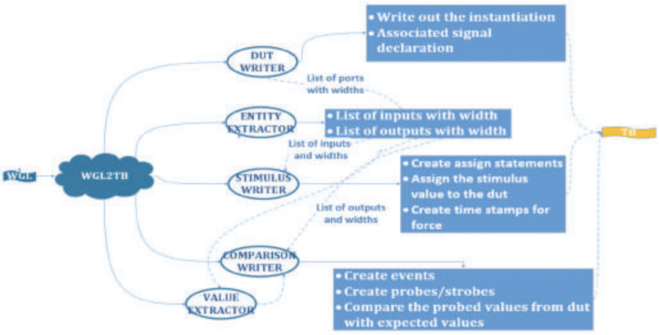
The verification environment and higher-level HDL environments have more safe levels on logical signals with proper initializations. There could be a chance of forces and deposits used for verifying the design with assumptions. These won't be present in WGL files. The real electrical and electronics problems associated with semiconductor come out once we port the WGL file on ATE. This stymied the debug and stabilizing the tester program. The reverse play of WGL is mandatory to be converted as HDL and gets simulated with HDL simulators to put right the patterns.

The idea of generating TB from WGL by Python comes out from facing the real issue in the project. Generally, the WGL will be generated and pass it to the ATE team. Then the ATE team will generate testbench for a particular WGL file to verify. We faced a large time gap between the generation of testbench from the testing team and then simulating it with the compilers to verify the generated WGL. There we need a quick fix to ensure the verification of WGL and to move on further to sign-off test. So, the python script has developed to generate TB and DV team simulated the generated TB with WGL in compilers. It is a flat model which means that TB doesn't have to be aware of any scan structures or DfT specifics.

APPROACH

The extraction of HDL test benches from ATE is too late for DfT engineers to sign-off tests. So, the preparation of WGL files and test engineers to do the conversion will not help to fix any design bugs before tapeout. We need an intermediate and powerful utility that can convert the WGL to Verilog testbench for doing quick check on the quality of our tests.

The back and forth WGL & HDL handling can be reduced much with this utility. WGL2TB is a python based script that can read WGL file and write out Verilog testbench. The following picture captures the modularized functionalities of the script. The subfunctions are developed with certain features to enhance and accommodate every syntax of WGL.

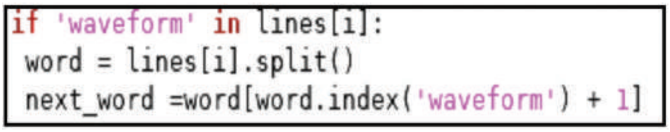

WGL
WGL2TB
DUT WRITER
ENTITY EXTRACTOR
STIMULUS WRITER
COMPARISON WRITER
VALUE EXTRACTOR
• Write out the instantiation
• Associated signal declaration
List of ports with widths
• List of inputs with width
• List of outputs with width
List of inputs and widths
• Create assign statements
• Assign the stimulus value to the dut
• Create time stamps for force
List of outputs and widths
• Create events
• Create probes/strobes
• Compare the probed values from dut with expected values
TB

Considering all above concerns, the algorithm is composed to automate the process of generating Testbench (TB) from WGL using Python script. The algorithm consists of five sub functions DUT Writer, Entity Extractor, Stimulus Writer, Value Extractor, and Comparison Writer.

Figure 1 depicts the different functions of the above steps. The different steps were explained as mentioned below.

Step 1: DUT Writer

The portion of the script or sub function extracts the module name from the WGL file, instantiation of DUT and declares the signal in TB.



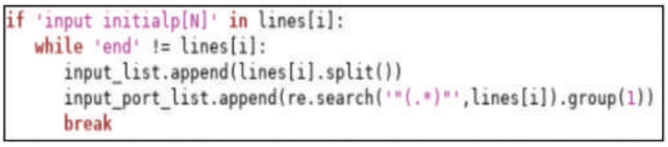
```
if 'waveform' in lines[i]:
 word = lines[i].split()
 next_word =word[word.index('waveform') + 1]
```

Fig 2: Extraction of module name from WGL in Python script

This makes the script more generic since the dut information is extracted from the WGL file. So this becomes more generic and independent.

Step 2: Entity Extractor

This portion of the script or sub function extracts all the input, output and input segregated in a list from WGL.



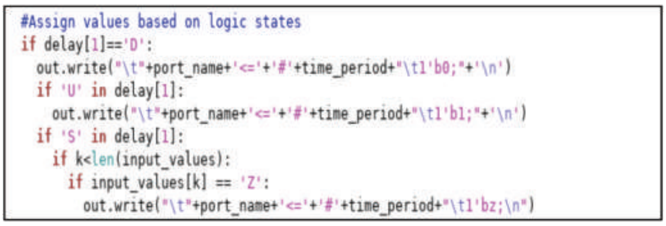
```
if 'input initialp[N]' in lines[i]:
   while 'end' != lines[i]:
      input_list.append(lines[i].split())
      input_port_list.append(re.search('"(.*)"',lines[i]).group(1))
      break
```

Fig 3: Extracting the inputs and segregated in a list

This helps the rest of the functions to use this information. In case of any new declarations or control generations in the script can use this information.

Step 3: Stimulus Writer

In this step, the delays are extracted from WGL and creating timestamps for the stimulus by script.

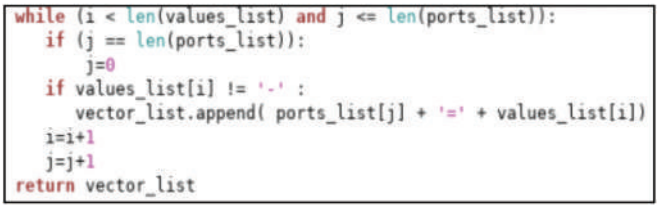


```
#Assign values based on logic states
if delay[1]=='D':
  out.write("\t"+port_name+'<='+'#'+time_period+"\t1'b0;"+'\n')
  if 'U' in delay[1]:
    out.write("\t"+port_name+'<='+'#'+time_period+"\t1'b1;"+'\n')
  if 'S' in delay[1]:
    if k<len(input_values):
      if input_values[k] == 'Z':
        out.write("\t"+port_name+'<='+'#'+time_period+"\t1'bz;\n")
```

Fig 4: Port name and the delay for each logical state transition is written to output_file

Step 4: Value Extractor

The Function of value extractor is to assign the corresponding values to the corresponding signals and segregated in a vector list.



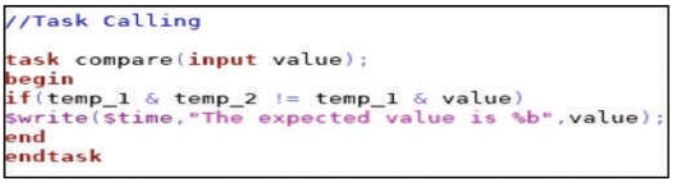
```
while (i < len(values_list) and j <= len(ports_list)):
    if (j == len(ports_list)):
        j=0
    if values_list[i] != '-' :
        vector_list.append( ports_list[j] + '=' + values_list[i])
    i=i+1
    j=j+1
return vector_list
```

Fig 5: Values and the ports are segregated in a vector list

The WGL file is read line by line and using the pattern matching statements and the particular lines are extracted.

Step 5: Comparison Writer

In this step, the actual output is compared with the expected output. The expected output is obtained from the WGL file.



```
//Task Calling

task compare(input value);
begin
if(temp_1 & temp_2 != temp_1 & value)
$write($time,"The expected value is %b",value);
end
endtask
```

Fig 6: Task for comparison

A task is written, or the event can be triggered in the testbench for comparison operation.

Advantages

- **It consumes very little time to generate the Testbench.**
- **The script is based on WGL syntax, so it is generic.**
- **Generated Testbench ensures the backward compatibility of the WGL files at ATE.**

Future Scope in progress

1. Further enhancement can be done to convert from simple Verilog testbench to SV or UVM based environments.
2. Parameterized periods would help the Design team to adjust the frequency operations during timing simulations.
3. Adding assertions to check the clocks or specified ports on specified cycles.
4. Compatibility of the script to read the STIL file.

Conclusion

The WGL2TB bridges the gap between ATE and simulators. The impending feel of WGL on event simulators increases the level of confidence. WGL2TB is handy for DfT and test engineers to regenerate debugging environment in a controlled and internally accessible simulator environments for any device.

2. TESSENT JTAG SEQUENCE GENERATOR

Author: Jeyaprabu Karuppusamy – Design Engineer2

ABSTRACT

The power up sequence of SoC's is complex with multiple power domains and clocking schemes. These sequences are completely functional dependent and mandatory for entering test modes. Though the conventional power enables controls are taken over by test override bits through internal registers, they need a proper sequence while accessing. In general, these registers are accessed through standard protocols such as IEEE1149.1 TAP or SPI or FCSI. Due to the complexity of internal sequences the register access needs to be in a certain order with variable widths. There are many access policies for the crypto engines and their keys, which can be accessed through test modes and functional modes. This complicates the test engineering to prepare tests or vectors for ATE. The approach might be simpler if a single controller or bootstrap is used for unlocking the device from functional mode to required test modes.

Keywords: IEEE1149.1, TAP, PLL, SPI, FCSI, SVF, RTL, VCO

Introduction

The present Era SoC's have more than 10 tap controllers (test controllers) being connected in a network to have an encrypted policy based unlocking sequences. The selection of a TAP in a hierarchy is challenging. Manually creating setup procedures for ATPG, MBIST tools are prone to human errors and time consuming. As we verify the designs multiple times throughout for development of RTL, the changes need to be precise and sooner to accommodate project schedules. The developed utility reads higher level excel or text-based IR, DR details and data to create the required procedures for tool specific sequences. The converter is developed using Practical Extraction and Reporting Language (PERL). It converts any number of Test Access Port (TAP) Instruction and Data Registers (IR-DR) sequences into Serial Vector Format (SVF) and Tessent TestKompress supported TestSetup Procedure file. Especially the PLL bring up for at-speed testing includes programming PLL configuration registers such as the divider values, controls to VCO, and controls to output buffers. This utility comes in handy while developing complex sequences and backtracing the sequences through simulation.

JTAG TAP controller contains an "Instruction Register (IR)" and one or more "Data Registers (DR)" that can be accessed over the 4-wire JTAG interface. At any given point in time of operation, the TAP controller will place one of these between the TDI and TDO lines. Data will be shifted into the register from TDI on the rising edge of TCK and data will be captured into TDO in the falling edge of TCK. So, to read data from a register between TDI and TDO, some dummy bits can be shifted into TDI that causes data present in the register to be pushed out via TDO.

The tessent test procedure file contains scan and clock procedures, and non-scan procedures. The scan and clock-related procedures inform the tool how to operate the scan chain and pulse clocks. The non-scan procedures can represent any type of pattern that the tool produces. The test_setup procedure is particularly useful for initializing system or boundary scan circuitry. Below shown figure illustrates the sequence generator operation.

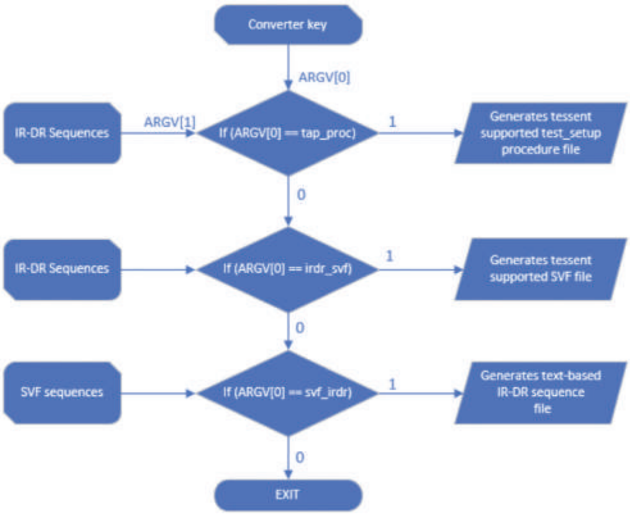
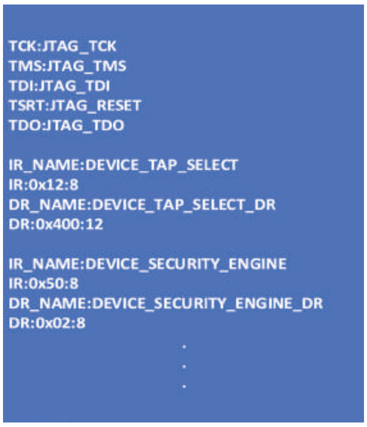


Figure 1: Sequence converter process

Approach

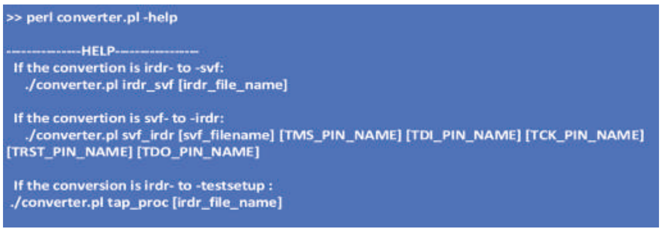
Sequence converter requires a higher-level excel or text-based IR and DR details with corresponding TAP ports with required syntax. The given figure captures an example of sequences. The initial 5 lines are indicating the pin mappings of the design for TAP controller. The IR_NAME, IR, DR_NAME, DR are keywords to mark them as IR and DR registers. The values and width of those registers are mentioned as per the required sequence. The script understands the DR, IR and converts cycle-based vectors and states of TMS, TDI, TDO, TRST, TCK signals.



```
TCK:JTAG_TCK
TMS:JTAG_TMS
TDI:JTAG_TDI
TSRT:JTAG_RESET
TDO:JTAG_TDO

IR_NAME:DEVICE_TAP_SELECT
IR:0x12:8
DR_NAME:DEVICE_TAP_SELECT_DR
DR:0x400:12

IR_NAME:DEVICE_SECURITY_ENGINE
IR:0x50:8
DR_NAME:DEVICE_SECURITY_ENGINE_DR
DR:0x02:8
                    .
                    .
                    .
```

Figure 2: Text-based Input sequences

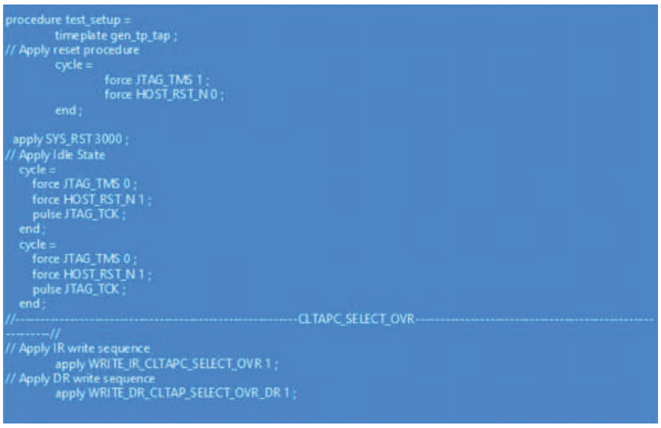
Sequence converter provides options to select the output file format such as '. svf' or '. testproc' or IR-DR sequence file by issuing corresponding switches.



```
>> perl converter.pl -help

--------------HELP---------------
 If the convertion is irdr- to -svf:
    ./converter.pl irdr_svf [irdr_file_name]

 If the convertion is svf- to -irdr:
    ./converter.pl svf_irdr [svf_filename] [TMS_PIN_NAME] [TDI_PIN_NAME] [TCK_PIN_NAME]
[TRST_PIN_NAME] [TDO_PIN_NAME]

 If the conversion is irdr- to -testsetup :
 ./converter.pl tap_proc [irdr_file_name]
```

Figure 3: The procedure to use the script

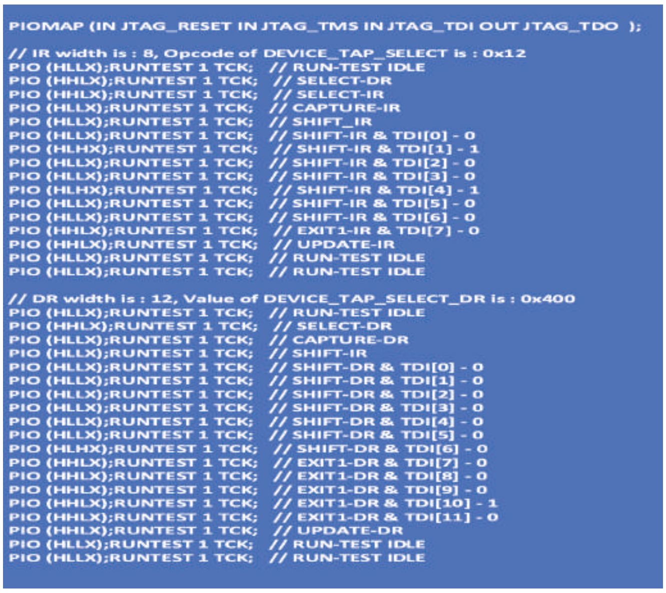
Instruction and Data register values are extracted from input sequence file and converted into svf or testproc if converter key is set to "irdr_svf" or "tap_proc". The following is the syntax of Tessent setup procedures. So, such a long and configuration sequences can be quickly written out as setup procedures. In case of at-speed testing we need to even program the PLLs for its desired frequencies before the testing started. So, the sequences can be added quickly without human errors.



```
procedure test_setup =
          timeplate gen_tp_tap ;
// Apply reset procedure
          cycle =
                    force JTAG_TMS 1 ;
                    force HOST_RST_N 0 ;
          end ;

 apply SYS_RST 3000 ;
// Apply Idle State
   cycle =
     force JTAG_TMS 0 ;
     force HOST_RST_N 1 ;
     pulse JTAG_TCK ;
   end ;
   cycle =
     force JTAG_TMS 0 ;
     force HOST_RST_N 1 ;
     pulse JTAG_TCK ;
   end ;
//--------------------------------------------------------CLTAPC_SELECT_OVR------------------------------------------
---------//
// Apply IR write sequence
          apply WRITE_IR_CLTAPC_SELECT_OVR 1 ;
// Apply DR write sequence
          apply WRITE_DR_CLTAP_SELECT_OVR_DR 1 ;
```

Figure 4: Generated test_setup file

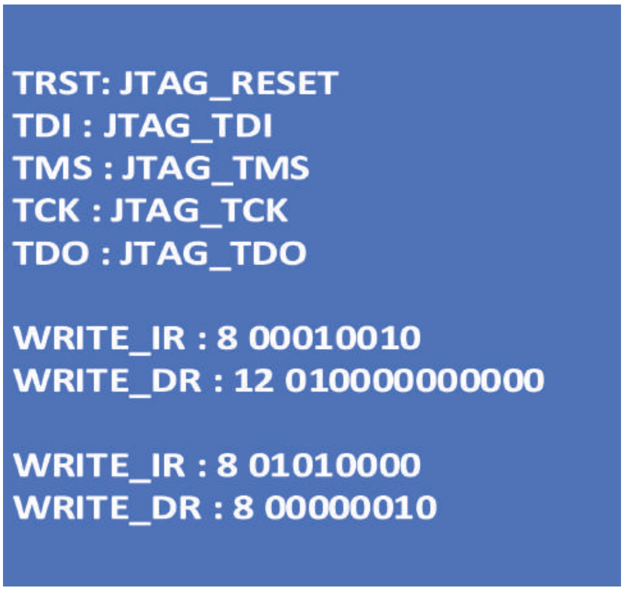
The following picture shows serial vector format which is generated from IR-DR sequences. This is very important with Tessent MBIST, BSCAN or LBIST pattern generation where the setup is like setup procedure in atpg tools.



```
PIOMAP (IN JTAG_RESET IN JTAG_TMS IN JTAG_TDI OUT JTAG_TDO  );

// IR width is : 8, Opcode of DEVICE_TAP_SELECT is : 0x12
PIO (HLLX);RUNTEST 1 TCK;   // RUN-TEST IDLE
PIO (HHLX);RUNTEST 1 TCK;   // SELECT-DR
PIO (HHLX);RUNTEST 1 TCK;   // SELECT-IR
PIO (HLLX);RUNTEST 1 TCK;   // CAPTURE-IR
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT_IR
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[0] - 0
PIO (HLHX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[1] - 1
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[2] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[3] - 0
PIO (HLHX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[4] - 1
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[5] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-IR & TDI[6] - 0
PIO (HHLX);RUNTEST 1 TCK;   // EXIT1-IR & TDI[7] - 0
PIO (HHLX);RUNTEST 1 TCK;   // UPDATE-IR
PIO (HLLX);RUNTEST 1 TCK;   // RUN-TEST IDLE
PIO (HLLX);RUNTEST 1 TCK;   // RUN-TEST IDLE

// DR width is : 12, Value of DEVICE_TAP_SELECT_DR is : 0x400
PIO (HLLX);RUNTEST 1 TCK;   // RUN-TEST IDLE
PIO (HHLX);RUNTEST 1 TCK;   // SELECT-DR
PIO (HLLX);RUNTEST 1 TCK;   // CAPTURE-DR
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-IR
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[0] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[1] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[2] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[3] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[4] - 0
PIO (HLLX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[5] - 0
PIO (HLHX);RUNTEST 1 TCK;   // SHIFT-DR & TDI[6] - 0
PIO (HHLX);RUNTEST 1 TCK;   // EXIT1-DR & TDI[7] - 0
PIO (HHLX);RUNTEST 1 TCK;   // EXIT1-DR & TDI[8] - 0
PIO (HHLX);RUNTEST 1 TCK;   // EXIT1-DR & TDI[9] - 0
PIO (HHLX);RUNTEST 1 TCK;   // EXIT1-DR & TDI[10] - 1
PIO (HHLX);RUNTEST 1 TCK;   // EXIT1-DR & TDI[11] - 0
PIO (HHLX);RUNTEST 1 TCK;   // UPDATE-DR
PIO (HLLX);RUNTEST 1 TCK;   // RUN-TEST IDLE
PIO (HLLX);RUNTEST 1 TCK;   // RUN-TEST IDLE
```

Figure 5: Generated SVF file

The serial vector format also can be converted back to higher level excel or text-based IR-DR sequence file if converter key is set to “svf_irdr”.



```
TRST: JTAG_RESET
TDI : JTAG_TDI
TMS : JTAG_TMS
TCK : JTAG_TCK
TDO : JTAG_TDO

WRITE_IR : 8 00010010
WRITE_DR : 12 010000000000

WRITE_IR : 8 01010000
WRITE_DR : 8 00000010
```

Figure 6: Generated text-based IR-DR sequences

Advantages

- DFT engineers can spent their valuable time in debugging real design issues rather debugging the test environment development with device setup.
- The generated .svf file is useful for other DFT tests such as MBIST/BSCAN/EFUSE/PLL pattern generation and ‘. testproc’ file is used to generate DFT SCAN patterns.
- Instruction mnemonics can be added to the pattern which helps test engineer to modify vectors (only device setup) during tester time if required.
- The converter only takes few milliseconds (~32ms) for hundred IR-DR sequences.
- This converter helps to generate functional vectors also If any of the functional debugs with the TAP controller.

Application in an industrial SoC

- Sequence generator is examined in an SoC were the chip has security unlock which needs to be unlocked using N number of instruction and data registers to enable the system’s test mode.
- Also, for scan and other test modes of operation, the chip needs to be initialized with appropriate state and with certain conditions via hierarchical TAP network.

Future Scope

- This could be expanded for SPI, FCSI, and any standard protocol-based sequences.
- The program conversion to a GUI based development with python is in progress to accommodate daisy chain, WTAP, iJTAG configurations with visual representations.

References

1) JTAG Basics

https://embeddedinn.xyz/articles/tutorial/JTAG-basics-and-usage-in-microcontroller-debugging/#:~:text=Introduction,and%2520Boundary%252DScan%2520Architecture%25E2%2580%259D

2) IEEE Standard for Test Access Port and Boundary-Scan Architecture

https://ieeexplore.ieee.org/document/6759738

3) Tessent® Shell User’s Manual Version 2020.1

TESSOLVE
A **Hero Electronix** Venture

Thank You !

Silicon Valley:

TessolveDTS Inc., 226 Airport Parkway,
Suite 300, San Jose, CA 95110
Tel : +1 408-865-0873
Fax : +1 408-865-0896
Email : sales@tessolve.com

San Diego:

TessolveDTS Inc.,
12348 High Bluff Drive,
Suite 100, San Diego, CA 92130.
Email : sales@tessolve.com

Richardson, TX:

TessolveDTS Inc.,
1702 N. Collins Boulevard, Suite 161,
Richardson, TX 75080
Tel : +1 972-290-0172
Fax: +1 972-437-9348
Email : sales@tessolve.com

Bengaluru:

Tessolve Semiconductor Pvt. Ltd.,
Plot No: 31 (P2), Electronic City Phase II,
Bangalore – 560 100, Karnataka, India.
Tel : +91 80 4181 2626,
+91 80 6816 2626
Email : sales@tessolve.com

Bengaluru:

Tessolve Semiconductor Private Limited
Indiqube South Island
Sy.No.32, Marenahalli 2nd Phase
JP Nagar, 24th Main, Ward No.177
Bangalore 560 078
Tel : + 91 80 66995800
Fax : +91 80 2668 6460
Email : sales@tessolve.com

Bengaluru:

Tessolve Semiconductor Pvt. Ltd.
Building Name: Smart Works (6th Floor)
Global Technology Park, Block "C"
Marathahalli Outer Ring Road,
Devarabeesanahalli Village, Bellandur,
Bengaluru- 560103
Email – sales@tessolve.com

Visakhapatnam:

Tessolve Semiconductor Pvt. Ltd.,
MVR's Vinayagar Trade center,
5th Floor, A-wing, VIP Rd, Asilmetta,
Visakhapatnam, Andhra Pradesh 530003.
Tel : 0891- 4805222, 0891- 4805333
Email : sales@tessolve.com

Coimbatore:

Tessolve Semiconductor Pvt. Ltd.,
Excellence – 5th Floor,
104, Race course road,
Coimbatore – 641018,
Tamilnadu, India.
Tel : +91 422 2221188, +91 422 2221199
Email : sales@tessolve.com

Tessolve Engineering Services Pte Ltd.,

Blk 20, Woodlands Link, #06/20
Singapore 738733
Tel : +65 6297-9613
Email : sales.sg@tessolve.com

**Tessolve Semiconductor Sdn.
Bhd. (1143041-M)**

Koridor Utara Malaysia,
Plot 36, Hilir Sungai Keluang 2,
Kawasan Perusahaan Bayan Lepas Fasa 4,
11900 Bayan Lepas, Pulau Pinang,
Office: +604 637 0672
Mobile: +6012 528 9996

Tessolve Semiconductor GmbH.,

Wöhlerstraße 29
30163 Hannover, Germany.
Email : hr_gmbh@tessolve.com
info_gmbh@tessolve.com

**Tessolve (Shanghai)
Engineering Services Pte. Ltd.**

667 Ziwei Road, Pudong New Area,
BLDG 172, RM 102
Shanghai, China 201210
Tel: +86-13127643713