# A Graph Theory Approach towards IJTAG Security via Controlled Scan Chain Isolation

Abhishek Das and Nur A. Touba
Computer Engineering Research Center,
University of Texas at Austin, TX 78712
*abhishekdas@utexas.edu, touba@ece.utexas.edu*

***Abstract*— The IEEE Std. 1687 (IJTAG) was designed to provide on-chip access to the various embedded instruments (e.g. built-in self test, sensors, etc.) in complex system-on-chip designs. IJTAG facilitates access to on-chip instruments from third party intellectual property providers with hidden test-data registers. Although access to on-chip instruments provides valuable data specifically for debug and diagnosis, it can potentially expose the design to untrusted sources and instruments that can sniff and possibly manipulate the data that is being shifted through the IJTAG network. This paper provides a comprehensive protection scheme against data sniffing and data integrity attacks by selectively isolating the data flowing through the IJTAG network. The proposed scheme is modeled as a graph coloring problem to optimize the number of isolation signals required to protect the design. It is shown that combining the proposed approach with other existing schemes can also bolster the security against unauthorized user access as well. The proposed countermeasure is shown to add minimal overhead in terms of area and power consumption.**

***Keywords*— *IEEE Std 1687, IJTAG, IJTAG security, design for test, scan chain***

## 1. INTRODUCTION

The complexity and density of integrated circuits (IC) have been rapidly increasing over the last decade. Higher density and smaller feature sizes have accelerated the growing complexity of an IC by enabling it to add more features. Today, many on-chip embedded instruments are used to meet the growing demands of time to market and chip quality. These embedded instruments facilitate test, debug and diagnosis. A few of the embedded instruments include trace buffers, test and debug controllers, physical sensors and embedded logic analyzers.

The IEEE Std 1149.1, also known as JTAG, defines the use of a test access port (TAP) to interface between the instruments using scan techniques. The growing complexity of circuits led to the development of IEEE Std 1687, also known as IJTAG, which allowed dynamic reconfiguration of the scan chain used to access the on-chip instruments. The dynamic reconfiguration of the scan chain is controlled by data shifted through the scan network using a Segment Insertion Bit (SIB). A SIB is a special scan cell which simply opens all the test data registers (TDRs) behind it, or opens a new scan segment, if the enable logic state is clocked into its update register. A simple IJTAG network with two instruments is shown in Fig. 1.

The IEEE Std 1687 uses the same TAP controller mechanism as an IEEE Std 1149.1 as a master TAP controller to control the IJTAG network. The TAP controller has these major phases. During shift-in phase, data is brought serially from an input pin and shifted through the scan segments across multiple clock cycles. The capture phase involves loading the outputs of an embedded instrument onto the test data registers (TDR) of the scan segment. And finally, the shift-out phase involves shifting out the data of the TDRs from the scan segment serially through an output pin across multiple clock cycles. The number of clock cycles needed to shift-in data and shift-out data depends on the length of the scan segment or in other words, the total number of TDRs in the chain. Apart from input test data and output captured data, the configuration bits from the TAP controller is also shifted through the IJTAG network which determines which SIBs are opened such that their scan segment or TDRs become accessible for testing and debug. The update phase is used to load the test vectors into the TDRs of the embedded instruments and the configuration bits into the SIBs.

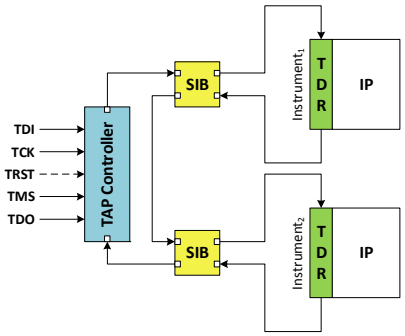


**Fig. 1.** IJTAG Network with two instruments.

Consider the Mingle IJTAG network from [Tšertov 16] as shown in Fig. 2 with 8 embedded instruments (W1 - W8) and 10 SIBs. The initial configuration is a scan chain with just two registers SIB1 and SIB2. Thus, the network path is SDI → SIB1 → SIB2 → SDO. If "10" is shifted in with the LSB shifted in first and then subsequent bits are shifted, during the update phase, then SIB1 becomes transparent such that SIB5 and SCB3 become accessible. We assume that during initial reset all registers are set to 0, thus SCB3 currently being 0 selects the output from SIB5 in SMUX3. Now, the network path becomes SDI → SIB5 → SCB3 → SIB1 → SIB2 → SDO. Next, we consider the scenario where we want to access instruments W7 and W8 for testing purposes. Shifting-in "1110" in the update phase serves two purposes. The SCB3 register updates to 1 thus

978-1-7281-1170-4/19/$31.00 ©2019 IEEE

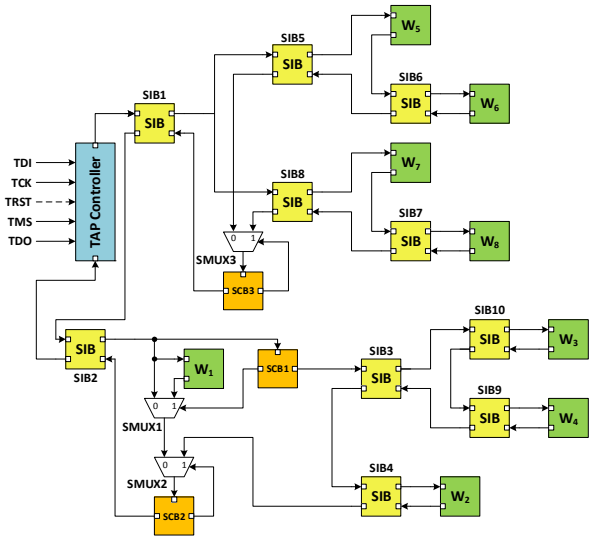


**Fig. 2.** IJTAG Network of Mingle benchmark.

selecting SIB8 as an input now, and SIB8 also receives a 1 which makes instrument W7 accessible. Thus, the current network path becomes SDI → W7 → SIB7 → SIB8 → SCB3 → SIB1 → SIB2 → SDO. If we assume that all instruments have their scan segment length as n, shifting-in "11110" followed by n zeroes also opens SIB7 thus making instrument W8 accessible. Thus, the final network path becomes SDI → W7 → W8 → SIB7 → SIB8 → SCB3 → SIB1 → SIB2 → SDO exposing the scan segments of instruments W7 and W8.

This improved accessibility of on-chip instruments comes at a cost of decreased security and safety of the on-chip embedded instruments. The embedded instruments generally have a wrapper with control and scan ports. The IJTAG network interfaces with these ports and thus the embedded instruments behave as a black box for the IJTAG integrator. The chip can have various level of vulnerabilities ranging from unauthorized access in the field during in-field operations to data tampering by an IP within the chip itself for numerous malicious intents. Since the embedded instruments are not exposed to the IJTAG integrator, any malicious logic hidden inside an instrument can potentially compromise the data being shifted through it. This paper proposes a new mechanism to prevent data modification and data sniffing by malicious embedded instruments. The proposed scheme can also be used to add an extra layer of protection against unauthorized access as well by increasing the complexity needed to access the chip. The rest of the paper is organized as follows. Section 2 describes the various security vulnerabilities in IEEE Std 1687 and the previous works that addresses these vulnerabilities. Section 3 describes the proposed scheme and analyzes its security features and hardware overhead. Section 4 evaluates the proposed scheme based on different hardware metrics. Finally, Sec. 5 presents the conclusion of this work.

## 2. Security Vulnerabilities and Previous Work

Attacks on IEEE Std 1687 can be classified into either of the following three broad categories.

- Unauthorized access
- Data transmission attack by an instrument in the scan chain
- Data sniffing by an instrument in the scan chain.

Each of these categories is further described in detail.

### *2.1 Unauthorized access*

In an unsecure IEEE 1687 scheme, an attacker who suspects that a chip contains an IEEE Std 1687 network can easily shift-in bits and ensure that each scan cell in the chain received a 1 (which would open a SIB, thus adding more scan chains in the path) at some point or another. This would increase the length of the scan chain and would be observed if a certain known sequence was shifted in. Such attacks are generally more dangerous when trace buffer or scan chain data can potentially be used to hack software encryption applications. Another scenario may be a potential theft or cloning of a chip ID or its access mechanism which could lead to production of counterfeit chips. This unauthorized access security vulnerability requires accessibility which is possible during in-field operations and sometimes maintenance as well.

In order to address the problem of unauthorized access, much work has been done to prevent such an attack. The most common solution is to use some kind of locking mechanism which doesn't allow a SIB to open until a certain key is identified along with the update value. [Dworak 13] proposed a Locking SIB (LSIB) that uses data scanned in through the network as key bits. These key bits must be set to a correct value for a corresponding LSIB to open. [Dworak 14] then introduced the concept of honeytraps and misdirection to make it even harder to unlock an LSIB. Obfuscation strategies are provided which delay or prevent the opening of an LSIB by increasing the search space that the attacker has to explore. [Liu 15] used an LFSR to generate a secret key which is used to open a SIB. This increases the difficulty of unauthorized access further due to the dynamic nature of the keys. [Baranowski 15] proposed fine-grained access management in reconfigurable scan networks (RSN). In this approach, each instrument is associated with a secret key. To open a set of instruments, the requesting entity is expected to know all respective secret keys. Thus, a challenge-response pair using a one-way hash function from an authorization instrument is used to prevent unauthorized access. The level of security provided is dependent on the size of challenge-response pairs.

### *2.2 Data Transmission Attack*

A data transmission attack is where an instrument on the chip itself can be malicious. A malicious instrument with its scan segment opened and connected for access can potentially alter configuration bits, input data as well as the output data being shifted through its scan segment. One type of attack possible by a malicious instrument is modifying the configuration bits that are being shifted through its scan segment. This can potentially result in either opening up new scan segments or not allowing access to instruments being tested. An outcome for this scenario can be test failures introduced by incorrect total scan length or

incorrect instruments being tested. This can lead to a perfectly healthy chip being rejected. Another consequence can be an increased time to market since there can be efforts to address the design functionality even though functionality is fine. The second type of attack relates to modifying the scan data being shifted through the scan segment of the malicious instrument. The output scan data from the capture of logic upstream or the input scan data to downstream logic can be modified, thus altering the output response or input test data. A consequence for such an action by a malicious instrument can be rejection of healthy chips as described previously. But another consequence of such a malicious action can also be failing to filter out defective chips. This has a much larger consequence since defective chips then can possibly fail in the field, thus harming customer relations or the overall reputation of the chip manufacturer or integrator.

[Kochte 17] proposed a scan pattern generation method to generate trustworthy access to sensitive scan segments. A bypass scan segment and a bypass MUX is used in this approach so that data moves through the bypass scan segment instead of the scan segment of the malicious instrument when the instrument is not trusted. The control signals from the scan segment is masked using a masking control MUX as well. An example of such a scheme is shown in Fig. 3. [Raiola 18] extended this model to construct a secure RSN based on user-defined cost functions. [Elnaggar 18] proposed a shadow scan segment with information flow tracking to compare the data moving through both the scan segment and the shadow segment and mark data as tainted if the data is different. The tainted data through various instruments are shifted out to identify the modified bits during an attack. In comparison, the proposed scheme does not use any additional scan segments or registers.

### *2.3 Data Sniffing*

Data sniffing simply refers to keeping track of the data flowing through an instrument's scan segments and making some kind of inference. Sensitive data related to software encryption algorithms or secret keys for various SIBs has the potential to be leaked through data sniffing. [Kan 16] proposed an approach with dual cipher streams and stub chain insertion to address this issue. The dual stream cipher obfuscates both the control sequences and the data access. The stub chain insertion into test data registers further aids in obfuscating the network topology.

## 3. PROPOSED SCHEME

The proposed scheme relies on the simple premise that if the scan segments of potentially malicious instruments are isolated such that no scan data passes through them, then it guarantees security against both data transmission attack as well as data sniffing. The proposed scheme assumes that the IEEE Std 1687 integrator is trustworthy. The isolation is achieved by means of an enable (disable) signal, a MUX and clock gating logic. An example of such a scheme for two connected scan segments of different instruments is shown in Fig. 4. There are two disable signals, $dis_1$ and $dis_2$, for the two scan segments. The combination of assertion and de-assertion of these two disable signals during the 8 cycles of shift-in and another 8 cycles of shift-out can guarantee that the scan segments $SS_1$ and $SS_2$ do not see each other's data. This prevents any malicious intent of modifying or sniffing each other's data. The only test data they see are the data that is meant specifically for their own scan segment.

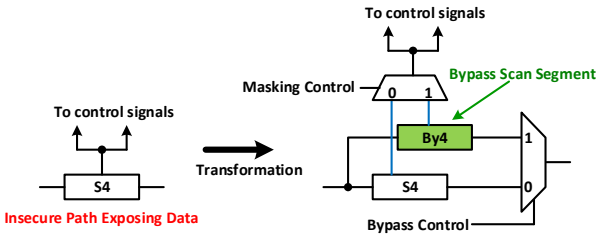


**Fig. 3.** Example Transformation of untrusted segment S4 as proposed in [Kochte 17].

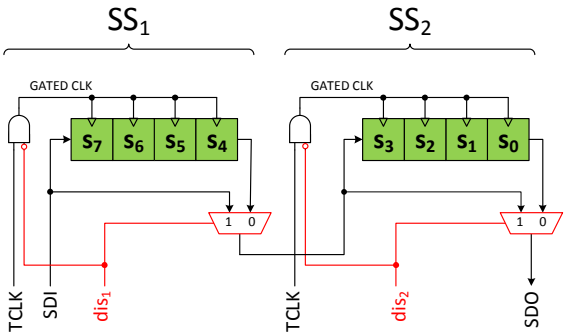


**Fig. 4.** Proposed Scheme with two scan segments and two disable ($dis_1$ and $dis_2$) signals

Scan segments can be isolated from each other through assertion and de-assertion of the two disable signals. Consider the example in Fig. 4, during shift-in $dis_1$ can be asserted to prevent the scan segment $SS_1$ to shift data, by gating the clock of the scan segments. During this phase, the MUX allows data to bypass scan segment $SS_1$ and get directly captured by scan segment $SS_2$. Thus, for the 4 cycles that $dis_1$ is kept high and $dis_2$ is kept low, the shift-in data is shifted directly to the scan segment $SS_2$. For the next 4 cycles, $dis_1$ can be kept low and $dis_2$ kept high. This gates off the clock for scan segment $SS_2$ preventing their data from shifting further. The incoming data is shifted into the scan segment $SS_1$ in these 4 cycles. Thus, in these 8 cycles the data was shifted into both the scan segments without scan segment $SS_1$ ever seeing what data is going into $SS_2$. Then capture cycle(s) can be initiated which triggers the functional logic and the functional outputs are captured in the scan segments. During shift-out, $dis_1$ can again be asserted first to gate-off the clock for scan segment $SS_1$. This ensures that the captured data in scan segment $SS_1$ stays within the scan segment while the captured data in scan segment $SS_2$ is being shifted out. After 4 cycles, $dis_1$ can be de-asserted and $dis_2$ can be asserted. This gates off the clock of scan segment $SS_2$ and enables the MUX bypassing $SS_2$. Thus, data from scan segment $SS_1$ is directly fed into the scan data out (SDO) pin without shifting

through scan segment $SS_2$.

In essence, the proposed scheme can be thought of as instrument groups having their own separate scan chain isolated from the scan segments in the rest of the chip. This enables testing of different instruments including malicious instruments in conjunction with other instruments without the worry of data being tampered or sniffed by malicious instruments. Also, compared to instruments simply having their own isolated scan chains, the proposed scheme is better because it needs only a single signal (enable/disable) for each potentially malicious group of instruments compared to multiple signals or potentially multiple test access port controllers. Furthermore, no changes need to be made to the shift in data, since the use of the isolation signal basically mimics the behavior of different scan chains within a single large connected chain.

*3.1 Isolation Signal Optimization*

Different instruments connected using the IJTAG standard will have different requirements of security based on the purpose of the embedded instruments. For example, a cryptographic instrument involved in generating essential keys for the security of the design ideally should not have its data passed through any other instruments since it can potentially be manipulated or sniffed for information. But it is possible that it needs to be tested in conjunction with other instruments. Similarly, trace buffers might contain important information that ideally should be avoided from malicious third-party IPs but is okay to shift-out its data from trusted instruments instead. Meanwhile, some instruments do not require any security and the data in it can basically be shifted through multiple IPs without any risk of security breach. These different security relationships can be used to optimize the number of isolation (disable/enable) signals in the design instead of naively using an isolation signal for each instrument.

This optimization problem can be modeled as a simple graph coloring problem. The definitions, construction and solution to the problem have been described below.

***Definition:*** A vertex in graph *G* represents an instrument in the design. Thus, all instruments have their individual vertices.

***Definition:*** An edge in graph *G* represents a security risk between the vertices (instruments) that it is connected to. Thus, if two instruments pose a security threat to each other's data and need to be isolated w.r.t each other, an edge needs to be drawn between the corresponding vertices. Instruments that are trustworthy w.r.t each other should not have an edge between their corresponding vertices.

***Proposition:*** The minimum number of colors required to color the vertices of graph *G* such that no two vertices connected by an edge have the same color, represents the optimal number of isolation signals.

The proof of the proposition is quite simple. Since all vertices that are connected pose a security threat to each other, they should ideally have their own isolation signals. Now it is possible that vertices with same color have different isolation signals, but that simply wastes resources. But vertices with different colors should never have the same isolation signals, since it then implies that data of instruments posing a security risk to each other passes through each of the instruments, defeating the purpose of isolation. The graph coloring problem is a NP complete problem and heuristic procedures can be used to find a good, but not necessarily optimal, solution [Kleinberg 06]. For a graph *G* with *n* vertices and *m* edges, a simple greedy algorithm is shown in Fig. 5.

1. *Q*: Queue of all vertices in Graph
2. Color first vertex $V_1$ with a color *NC*
3. Add color *NC* to the list *ColorsUsed*
4. For each remaining vertex *V* in *Q*
   - 4.1 Check all vertices connected to *V*
   - 4.2 Color *V* with a color C from *ColorsUsed* such that none of the vertices connected to *V* are colored *C* .
   - 4.3 If all colors from *ColorsUsed* are in use by connected vertices, assign a new color to *V*.
   - 4.4 Update *ColorsUsed*
5. *ColorsUsed*: Final list of all the colors used

**Fig. 5.** A simple Graph coloring algorithm.

Instruments marked with the same color in the Graph coloring problem are isolated with the same isolation signal. We consider an example from the Mingle benchmark shown in Fig. 2. We consider the case where all SIBs are opened i.e. all instruments need to be accessed and tested. Let $W_1$ represent part of a cryptographic engine which needs to be completely isolated from all other instruments. Let instruments $W_2$, $W_3$ and $W_4$ be from a common source such that they do not pose a threat to each other but can potentially corrupt all other instrument's data stream. Let the remaining instruments be from a trusted source with no security risk. The graph constructed using these security relationships is shown in Fig. 6. As seen from Fig. 6, the minimum number of colors required is 3. Thus, the optimal number of isolation signals required is 3.

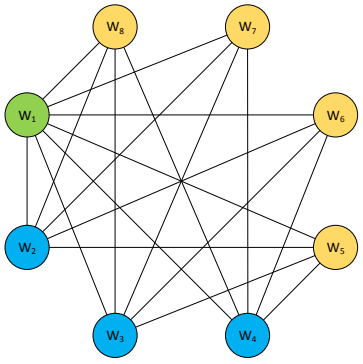


**Fig. 6.** Colored graph for Mingle benchmark.

The security relationships of the different instruments or the importance of an instrument is based on the IJTAG integrator's decision. The proposed scheme assumes that the IJTAG integrator is trustworthy and will be able to derive the data

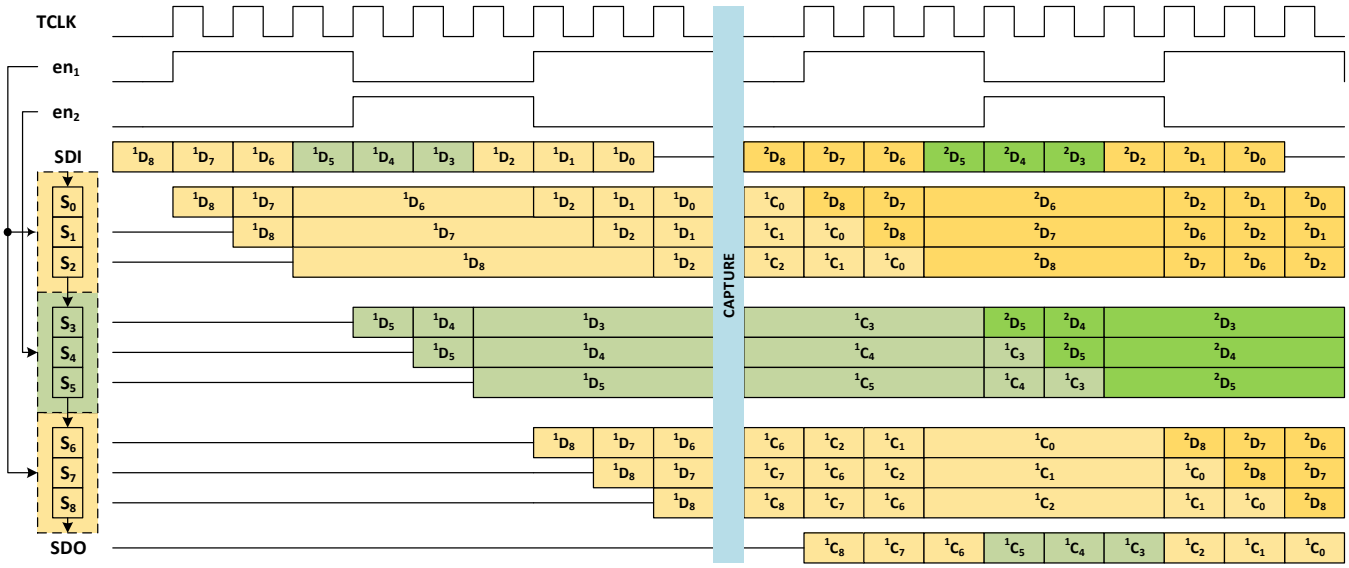


**Fig. 7.** Example of data shifts using scan isolation for a design with 3 scan segments and 2 isolation signals.

segments that each instrument can have trusted access to. Based on this trusted access relationship of instruments, the number of isolation signals can be found out using the graph coloring procedure described previously. These isolation signals need to be integrated through custom tooling by the IJTAG integrator and are not part of the IJTAG instruction itself. This is for easy control of the scan data being shifted in and out of the instruments. This also provides the IJTAG integrator added flexibility on how the test data can be shifted through efficiently in as less time as possible.

The isolation signal with the clock gating logic and the MUX can easily be built into the TDR itself as an added security measure and to allow convenient integration. Additionally, an enable signal generation logic can be placed by the IJTAG integrator which basically generates the different isolation signals based on the length of the different scan segments and the security relationship between the scan segments.

### *3.2 Security Analysis*

The proposed scheme guarantees protection from both data transmission attacks as well as data sniffing since malicious instruments simply see data intended for them in their scan segment and no other data passes through their scan segments. Thus, there aren't any means by which data can be tampered or sniffed by a malicious instrument for the proposed scheme. The isolation provided by the proposed scheme thus prevents any kind of data integrity or data sniffing attacks. One thing to note is that the order of the data being shifted in/out doesn't change and shift-outs can still be overlapped with shift-ins to reduce the number of shift cycles. An example of data shift-in and shift-out from 3 scan chains with two isolation signals is shown in Fig. 7. In this case, for ${}^{x}D_{y}$, $x$ represents the set of data being shifted-in and $y$ represents the corresponding TDR it is meant for. Similarly, for ${}^{p}C_{q}$, $p$ represents the set of outputs being shifted-out and $q$ represents the corresponding TDR that produced it.

Now, it is possible that the potentially malicious instruments can simply be kept off the scan chain by keeping the SIB, that gives access to it, closed. The main issue with this approach is that it prevents testing of such instruments in conjunction with other instruments. This might specifically impede debug and repair wherein such tests might be required to diagnose different faults and its effect on other instruments.

The schemes preventing unauthorized access in [Dworak 13], [Dworak 14], [Baranowski 15] and [Liu 15] are orthogonal to the proposed scheme. Thus, they can be used in tandem with the proposed scheme to avoid unauthorized access. The proposed scheme can potentially increase the difficulty of unauthorized access by simply changing the active polarity of the enable (disable) signals for any random number of scan segments. For instance, in the example of Fig. 4, instead of using $dis_2$ to disable the scan $SS_2$, the polarity of the disable signal can be reversed such that the clock gating and bypass of scan segment occurs when $dis_2$ is de-asserted instead of being asserted. Thus, for an unauthorized user trying to attack the system, they also have to guess the correct combination of which enable signals are active low and which enable signals are active high to gather any useful information. The random combination of active-high and active-low enable (disable) signals can dissuade unauthorized access since depending on the number of instruments isolated, the total number of combinations to guess for even a decipherable change can be high for large circuits. If $b$ isolation signals are being used in a design, the worst-case number of combinations will be $2^b$.

## 4. EVALUATION

The proposed scheme was implemented for the ITC 2016 benchmarks [Tšertov 16]. The benchmarks were modified accordingly to accurately reflect the proposed scheme and were synthesized using the NCSU FreePDK45 45nm library. The area overhead and power overhead results with all the instruments

**Table I:** Area overhead and power overhead of proposed scheme.

| Benchmark | Original Results | | | With Proposed Scheme | | | Overhead | | |
|---|---|---|---|---|---|---|---|---|---|
| | *Area ($\mu m^2$)* | *$P_{dyn}$ (mW)* | *$P_{leak}$ ($\mu W$)* | *Area ($\mu m^2$)* | *$P_{dyn}$ (mW)* | *$P_{leak}$ ($\mu W$)* | *Area%* | *Pdyn%* | *Pleak%* |
| BasicSCB | 9733.28 | 0.932476 | 66.7288 | 9826.67 | 1.1397 | 67.6425 | 0.959492 | 22.223 | 1.369274 |
| Mingle | 14895.6 | 1.6603 | 102.7742 | 14975.36 | 1.8625 | 103.061 | 0.53546 | 12.17852 | 0.279058 |
| TreeFlat | 5989.7 | 0.544578 | 40.7322 | 6010.32 | 0.559815 | 40.8991 | 0.344258 | 2.798039 | 0.40975 |
| TrapOrFlap | 47184.36 | 2.5551 | 256.8188 | 47251 | 2.5697 | 257.1893 | 0.141233 | 0.571406 | 0.144265 |

wrapped with the proposed countermeasure, is shown in Table I. As can be seen from the table, the proposed scheme adds negligible area overhead. The dynamic and leakage power overhead are also very small for most of the benchmarks. Specifically, for the complex *TrapOrFlap* benchmark, the power overhead is almost negligible.

For the proposed countermeasure, all instruments need to be wrapped with the additional clock-gate and MUX as discussed in Sec. 3. The only variable is the number of isolation signals which can be optimized as discussed in Sec. 3.1. The cost of adding the proposed countermeasure to each instrument is minimal as seen from Table I. Compared to schemes like [Kochte 17] and [Elnaggar 18], the proposed scheme does not use any additional scan registers at all, which reduces both the area and power overhead. The only overhead the proposed scheme incurs is in terms of an additional pin for the isolation signal in each scan segment as well as the different isolation signals that need to be routed.

## 5. CONCLUSIONS

In this paper, a new comprehensive approach to IJTAG security is proposed through isolation of scan chain data during shift-in and shift-out phase. The proposed scheme prevents any type of data integrity or manipulation attack and data sniffing attack by isolating the data through such potentially malicious instruments and not letting it pass through their scan chains. The proposed approach adds minimal hardware area and power overhead to the existing design. The optimum number of isolation signals required to protect the design against malicious instruments is modeled as a graph coloring problem. Combined with other unauthorized access prevention schemes, the proposed approach adds to their layer of security by increasing the complexity required to make an unauthorized access. Thus, the proposed scheme is highly suitable for modern complex system-on-chip design with IEEE Std. 1687, deterring any kind of malicious attack.

## ACKNOWLEDGMENTS

This research was supported in part by the National Science Foundation under Grant No. CCF-1617665.

## REFERENCES

[Baranowski 15] R. Baranowski, M. A. Kochte and H. J. Wunderlich, "Fine-Grained Access Management in Reconfigurable Scan Networks," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 6, pp. 937 – 946, Jun. 2015.

[Dworak 13] J. Dworak, A. Crouch, J. Potter, A. Zygmontowicz and M. Thornton, "Don't forget to lock your SIB: hiding instruments using P1687," in *Proc. of IEEE International Test Conference (ITC)*, paper 6.2, 2013.

[Dworak 14] A. Zygmontowicz, J. Dworak, A. Crouch and J. Potter, "Making it harder to unlock an LSIB: Honeytraps and misdirection in a P1687 network," in *Proc. of Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1-6, 2014.

[Elnaggar 18] R. Elnaggar, R. Karri and K. Chakrabarty, "Securing IJTAG against data-integrity attacks," in *Proc. of IEEE VLSI Test Symposium (VTS)*, pp. 1 – 6, 2018.

[Kan 16] S. Kan, J. Dworak and J. G. Dunham, "Echeloned IJTAG data protection," in *Proc. of IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*, pp. 1 – 6, 2016.

[Kleinberg 06] J. Kleinberg and E. Tardos, *Algortihm Design*. Boston, MA, USA: Pearson/Addison-Wesley, 2006.

[Kochte 17] M. A. Kochte, R. Baranowski and H. J. Wunderlich, "Trustworthy reconfigurable access to on-chip infrastructure," in *Proc. of IEEE International Test Conference in Asia (ITC-Asia)*, pp. 119 – 124, 2017.

[Liu 15] H. Liu and V. D. Agrawal, "Securing IEEE 1687-2014 Standard Instrumentation Access by LFSR Key," in *Proc. of IEEE Asian Test Symposium (ATS)*, pp. 91-96, 2015.

[Raiola 18] P. Raiola, M. A. Kochte, A. Atteya, L. R. Gómez, H-J Wunderlich, B. Becker and M. Sauer, "Detecting and Resolving Security Violations in Reconfigurable Scan Networks," in *Proc. of IEEE International Symposium on On-Line Testing And Robust System Design (IOLTS)*, pp. 91-96, 2018.

[Tšertov 16] A. Tšertov, A. Jutman, S. Devadze, M. S. Reorda, E. Larsson, F. G. Zadegan, R. Cantoro, M. Montazeri and R. Krenz-Baath, "A suite of IEEE 1687 benchmark networks," in *Proc. of IEEE International Test Conference (ITC)*, paper 6.1, 2016.