ISSN 2186-5647

2-23

RTL におけるアンチ SAT に基づくコントローラの論理暗号化の評価

日大生産工(学部) ○辻川 敦也　日大生産工 細川 利典　京産大 吉村 正義

1. はじめに

近年，大規模集積回路(Large Scale Integrated circuits: LSI)の大規模化に伴い，LSI 設計会社 1 社のみで LSI の設計を行うことが困難となっている．それゆえ，IP ベンダより IP コア(Intellectual Property Core)と呼ばれる LSI を構成するための部分的な回路情報を購入し，必要な部分のみを自社で設計する手法が多く用いられている．一方，IP コアは著作権侵害が容易に行えるという欠点を持つため，著作権保護を行う必要がある.このような IP コアの著作権保護の一手法として，論理暗号化手法(Logic Locking)[1]が提案されている．論理暗号化とは元の論理回路に鍵入力と呼ばれる外部入力を追加し，鍵入力に正しい値が設定された場合にのみ正しい動作を行うように回路を変更する手法である．

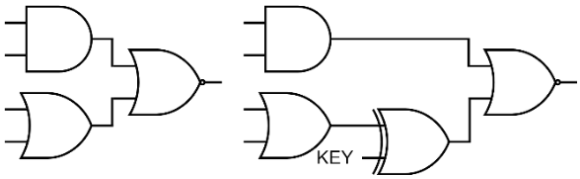
しかしながら，既存の論理暗号化手法[5][6]は SAT 攻撃によって容易に正しい鍵が特定されることが報告されている[2]．SAT 攻撃は，効率よく鍵探索空間を狭める識別入力パターン(Distinguishing Input Pattern: DIP)を繰り返し求めながら，正しい鍵を特定する攻撃アルゴリズムである．

SAT 攻撃に対する防御手法[3][4][9]がいくつか提案されている．この防御手法は，回路中にデコーダを挿入することで，1 回の攻撃で 1 つの鍵入力候補しか削除できないため，SAT 攻撃の鍵探索効率を著しく低下させることが可能である．一方，この防御手法は，暗号化部分がデコーダであるという構造的特徴を持つ．そのため，構造的特徴を特定してデコーダを削除することで，正しい鍵が特定される可能性がある．それゆえ，本論文では，レジスタ転送レベル(Register Transfer Level: RTL)において，コントローラ拡大[10][11][12]を用いることで，回路が構造的特徴を持たない SAT 攻撃に耐性のある論理暗号化手法を提案する．

2. 論理暗号化と SAT 攻撃

2.1 論理暗号化

論理暗号化が施された回路には新たに鍵入力が追加され，正しい鍵が挿入された時のみ，正しい動作をするように設計される．論理暗号化を用いた手法[1][5][6]はそれぞれのアルゴリズムに基づき指定された数だけ回路中の信号線に XOR/XNOR ゲートを挿入する．図 1(a)に元の回路，図 1(b)に XOR 挿入による論理暗号化を施した回路を示す．論理暗号化が施された回路には鍵入力 KEY が追加される。XOR ゲートに入力している鍵入力を'0'に，XNOR ゲートに入力している鍵入力を'1'にすることで暗号化された回路の動作を元の回路と同一にすることが可能である．XOR ゲートに'1'，XNOR ゲートに'0'を入力すると，出力はもう 1 つの入力の値を反転させた値となる．そのため，鍵として，XOR ゲートに'0'，XNOR ゲートに'1'を入力することで，元の回路と等価な値を伝搬させることが可能である．



(a)元の回路　　(b)暗号化済み回路

図 1. XOR ゲート挿入による論理暗号化

2.2 SAT 攻撃

近年，論理暗号化の安全性は SAT 攻撃によって脅かされている[2]．SAT 攻撃は，SAT ソルバを使用して鍵空間を削除することで正しい鍵を求める攻撃アルゴリズムである．SAT 攻撃の攻撃者は，論理暗号化された回路の正しい鍵を入手することが目的であり，VLSI 設計会社や VLSI 製造工場であると仮定する．攻撃者は，論理暗号化されたゲートレベルのネットリストであるオラクルと呼ばれる論理暗号化が解除された活性化済み VLSI を有する．

SAT 攻撃のアルゴリズムについて説明する．SAT 攻撃の入力は論理暗号化済みのゲートレベルのネットリストCとオラクルVであり，出力は正しい鍵入力K である．まず適当な方法で 1 つの入力パターンを求め，オラクルVを使用して，識別入力パターンXに対する正しい出力Yを求める．次に，論理暗号化済みのゲートレベルのネットリストCを 2 つ使用して，識別入力パターンXに対して 2 つの鍵入力k_1とk_2がそれぞれ異なる出力値Y_1とY_2を出力するという条件の CNF 式$F_i = C(X, k_1, Y_1) \wedge C(X, k_2, Y_2)$を作成し，SAT ソルバを使用して$X$，$k_1$，$k_2$を求める．求められた場合，求めた$X$に対する正しい出力値$Y$を求め，入出力応答を満たす鍵入力の条件を$F_i$に加える．求められなかった場合は求められたすべての識別入力パターンとその出力値のペアを満たす鍵入力を解とする．

この攻撃は，DIP を繰り返し求めて使用することによって不要な鍵空間を削除していく．DIP は，2 つの鍵入力k_1とk_2がそれぞれ異なる出力値を生成する入力値である．出力値が異なる場合，少なくとも 1 つの鍵入力が正しくないことがわかるため，1 つの DIP で複数の誤った鍵入力を除外することが可能である．

2.3 対策手法

SAT 攻撃を防御するために，回路中に Key Decoder を挿入する論理暗号化手法[3][4]が提案

An Evaluation of Logic Locking of Controllers Based on Anti-SAT at RTL

Atsuya TSUJIKAWA, Toshinori HOSOKAWA, and Masayoshi YOSHIMURA

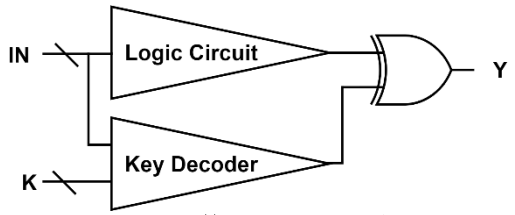


図 2．対策手法の論理構造

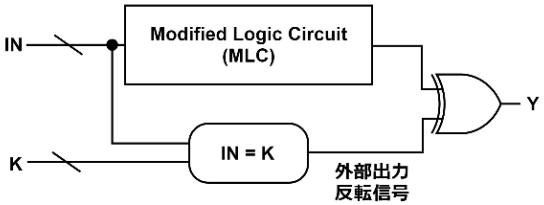


図 3．TTLock の論理構造

された．図 2 に対策手法の論理構造を示す．この論理暗号化手法は，1 つの識別入力パターンにより削除される鍵入力候補を 1 つ削減することに基づいている．外部入力と鍵入力を入力する Key Decoder 回路によって，どの外部入力と鍵入力の組合せが入力されると外部出力を反転させるかを制御する．Key Decoder によって，誤った外部出力値を出力する外部入力と鍵入力の組合せを 1 つのみに設計することで，SAT 攻撃への耐性を高める．

これらの論理暗号化手法は SAT 攻撃への耐性を高める一方，暗号化部分がデコーダ構造となるという構造的な特徴を持つため，構造的な特徴に対する攻撃への脆弱性を有する[7][8]. 外部出力値が誤っている時における信号線の 0 である確率と 1 である確率を求めることで，挿入された Key Decoder の出力信号線を特定することが可能であり，その信号線を解析することで，鍵入力の時に外部出力が反転するか否かを特定することができる．

SAT 攻撃と構造的な特徴に対する攻撃に耐性のある論理暗号化手法として，TTLock[9]が提案された．図 3 に TTLock の論理構造を示す．この論理暗号化手法は，修正済み論理回路(Modified Logic Circuit: MLC)とデコーダ回路である比較器の 2 つから構成される．MLC は外部入力パターンを 1 つ選択し，選択した入力パターンのみを反転するように修正した論理回路である．比較器では，入力された外部入力パターンと鍵入力値が一致するかを判定し，一致した場合は'1'を，一致しない場合は'0'を出力する．MLC で選択した外部入力の場合は，鍵と一致した場合正しい外部出力値を出力し，一致しない場合は反転した外部出力値を出力する．選択されなかった外部入力パターンの場合は，鍵と一致した場合反転した外部出力値を出力し，一致しない場合は，正しい外部出力値を出力する．それゆえ，MLC で出力が反転する外部入力パターンに割当てられた鍵入力は，正しい鍵入力と判断できる．

TTLock では，MLC をどのように生成するかが課題である．デコーダ部である比較器を特定されると，正しい鍵入力を特定される恐れがある．そのため，再度論理合成を行うことで，比較器の特定を困難にしている．

3．提案手法

既存の SAT 攻撃に耐性のある論理暗号化手法は回路に構造的特徴を持たせるため，構造的特徴を狙う攻撃手法に対して脆弱である．それゆえ，本章では，構造的特徴のない SAT 攻撃に耐性のある論理暗号化手法を提案する．本提案手法は，TTLock を 2 つの手法で改良することで，SAT 攻撃で現実時間内での正しい鍵の探索を困難にする論理暗号化手法である．1 つ目の手法は，RTL 回路での TTLock の実装である. 2 つ目の手法は，コントローラ拡大による鍵入力数の増加である．

また，コントローラは，論理合成によって，デコーダ回路となりやすい．それゆえ，コントローラに対して論理暗号化を行い，論理合成を行う．論理合成により生成されたゲートレベルの回路は，元のコントローラによるデコーダ構造と論理暗号化によるデコーダ構造の両方を持つため，論理暗号化部分のデコーダの特定が難しくなる．

3.1 RTL での TTLock の実装

本手法は RTL を対象とする論理暗号化手法であり，本手法で対象とする RTL 回路はコントローラとデータパスから構成されていると仮定する．コントローラとデータパスは互いに信号線で接続されている．コントローラからデータパスへの信号線は，制御信号線であり，データパスからコントローラへの信号線は，状態信号線である．コントローラは現在状態と状態信号に基づいて，次状態と制御信号を出力する．

TTLock は割当てた鍵が正しい鍵の場合は，鍵入力値と割当てた鍵が一致した場合に正しい出力値を出力し，一致しない場合は異なる出力値を出力する．割当てた鍵が不正な鍵の場合は，割当てた鍵と鍵入力が一致した場合に不正な出力値を出力し，一致しない場合は正しい出力値を出力する. 本手法では，これらの TTLock の概念を RTL で実装を行う．論理暗号化が施された RTL 回路は，コントローラに直接鍵入力が追加される．まず，コントローラの現在状態と状態信号のペアごとに異なる鍵入力を割当てる．次に，現在状態と状態信号のペアに対して，割当てられた鍵と誤った鍵入力値が一致しているか否かを判定する．もし割当てられた鍵入力値と一致した場合，論理暗号化前と異なる次状態と制御信号を出力する．もし割当てられた鍵と誤った鍵入力値が一致しない場合，通常の次状態と制御信号を出力する．ただし，割当てられた鍵が正しい鍵入力値である時，入力された鍵入力値が正しい鍵入力値と一致する場合，論理暗号化前と同じ次状態と制御信号を

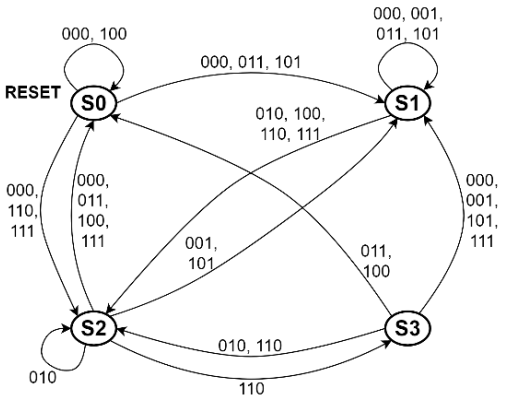


図 4. dk15 の状態遷移図

出力し，正しい鍵入力値と一致しない場合，論理暗号化前と異なる次状態と制御信号を出力する．

3.2 コントローラ拡大

鍵入力ビット数は暗号強度を左右するため，任意に設定されるべきである．しかしながら，本提案手法では，SAT 攻撃への耐性を高めるために，鍵入力はコントローラの状態遷移ごとに割当てられる．つまり，コントローラの状態遷移数が鍵入力数の上限となる．それゆえ，本手法では，コントローラ拡大を用いることで，状態数及び状態遷移数を増加させ，鍵入力数の増加を図る．コントローラ拡大とは，有限状態機械であるコントローラに任意の状態や状態遷移を追加する設計手法のことである．本手法で用いるコントローラ拡大は，拡大前後の入出力は同一である。また，内部状態のみ異なり，拡大前後で順序回路として等価な回路である．

以下の(1)-(5)に，本手法で行うコントローラ拡大の手順を示す．

(1)コントローラ全体を複製し，状態数を増加させる
(2)複製したコントローラごとにグループ ID を付ける
(3)元のコントローラから任意の状態遷移を 1 つ選択する
(4)グループ ID の円順列を生成する．
(5)(3)で選択された状態遷移の遷移先を，グループ ID が(4)で生成した円順列の次のグループの同名状態に変更する

本コントローラ拡大手法を MCNC ベンチマーク回路の dk15 を例にとって説明する．図 4 に拡大前の dk15 のコントローラ，図 5 に拡大後のコントローラを示す．このコントローラは，状態信号数が 3，状態数が 4 であり，状態レジスタビット幅は 2 である．したがって，鍵入力ビット幅は 5 となる．このコントローラに対して 7 ビットの鍵を入力することを考える．状態レジスタビット幅を 2 増やす必要があるため，コントローラを 4 つ(2^2)コピーする．4 つのコントローラにそれぞれのグループ ID(G0，G1，G2，G3)を付ける．次に任意のコントローラの状態遷移を 1 つ選択する．図 5 の例で，状態 S3 から S1 へ遷移する状態遷移を選択したとする．次に円順列 G0→G1→G2→G3→G0 を生成する．G0 の選択された状態遷移の遷移先を次のグループ G1 の同名状態となるように変更する．すべてのグループに

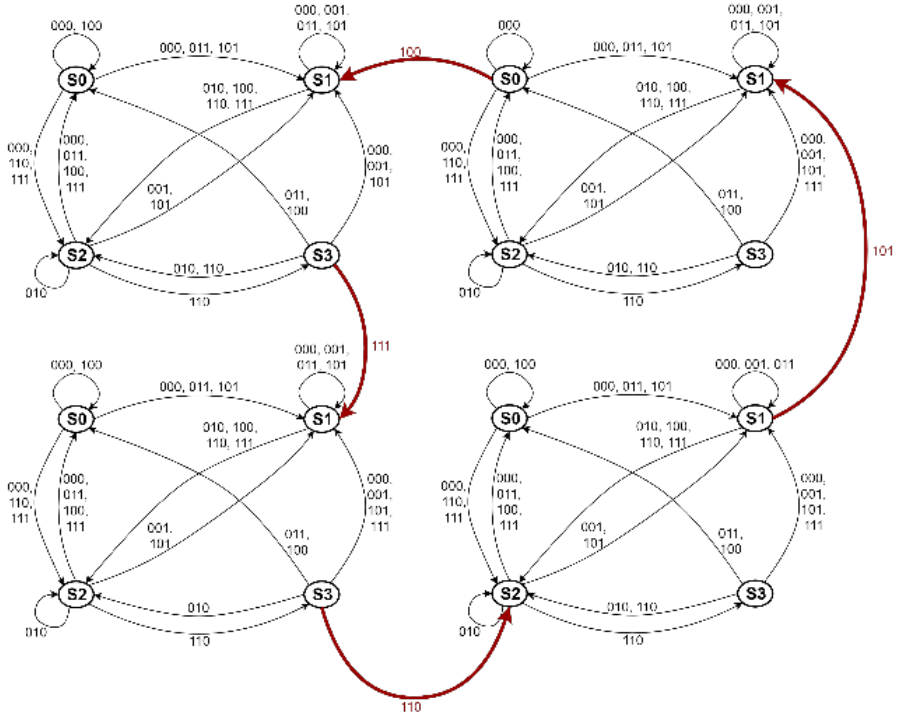


図 5. コントローラ拡大後の dk15 の状態遷移図

表 1. 論理暗号化結果

回路名	SAT 攻撃結果		信号線数	面積	面積オーバヘッド(%)
	実行回数	実行時間(s)			
dk15_RLL_32	12	1.32	3851	10,271	101
dk15_提案手法_8	256	4482.69	6801	11,625	114

対して，同様の処理を行う．これらの操作により状態数を 16 に増やし，7 ビットの鍵を入力することが可能となる。

4. 実験結果

本論文では，従来の論理暗号化手法と提案手法に対して，SAT 攻撃と構造的特徴の評価を行った．使用した回路は MCNC ベンチマーク回路と自作の 32 ビットデータパスを組合せた回路である．論理合成及び面積評価は Synopsys 社の Design Compiler を用いた．SAT 攻撃中の SAT ソルバは MiniSAT[13]を用いている．表 1 には dk15 の暗号化手法ごとの構造的特徴の評価， SAT 攻撃の評価を示す．実験結果の回路名は"オリジナル回路名_暗号化手法_鍵ビットの数"で表している．表中の実行回数，実行時間は、正しい鍵が求まるまでの SAT 攻撃の実行回数，実行時間である．面積オーバヘッドは，暗号化前の dk15 における論理合成後の回路の面積と比較したオーバヘッドである．

RLL[1]では 32 ビットの鍵の探索が 12 回，1.32 秒で求まった．対して提案手法では，鍵のビット幅が 8 ビットと従来手法より少ないにも関わらず正しい鍵の解読に 256 回，4482.69 秒を要するという結果になった．以上の結果から，提案手法の SAT 攻撃への耐性の有効性を示すことができた．しかし面積オーバヘッドが，RLL では鍵ビット幅 32 ビットで 101%なのに対し，提案手法では鍵ビット幅が 8 ビットであるのに 114%と大きくなってしまった．

5. おわりに

本論文では，コントローラ拡大を用いた SAT 攻撃に耐性のある論理暗号化手法を提案した．提案手法では正しい鍵の解読に実行回数 256 回という結果が得られたため，提案手法の有効性を示すことができた．今後の課題として鍵ビット数の拡大することと，拡大した際の面積オーバヘッドの削減が挙げられる

文献

[1] J.A.Roy, E Koushanfar, and I.L. Markov , “Ending pirace of integrated circuits , ” computer , vol43 , no10 , pp.30-38,Oct2010.

[2] P. Subramanyan, S. Ray, and S. Malik,“Evaluating the security of logic encryption algorithms,” in Hardware Oriented Security and Trust (HOST), 2015 IEEE International Symposium on. IEEE, 2015, pp.137-143.

[3] M. Yasin, B. Mazumder, J. J. Rajendran, and O. Sinanoglu , “ SARlock : SAT Attack Resistant Logic Locking , ”IEEE International Symposium on Hard Hardware Oriented Security and Embedded Systems, pp.127-146 , 2017.

[4] Y. Xie and A. Srivastava, “Mitigating SAT Attack on Logic Locking , ” International Conference on Cryptographic Hardware and Embedded System, pp.127-146, 2016.

[5] J. Rajendran, H. Zhang, C. Zhang, G. Rose, Y. Pino, O. Sinanoglu, and R. Karri, “Fault Analysis-Based Logic Encryption”, IEEE Transactions on Computers, vol. 64, no. 2, pp. 410-424, 2015.

[6] M. Yasin, J. Rajendran, O. Sinanoglu, and R. Karri, ,”On Improving the Security of Logic Locking”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and System, vol.35, no. 9, pp.1411-1424, 2016.

[7] M Yasin, B Mazumdar, O Snanoglu and J Rajendran ,”Removal Attacks on Logic Locking and Camouflaging Techniques ” , IEEE Asia and South Pacific Design Automation Conference 2017.

[8] M. Yasin, B. Mazumdar, O. Sinanoglu, and J. Rajendran, “ Security Analysis of Anti-SAT , ” IEEE Asia and South Pacifi Design Automation Conference, pp. 342-347, 2016.

[9] M. Yasin, A. Sengupta, B. C. Schafer, Y. Makris, O. Sinanoglu, and J. V. Rajendran. “What to lock? : Functional and parametric locking, ” In Proceedings of the on Great Lakes Symposium on VLSI 2017.

[10] T. Masuda, J. Nishimaki, T. Hosokawa and H. Fujiwara, “A Test Generation Method for Datapaths Using Easily Testable Functional Time Expansion Models and Controller Augmentation,” IEEE the 24th Asian Test Symposium (ATS’15), pp.37-42, Nov. 2015.

[11] S. Ohtake, T. Masuzawa, and H. Fujiwara, “A non-scan approach to DFT for Controllers Achieving 100% Fault Efficiency,” Journal of electronic Testing : Theory and Applications (JETTA), Vol. 16, No. 5, pp.553-566, Oct. 2000.

[12] L.M.Flottes, B.Rouzeyre, L.Volpe, “A Controller Resynthesis Based Methods for Improving Datapath Testability,” IEEE International Symposium on Circuits and Systems, pp. 347-350, May 2000.

[13] N. Een and N. Sorensson, “An extensible SAT-solver,” in 6th Int. Conf. on Theory and Applications of Satisfability Testing (SAT2003), pp. 502-518, 2003.