様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　元 年　 ６ 月 ２０ 日現在

機関番号：１３９０４
研究種目：基盤研究(C)（一般）
研究期間：2016～2018
課題番号：１６Ｋ００００７２

研究課題名（和文）ソフトウェアの論理回路化による知的財産保護に関する研究

研究課題名（英文）Protection of Intellectual Property by the logic circuit converted from software

研究代表者
市川　周一（Ichikawa, Shuichi）

豊橋技術科学大学・工学（系）研究科（研究院）・教授

研究者番号：７０２６２８５５

交付決定額（研究期間全体）：（直接経費）　　3,500,000円

研究成果の概要（和文）：ソフトウェアには貴重な知的財産が多く含まれており，その保護は極めて重要である．本研究ではソフトウェアの一部を論理回路化し，FPGA等の再構成可能論理回路に隠蔽する手法について検討した．本研究ではC言語プログラムの一部を高位合成(HLS)ツールにより論理回路に変換するが，コンパイラ基盤LLVM上に実装された汎用難読化ツールoLLVMを併用することにより難読化論理回路を生成できることを示した．ソフトウェアを難読化する別手法として，命令レジスタファイル(IRF)を利用してセキュアプロセッサ実現する手法についても検討・評価した．

研究成果の学術的意義や社会的意義
制御システムや組込みシステムには，企業の持つ技術的ノウハウが多く含まれている．しかしソフトウェアは複製や解析が容易であるため，知財の流出につながりやすい．流出した知財が競合製品に使用されると企業に多大な経済的損失をもたらすため，知財保護は極めて重要である．
本研究ではソフトウェアの一部を論理回路化し，FPGA等の再構成可能論理回路に隠蔽する手法について検討した．FPGAを用いることにより，ソフトウェアをハードウェア化してもシステムの柔軟性が保たれる．ソフトウェア難読化ツールを併用することで，難読化ハードウェアが生成可能であることを示した．

研究成果の概要（英文）：Software includes valuable intellectual property, and thus it is important to protect software from various analysis and plagiarism. In this study, the author proposed to conceal the important part of software by converting it into logic circuit. In this study, the C code is converted into the corresponding obfuscated logic circuit by using High Level Synthesis (HLS) with a general purpose obfuscation tool oLLVM. Another method to protect software, the secure processor implemented with Instruction Register File (IRF), is also proposed and examined.

研究分野：専用計算システムアーキテクチャ．特に，並列計算機方式，並列処理，FPGA応用，セキュリティ

キーワード：ソフトウェア保護　難読化　セキュアプロセッサ　FPGA　専用回路　制御プログラム　組込みソフトウェア

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９、ＣＫ－１９（共通）

１．研究開始当初の背景

制御システムや組込みシステムには，企業の持つ技術的ノウハウやトレードシークレットが含まれている．ある意味，企業にとってそれらの知的財産は製品自体より重要なものである．しかし，ソフトウェアは複製や解析が容易であるため，知的財産の流出につながりやすい．流出した知的財産が複製品や競合製品に使用されると，企業に多大な経済的損失をもたらす可能性がある．従って，こうした知的財産の保護は極めて重要な課題である．

ソフトウェアの保護に関する研究は，これまでも多く行われてきた．

- ソフトウェアの難読化

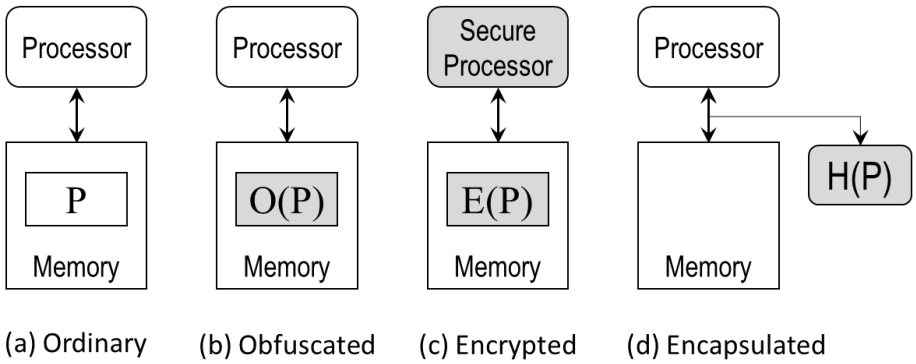
通常のシステムでプログラム P を実行すると（下図(a)），メモリ上の P を解析される恐れがある．ソフトウェアの機能を変えず，機能等価な変換を施すことにより，解析を難しくする手法を難読化とよぶ．下図(b)で，O(P)は難読化後のプログラム P を表す．

Barak(2012)は，プログラムを Virtual Black Box として Oracle Access が可能であれば，実行時間の二乗程度の時間でソースコードを再構築できると述べた．誤解を恐れず単純化すれば，復元が計算困難な難読化は存在しないという結論である．前提条件の変更を含め幾つかの後続研究が行われ，悲観論と楽観論が交錯しているが，たとえ復元困難な難読化がないとしても，解析に要する時間を増やして攻撃者の経済的動機を喪失させることは可能であり，実用上重要である．

- セキュアプロセッサ

ソフトウェア P をメモリ上で暗号化 E(P)すれば，復号鍵がない限り解析も実行も困難になる．しかし通常のプロセッサを使う限り，実行前にメモリ上で復号する必要があるので，復号後のソフトウェアを複製あるいは解析することができる．こうした攻撃を防ぐためには，プロセッサ内部にハードウェアサポートが必要である．セキュリティ保護機能を持つプロセッサをセキュアプロセッサという（下図(c)）．

セキュアプロセッサの研究も近年広く行われており，申請者も 10 年余り前から研究を続けている．セキュアプロセッサは将来的に有望な手法であるが，ユーザが実績ある汎用プロセッサを捨てることは難しく，現時点では広範に採用することは困難である．



２．研究の目的

制御ソフトウェアや組込みソフトウェアの保護は社会的に極めて重要である．しかし，ソフトウェアの難読化では保護が不十分であり，セキュアプロセッサの採用はコストや汎用性の面で難しい．本研究では，ソフトウェア（の一部）を論理回路化して隠蔽することにより，知的財産を保護する手法を検討する（上図(d)）．本研究は，汎用プロセッサに再構成可能論理を付加するという技術トレンドに一致しており，将来性と重要性の高いテーマである．

本研究の目的を列挙すれば以下のようになる．(1) 保護レベルの定量化，(2) 定量化指標を用いた論理回路化対象の選択手法，(3) CAD や FPGA など実装技術を考慮した保護手法の考案，(4) 現実的な問題への適用評価．

３．研究の方法

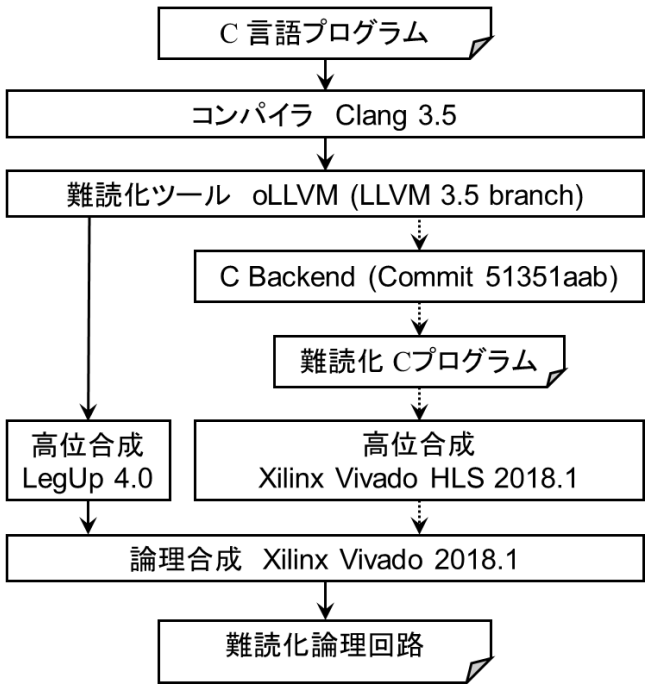
3.1　ソフトウェアの論理回路化手法（特に難読化論理回路の生成）

従来の研究では，論理回路を難読化するために専用ツールを自主開発していた（学会発表[3]）．しかし，専用ツールの開発には大きなコストが必要で，また開発プラットホームの変更に対する対応が難しかった．

一方で近年，高位合成(HLS)技術の進歩により C 言語記述から論理回路を生成できるようになった．さらに 2015 年，コンパイラ基盤 LLVM を用いたソフトウェア難読化ツール oLLVM が公開されたため，C 言語プログラムを難読化することが容易になった．ソフトウェア難読化ツールと HLS ツールを組合せることにより，難読化論理回路を生成できる可能性があるが，著者の知る限り，これまで実現・評価した研究は存在しなかった．

そこで本研究では，ベンチマークプログラム CHStone を oLLVM と HLS で難読化論理回路に

変換することが可能か検討し，その評価を行った．その処理フローを以下の図に示す（出典：発表論文[1]）．図の右側のフローは最初に試みた処理方法（発表論文[3]），左側のフローは後続研究（発表論文[1]）で試みた処理方法である．



3.2　セキュアプロセッサの低コスト実装手法

1 章で「セキュアプロセッサは将来有望だが現状採用困難」と述べた．しかし実装方法を工夫することによって，既存プロセッサとの互換性を保ちつつ，低コスト・低オーバヘッドで実現することができれば，現実的な選択肢になりうる．

著者は 2008 年にプロセッサ多様化手法を提案した(Shuichi Ichikawa et al. IEICE Trans. Fundamentals, Vol. E91-A, No.1, pp. 211—220, 2008)．一方 Hines らは，命令フェッチレートを向上させるため命令レジスタファイル(IRF)技術を提案した(2005 年). IRF をプロセッサ多様化に流用することにより, 性能を向上させつつプロセッサ多様化によるセキュアプロセッサが実現できる．

Oblivious RAM (ORAM)は，メモリアクセスによる情報漏洩を避ける技術である．プログラムの表現を隠蔽したとしても, 動作時のメモリアクセスを監視することによりプログラムの動作について知見を得ることができる．ORAM を作成する際には乱数（疑似乱数）が必要になるが，乱数品質と安全性の関係についてはこれまで研究されていなかった. 安全性を保ったまま実現コストを抑制できるような乱数生成方法を検討する必要がある．

LSI 製造上の個体差により各プロセッサに唯一の ID を与える技術として，PUF (Physically Unclonable Function)が注目されている．PUF の利用により，論理回路やソフトウェアを特定の個体だけで動作させることができる（即ち複製や盗用防止ができる）．PUF を作成するために乱数生成回路(TRNG)の出力の偏りを利用することができるので，著者らが研究してきたラッチ型 TRNG 技術を応用することができる．

４．研究成果

4.1　ソフトウェアの論理回路化手法

学会発表[3][5]では高位合成によりソフトウェアの論理回路化を試みた．さらに雑誌論文[1][3]ではソフトウェア難読化ツールと高位合成を併用して難読化論理回路を生成することを試みた．これらのツールは元々独立に作成されているため，併用することは技術的に難しく，色々な問題を回避・解決する必要があった．現在のソフトウェアには多くの問題があり一部の組合せでは不具合が生じるが，基本的には難読化論理回路の生成が可能であることが確認された．ベンチマークプログラム CHStone を使用して，難読化による性能オーバヘッドも定量的に評価することができた．

FPGA 技術において, 論理回路の一部を動的に（電源を入れたまま）書き換える技術を Dynamic Partial Reconfiguration (DPR)という．学会発表[4]および雑誌論文[2]では，FPGA 技術を前提として, プロセッサの周辺回路を動的に書き替えることによりシステムの耐故障性を高める手法について検討した．この手法はソフトウェアの論理回路化による知財隠蔽に応用できる．即ち，ソフトウェアの一部を論理回路（あるいは難読化論理回路）として生成しておき，実行するプログラ

ム毎に周辺回路に構成することが可能になる．

4.2　セキュアプロセッサの低コスト実装手法

雑誌論文[4]と学会発表[1]では，ラッチ型 TRNG のラッチ数を減らす（コスト削減）技術について，評価結果を述べている．この知見の延長上にラッチ型 PUF が実現でき，最終的に知財保護に適用できるものと期待される．

雑誌論文[5]は，IRF を用いたプロセッサ多様化手法について検討している．雑誌論文[6]および学会発表[2]では，メモリアクセスパタンからの情報漏れを防ぎつつ，実現コストを削減する方法について検討した．

4.3　今後の課題

当初の研究目的 4 つのうち，項目(1) 保護レベルの定量化，項目(2) 定量化指標を用いた論理回路化対象の選択手法については，研究期間内に研究成果を発表するには至らなかった．これらについては，使用するツールを吟味し，処理フローを確立して初めて研究を開始できるという制約があったためである．

項目(1)については未発表であるが評価が進められており，近々発表できると考えている．項目(2)については更に先の達成目標となるため，今後の課題としたい．

５．主な発表論文等

〔雑誌論文〕（計　６件）

[1] 山田翔太郎, 市川周一, 藤枝直輝： "LegUp と oLLVM による難読化制御論理回路の実装," 電気学会論文誌 D, vol. 139, no. 9, (to appear) (2019). （査読有）

[2] 荻堂盛也, 山田親稔, 宮城桂, 市川周一, 藤枝直輝: "部分再構成を用いたプロセッサの耐故障化手法に関する検討," 電気学会論文誌 D, vol. 139, no. 2, pp. 187--192 (2019). （査読有）
DOI: 10.1541/ieejias.139.187

[3] 松岡佑海, 藤枝直輝, 市川周一: "難読化ツール oLLVM を用いたハードウェア難読化手法の評価," 電気学会論文誌 D, vol. 139, no. 2, pp. 111--118 (2019). （査読有）
DOI: 10.1541/ieejias.139.111

[4] Naoki Fujieda, Shuichi Ichikawa: "A Latch-latch Composition of Metastability-based True Random Number Generator for Xilinx FPGAs," IEICE Electronics Express (ELEX), Vol. 15, No. 10, pp. 1--12 (2018). （査読有）DOI: 10.1587/elex.15.20180386

[5] Naoki Fujieda, Kiyohiro Sato, Ryodai Iwamoto, Shuichi Ichikawa: "Evaluation of Register Number Abstraction for Enhanced Instruction Register Files," IEICE Transactions on Information and Systems, Vol. E101-D, No. 6, pp. 1521--1531 (2018). （査読有）
DOI: 10.1587/transinf.2017EDP7221

[6] Naoki Fujieda, Ryo Yamauchi, Hiroki Fujita, Shuichi Ichikawa: "A Virtual Cache for Overlapped Memory Accesses of Path ORAM," International Journal of Networking and Computing, vol. 7, no. 2, pp. 106--123 (2017). （査読有）
http://www.ijnc.org/index.php/ijnc/article/view/145

〔学会発表〕（計　５件）

[1] Naoki Fujieda, Hitomi Kishibe, Shuichi Ichikawa: "A light-weight implementation of latch-based true random number generator," Proc. 15th International Wireless Communications & Mobile Computing Conference (IWCMC 2019) (to appear). （査読有）

[2] Hiroki Fujita, Naoki Fujieda, Shuichi Ichikawa: "An Analysis on Randomness of Path ORAM for Light-weight Implementation," Proc. Sixth International Symposium on Computing and Networking Workshops (CANDARW 2018), pp. 163--165 (2018). （査読有）

[3] Yoshiki Ishigaki, Naoki Fujieda, Yuumi Matsuoka, Kazuki Uyama, Shuichi Ichikawa: "An Obfuscated Hardwired Sequence Control System Generated by High Level Synthesis," Proc. Fifth International Symposium on Computing and Networking (CANDAR 2017), pp. 323--325 (2017). （査読有）

[4] Seiya Ogido, Chikatoshi Yamada, Kei Miyagi, Shuichi Ichikawa: "A Study of a Fault-tolerant System Using Dynamic Partial Reconfiguration," Proc. Fifth International Symposium on Computing and Networking (CANDAR 2017), pp. 600--602 (2017). （査読有）

[5] Naoki Fujieda, Shuichi Ichikawa, Yoshiki Ishigaki, Tasuku Tanaka: "Evaluation of the hardwired sequence control system generated by high-level synthesis," Proc. 26th IEEE International Symposium on Industrial Electronics (ISIE 2017), pp. 1261--1267 (2017). （査読有）

〔その他〕

研究者個人の論文リスト.
http://www.ccs.ee.tut.ac.jp/~ichikawa/research/papers.html
可能な限り，本文の PDF **や，ダウンロード可能な** URL **を合わせて掲載している.**

６．研究組織

(1) **研究分担者　　なし**

(2) **研究協力者**
研究協力者氏名：　藤枝 直輝
ローマ字氏名： Naoki Fujieda
（2013 **年** 4 **月～**2019 **年** 3 **月**: **豊橋技術科学大学 電気・電子情報工学系 助教,**
2019 **年** 4 **月より 愛知工業大学 工学部 電気学科 電子情報工学専攻 講師）**

※科研費による研究は、研究者の自覚と責任において実施するものです。そのため、研究の実施や研究成果の公表等については、国の要請等に基づくものではなく、その研究成果に関する見解や責任は、研究者個人に帰属されます。