修士論文

# IEEE802.11ah送信機の物理層における小規模な暗号化回路の開発と評価

吉田　怜矢

2019年1月31日

奈良先端科学技術大学院大学
情報科学研究科

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士 (工学) 授与の要件として提出した修士論文である。

吉田　怜矢

審査委員：

| | |
|---|---|
| 中島 康彦 教授 教授 | （主指導教員） |
| 岡田 実 教授 教授 | （副指導教員） |
| 中田 尚 准教授 | （副指導教員） |
| Tran Thi Hong 助教 | （副指導教員） |
| Renyuan Zhang 助教 | （副指導教員） |

# IEEE802.11ah 送信機の物理層における小規模な暗号化回路の開発と評価*

吉田　怜矢


## 内容梗概

IoT 技術はスマート社会を発展させるための重要な技術であり，IoT に用いられる通信のセキュリティは非常に重要である．現在，IEEE802.11ah などのほとんどの無線通信規格でのセキュリティは MAC 層でのデータ暗号化が採用されている．しかし MAC 層でのデータ暗号化では大量のデータ収集によって暗号化キーが破られる可能性がある．無線通信セキュリティを向上させるために PHY 層でのデータ暗号化は最近の研究の動向である．PHY 層における暗号化方法の一つに Phase Encryption があり，これは暗号化が変調の後に行われるものである．しかしこの方法の従来のアルゴリズムは複雑な計算が必要であり，複雑な回路は回路面積を大きくし電力を大量に消費するため省電力が求められる IoT 無線の物理層には向いていない．本研究では小規模な回路で Phase Encryption を実現できるアルゴリズムを提案した．従来手法と提案手法，両者の比較のためにそれぞれのアルゴリズムの回路を設計し，実装した．その結果，提案手法の回路は従来手法の回路に比べて 1/34 の面積で実装できた．



## キーワード

IoT,Phase Encryption,Wi-Fi,ASIC implementation


*奈良先端科学技術大学院大学 情報科学研究科 修士論文, 2019 年 1 月 31 日.

# Development and Evaluation of low complexity encryption circuit in PHY layer of IEEE802.11ah transmitter *

Satoya Yoshida

**Abstract**

IoT technology has been a important technology for developing smart society. And security of IoT communication is important. Currently, most of wireless comunication standards such as 802.11ah define the data encryption at MAC layer in which encryption key can be cracked by massive data collection. Encryption in PHY layer is a reseacrh trend to improve wireless communication security. One of the well-known encryption method in PHY layer is Phase Encryption, in which data is encrypted after modulation. However, conventional algorithm of this method is high complex when implementing in hardware. Whereas, IoT device demands low power consumption and low cost circuit.

In this thesis, I propose a low complex encryption method in PHY layer. I also develop hardware architectures of the proposed Phase Encryption algorithm and the conventional Phase Encryption algorithm. The circuit implementation results show that with the propose method I can reduce the hardware area by 34 times as compare of using the conventional Phase Encryption.

**Keywords:**

IoT,Phase Encryption,Wi-Fi,ASIC implementation

*Master's Thesis, Graduate School of Information Science, Nara Institute of Science and Technology, January 31, 2019.

# 目 次

# 図 目 次

# 表 目 次

# 1. はじめに

本章では研究の背景と目的，本稿の構成について述べる．

## 1.1 背景

近年モノのインターネット (Internet of Thing，IoT) が社会において耳目をあつめている．IoT は人やモノをインターネットに接続し，それらから得られる膨大な情報をセンサなどによって数値化し取得することによって集めた情報を元に相互に制御を行う仕組みである．たとえば IoT を活用した都市はスマートシティ [2] と呼ばれるものがある．

都市はざまざまな問題を抱えているがその一つに電力の需要過多で停電を引き起こす，供給過多により燃料資源を使いすぎるなどといったエネルギー問題がある．スマートシティでは都市内に設置された温度，湿度，照度センサ群から情報を集め天候を予測したり，各家庭や工場オフィスの電力計から電力消費をリアルタイムに監視することで必要な電力需要を見積もることができ，エネルギー問題を解決している．

このように IoT 機器は自動車やインフラストラクチャー，医療機器，工場など広い範囲に導入されており，もしそれらが IoT 機器を通じてアタックされた場合，その影響は大きく人の命を巻き込む事態も起こりえる．さらに IoT 機器はディスプレイなど出力デバイスがないものも多く，人による監視が難しい問題もある．実際に多数の IoT 機器に Mirai[3] と呼ばれるマルウェアが感染し，DDoS 攻撃を行うために利用されていたという事例 [4] も発生している．よって IoT 機器のセキュリティの強化が重要な課題である．

## 1.2 目的

前節ではIoT機器のセキュリティの強化が重要な課題であると述べた．IoTを実現するために膨大な数のセンサが必要であり同時にそのデータを集めるためのネットワークも必要であり，そのための基盤技術としてワイヤレス・センサ・ネットワーク(WSN)技術がある．WSNはネットワーク無線を使用することにより配線の必要が無くなりケーブルや敷設の費用が節約でき，設置する場所を選ばないのが利点である．しかしセンサノードの大きさや設置する場所によっては安定した電源を得られるとは限らず，出力が低い太陽電池や容量が限られた電池などを使用しなければならない．またIoTモジュール一つの消費電力は微小だが，IoTモジュールの数は世界全体で2020年には約300億[5]に拡大する見通しであり，天文学的な数量のIoTモジュールが消費する電力は膨大なものになる．よってIoT機器には省電力なものが求められるが，これを満たすIoT無線モジュール向けの通信規格としてはIEEE802.11ah(以降11.ah)がある．しかし現在，11ahの物理層においてはセキュリティとして暗号化は制定されていない．MAC層やその上位のプロトコルが提供する暗号化は高い計算リソースがあれば破ることができるものもある．[6] [7] 対して物理層が提供する暗号化は特定の二者間の通信で交わされる暗号文のみを受け取れないため統計的計算による暗号解読は難関である．そのため11ahの次の改定または11ahをベースとしたよりセキュリティの高い規格が制定されることをを見据え，Phase Encryptionと呼ばれる暗号化を物理層に実装しようと考えた．しかし従来提唱されているPhase Encryptionアルゴリズムは回路実装においては複雑な計算が多く，省電力が求められるIoTの用途には適していない．

そこでPhase Encryption暗号化回路の小規模な実装を目標に新しいアルゴリズムを提案し，そのためのアーキテクチャを設計する．また比較のため従来アルゴリズムの回路アーキテクチャを設計する．その後アーキテクチャをverilogHDLで実装し，Phase Encryption回路の提案手法と従来手法の面積の比較を行う．

## 1.3 構成

第2章は前提知識としてLPWA(Low Power Wide Area)無線，IEEE802規格，IEEE802.11ah規格の物理層とMAC層，位相振幅変調，Phase Encryptionについ

て述べる．第 3 章は Phase Encryption の関連研究としてシンボルを位相，振幅に換した後暗号化する手法とシンボルの符号ビットのみを暗号化する手法について述べる．第 4 章では提案する新しいアルゴリズムについて述べる．第 5 章では提案するアルゴリズムと従来のアルゴリズムの回路への設計について述べる．第 6 章では提案するアルゴリズムと従来のアルゴリズムの回路面積を比較しその考察について述べる．第 7 章でまとめとする．

# 2. 前提知識

## 2.1 LPWA 無線通信規格

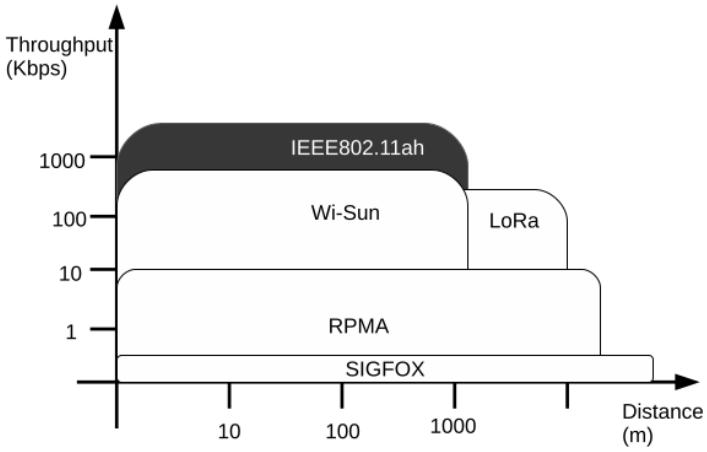


図 1: LPWA の距離とスループット

IoT 向けの無線には低消費電力で長距離通信が必要とされており，低消費電力・長距離通信が可能な無線通信規格を概して Low Power, Wide Area の頭文字を取って LPWA と呼称される．図 1 は主要な無線通信規格通信範囲とスループットを示したものである．また LPWA の中でも周波数が 1GHz 弱の電波を使用するものを SubG と呼び，SubG の中でもスループットが 100bps と比較的低く伝送距離が 50km の SIGFOX，伝送距離が 10km と下がるが 250kbps のスループットをもつ

LoRa，1km ほどだが他の SubG より比較的高いスループットを持つ IEEE802.11ah や Wi-Sun などがある．

- 幅広い通信レートに対応
- 比較的スループットが高い
- 接続が IP ベース

といったアドバンテージがあり，既存の Wi-Fi 技術の利用者をそのまま取り込むことができる．既存の IP ネットワークへの親和性が高いことで，802.11ah を用いたデバイスを誰もが開発しやすい．なお，他の Wi-Fi 規格とは周波数が異なり，また MAC フレームも従来のものからコンパクトな構成に変更されているため，直接的な接続の下位互換性はない．MAC フレームについては 802.11ac までの構造をベースとしており，既存のイーサネット 802.11ah 間のプロトコル変換は容易であると予想される．

## 2.2 IEEE802.11ah

IEEE802,11ah の MAC 層と PHY 層について述べる．MAC 層は IEEE802.11ac の MAC 層に省電力化に関連する機能が追加されたものである．物理層は 11.ac のものが使用されている．

### 2.2.1 MAC 層

MAC 層では省電力化のために IEEE802.11ah において RAW と TWT という機能が追加された.TWT は Target Wake Time と呼ばれており，無線子機 ( 以下ノード ) の通信モジュールを使用しないときだけスリープさせるノードの消費電力が低くなり，かつ通信を行うノードを減らせるので周波数帯を節約することができる．また TWT は明示的かどうか (明示的 TWT は通信を行うたび次の通信のタイミングを決定する, 暗黙的ならば通信タイミングは定期的) と announced であるかどうか (announced ならばアクセスポイント (以下 AP) からのビーコンで眠ってい

たノードに通信タイミングを知らせる．unannounced ならば通信タイミングは知らせない) の 2 つの分類がある．

RAW は Restricted Access window の略称で，短期間にノードが AP にアクセスしないようにする機能である．この機能は時間をウィンドウごとに分割し, ウィンドウもスロットごとに分割し, そしてスロットごとにノードを割り当てている. AP へのアクセスが許されたスロット ( AP からのビーコンを受け取った window 内のスロット) に属するノードのみが AP との通信を行うので，AP へのノードのアクセスの集中が防げ，衝突が発生しにくくなる．

### 2.2.2 物理層

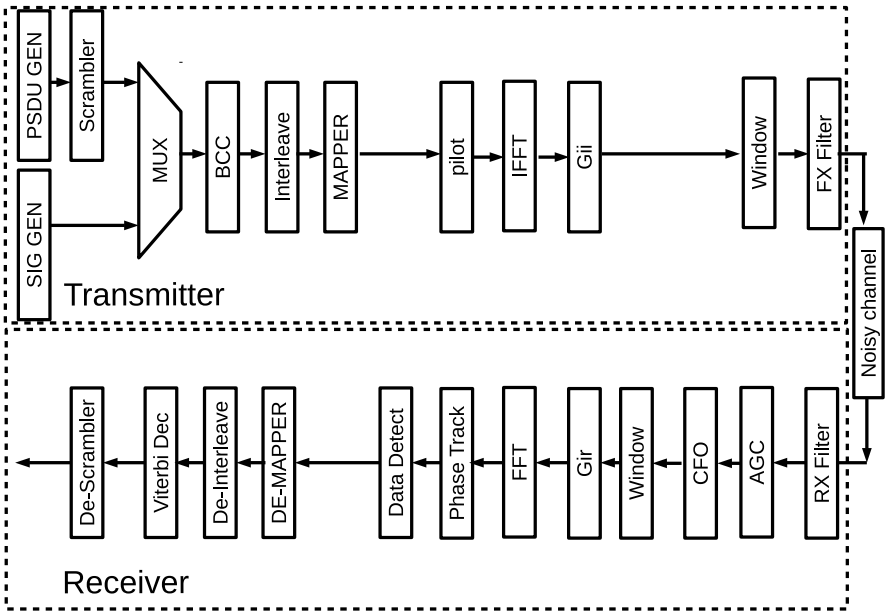


図 2: 物理層のブロック図

送信側の物理層 (図 2) は

**scrambler** ... 長い 0 と 1 のシーケンスによって，送信同期の問題が発生する可能性があり，0 または 1 が連続で出ないようランダマイズを行う．IEEE 802.11.ac ではクロック前の出力と 7 クロック前の出力の排他的論理和をとったものを入力と排他的論理和を取り，出力している．

**BCC(Binary Convolutional Code) エンコーダ** ... 入力が 6 つのシフトレジスタと排他的論理和で構成おり，1 クロックごとに入力が 1 ビットずつシフトされこの動作を繰り返しながら 2 ビットづつ出力する IEEE 802.11ac に用いられる拘束長 $k = 7$，符号化率 $r = 1/2$ のエンコーダが実現されている．

**Interleave** ... 通信路において通信状態が連続して悪化し，符号誤りが一箇所に集中した場合 (バースト誤り) 誤り訂正符号よって訂正ができなくなることがある．Interleave では符号化後の順番を入れ替え，連続した誤りを時間的に分散させている．

**MAPPER** ...Interleave から出力された数クロックのシリアル入力をパラレルにしたもの変調を行う．本稿では mapper と pilot の間に物理層の暗号化を行う. また変調については 2.3 を参照すること．

**pilot** ...OFDM を使用したとき，Pilot ではフェージングにより OFDM 信号の復調のためにパイロット信号が付与する．

**IFFT** ... 変調が行われた複数のサブキャリアを逆フーリエ変換を行い一つの OFDM 信号にする

**GII** ... 遅延波の影響を避けるために OFDM 信号の一部をコピーして追加するガードインターバルを追加する．

**Window** ... 窓関数

また受信側は基本的に逆の処理になっている．

本稿では送信側において Phase Encryption の実装が行われる前のブロックである MAPPER に注目して研究を行った．

## 2.3 位相振幅変調 [1]

OFDM は 2 次変調で一次変調では QAM(直交振幅変調) が用いられる．QAM は入力されたデータビットを複素数平面上に配置されたシンボルに転置する．この転置のための複素数平面をコンストレーションと呼び，この転置のために MAPPER ブロックが用いられる．図 3 にはコンストレーションにおける原点とシンボルの距離である振幅 $A$，原点とシンボルを結ぶ線と I 軸の角度が位相 $p$ となるコンストレーション上のシンボルを搬送波 $S(t)$(正弦波とする) に変換するための式は

$$S(t) = A\sin(2\pi f_0 t + p) \tag{1}$$

で示される．

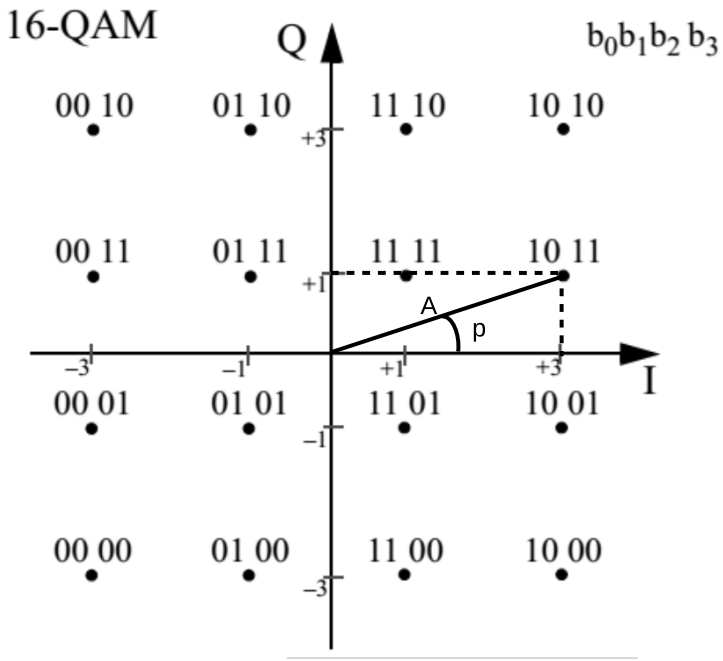


図 3: コンストレーション

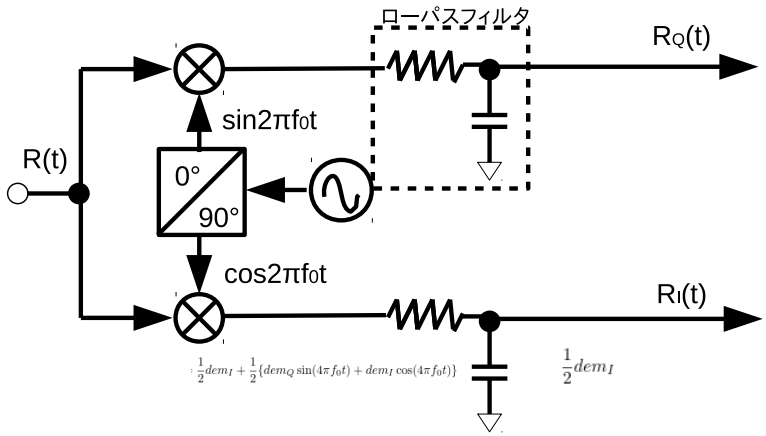


図 4: 受信波からの RQ の検出回路

実際に受信した搬送波の検出において，図 4 Q 成分を $dem_Q$，I 成分を $dem_I$ としたとき式 (1) は

$$S(t) = dem_Q \sin(2\pi f_0 t) + dem_I \cos(2\pi f_0 t) \tag{2}$$

と変形することができる．

*S(t)* に同じ周波数を持つ余弦信号をかけて

$$R_I(t) = dem_Q \sin(2\pi f_0 t)\cos(2\pi f_0 t) + dem_I \cos(2\pi f_0 t)\cos(2\pi f_0 t)$$

$$= \frac{1}{2} dem_I + \frac{1}{2}\{dem_Q \sin(4\pi f_0 t) + dem_I \cos(4\pi f_0 t)\} \tag{3}$$

式 (3) の $R_I(t)$ の 2 項目の成分をローパスフィルタで除くことで I 成分を検出することができる．また Q 成分は *S(t)* に余弦信号と 90° 位相が異なる正弦信号をかけることで同じように検出することができる．

## 2.4 Phase Encryption

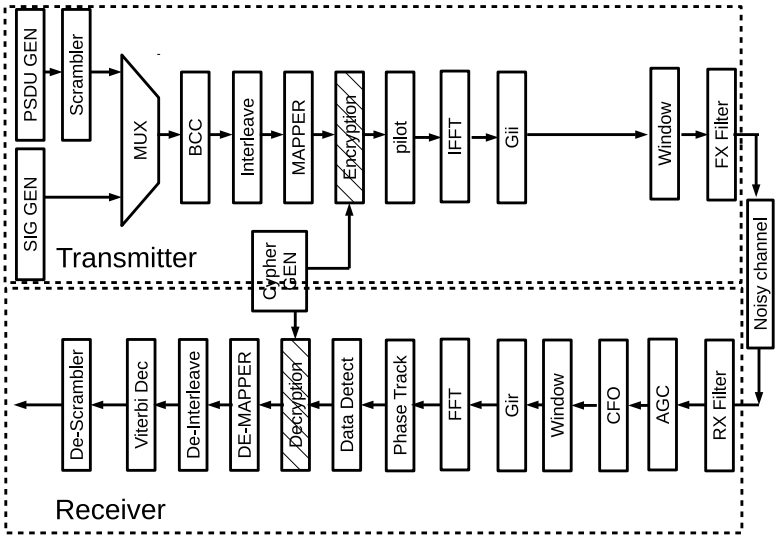


図 5: Phase Encryption

物理層においての暗号化は RC4 などのストリーム暗号によって行われるが，その手法としては従来良く使用されている変調の前に暗号化を行う手法と，Phase Encryption 図 5 と呼ばれる変調の後に暗号化を行う手法の 2 種類がある．

Phase Encryption では，変調されたシンボル $dem_I$，$dem_I$ それぞれのビットとストリーム暗号との排他的論理和をとって暗号化する．この暗号化手法の利点として

は物理層の中でも更に低いレベルを秘匿化できより安全であると考えられる．しかし変調後のシンボルの出力の集合を構成する元は連続しておらず (例えば 16QAM の場合-3,-1,1,3)，すべてのシンボルのビットを暗号化した場合暗号化後に出力しうる集合は暗号化前と同じではない (16QAM の場合-3,-2,-1,0,1,2,3)．そのため暗号化後の変調信号はコンストレーション上のシンボルに乗らず，復調が難しくなり誤り率が増加してしまう．図 6 は 8MSB Encrypt の線がこの暗号化手法のビットエラー率を示す．[8]

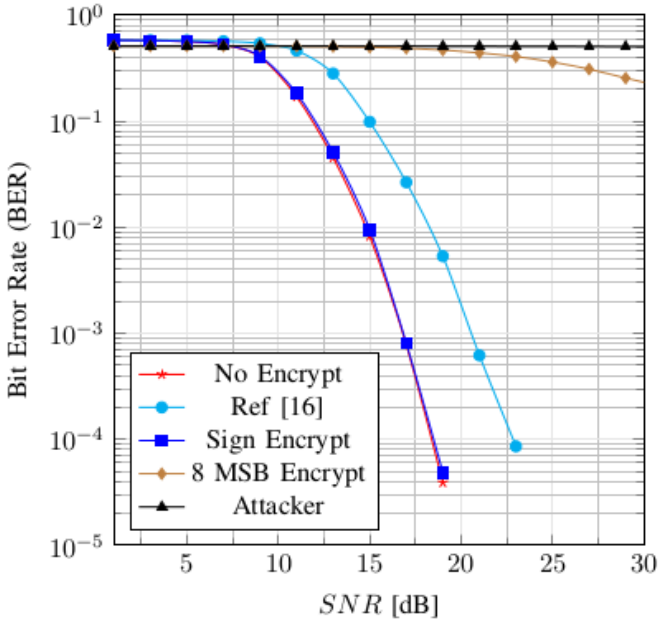


図 6: 16QAM を用いた時の各 Phase Encryption 手法のビットエラー率

関連研究としてビットエラーを発生しにくくする Phase Encryption の手法としてコンストレーション上のシンボルの位置の座標を直交座標から極座標に変換し，極座標成分である位相と振幅の暗号化を行うことによる手法 (3 章 1 節)[9] と，直交座標の I 成分と Q 成分の符号ビットのみを暗号化する手法 (3 章 2 節)[10] を紹介する．

# 3. 関連研究

## 3.1 位相と振幅に変換後に暗号化を行う手法

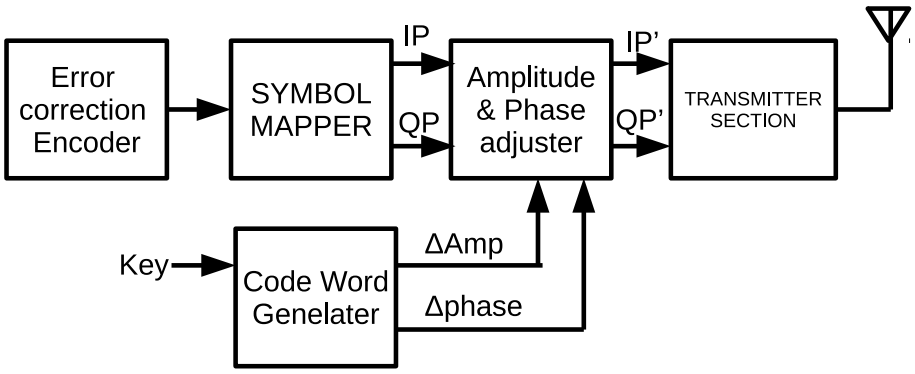


図 7: 位相と振幅に変換後に暗号化を行う手法のブロック図

この手法ではシンボル MAPPER で変調された数ビット (変調方式によって異なる) のシンボルを位相と振幅に変換した後，位相と振幅の暗号化を行うことで Phase Encryption を実現している．シンボルを図 3 のようにシンボルの直交座標から極座標系に変換される．その後，極座標系の位相と振幅はそれぞれ暗号化される．

図 7 はこの手法における一連の流れである．Error correction Encoder は図 7 の BCC，Interleave にあたる．符号化されたデータは SYMBOLMAPPER モジュールにおいて直交座標系である IP，QP に変換される．図 7 の Encryption モジュールにあたる Amplitude & Phase adjuster において暗号化シンボル IP’，QP’ に暗号化される．図 7 の Encryption 以降のモジュールに相当する TRANCEMITTER SECTION モジュールに入力され搬送波が生成される．

Amplitude & Phase adjuster での位相と振幅の暗号化方法は Code Word Generator によって生成される暗号化ストリーム $\Delta amp$，$\Delta phase$ を用いる．振幅を $amp$，位相を $phase$ とすると

$$amp' = amp + \Delta amp \tag{4}$$

$$phase' = phase + \Delta phase \tag{5}$$

暗号化された振幅 amp’，位相 phase’ が上の式で生成できる．また RC4 などの暗号化モジュールが生成するキーストリームが $N$ ビットとすると，位相，振幅にそれぞれ $\frac{1}{2}N$ が $\Delta amp$，$\Delta phase$ 生成に利用される．

$\Delta phase$ の生成に割り振られた $\frac{1}{2}N$ ビットの値を $n_{phase}$ とすると

$$\Delta phase = 2\pi \frac{n_{phase}}{2^{\frac{1}{2}N} - 1} \tag{6}$$

$\Delta amp$ の生成に割り振られた $\frac{1}{2}N$ ビットの値の最上位ビットを $\mathrm{n}_{MSB\_amp}$，残りの $\frac{1}{2}N - 1$ ビットを $\mathrm{n}_{amp}$，|R|をコンストレーションにおける原点とシンボルとの最長距離とすると，

$$\Delta amp = \begin{cases} |R|\frac{n_{amp}}{2^{\frac{1}{2}N}-1} & (n_{MSB\_amp} = 0) \\ -|R|\frac{n_{amp}}{2^{\frac{1}{2}N}-1} & (n_{MSB\_amp} = 1) \end{cases} \tag{7}$$

となるしかし直交座標から極座標系に変換において

$$phase = \tan^{-1}(\frac{QP}{IP}) \tag{8}$$

$$amp = \sqrt{IP^2 + QP^2} \tag{9}$$

また暗号化後，*amp’*，*phase’* を極座標系から直交座標系に変換するとき，

$$IP' = amp' \cdot \cos(phase') \tag{10}$$

$$QP' = amp' \cdot \cos(phase') \tag{11}$$

図 6 では Ref[16] の線がこの暗号化手法のビットエラー率を示し 8 MSB Encrypt よりビットエラー率が軽減されていることがわかる．しかし三角関数や平方根などのこれらの処理を実現する回路は加減算より複雑であり，回路が巨大化してしまうというデメリットが生じる．なおこのアルゴリズムは回路アーキテクチャがまだ設計されていないので，5 章において設計を行う．

## 3.2 符号ビットのみの Phase Encryption

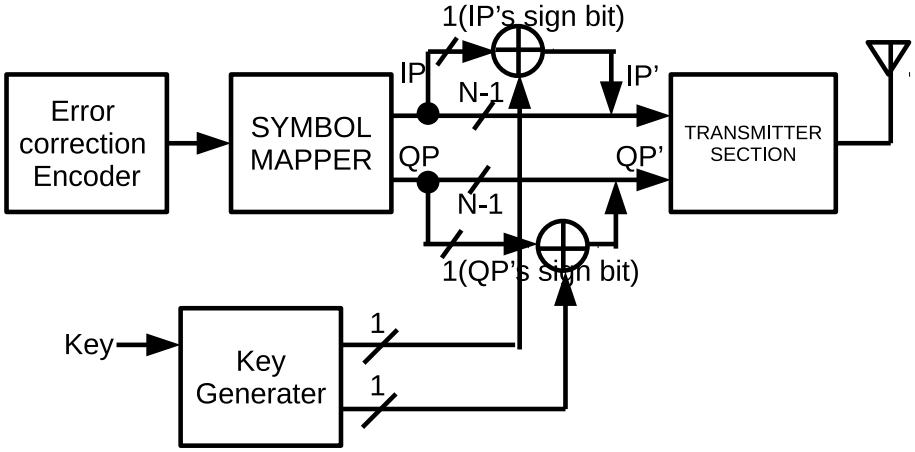


図 8: 符号ビットのみを暗号化する手法のブロック図

図 8 は符号ビットのみの Phase Encryption の一連の流れを示している．図 8 の Error correction Encoder，SYMBOL MAPPER，TRANSMITTER SECTION は図 7 のものと同じである．RC4 などのキーストリームが式 (4)，(5) によって $\Delta amp$, $\Delta phase$ を生成する図 7 の Code Word Generator と違い，図 8 の Key Generator はキーストリームがそのまま出力される．

生成される 2 ビットのキーストリーム SYMBOL モジュールで生成された IPQP のそれぞれの符号ビットと排他的論理和をとり暗号化する．IPQP を暗号化しても生成された IP'，QP' はコンストレーション上に遷移し，誤り率は低く済む．しかし IPQP 暗号化されたシンボルの候補数は 4 通り ((1,1) の場合 (1,1),(1,-1),(-1,1),(-1,-1)) であるためシンボル数が 4 つの QPSK では安全である．それ以上のシンボルを持つ 16QAM，64QAM では暗号化されない IP，QP のビットが存在するため安全ではない．

# 4. 提案手法

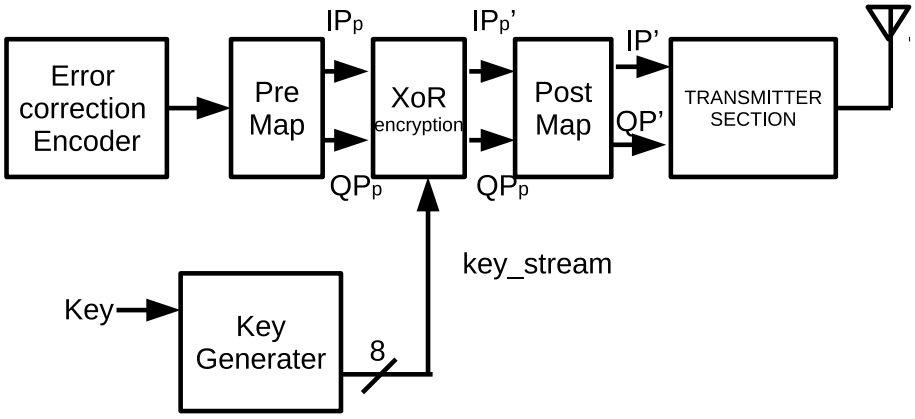


図 9: 2 段階の mapper を用いた Phase Encryption

関連研究の位相と振幅に変換後に暗号化を行う手法では Phase Encryption 実現のために Phase Encryption を使用しない場合に比べて回路面積が大きくなってしまうという欠点がある．ここではこの欠点を解消するために，2 つの Mapper を使用して段階的に Phase Encryption を行う方法を提案する．

この手法は図 9 に示すように 2 つの Mapper を使用する．ひとつ目 Mapper は PreMap と呼び上流から流れてくるデータビットを連続した擬似シンボル $IP_p$，$QP_p$ に変換する Mapper である．PreMap によって生成された擬似シンボル $IP_p$，$QP_p$ は Key Generator から生成される key_stream によって排他的論理和がとられ暗号化された疑似シンボル $IP_p$'，$QP_p$' が生成される．この擬似シンボルの集合は連続した正の整数で構成されており，この集合の元はすべてのビットとランダムに生成される key_stream と排他的論理和をとっても暗号化後に再び戻ってくる性質をもつ．

例えば16QAM の疑似シンボルは{0(00),1(01),2(10),3(11)}の集合であり Key Generator から生成される key_stream{0(00),1(01),2(10),3(11)}とどの組み合わせで排他的論理和をとっても，再び同じ集合{0(00),1(01),2(10),3(11)}に戻ってくることがわかる．

次に疑似シンボルでは搬送波を生成できないので，2 つ目の Mapper は PostMap と呼び，PostMap は暗号化された擬似シンボル $IP_p$'，$QP_p$' を暗号化された正規の

シンボル IP’，QP’ に変換する Mapper である．

例えば PostMap では 16QAM の疑似シンボルから正規シンボルは/0→ −3, 1 → −1, 2 → 1, 3 → 3/と転置される．この手法であるとコンストレーションのシンボルから暗号化後もずれることはないので，ビット誤り率も図 6$SignEncrypt$ や暗号化を行わない $NoEncrypt$ と理論的には同じであると考えられる．シンボルを位相と振幅に変換し，暗号化後位相振幅をシンボルに戻す回路を必要とする従来の $PhaseEncryption$ の回路に比べて，提案する手法は 2 つの $mapper$ で実装ができるため大幅な回路面積削減を期待できる．

# 5. 実装

3 章 1 節で示した位相と振幅に変換後に暗号化を行う手法 (以下従来手法) と 4 章で示した提案手法の設計を行う．

## 5.1 コンストレーション図

提案手法も従来手法も変調は以下のコンストレーション図に基づいて行われる．また [11] の画像を図 10，11，12，13，14 として使用した．

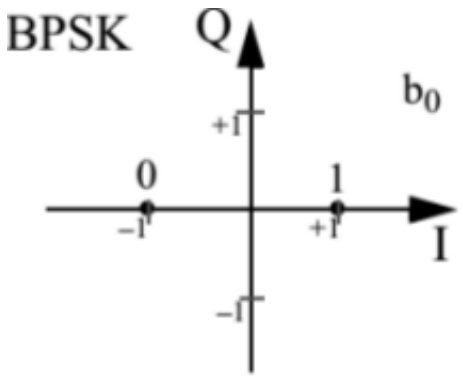


図 10: bpsk のコンストレーション図

BPSK は同時に 1 ビットの入力でシンボルを返す．コンストレーション上のシンボル数は 2.

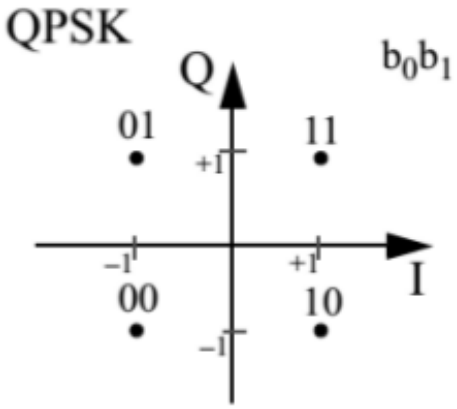


図 11: qpsk のコンストレーション図

BPSK は同時に 2 ビットの入力でシンボルを返すので BPSK にくらべて 2 倍効率がよい．コンストレーション上のシンボル数は 4.

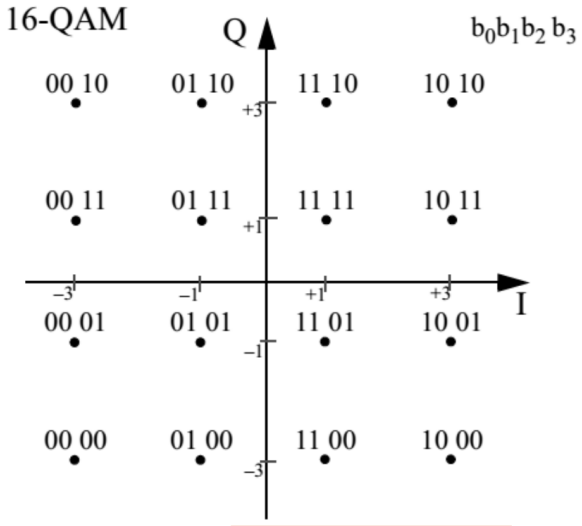


図 12: 16QAM のコンストレーション図

16QAM は同時に 4 ビットの入力でシンボルを返すので QPSK にくらべて 2 倍効率がよい．コンストレーション上のシンボル数は 16.

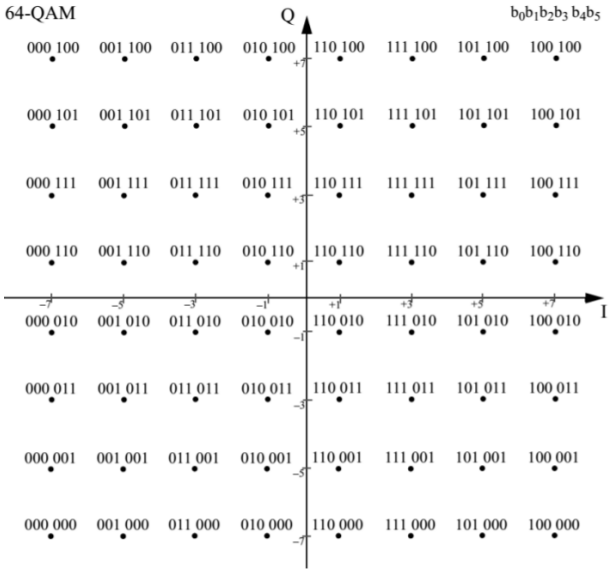


図 13: 64QAM のコンストレーション図

64QAM は同時に 6 ビットの入力でシンボルを返すので 16QAM にくらべて 1.5 倍効率がよい．コンストレーション上のシンボル数は 64.

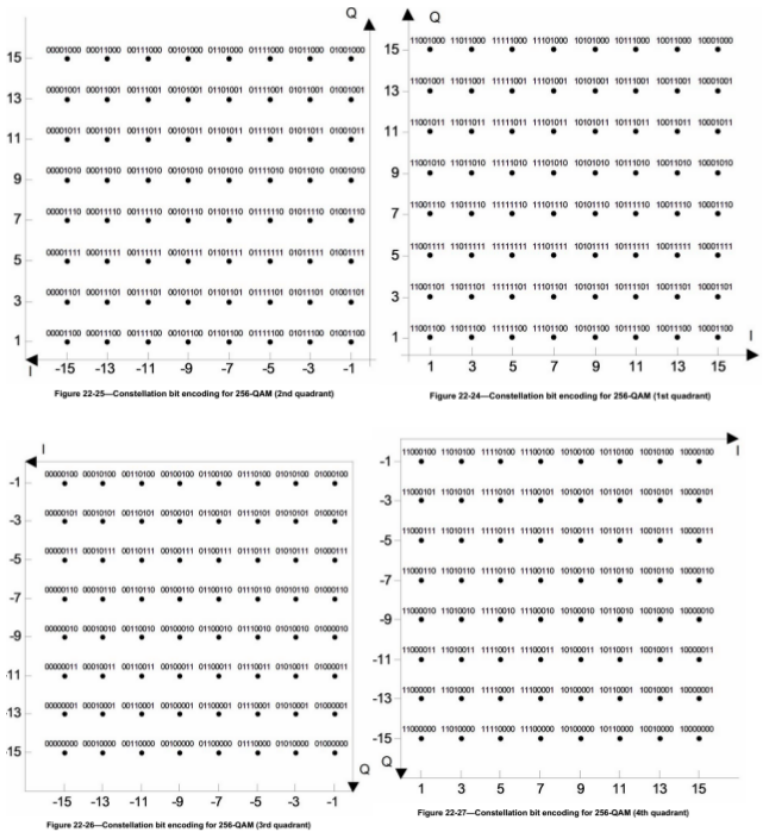


図 14: 256QAM のコンストレーション図

64QAM は同時に 8 ビットの入力でシンボルを返すので 64QAM にくらべて 1.3 倍効率がよい．コンストレーション上のシンボル数は 256.

## 5.2 従来手法

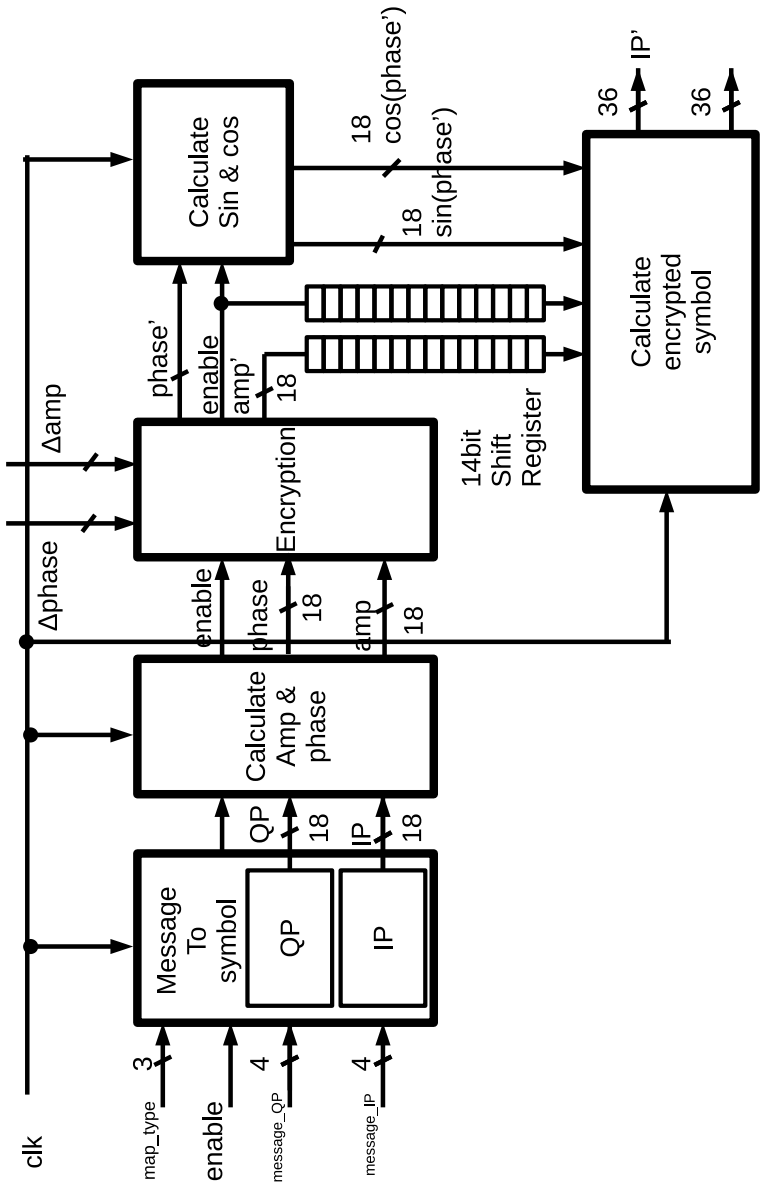


図 15: 従来の Phase Encryption の内部ブロック図

図 18 は 3 章 1 節で解説した従来の Phase Encryption の設計の内部ブロック図である．大まかな流れとしては各 4 ビットの message_IP，message_QP が Message To Symbol モジュールに入力され，シンボルの直交座標である IP，QP が出力される．シンボル IP，QP は Calculate amp&phase モジュールに入力され，シンボルの極座標系である振幅 amp，位相 phase に変換される．XoR Encryption モジュールにおいて phase に $\Delta phase$，amp に $\Delta amp$ が加算され amp’，phase’ と暗号化される．phase’ は Calculate sin&cos モジュールにおいて sin(phase’)，cos(phase’) に変換されるが，その変換に 14 クロックかかる．amp’ はそれまで 14 個の 18 ビットレジスタで構成されるシフトレジスタで，14 クロック待機してから sin(phase’)，cos(phase’) と同時に Calculate encrypted Symbol モジュールに入力される．Calculate encrypted symbol モジュールで amp’ と sin(phase’)，cos(phase’) の乗算が行われ暗号化された直交座標系シンボル IP’，QP’ が出力される．なお，enable はデータが有効か無効かを示す線で enable が 0 の場合各モジュールの入力は無効である．

また IP，QP，IP’，QP’，amp，amp’，$\Delta amp$ のビットフォーマットは図 16 で，phase，phase’，$\Delta phase$，sin(phase’)，cos(phase’) のビットフォーマットは図 17 である．両方とも符号つき 18 ビットの小数である．

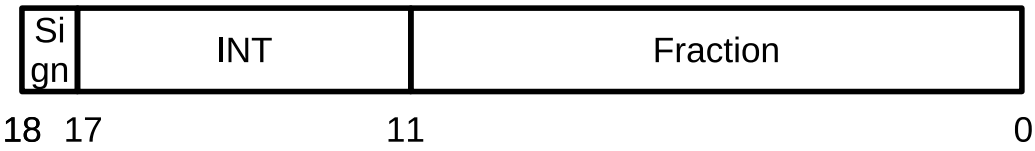


図 16: IP，QP，IP’，QP’，amp，amp’，$\Delta amp$ のビットフォーマット


図 17: phase，phase’，$\Delta phase$，sin(phase’)，cos(phase’) のビットフォーマット

1 節は Message To Symbol モジュール，2 節は Calculate amp&phase モジュール，3 節は Calculate sin&cos モジュールの解説を行う．Calculate crypted Symbol は単なる加算器，XoR Encryption は乗算器であるので解説は行わない．

### 5.2.1 Message To Symbol モジュール

Message To Symbol モジュールは下の表 2，3，4，5，6 にしたがって IP，QP が出力される Mapper である．

このモジュールでは変調方式を示す 3 ビットの map_type を入力すると，それに対応したシンボルである IP，QP が出力される．

表の input が入力されるデータビットであり，図 18 における message_IP，message_QP に相当する．また output はコンストレーション図に input を入れた結果である．しかし平均電力をにするために 1 にするために output に表 1 に示されている変調方式に対応する正規化係数がかけられる．結果，18 ビットの Nomalized output が図 18 における IP，QP として出力される．

| 変調方式 | 正規化係数 |
|---|---|
| BPSK | 1 |
| QPSK | $1/\sqrt{2}$ |
| 16QAM | $1/\sqrt{10}$ |
| 64QAM | $1/\sqrt{42}$ |
| 256QAM | $1/\sqrt{170}$ |

表 1: 変調方式に対する正規化係数

| input | output | Nomalized output(18bit decimal) |
|---|---|---|
| 0 | -1 | -1(260096) |
| 1 | 1 | 1(2048) |

表 2: BPSK の場合の Mapper

| input | output | Nomalized output(18bit decimal) |
|---|---|---|
| 0 | -1 | -1(260695) |
| 1 | 1 | 1(1448) |

表 3: QPSK の場合の Mapper

| input | output | Nomalized output(18bit decimal) |
|---|---|---|
| 00 | -3 | -0.9487(260201) |
| 01 | -1 | -0.3162(261496) |
| 11 | 1 | 0.3162(647) |
| 10 | 3 | 0.9487(1942) |

表 4: 16QAM の場合の Mapper

| input | output | Nomalized output(18bit decimal) |
|---|---|---|
| 000 | -7 | -1.0801(259931) |
| 001 | -5 | -0.7715(260563) |
| 011 | -3 | -0.4629(261195) |
| 010 | -1 | -0.1543(261827) |
| 110 | 1 | 0.1543(316) |
| 111 | 3 | 0.4629(948) |
| 101 | 5 | 0.7715(1580) |
| 100 | 7 | 1.0801(2212) |

表 5: 64QAM の場合の Mapper

| input | output | Nomalized output(18bit decimal) |
|---|---|---|
| 0000 | -15 | -1.1504(259787) |
| 0001 | -13 | -0.9971(260101) |
| 0011 | -11 | -0.8437(260416) |
| 0010 | -9 | -0.6903(260730) |
| 0110 | -7 | -0.5369(261004) |
| 0111 | -5 | -0.3835(261358) |
| 0101 | -3 | -0.2301(261672) |
| 0100 | -1 | -0.0767(261986) |
| 1100 | 1 | 0.0767(157) |
| 1101 | 3 | 0.2301(471) |
| 1111 | 5 | 0.3835(785) |
| 1110 | 7 | 0.5369(1099) |
| 1010 | 9 | 0.6903(1413) |
| 1011 | 11 | 0.8437(1727) |
| 1001 | 13 | 0.9971(2042) |
| 1000 | 15 | 1.1504(2356) |

表 6: 256QAM の場合の Mapper

### 5.2.2 Calculate amp & phase モジュール

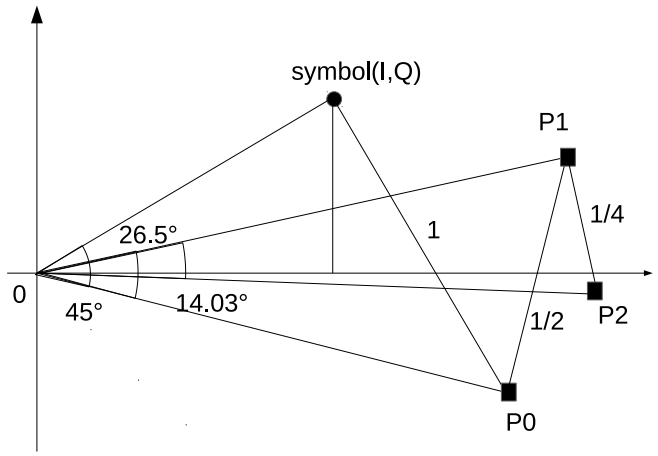


図 18: Cordic アルゴリズムを使用した位相計算

| i | | output | 絶対値ベクトル |
|---|---|---|---|
| 0 | Angle0 | 45 ° | 1.414213562 |
| 1 | Angle1 | 26.56505118 ° | 1.58113883 |
| 2 | Angle2 | 14.03624347 ° | 1.629800601 |
| 3 | Angle3 | 7.125016349 ° | 1.642484066 |
| 4 | Angle4 | 3.576334375 ° | 1.645688916 |
| 5 | Angle5 | 1.789910608 ° | 1.646492279 |
| 6 | Angle6 | 0.89517371 ° | 1.646693254 |
| 7 | Angle7 | 0.447614171 ° | 1.646743507 |
| 8 | Angle8 | 0.2238105 ° | 1.64675607 |
| 9 | Angle9 | 0.111905677 ° | 1.646759211 |
| 10 | Angle10 | 0.055952892 ° | 1.646759996 |
| 11 | Angle11 | 0.027976453 ° | 1.646760193 |

表 7: i 番目の三角形とその絶対値ベクトル

図 18 に三角形 $\Delta 0P0P1$ があったとき，0-P1 は三角形の斜辺，P1-P0 は対辺，0-P1 は隣辺とする．

Calculate amp & phase モジュールは Message To Symbol モジュールから出力された IP，QP を振幅 amp, 位相 phase に変換するモジュールである．この変換を実装するために CORDIC(COordinate Rotation DIgital Computer)[12] [13] というアルゴリズムを用いている．

CORDIC は足し算引き算のみで sin，cos，$tan\_1$，sinh，cosh，exp，log などの関数値をアルゴリズムで求めることができる．

このアルゴリズムは比較的小規模な回路で実現できるため，関数電卓などで使用されている．

このモジュールでは入力されたシンボルである直行座標系の IP，QP 成分から位相と振幅を導き出すために CORDIC アルゴリズム使用している．CORDIC アルゴリズムでは図 19 のように直角三角形を隣辺と対辺の比率が 1，1/2，1/4，1/8... となるように減らしながら 12 回繰り返し足し引きし，直角三角形の斜辺を逐次的に I 軸に近づける．12 回の直角三角形の角の足し引きで角度 (位相) を求めることができる．またベクトル絶対値と I 成分を乗算することで振幅を求めることができる．

直角三角形の斜辺を I 軸に近づけたい

$I_{i-1} > 0$ とき直角三角形 i-1 の最長辺に直角三角形 i の隣辺を下向きに接続する．$P_i$ の座標を $(I_i,Q_i)$，原点と直角三角形 i のなす角 $\phi_i$ と表現するとき

$$I_i = I_{i-1} + (\frac{1}{2})^{i-1} Q_{i-1} \tag{12}$$

$$Q_i = Q_{i-1} - (\frac{1}{2})^{i-1} I_{i-1} \tag{13}$$

$$\phi_i = \phi_{i-1} + Angle_i \tag{14}$$

$I_{i-1} < 0$ とき現在の直角三角形 i の最長辺に次の直角三角形 i-1 の隣辺を下向きに接続する．

$$I_i = I_{i-1} - (\frac{1}{2})^{i-1} Q_{i-1} \tag{15}$$

$$Q_i = Q_{i-1} + (\frac{1}{2})^{i-1} I_{i-1} \tag{16}$$

$$\phi_i = \phi_{i-1} - Angle_i \tag{17}$$

また 1 回目の直角三角形の斜辺の長さが $\sqrt{1+1} = \sqrt{2}$ に対する $n$ 回目の直角三角形 n の斜辺の長さである絶対値ベクトル $|f_n|$ は

$$|f_n| \quad = \sqrt{1 + (\frac{1}{1})^2} * \sqrt{1 + (\frac{1}{2})^2} * \cdots * \sqrt{1 + (\frac{1}{2^{n-1}})^2} \tag{18}$$

$$\tag{19}$$

12 回目の直角三角形 12 の絶対値ベクトルである $|f_{12}|$ は 1.646 である．そして直角三角形 12 の斜辺は I 軸に収束しているため $I_{12}$ は最初のシンボルの座標と原点の長さ (振幅) の 1.646 倍であると言える．よって振幅 $amp$ は

$$amp = I_{12} * 0.6072 \tag{20}$$

と表すことができる．

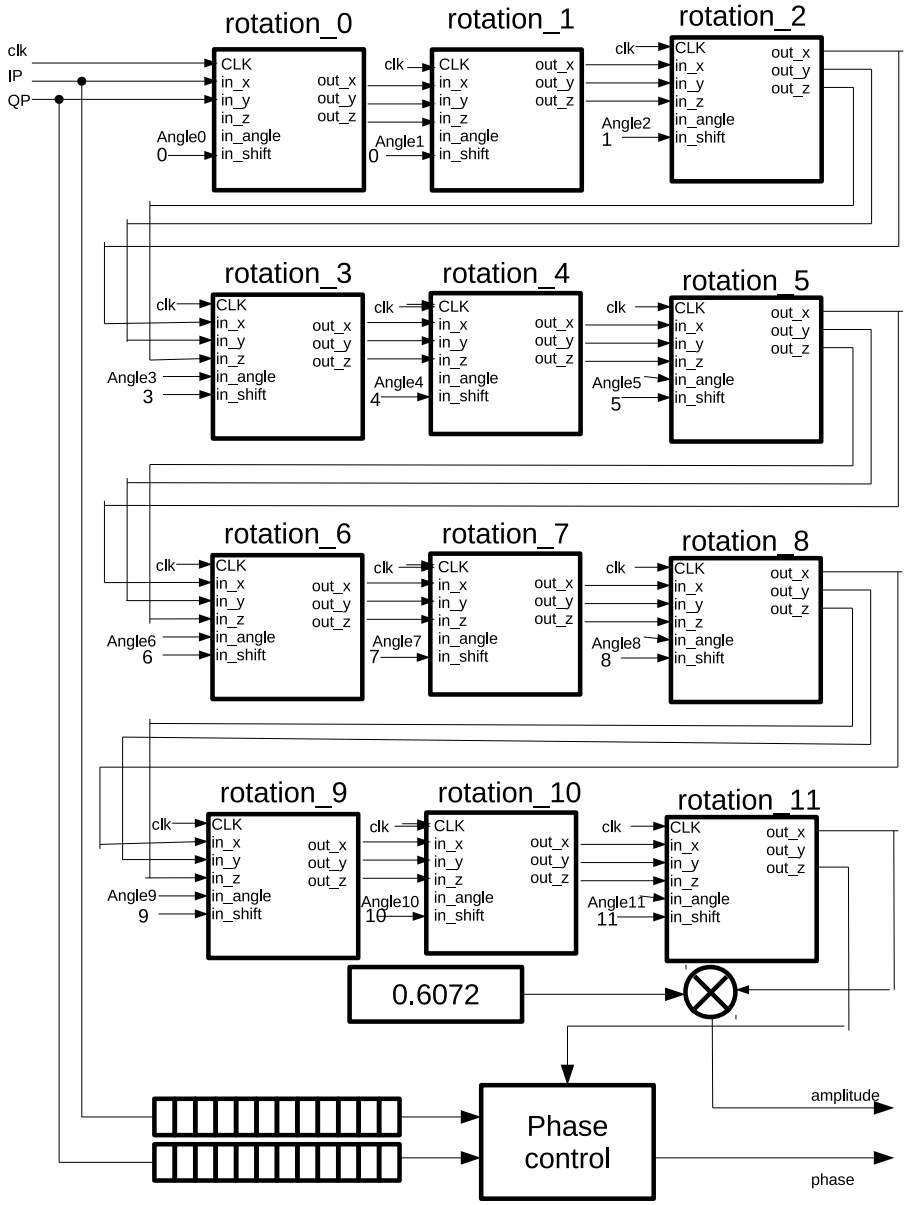


図 19: Calculate amp & phase モジュールのブロック図

図 19 は Calculate amp & phase モジュールのブロック図である．18 ビットの IP,QP が rotation に与えられる式 (4)〜(9) を計算した後，次の rotation に入力として与えられる．こうして rotation0〜rotation11 逐次的に計算された後，rotation11 の出力 out_x は 0.6072 と乗算され 18 ビットの amp として出力される．rotation11 の出力 out_y は IP，QP が入力されて 12 クロック後に出力される．12 つのシフトレジスタによって 12 クロック遅延させられた IP，QP と同じタイミングで Phase control に入力され phase として出力される．phase control は 12 クロック遅延させられた IP，QP の符号ビットから IP,QP がどの象限にいるのか判断し，表 8 に基づいて 18bit の phase として出力している．

| 象限 | phase |
|---|---|
| 1,4 | out_z |
| 2 | - $\pi$ +out_z |
| 3 | $\pi$ +out_z |

表 8: 象限に対する角度

モジュール rotation は式 (12)〜(17) を実装している．rotation の入力は 18bit の in_ x,in_ y,in_ z,in_angle がある．

$in_x$は$P_i$ の I 軸の位置，$in_y$は$P_i$ の Q 軸の位置でこれらは P の座標である $in_z$は今までの直角三角形の角度の累計である．また $in\_angle$ は表 7の $angle\_i$ の角度が入力される．

例えば rotation3 では 8bit の 7.125016349° が Angle3 として入力されている．他の入力としては 4bit の $in_{shift}$がある．

$in_{shift}$は $i$ 番目の $rotation$ モジュール内で例えば式 (12) が実行される．

このとき $in_y$に$\frac{1}{2}^i$がかけられるがこの処理のために $i$ ビットだけ右シフトする必要があり，何ビット右シフトすればよいのかを決定するために $in_{shift}$が入力される．

### 5.2.3 Calculate sin&cos モジュール

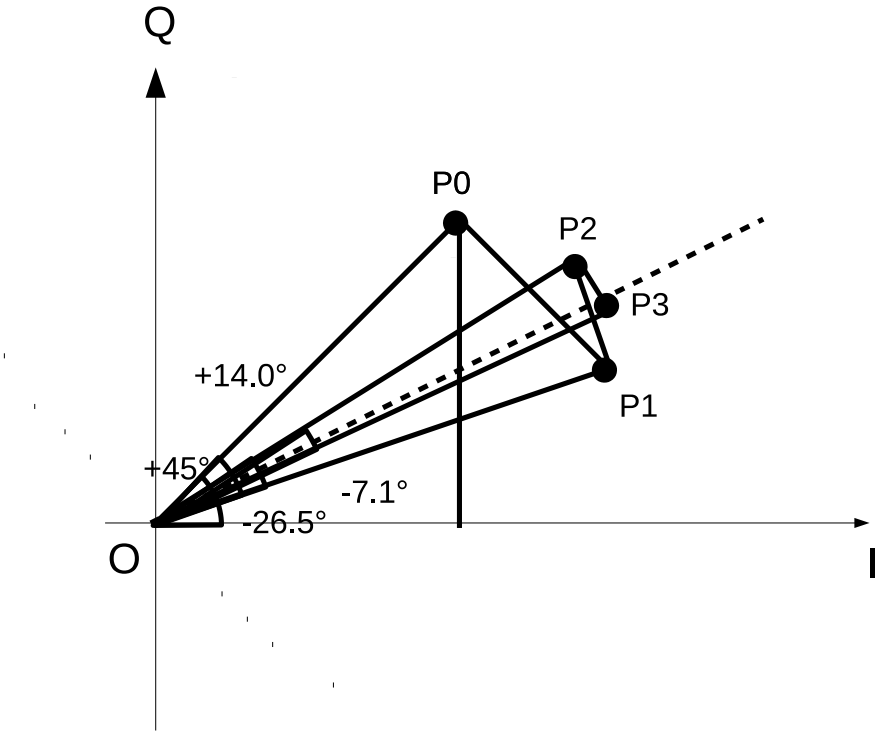


図 20: CORDIC アルゴリズムを使用した sin30，cos30

Calculate amp & phase モジュールでは CORDIC アルゴリズムは直角三角形の斜辺を I 軸に逐次的に近づけていくことでシンボル座標から角度位相を導出していた．Calculate sin&cos モジュールで使用される cordic アルゴリズムでは角度 (位相)$\theta$ が入力である．角 $\theta$ を持つ直角三角形の斜辺に直角三角形 i の斜辺を逐次的に近づけていくことで角度 (位相)$\theta$ から $\sin\theta$，$\cos\theta$ を導いている．

$\phi_{i-1}$ を直角三関係 i-1 の隣辺と I 軸のなす角とすると $\phi_{i-1} > \theta$ とき直角三角形 i-1 の斜辺に直角三角形 i の隣辺を下向きに接続する．$\phi_i$ の座標を $(I_i,Q_i)$，

$$I_i = I_{i-1} - (\frac{1}{2})^{i-1} Q_{i-1} \tag{21}$$

$$Q_i = Q_{i-1} + (\frac{1}{2})^{i-1} I_{i-1} \tag{22}$$

$$\phi_i = \phi_{i-1} + Angle_i \tag{23}$$

$\phi_{i-1} > \theta$ とき現在の直角三角形 i の最長辺に次の直角三角形 i-1 の隣辺を上向きに接続する．

$$I_i = I_{i-1} + (\frac{1}{2})^{i-1} Q_{i-1} \tag{24}$$

$$Q_i = Q_{i-1} - (\frac{1}{2})^{i-1} I_{i-1} \tag{25}$$

$$\phi_i = \phi_{i-1} - Angle_i \tag{26}$$

図 20 では 4 回繰り替えし $\sin 30, \cos 30$．まず45の直角三角形 0 が配置される．次に45 30 なので I 軸と P0 の間に P1 が来るように 0，P0，P1 からなる直角三角形 1 が配置される．次に $(45 - 26.5)30$ なので Q 軸と P1 の間に P2 が来るように 0，P1，P2 からなる直角三角形 2 が配置される．次に $(45 - 26.5 + 14.0)30$ なので I 軸と P2 の間に P3 が来るように 0，P2，P3 からなる直角三角形 3 が配置される．

CORDIC では上記のような処理をくりかえして直角三角形 i の 0，Pi の辺を任意の角度に近づけていく．

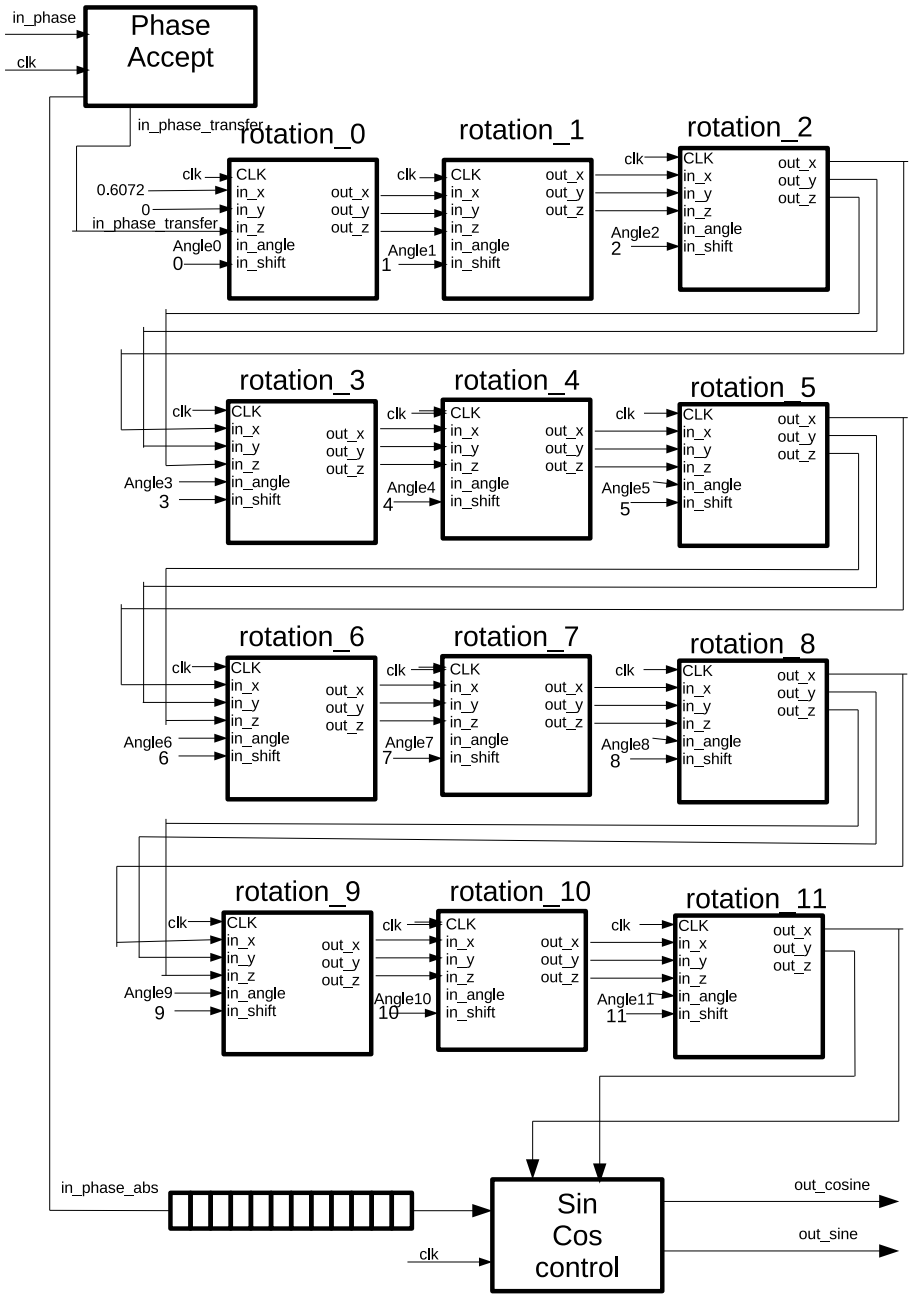


図 21: Calculate sin&cos モジュールのブロック図

図 21 は Calculate sin&cos モジュールのブロック図である．18 ビットの入力された位相 in_phase は Phase Accept に入力され CORDEC で処理できるように位相の象限におうじて正規化され in_phase_trancsfer として出力される．

rotation0 には in_z として in_phase_trancsfer が，in_x として 0.6072 が，in_y として 0 が，入力される．また in_angle には表 7 の angle0 の値が入力される．こうして rotation0〜rotation11 で逐次的に計算された後，rotation11 の出力 out_x，out_y を絶対値として位相 $\mathrm{in}_{phase}$ の象限に応じて符号が付加され $out\_cosine$, $out\_sine$ として出力される．

### 5.3 提案手法

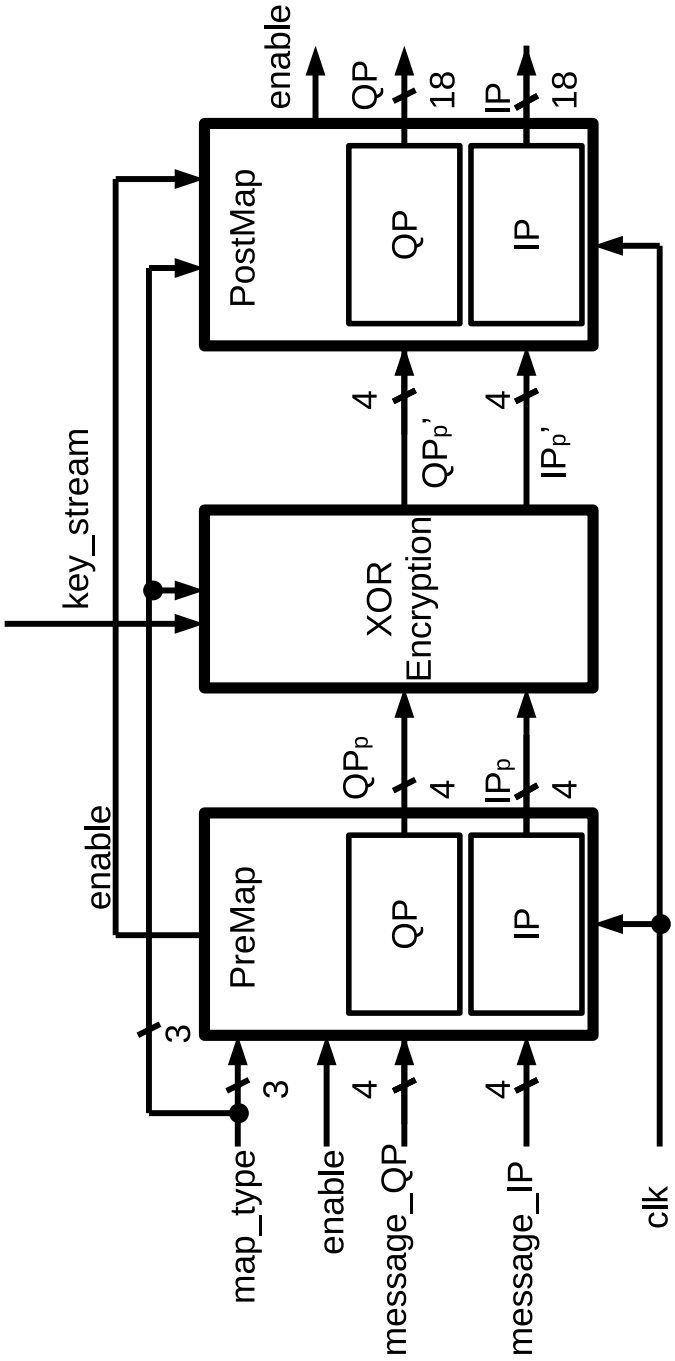


図 22: 提案手法の Phase Encryption のブロック図

図 22 は 4 章で解説した提案する phase encryption の設計の内部ブロック図である．大まかな流れとしては，各 4 ビットの message_IP，message_QP が PreMap モジュールに入力され符号は常に正である 4 ビットの疑似シンボル $\mathrm{IP}_p, QP_p$が出力される．疑似シンボル $IP_p, QP_p$は $XoR\ Encryption$ モジュールに入力され，暗号化された $QP'_p, IP'_p$が出力される．$QP'_p, IP'_p$は $PostMap$ に入力されることで暗号化された疑似シンボルから暗号化された正規のシンボル $IP', QP'$に変換される．なお，$enable$ はデータが有効か無効かを示す線で $enable$ が 0 の場合各モジュールの入力は無効である．また $IP, QP$ のビットフォーマットは図 16である．

XoR Encryption では疑似シンボル $\mathrm{IP}_p, QP_p$がそれぞれ，$QP_p[3] \oplus x[7] \sim QP_p[0] \oplus x[4], IP_p[3] \oplus x[3] \sim IP_p[0] \oplus x[0]$ のように 8 ビットの $x$ と各ビットづつ $xor$ がとられ暗号化を行っている．$x$ はキーストリーム $k$ から表 9のように決められる．例えば $16QAM$ の場合は 8 ビットのキーストリームのうち 0, 1, 4, 5 ビット目のみが有効となり，あとは 0 として扱われる．

| 変調 | x[7] | x[6] | x[5] | x[4] | x[3] | x[2] | x[1] | x[0] |
|---|---|---|---|---|---|---|---|---|
| BPSK | 0 | 0 | 0 | 0 | 0 | 0 | 0 | k[0] |
| QPSK | 0 | 0 | 0 | k[4] | 0 | 0 | 0 | k[0] |
| 16QAM | 0 | 0 | k[5] | k[4] | 0 | 0 | k[1] | k[0] |
| 64QAM | 0 | k[6] | k[5] | k[4] | 0 | k[2] | k[1] | k[0] |
| 256QAM | k[7] | k[6] | k[5] | k[4] | k[3] | k[2] | k[1] | k[0] |

表 9: 変調に対するキーストリームの有効ビット

次に PreMap モジュール，PostMap モジュールについて述べる．

### 5.3.1 PreMap&PostMap

下の表の message input は図 22 の各 4 ビットの message_IP，message_QP に相当する．PreMap output は図 22 の各 4 ビットの疑似シンボル $\mathrm{IP}_p, QP_p$または暗号化された疑似シンボル $QP'_p, IP'_p$に相当する．$Nomalized\ output$ は図 22の暗号化された正規シンボル $IP', QP'$に相当する．

PreMap では message input に対応した PreMap output が $\mathrm{IP}_p$, $QP_p$ として出力される $Mapper$ である．また疑似シンボルの暗号化後も $PreMap\ output$ の行のいずれかの値に遷移する．

PostMap では PreMap output に対応した Nomalized output が IP’，QP’ として出力される Mapper である．

| message input | PreMap output | PostMap output | Nomalized output(18bit decimal) |
|---|---|---|---|
| 0 | 0 | -1 | -1(260096) |
| 1 | 1 | 1 | 1(2048) |

表 10: BPSK の場合の Mapper

| message input | PreMap output | PostMap output | Nomalized output(18bit decimal) |
|---|---|---|---|
| 0 | 0 | -1 | -1(260695) |
| 1 | 1 | 1 | 1(1448) |

表 11: QPSK の場合の Mapper

| message input | PreMap output | PostMap output | Nomalized output(18bit decimal) |
|---|---|---|---|
| 00 | 0 | -3 | -0.9487(260201) |
| 01 | 1 | -1 | -0.3162(261496) |
| 11 | 2 | 1 | 0.3162(647) |
| 10 | 3 | 3 | 0.9487(1942) |

表 12: 16QAM の場合の Mapper

| message input | PreMap output | PostMap output | Nomalized output(18bit decimal) |
|---|---|---|---|
| 000 | 0 | -7 | -1.0801(259931) |
| 001 | 1 | -5 | -0.7715(260563) |
| 011 | 2 | -3 | -0.4629(261195) |
| 010 | 3 | -1 | -0.1543(261827) |
| 110 | 4 | 1 | 0.1543(316) |
| 111 | 5 | 3 | 0.4629(948) |
| 101 | 6 | 5 | 0.7715(1580) |
| 100 | 7 | 7 | 1.0801(2212) |

表 13: 64QAM の場合の Mapper

| message input | PreMap output | PostMap output | Nomalized output(18bit decimal) |
|---|---|---|---|
| 0000 | 0 | -15 | -1.1504(259787) |
| 0001 | 1 | -13 | -0.9971(260101) |
| 0011 | 2 | -11 | -0.8437(260416) |
| 0010 | 3 | -9 | -0.6903(260730) |
| 0110 | 4 | -7 | -0.5369(261004) |
| 0111 | 5 | -5 | -0.3835(261358) |
| 0101 | 6 | -3 | -0.2301(261672) |
| 0100 | 7 | -1 | -0.0767(261986) |
| 1100 | 8 | 1 | 0.0767(157) |
| 1101 | 9 | 3 | 0.2301(471) |
| 1111 | 10 | 5 | 0.3835(785) |
| 1110 | 11 | 7 | 0.5369(1099) |
| 1010 | 12 | 9 | 0.6903(1413) |
| 1011 | 13 | 11 | 0.8437(1727) |
| 1001 | 14 | 13 | 0.9971(2042) |
| 1000 | 15 | 15 | 1.1504(2356) |

表 14: 256QAM の場合の Mapper

# 6. 評価

前章にて設計した Phase Encryption を実現するための従来手法と提案手法についての設計を行った．これらを verilog 言語による回路のソースコードを論理合成することによって回路の面積を測定した．測定した従来手法の回路と提案手法の回路を比較し，どれほど小規模化できたか比較，評価した．

## 6.1 評価実験

使用したツールは Synopsys 社の Design Compiler を用いた．実装に用いたプロセスルールは ROHM 社の $0.18\mu m^2$ で ASIC 回路として実装した．

動作周波数を 10MHz と設定した．

またここで述べる回路面積はゲートの総面積のことで配線は含まれてはいない

表 15 は従来手法と提案手法の Phase Encryption の回路の面積である．combination は組み合わせ回路 sequential は順序回路を表す．16，17 は従来手法と提案手法の回路におけるモジュール別の回路面積を示す．なお calclate amp&phase の回路面積は amp と sin(phase)，cos(phase) の乗算を行うモジュールである Calculate encrypted symbol の回路面積を含んだ結果となっている．

| 回路実装手法 | circuit | 面積 ($\mu m^2$) |
|---|---|---|
| | | |
| 従来の Phase Encryption 回路 | Total | 62355.35 |
| | combinational | 44618.02 |
| | sequential | 17737.32 |
| 提案した Phase Encryption 回路 | Total | 1800.33 |
| | combinational | 1230.61 |
| | sequential | 569.71 |

表 15: 手法別の回路面積

| module | circuit | 面積 ($\mu m^2$) |
|---|---|---|
| Mapper | Total | 1634.54 |
| | combinational | 1200.47 |
| | sequential | 434.06 |
| Calculate sine&cosine | Total | 32929.02 |
| | combinational | 20835.35 |
| | sequential | 12093.66 |
| calclate amp&phase | Total | 28799.33 |
| | combination | 20458.63 |
| | sequential | 8319.67 |

表 16: 従来手法の各モジュールの回路面積

| module | circuit | 面積 ($\mu m^2$) |
|---|---|---|
| PreMap | Total | 204.22 |
| | combinational | 107.76 |
| | sequential | 96.45 |
| PostMap | Total | 1552.39 |
| | combinational | 1118.33 |
| | sequential | 434.06 |
| XOR Encryption | Total | 72.34 |
| | combinational | 72.34 |
| | sequential | 0 |

表 17: 提案手法の各モジュールの回路面積

## 6.2 比較と考察

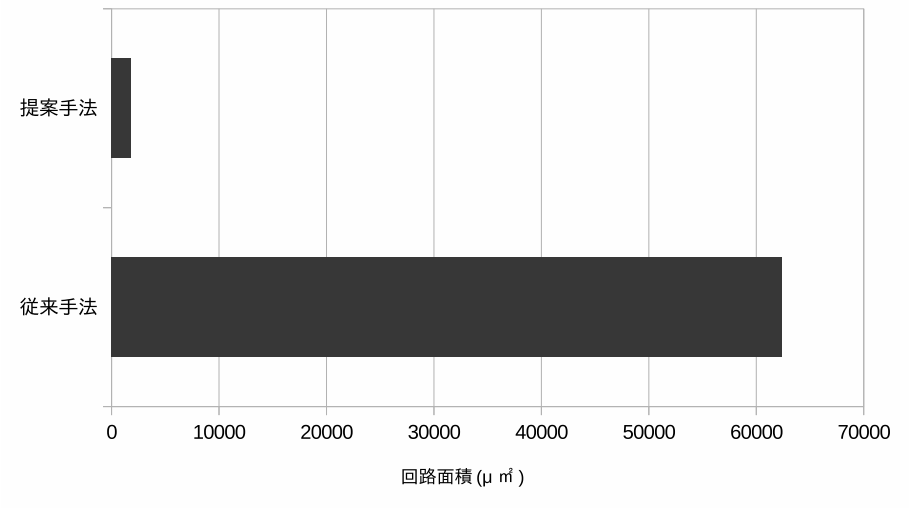


図 23: 回路面積の比較

図 23 を見ると提案手法の面積は従来手法に比べて 1/34 になっていることがわかる．CMOS 回路の消費電力 P は A(g) がゲート g の信号比率，f(g) がゲート g の周波数，V(g) ゲート g の電圧 C，(g) がゲート g の静電容量とすると G は回路のゲートの集合とすると

$$P = \sum_{g \subseteq G} A(g) \cdot f(g) \cdot C(g) \cdot V(g)^2 \tag{27}$$

に近似できる．[14]

今回周波数と電圧，静電容量はすべてのゲートで同じで回路の消費電力はゲート数におおむね比例するため，提案手法の回路面積の削減によって消費電力も削減できたと考えられる．

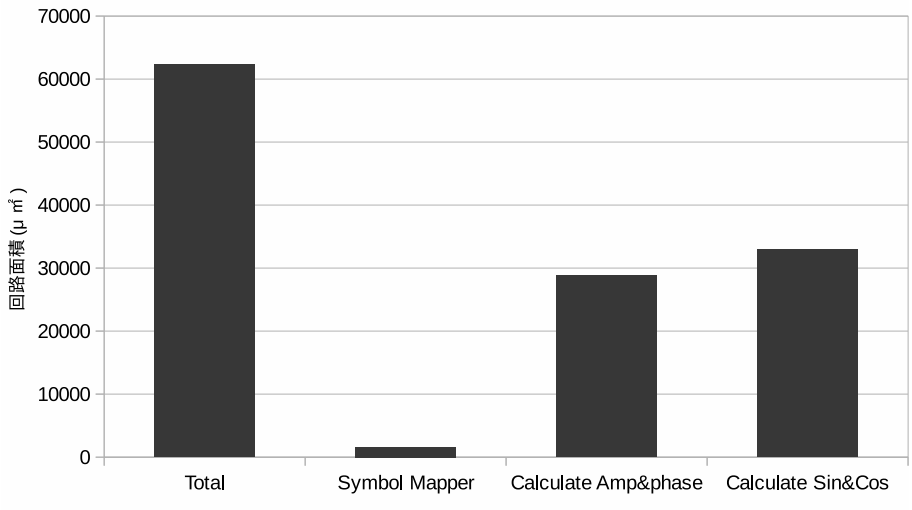


図 24: 従来手法のモジュールごとの回路面積の内訳

図24から極座標系と直交座標系の相互の変換を行うためのCalculate sine&cosine，calculate amp&phaseの両モジュールが従来手法の回路の99%を占めていることがわかる．

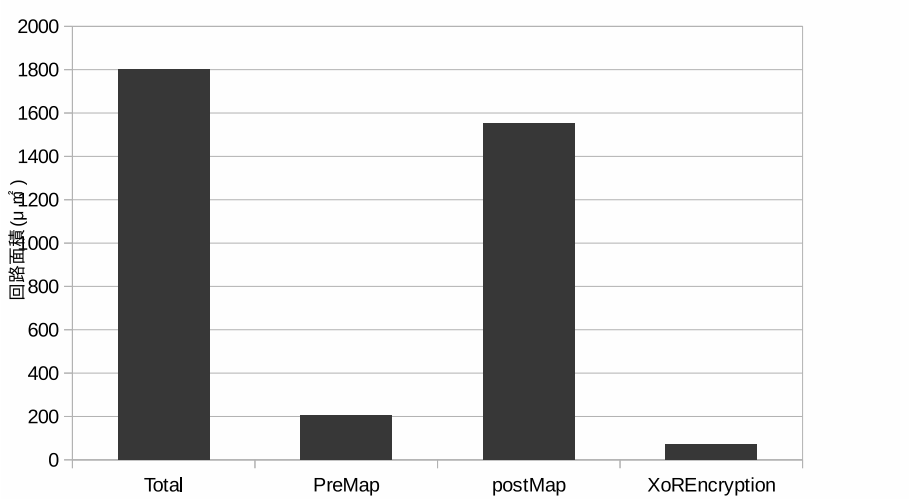


図 25: 提案手法のモジュールごとの回路面積の内訳

図25からPostMapモジュールが提案手法の回路の86%を占める対してPreMapモジュールが11%，XoR Encryptionモジュールが4%と小さいことがわかる．PostMap

モジュールは従来手法の SymbolMapper(Phase Encryption を使用しなくても物理層に実装される) とほぼ同じ大きさであるから提案手法によってごく小さい回路の追加で Phase Encryption が実現できたと言える.

# 7. おわりに

本稿ではIEEE802.11ah物理層のセキュリティとして変調より上位の層を隠蔽することができるPhase Encryptionの実装を行うにあたり，従来のアルゴリズムを実現するため回路を設計した．このアルゴリズムでは変調を行った後，シンボルを直交座標系から極座標系に変換し，各成分である振幅と位相を暗号化し，極座標系の各成分を直交座標系に再変換を行い暗号化された直交座標成分を生成する．しかし直交座標系から極座標系，またその逆の変換は三角関数計算を求める必要があり，そのための回路は複雑で巨大な回路面積を必要とした．

この問題の解決のために変調シンボルを直接暗号化できないかと考え，新しいPhase Encryptionのアルゴリズムを提案した．このアルゴリズムでは連続した整数成分を持つ疑似シンボルに変調し暗号化する．この疑似シンボルの集合の元は暗号化しても再び同じ集合に戻ってくる性質を持つ．そのため暗号化後シンボルでないもの遷移するというデメリットを持たない．暗号化後の変調シンボルが正規の変調シンボルにマッピングすることによって暗号化された正規の変調シンボルが生成できる．このアルゴリズムは設計において疑似シンボルを生成するPreMap，正規シンボルを生成するPostMap2つのMapperしか必要としない

提案手法と従来手法をVerilog HDLで実装して論理合成を行い，面積を見積もった．結果，提案手法の回路面積は従来手法の回路面積の1/34まで削減できた．

# 業績

1. 吉田 怜矢, トランティ ホン, 中島　康彦. "IEEE802.11ah における RTS/CTS のパフォーマンスの調査". 信学技報 IEICE Technical Report. RCS2018-3. 2018 年 4 月.

2. Duc-Phuc Nguyen, Dinh-Dung Le, Dai-Long Hoang, Satoya Yoshida, Tran Thi Hong, Yasuhiko Nakashima. "A Precise Indoor Localization System with Fixed Visible Light Communication LEDs for Smart Shopping". 2017 International Conference for Top and Emerging Computer Scientists (ICTECS 2017). 2017 年 12 月.

# 謝辞

研究活動を通じ，多大なるご指導ご鞭撻をいただいた本学の中島康彦教授に深く感謝申し上げます．本稿をご精読いただき研究活動の場などで貴重なご意見ご助言を頂いた岡田実教授に深く感謝の意を表します．研究活動や報告の場に多くのご助言をいただいた本学の中田尚准教授に深く感謝いたします．また日ごろの研究活動におけるご指導を与えてくださった TRAN Thi Hong 助教授に心から感謝申し上げます．また発表の場でご助言頂いた Renyuan ZHANG 助教授， 木村 睦先生に感謝の意を申し上げます．また研究活動から日常生活にいたるまで 2 年間をともに過ごし支えてくれた同輩の上竹規之氏， 菊谷 雄真氏，平賀 由利亜氏，山根弘樹氏そして Hoang Gia Vu 氏，福岡久和氏，山野龍佑氏，Nguyen Duc Phuc 氏，一倉 孝宏氏， Hoang Dai Long 氏， Le Dinh Dung 氏，Tati Erlina 氏， Nguyen Van Tinh 氏， Khong Thi Thu Thao 氏，Tran Thi Diem 氏，池田 裕哉氏，岩本淳氏， 西本 宏樹氏，新谷 隆太氏，Pham Hoai Luan 氏をはじめとするコンピューティングアーキテクチャ研究室の皆様には感謝の念が絶えません．最後に大学院生活を支えてくださりどんな時も暖かく見守ってくれた家族に感謝いたします．

# 参考文献


[1] 石井聡. トランジスタ技術 11 月号. 2013.

[2] 富士通株式会社. スマートシティへの取組. http://www.soumu.go.jp/main_content/000377859.pdf(2019-1-17 に閲覧)".

[3] 啓悟長柄. 組込みシステム向けマルウェア mirai の攻撃性能評価. Technical Report 41, mar 2017.

[4] Linux iot デバイスを狙う「mirai」ボットネットの拡散と ddos 攻撃に注意. "https://eset-info.canon-its.jp/malware_infonews/detail/161006.html(2019-1-15 に閲覧)".

[5] 総務省. 第 1 部第 3 節 iot 化する情報通信産業. http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h29/html/nc133100.html(2019-1-17 に閲覧)".

[6] Devesh Bundhoo. M. Razvi Doomun, K.M. Sunjiv Soyjaudah. Energy consumption and computational analysis of rijndael-aes. In *IEEE/IFIP Int. Conf. Cen-tral Asia Internet*, 2007.

[7] B. Sun R. Wang Y. Xiao, H. Chen and S. Sethi. Mac security and security overhead analysis in the ieee 802.15.4 wireless sensor networks. In *J. Wireless Commun.*, 2006.

[8] R. Wang Dai Long Hoang, Thi Hong Tran and Yasuhiko Nakashima. Performance evaluation of 802.11ah physical layer phase encryption for iot applications. In *International Conference on Advanced Technologies for Communications.*, 2018.

[9] Micheal Kloos. Method and apparatus for encryption of over-the-air communications in a wireless communication system . US7693284B2., 2010.

[10] Guang Gong. Fei Huo. Xor encryption versus phase encryption, an in-depth analysis. In *2014 IEEE International Symposium.*, Vol. vol.57 .

[11] 802.11 Working Group of LAN/MAN Standards Committee of IEEE Computer Society. *Part 11:Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) Specification.* 2016.

[12] Ray Andraka . A survey of cordic algorithms for fpga based computers., 1998.

[13] 東海大学理学部 遠藤研究室. 関数電卓のしくみ (cordic アルゴルズム). http://teamcoil.sp.u-tokai.ac.jp/calculators/column/100224/(2019-1-17 に閲覧), 2010.

[14] 石原亨. ソフトウェアに対する電力見積りと電力削減技術. In *Fundamentals Review Vol2,No.3.* 電子情報通信学会, 2009.