

Race Logic 高性能化に向けた
ナノフォトリック・デバイスを用いた実装

浅井里奈

平成28年1月

情報知能工学科

Race Logic 高性能化に向けた ナノフォトニック・デバイスを用いた実装

浅井里奈

内容梗概

配列アラインメントナノフォトニクス高性能化

目次

1	はじめに	1
2	Race Logic とその実装に関する課題	3
2.1	Race Logic	3
2.2	配列アラインメント	5
2.3	CMOS による Race Logic 実装	8
3	ナノフォトニック・デバイスを用いた Race Logic 実装の提案	12
3.1	光デバイスについて	12
3.1.1	光デバイスの特徴	12
3.1.2	光素子の説明	13
3.1.3	ナノフォトニクス of 性質	18
3.2	提案回路	20
4	検証・評価	24
4.1	検証	24
4.2	評価	25
5	おわりに	26
	謝辞	27
	参考文献	28

第 1 章

はじめに

高性能化や低消費電力を実現させるために、多くのアプリケーションにおいて専用アクセラレータが検討されてきた。その中でも、動的計画法（Dynamic Programming, DP）によって解くことができる最適化問題の専用アクセラレータとして、”Race Logic” と呼ばれる新しいコンピューティングのアプローチが提案された [1]。Race Logic の有効性は、CMOS を用いて実装された DNA のグローバル配列アラインメントスコアを求める回路にて明らかにされている。

一方、ナノフォトニクスと呼ばれる新しい光素子技術が注目を集めている。このナノフォトニクスを用いて機能を実現したデバイスをナノフォトニック・デバイスという。ナノフォトニック・デバイスは光速度で演算を実現できる素子として注目されており、パターン検出機構や加算器などのアーキテクチャの検討がされている。

本研究では、ナノフォトニック・デバイスを用いたアーキテクチャの検討一環として Race Logic の考えに基づくアクセラレータを構成することを目的とした。本論文では、ナノフォトニック・デバイスを用いた DNA のグローバル配列アラインメントスコアを求める回路を提案し、シミュレータを用いた回路の機能検証とモデル式を用いた評価を行うことでその有効性と実現に向けての問題点について検討する。

本論文の構成は以下の通りである。第 2 章で Race Logic 及び配列アラインメントの基本原則を説明し、CMOS を用いて実装された Race Logic に基づく回路例について述べる。第 3 章では、光デバイスとナノフォトニクスの基本事項を説明し、ナノフォトニック・デバイス

を用いた Race Logic に基づく回路を提案する．第 4 章では提案した回路に対してシミュレータを用いた検証とモデル式を用いた評価を行い，第 5 章でまとめを行う．

第 2 章

Race Logic とその実装に関する課題

本章では、まず Race Logic の基本原理を説明する。具体的な評価の対象アプリケーションである配列アラインメントと CMOS による実装例をまとめ、その後 Race Logic の検討において解決すべき課題について述べる。

2.1 Race Logic

Race Logic の基本概念は、回路に設定された競争条件を利用して、計算を実行することである。Race Logic の計算は有向非巡回グラフ (Directed Acyclic Graph, DAG) の最短・最長経路探索に帰結する。有向非巡回グラフとは、グラフ理論における閉路のない有向グラフの事である。その例を図を 2.1 に示す。有向グラフは頂点と有向辺から構成され、辺は頂点同士をつなぐが、ある頂点 v から出発し辺をたどっても、頂点 v に戻ってこないというものである。Race logic では最短・最長経路を探索するために、各パスをにおいて条件に合わせて重み付けを行い、出力時にその重みの総和を見る。その総和の最小値を見るか最大値を見るかが、最短・最長経路探索にそれぞれ対応している。

重みに遅延時間を選択することによって、各パスを通過する信号毎に出力のタイミングが変わってくる。信号が Race Logic を用いた回路に入力されてから出力されるまでの遅延時間を計測することが、パスの重みの総和を見ることに等しい。最短経路検索においては一番早く出力された信号の出力タイミングを、最長経路検索においては一番遅く出力された信号

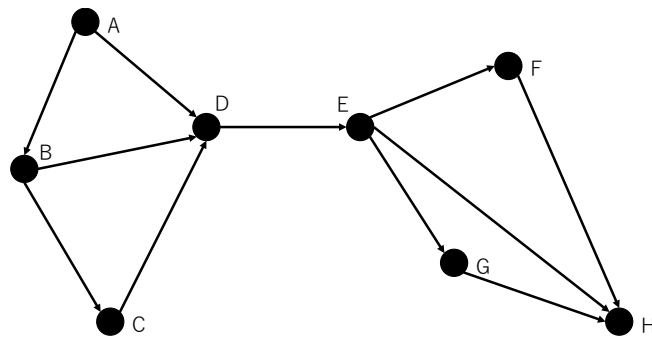


図 2.1: 有向非巡回グラフの例

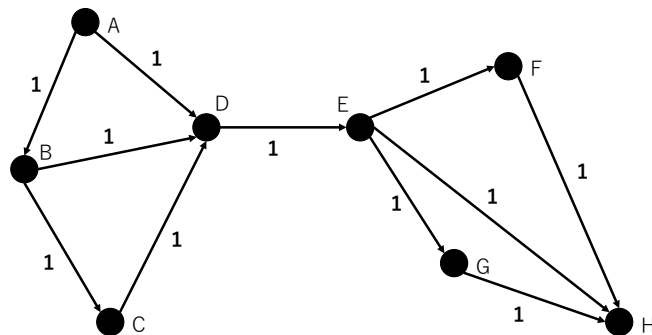


図 2.2: Race Logic の例

の出力タイミングのみを見る．つまり，出力のタイミングを競うレースに勝利した信号の出力タイミングのみを見る，ということであり，Race Logic の名前の所以でもある．出力信号の遅延時間がとある情報を持つ計算結果となる．例として，図 2.2 の頂点 A に信号を入力し，頂点 H から出力されるものを考える．これは図 2.1 の一辺を通過する際の遅延時間を 1 としたものである．頂点 A に信号を入力してから頂点 H に至る経路は， $A \rightarrow D \rightarrow E \rightarrow F \rightarrow G$ ， $A \rightarrow D \rightarrow E \rightarrow G \rightarrow H$ ， $A \rightarrow D \rightarrow E \rightarrow H$ ， $A \rightarrow B \rightarrow D \rightarrow E \rightarrow F \rightarrow G$ ， $A \rightarrow B \rightarrow D \rightarrow E \rightarrow G \rightarrow H$ ， $A \rightarrow B \rightarrow D \rightarrow E \rightarrow H$ ， $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow G$ ， $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow G \rightarrow H$ ， $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow H$ の 9 通り考えられ，各経路を通った信号の遅延時間はそれぞれ 4, 4, 3, 5, 5, 4, 6, 6, 5 である．よって最短経路は $A \rightarrow D \rightarrow E \rightarrow H$ であり，一番最初に出力される信号の遅延時間は 3 である．一辺の通過の際の遅延時間を 1 としたので，一番最初に出力された信号の遅延時間は頂点 A から頂点 H までの距離を表している．

このように出力信号の遅延時間はとある情報を持っており，どのような情報を持つかは応



図 2.3: グローバル配列アラインメントとローカル配列アラインメント

用するアプリケーションによって異なる。

2.2 配列アラインメント

生物学の分野において注目されている生物配列 (DNA の塩基配列とタンパク質アミノ酸配列) の文字列処理 (配列情報解析) [2, 3] の中でも、今回は配列アラインメントに焦点を当てる。

DNA やタンパク質はユニットと名付けられた単位の物質が一行に並んだ高分子である。ここでいうユニットとは、DNA においては 4 種の核酸、タンパク質においては 20 種類のアミノ酸である。それぞれのユニットを文字とし DNA やタンパク質の配列を単なる文字列だとみなして処理をしてもある種の本質は失われないという考えに基づき、文字列処理をすることで生物配列の解析を行なっている。DNA の塩基配列やタンパク質アミノ酸配列の研究は、バイオインフォマティクスの最重要課題の一つとして取り組まれてきた。配列情報解析の重要な対象であるゲノム塩基配列は、すでに 200 種類以上が決定されており、さらに多くの解析が進行中であるといわれている [4]。

生物配列の文字列処理の中で、DNA 配列中に同じ順序で並んでいるユニットのパターンを見つける配列アラインメントがある [5]。アラインメントとは、複数の配列を入力として配列要素の間に最適な対応関係を求める処理である。配列アラインメントの中には、グローバル配列アラインメントとローカル配列アラインメントというものがある。グローバル配列アラインメントは、ある 2 つ以上の配列全体の間でのアラインメントを見つけることであり、ローカル配列アラインメントは、ある 2 つ以上の配列の一部で最も一致度が高い部分でのアラインメントを見つけることである。配列 P と配列 Q のグローバル配列アラインメントとローカル配列アラインメントを例として図 2.3 に示す。

配列アラインメントには、動的計画法による解法として Needleman-Wunsch アルゴリズム

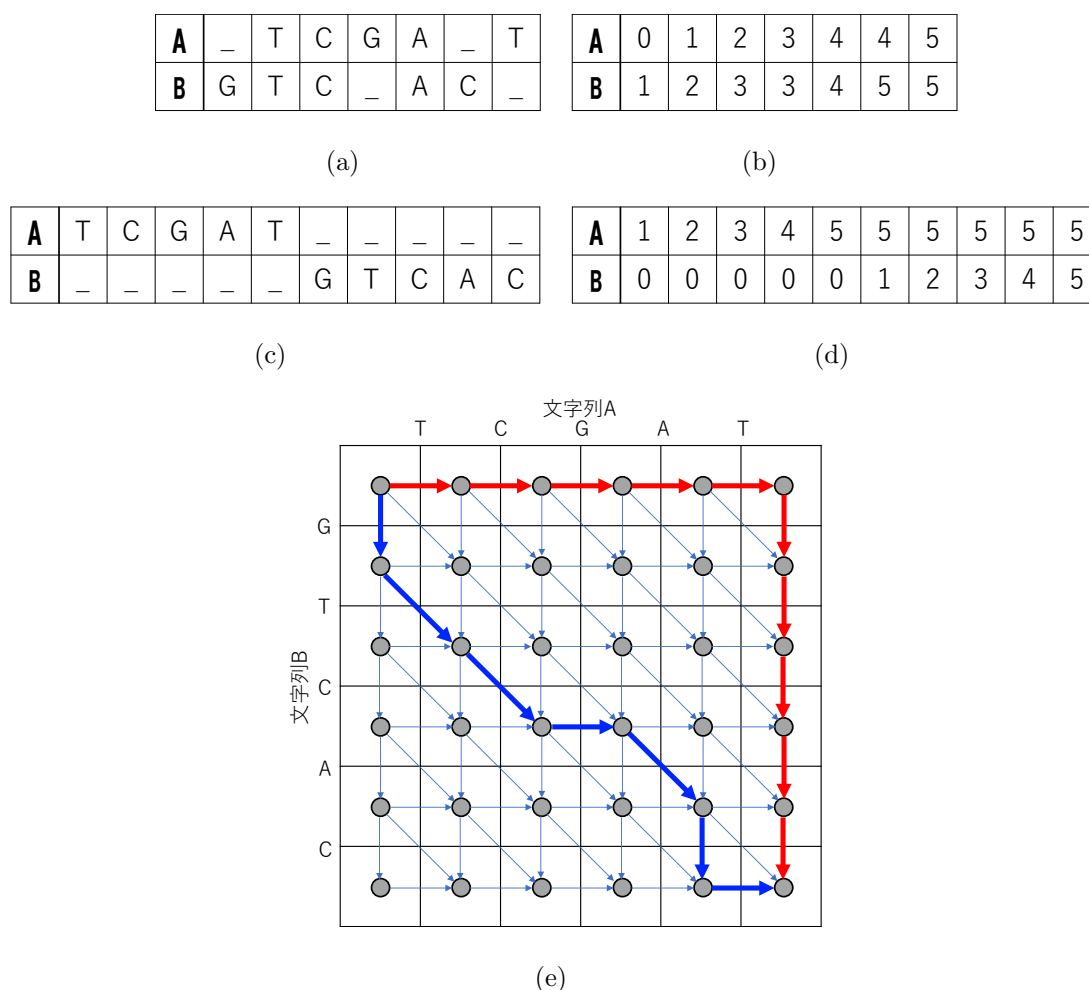


図 2.4: 編集距離と編集グラフ

ム [6] や Smith-Waterman アルゴリズム [7] が存在する．配列アラインメントは生物学において重要な手法であり，計算機を用いた処理の高速化は従来より多くの研究がなされてきた [8-12]．

文字列の類似度を知るための典型的な手法は，情報理論に由来する編集距離 (edit distance) である．編集距離は，一方の文字列をもう一方の文字列に変形するのに必要な手順である一文字の挿入・削除・置換にそれぞれ編集スコアを割り当て，そのスコアの和として定義される．編集距離の理解を助けるために，長さ $N = 5$ の文字列 $A = \text{"TCGAT"}$ と長さ $M = 5$ の文字列 $B = \text{"GTCAC"}$ を考える．図 2.4(a) と図 2.4(c) は，文字列 A を B に変換する 2 つの方法を示している．上の行のスペースは挿入を表し，下の行のスペースは削除を表す．両方の行に同じ文字がある列は一致と呼ぶ．図 2.4(a) の方法は文字 G と T を削除し， G と C を挿

入する一方で、図 2.4(c) の方法は文字列 A を完全に削除して文字列 B を挿入している。

図 2.4(b) および図 2.4(d) は、2つのアラインメント方法の代替表現である。任意の位置の数字は、図 2.4(a) および図 2.4(c) の方法においてその位置までに存在する記号の数を示している。この表示は各列の数値が図 2.4(e) に示す 2次元の編集グラフの座標と考えることができる。このグラフを編集グラフという。編集グラフは2つの文字列間において、取りうる限りの配置の二次元表現である有向非巡回グラフ (DAG) である。全てのエッジが編集操作に対応していて、垂直の矢印は挿入を、水平の矢印は削除を、斜めの矢印は一致を表している。任意のアラインメントはこのグラフのパスで表現できる。例えば、図 2.4(e) の青と赤の矢印は、それぞれ図 2.4(a) および図 2.4(c) に示す2つのアライメントに対応している。

2つの文字列間の編集距離は、動的計画法を用いて計算できる。動的計画法は小さな部分問題から始めて次第に大きな問題を漸進的に解決し、各ステップはそれ以前の計算の結果に依存している。編集グラフ上の各ノードは、部分問題の最適解に対応するスコア、すなわち最初のノードから自身への最短経路に対応するスコアを計算している。隣接するノードは、計算が対角線に沿って進むにつれてそれ以前の最適解を利用して自身のスコアを計算する。編集グラフ自体は最初のノードから最後のノードまでの経路として、表現される可能性のあるすべてのアライメントから構成されている。よって、上記の方法は比較対象の文字列間の最適なアライメントについて空間全体の検索が保証されている。

任意の2つの文字列が与えられた場合、多数の異なるパスがあり、それぞれが独自のアラインメントを持つ。ある特定のアライメントの相対的なメリットを決定するためにスコアマトリックスが導入される。このスコアマトリックスは効果的に編集グラフの各エッジの重みを定義することができる。スコアマトリックスとはあるユニット間の一致不一致や置換確率から求められるスコアからなる行列である。一致・不一致スコアと対数オッズスコアの式を式 2.1 と式 2.2 にそれぞれ示す。

$$S(A, B) = \begin{cases} \alpha & A = B \\ \beta & A \neq B \end{cases} \quad (2.1)$$

$$S(A, B) = \log \frac{q(A, B)}{p(A)p(B)} \quad (2.2)$$

ここで $q(A, B)$ は進化上の関係から A と B の対応が生じた確率、 $p(A)$ は偶然に A が生じた確率、 $p(A)p(B)$ は偶然に A と B の対応が生じた確率である。図 2.5 には式 2.1 に具体的な数値

	A	C	T	G
A	2	1	1	1
C	1	2	1	1
T	1	1	2	1
G	1	1	1	2

	A	C	T	G
A	1	2	2	2
C	2	1	2	2
T	2	2	1	2
G	2	2	2	1

(a) 最長経路探索に用いられるスコアマトリックス (b) 最短経路探索に用いられるスコアマトリックス

図 2.5: スコアマトリックスの例

を当てはめた場合のスコアマトリックス例を示す。一般的に、不一致のペナルティは特定の文字のペアにも依存することに注意が必要である。

あるノードにおけるスコアの最大値と最小値を求める関数は式 2.3a と式 2.3b のように書ける。

$$S_{i,j} = \max \begin{cases} S_{i-1,j} + \delta(-, P_j) \\ S_{i,j-1} + \delta(Q_i, -) \\ S_{i-1,j-1} + \delta(Q_i, P_j) \end{cases} \quad (2.3a)$$

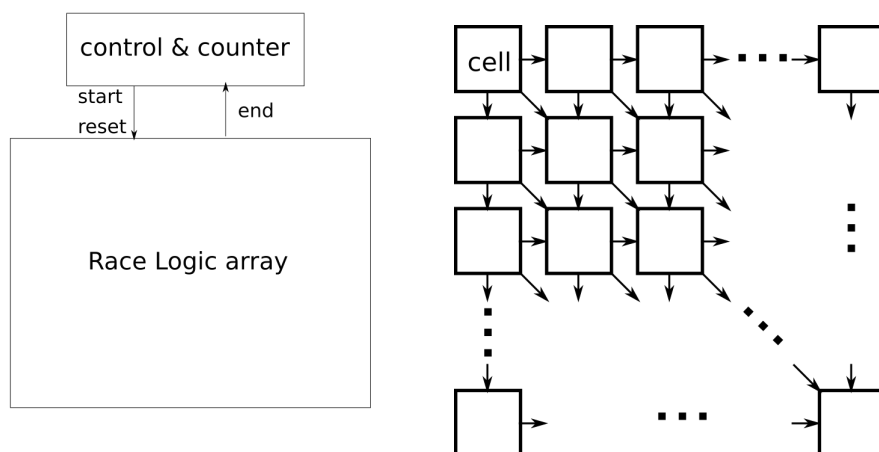
$$S_{i,j} = \min \begin{cases} S_{i-1,j} + \delta(-, P_j) \\ S_{i,j-1} + \delta(Q_i, -) \\ S_{i-1,j-1} + \delta(Q_i, P_j) \end{cases} \quad (2.3b)$$

i と j は図 2.4(e) に示す行と列のインデックスである。

アラインメントのメリットを決定することとはつまり、図 2.5(a) のマトリックスと式 2.3a を用いて最長経路を探索すること、または図 2.5(b) のマトリックスと式 2.3b を用いて最短経路を探索することと等しい。

2.3 CMOS による Race Logic 実装

本節では、CMOS を用いた Race Logic 実装例として、配列アラインメントアクセラレータを説明する。Race Logic の考えを用いて実装された配列アラインメントアーキテクチャの



(a) Race Logic を用いた配列アラインメントアクセラレータの構成

(b) Race Logic Array の構成

図 2.6: 配列アラインメントアクセラレータの基本構造

基本構造を図 2.6 に示す。この Race Logic Array はセルと呼ばれる単位のユニットが繰り返される構造をとっている。図 2.4(e) に示す編集グラフに対して、Race Logic Array が編集グラフ全体に、セルがノードに対応する。セルは上・斜上・左のセルから信号の入力を受け付ける。信号が入力された後、設定された条件に合わせて適切な処理をした後に、次のセルへと信号を出力する。実装の選択肢として、セルへの信号伝搬をクロックと同期させる同期型と、セルへの信号伝搬をクロックと同期させない非同期型とがある。

最短経路探索を行うことで DNA グローバル配列アラインメントスコアを得る配列アラインメントアクセラレータについて、同期型と非同期型を見ていく。

● CMOS による同期型 Race Logic 実装

CMOS で実装された同期型 Race Logic のセルの構造を図 2.7 に示す。

ブール値“1”の信号は左・斜上・上のセルのいずれからとも入力される。入力された信号は OR ゲートを通過して、飽和アップカウンタにおいて N クロックサイクルに 0 をカウントする。この飽和アップカウンタをクロックと同期させる。これにより、1 つのセルを通過し、右・下のセルへと伝搬する際に 1 クロックサイクルを要する。各着色ゲートの出力は、所望の重量に達した時点でトリガーする特定の重量を表しており、アルファベットの符号化を入力とする MUX から所望の重量を選択することができる。

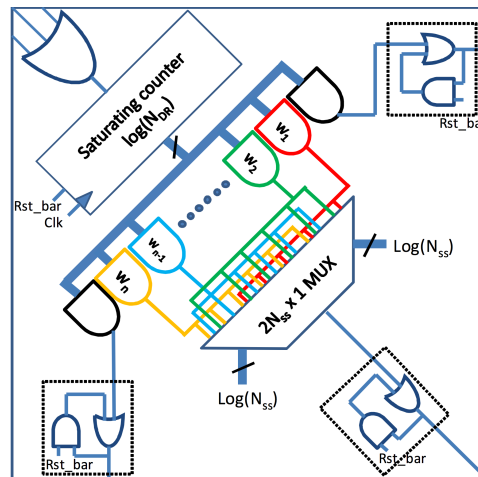


図 2.7: CMOS で実装された同期型 Race Logic のセル構造 [1]

生成される出力信号がパルスではなく固定ブール値“1”であることを確実にするために、到着回路のセットが配置され、各計算の最後にリセットされている。

図 2.7 のセルを繰り返した構造を持つアレイに信号が入力された時から出力信号を得るまでのクロック数をカウントするカウンタがアレイ外部に存在する。このカウンタが計測した値が最短経路をパスした時のクロック数となる。

- CMOS による非同期型 Race Logic 実装

CMOS で実装された非同期型 Race Logic のセルの構造を図 2.8 に示す。

ブール値“1”の信号は左・斜上・上のセルのいずれからも入力される。入力された信号は OR ゲート、リセットのための AND ゲートを通過する。アレイのリセットのタイミングは、全ての遅延素子が確実にリセットされるように外部から調整される。AND ゲートを通過後に次のセルへの伝搬に向けて分けられ、それぞれの経路で遅延素子を通過する。各着色した経路では遅延素子によって起こる遅延に変化をつけている。同期型と同様、アルファベットの符号化を入力とする MUX から所望の遅延を選択することができる。ダミーのパスゲートは、全ての遅延経路に亘って同様の遅延を保証するために斜下へのパス以外に追加されている。

図 2.8 のセルを繰り返した構造を持つアレイに信号が入力された時から出力信号を得るまでの遅延時間をカウントするカウンタがアレイ外部に存在する。

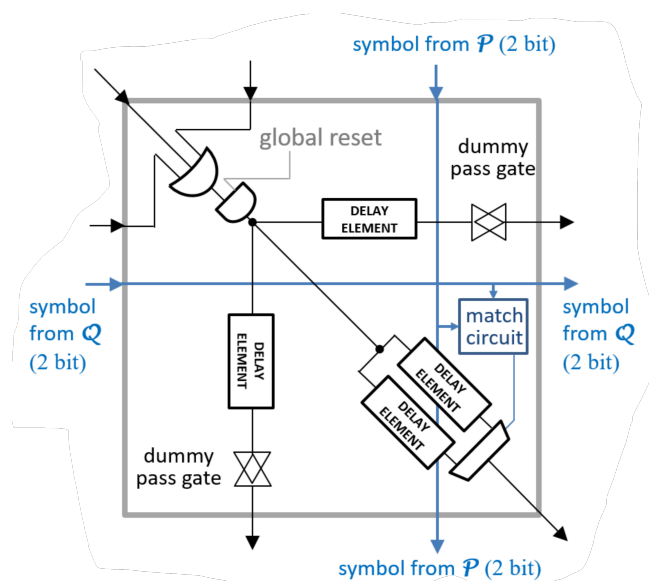


図 2.8: CMOS で実装された非同期型 Race Logic のセル構造 [13]

上記の回路については、シミュレーションをもってその有効性が明らかにされ、性能・面積・消費電力密度などが報告されている。

第 3 章

ナノフォトニック・デバイスを用いた Race Logic 実装の提案

本章では，本提案の光 Race Logic を理解する上で必要な光デバイスに関する基本事項を説明し，ナノフォトニック・デバイスによる Race Logic 実装について述べる．

3.1 光デバイスについて

本節では，まず光デバイスの特徴について述べる．その後，代表的な光素子について説明し，最後にナノフォトニクスについて解説する．

3.1.1 光デバイスの特徴

以下に光デバイスの特徴をまとめ，詳細を説明する．光デバイスは光が信号を伝搬する素子全般のことを指す．一方，本論文では電気デバイスとは電気が信号を伝搬する CMOS トランジスタを指すものと定義する．

- デバイスサイズ

現状，光デバイスのゲート長は $cm \sim mm$ オーダーのスケールである．後述するナノフォトニクスを用いたとしても，そのスケールのオーダーは μm である．

- 信号の周波数帯域

光デバイスにおいて、信号の伝搬は光信号が通過するか否かで行われる。電気デバイスのように時定数によって周波数帯域が制限されることがない。よって、その周波数帯域は広帯域であると言える。

- データの蓄積

電気デバイスは、電荷を貯めることでデータを保持できる。光デバイスは光を留めておくことが難しいため、データの蓄積は困難である。電気デバイスの方が、光デバイスよりもデータの蓄積が容易であると言える。

- スループット

データ処理やネットワークにおけるスループットについて述べる。スループットとは単位時間あたりの処理量や処理可能なデータ量のことである。光信号は多重性と呼ばれる複数の異なる周波数を多重して送る事ができる性質を持つ。光デバイスは光信号の広帯域性や、波長多重、位相多重と言った多重性を利用して、伝搬信号自体の情報量を増加させることで、データ転送速度を向上させることが可能である。

電気デバイスはデバイスの小型化が可能という特徴から集積度を上げることが可能である。それに比べ、現状では光デバイスはデバイスの小型化に向いておらず、集積度を上げることが困難であった。よって演算には電気デバイスが用いられてきた。一方、光デバイスは伝搬信号の情報量が大きく、信号の移動速度も速いという特徴から通信に使われてきた。

3.1.2 光素子の説明

本項では、代表的な光素子について説明する。その説明にあたり、語句を定義する。

- 光伝搬信号

光デバイスおよび、そのデバイスを用いて構成した回路において、情報を伝搬する光信号を指す。

- 光伝搬入力信号および光伝搬出力信号

光デバイスおよび、そのデバイスを用いて構成した回路において、入力される光伝搬

信号を光伝搬入力信号，出力される光伝搬信号を光伝搬出力信号と呼ぶ．光入力信号および光出力信号と略す．

- 光伝搬入力信号強度および光伝搬出力信号強度

光伝搬入力信号および光伝搬出力信号の信号強度を指す．単位は [W] である．光入力信号強度および光出力信号強度と略す．

- 光制御信号

光デバイスを制御するための光信号を指す．

- 光制御信号強度

光制御信号の信号強度を指す．単位は [W] である．

- 電気制御信号

光デバイスを制御するための電気信号を指す．

- 電気制御信号強度

電気制御信号の信号強度を指す．単位は [V] である．

光スイッチ

光スイッチとは，光伝搬信号を通過させるか否かでオン動作およびオフ動作をする光デバイスである．光スイッチの性能指標として，よく用いられるのが漏れ率および透過率，消光比である．光スイッチへの光入力信号強度を P_{in} ，光出力信号強度を P_{out} とした場合の，光入力信号強度と光出力信号強度の関係を図 3.1 に示す．図の縦軸は光出力信号強度，横軸は時間を表している．理想的なスイッチでは，光伝搬信号の漏れが無い場合光出力信号強度は図の青線に示す関係を取る．しかしながら，実際にはオン動作とオフ動作どちらの場合でも光伝搬信号の漏れがあるため，光出力信号強度は図の赤線に示す関係を取る．以下にそれぞれの性能指標の定義を示す．

- 漏れ率，透過率

漏れ率とは，スイッチがオン動作とオフ動作の際，それぞれどの程度の光伝搬信号が漏れるかということを表す指標である．図 3.1 に示す α は，スイッチがオン動作の際

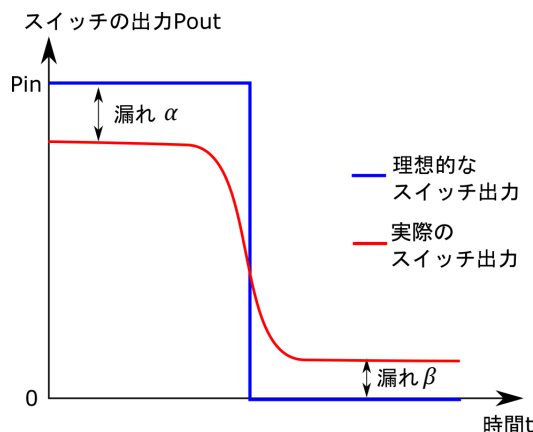


図 3.1: 光スイッチの光入力信号強度と光出力信号強度

にスイッチから回路外へ光伝搬信号がどの程度漏れ出すかを表す漏れ率である。また、スイッチがオン動作の際にどの程度の光伝搬信号を透過させられるかを表す指標を透過率と呼び、漏れ率 α を用いて表すと $1 - \alpha$ となる。光スイッチへの光入力信号強度を P_{in} 、オン動作をする際の光出力信号強度を P_{1out} とすると、 $\frac{P_{1out}}{P_{in}} = 1 - \alpha$ である。図 3.1 に示す β は、スイッチがオフ動作の際に光伝搬信号を遮断しきれずに、どの程度出力へ漏れ出すかを表す漏れ率である。光スイッチへの光入力信号強度を P_{in} 、オフ動作をする際の光出力信号強度を P_{0out} とすると、 $\frac{P_{0out}}{P_{in}} = \beta$ である。 α 、 β の値が小さいほどスイッチの性能が高いと言える。

- 消光比

消光比とは、スイッチの光出力信号が 1 と 0 の場合の光出力信号強度比である。透過率 $1 - \alpha$ および漏れ率 β を用いると、式 (3.1) で表される。

$$\text{消光比} = \frac{1 - \alpha}{\beta} \quad (3.1)$$

消光比は、透過率および漏れ率を用いて議論することが可能であるとわかる。よって本論文では透過率、漏れ率に着目し、これらをスイッチ性能として議論する。

受光器

光の素粒子は一般に光子 (フォトン) と呼ばれる。全ての粒子が波動性を持つことを、粒子と波動の二重性と言う。光子も粒子性と波動性の 2 つの性質を持つ [14]。光子のエネルギーは光の周波数 (波長) で決定する。

$$E = h\nu \quad (3.2)$$

E は光子のエネルギー、 h はプランク定数、 ν は光の周波数である。光の強度は光子の数によって決定する。

物質中の電子のエネルギーは、取り得るエネルギー準位が限定されている。そのエネルギー準位は帯構造を取り、図 3.2 に示すようにそれぞれ伝導帯、禁制帯、価電子帯と呼ばれる。伝導帯とは、電子が占めているエネルギー帯のうち最も高いエネルギー準位を示すエネルギー帯である。この伝導帯は電子が充填されておらず、このエネルギー帯に存在する電子は自由電子として振る舞う。価電子帯は価電子によって充填されたエネルギー帯である。禁制帯とは電子が存在できないエネルギー帯である。この禁制帯の幅が図 3.2 に示す E_g であり、エネルギーギャップと呼ばれる。半導体物質において、エネルギーギャップを超えるのに十分なエネルギーを持った光子 1 つが入射した際に、自由電子と正孔のペア 1 つを生成する。この現象を吸収という。光子のエネルギーはその光の周波数で決まるため、エネルギーギャップの大きさに対応した周波数がある。逆の現象が、放出である。これは、自由電子と正孔が再結合した際に、そのエネルギーギャップ $E_g = h\nu$ に相当するエネルギーを持つ光子を放出する現象である。図 3.3 に吸収と放出の様子を示す。図 3.3(a) におけるエネルギーギャップが、緑の光の光子の持つエネルギーと等しいとする。この際、緑の光を入射すると電子正孔対が生成される。しかしながら、赤の光は緑の光よりも周波数が小さいため、光子のエネルギーが緑の光と比べて小さい。よって赤の光を入射しても電子は伝導帯へと励起することができず、電子正孔対は生成されない。受光器であるフォトダイオードはこの吸収の現象を利用して光を検出する。

フォトダイオードは p 型半導体と真性半導体と n 型半導体を接合した pin 接合という構造を持ち、空乏層で発生した電子や正孔が移動することで電流が流れる。この電流のことを光電流と呼ぶ。流れる光電流の大きさは光の強度に比例する。図 3.4 に受光器のエネルギーバンド図を示す。

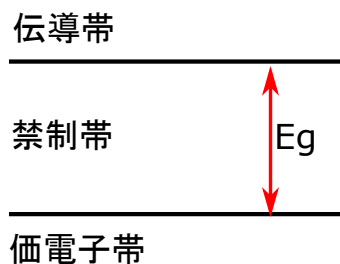


図 3.2: 半導体のエネルギーバンド図

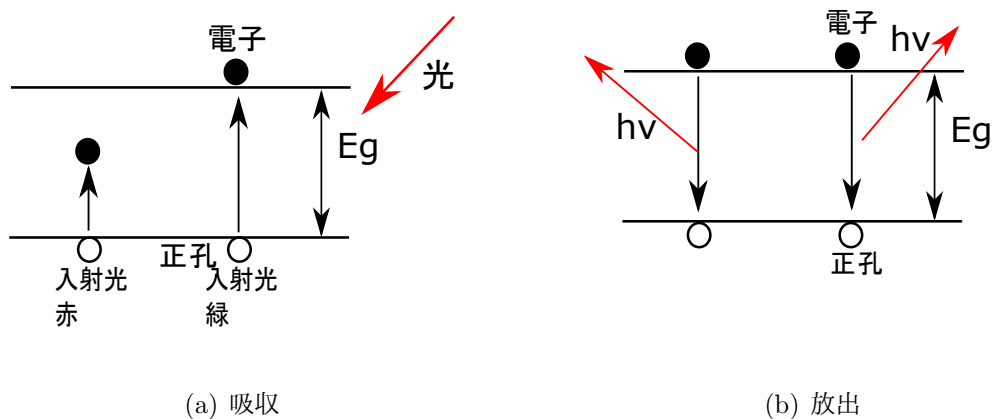


図 3.3: 光の吸収と放出

受光器の性能は受光感度として表される．受光感度は，光入力信号強度を $[W]$ ，光電流を $[A]$ で表した場合，両者の比で表される．受光感度は式 (3.3) で表される．式 (3.3) によって導かれる値が 1 に近い程，感度の良い受光器であることを示す．

$$\text{受光感度} = \frac{A}{W} \quad (3.3)$$

もう一つ重要な性能指標が受光器の最小受光感度である．最小受光感度とは，受光器が検出可能な最小の信号強度のことである．

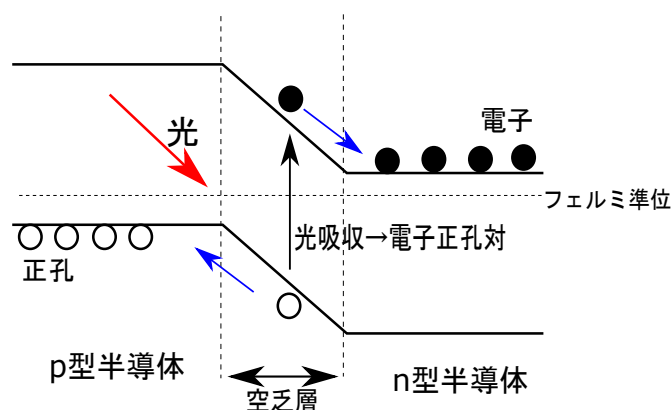


図 3.4: 受光器のエネルギーバンド図

光カプラ

光カプラとは、1つの入力端子に入射した光伝搬信号を複数の出力端子に出射する分岐・分配機能と複数の入力端子に入射した光伝搬信号を1つの出力端子に出射する結合機能を持つデバイスである。光分岐結合器、光分岐挿入器とも呼ばれる。光カプラの分類を図3.5に示す。4端子で2入力2出力または3入力1出力のものが光方向性結合器(図3.5(a))、1入力N出力のものが光分配器もしくは光分岐器(図3.5(b))、N入力1出力のものが光結合器(図3.5(c))、N入力M出力のものが光スターカプラ(図3.5(d))と呼ばれている。光結合器の多くが入出力端子を逆にすることで光分配器としても使うことができる[15]。また、光の結合・分配の際には損失が発生する。

光遅延素子

光遅延素子は光伝搬信号に伝搬遅延を付加できる素子のことを指す。その実現は様々な方法があり、いずれの方法を用いても遅延を付加する際には損失が発生する。

3.1.3 ナノフォトニクス of 性質

ナノフォトニクスとは、ナノ加工技術をベースとして、近接場光の性質を活かした技術である。近接場光とは物質の表面付近に局在する非伝播な電磁場であり、その局在範囲は光の

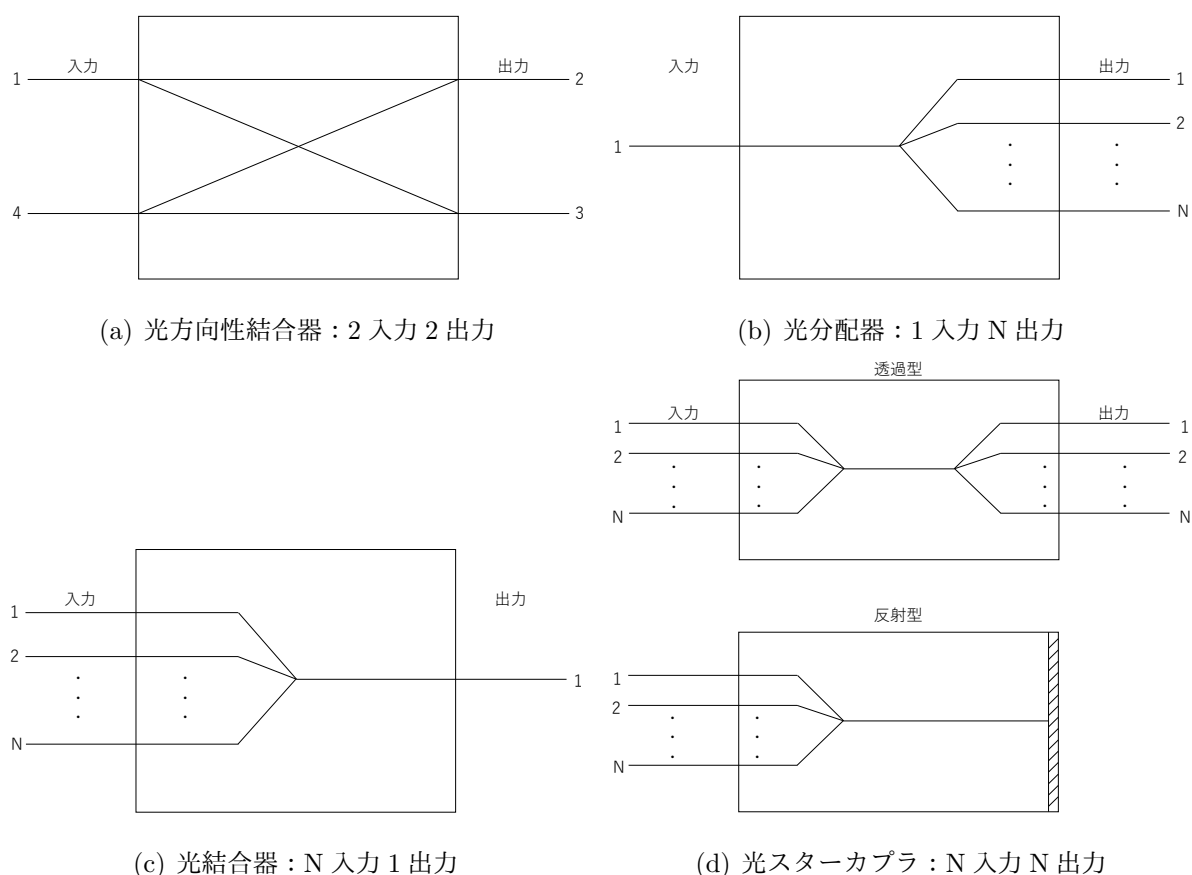


図 3.5: 光カプラの分類

波長と同程度かそれに比べ小さい．近接場光の概要を図 3.6 に示す．物質から遠ざかるにつれて電磁場は減少するため，その特徴からエバネッセント光とも呼ばれる．屈折率の大きい媒質から屈折率の小さい媒質に光を入射させる．この場合ある角度を超えると，光は境界面を通過せず全て反射する．この現象を全反射と言う．物質表面に全反射が起こるように光を入射した際，反射が起こっている物質境界面付近では局在する電磁場が発生する．この電磁場光がエバネッセント光である．エバネッセント光が発生した際，境界面から離れる方向に電磁場が弱くなる．図 3.7 はエバネッセント光が発生する様子である．従来の光通信で広く使われている光素子の素材はガラスである．これを半導体微細加工技術を用いて，半導体へと置き換える．この技術によって半導体内を光が伝送できる．ガラスから半導体へと素材を変えただけでは，光素子のゲート長のスケールは cm から mm のサイズに小さくなる．これを μm のスケールにするためには，半導体などのナノ加工技術がベースとしてある．半導体のナノ加工技術が素子の加工技術のベースになり，近接場光の局在性をはじめとする近接場

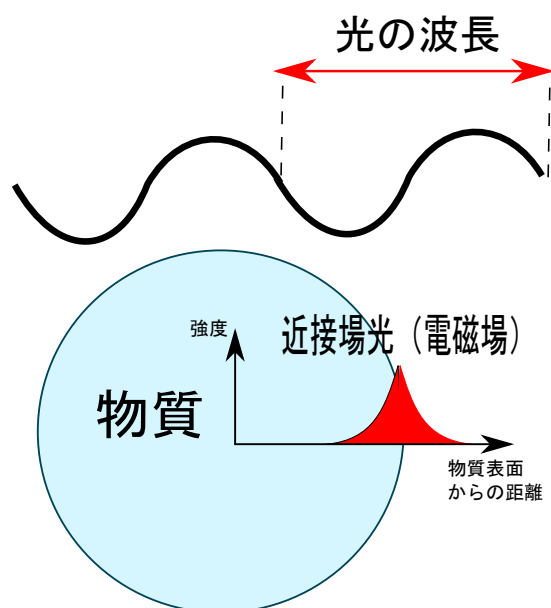


図 3.6: 近接場光

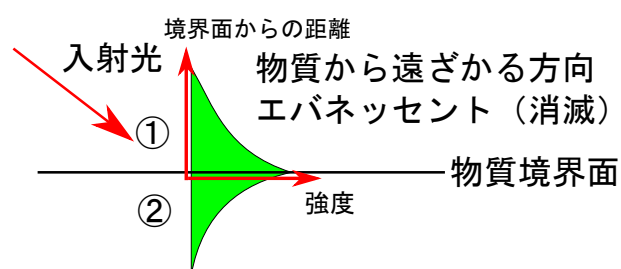


図 3.7: エバネッセント光

光にしかない特徴を活かして，光信号をナノレベルで制御することが可能になって成り立つ技術がナノフォトニクスである．

3.2 提案回路

ナノフォトニック・デバイスを用いた Race logic（以下，光 Race Logic）回路のセル構造を考えていく．図 3.8 に光 Race Logic 回路のセルの概要を示す．

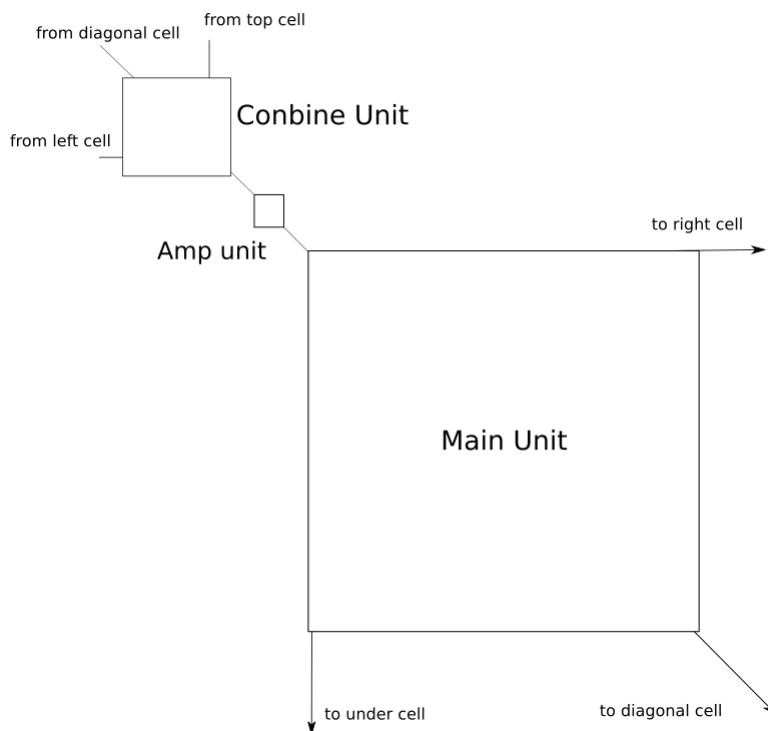


図 3.8: 光 Race logic 回路におけるセルの概要

セルは大きく分けて3つのユニットに分けて考えることができる。まずは入力ユニットで、これは左・斜上・上のセルからの入力経路を持つ。入力の次は合波ユニットで、ここは合波とそれに付随する処理を行うユニットである。最長経路探索や最短経路探索、回路に設定する競争条件に合わせてその処理・構成が変わる。その後、光伝搬信号はアンプユニットで光強度を増幅された後、メインユニット到達する。メインユニットでは光伝搬信号の分波と任意の処理を施して、光伝搬信号を次のセルへと出力する。このユニットの構成と処理は回路に設定された競争条件によって変化する。また、同期・非同期の処理に関しては合波ユニットもしくはメインユニット、あるいはその両方の構成で設定する。対象とするアプリケーションや設定する競争条件によって、光 Race Logic 回路の構成は詳細に決定する。

今回は DNA のグローバル配列アラインメントスコア計算を対象とした非同期型光 Race Logic 回路を提案する。計算に用いるスコアマトリクスは式 2.1 を用いて求めた。そのスコアマトリクスを図 3.9 に示す。最下行と最右列はギャップスコアを表している。

	A	C	T	G	-
A	1	∞	∞	∞	1
C	∞	1	∞	∞	1
T	∞	∞	1	∞	1
G	∞	∞	∞	1	1
-	1	1	1	1	1

図 3.9: 使用するスコアマトリクス

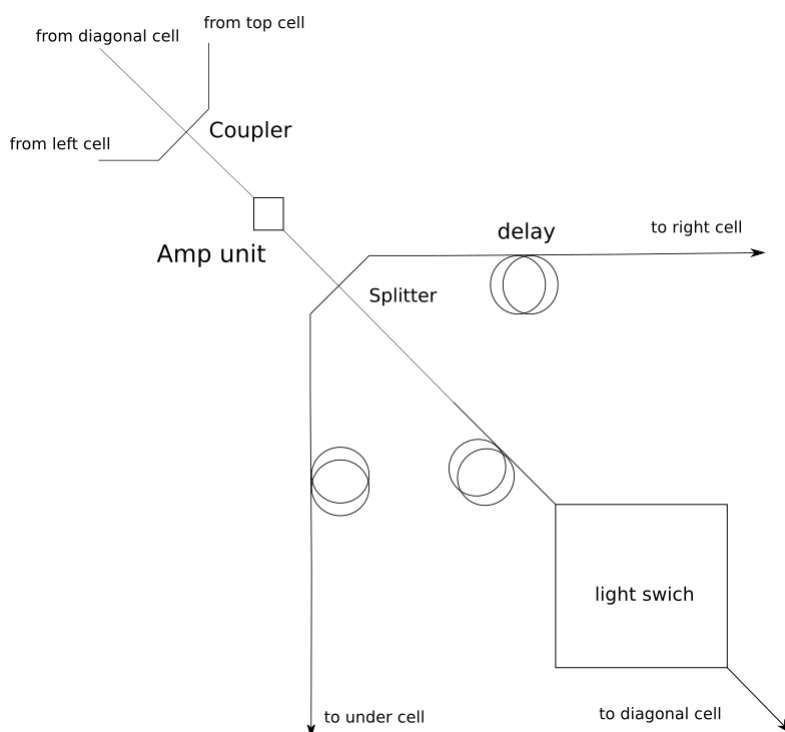


図 3.10: 提案回路のセル構造

図 3.10 に提案する回路のセル構造を示す．提案回路のセルにおいて，合波ユニットでは 3 入力 1 出力の光カプラを用いて合波が行われる．この時，3 方向から同時に光信号が入力された場合，それぞれの光入力信号が光カプラに到達するまでの時間が同じになるよう調整する必要がある．メインユニットでは 1 入力 3 出力の光カプラを用いて右，下，斜め下へに分波され，各伝搬経路ごとにスコアマトリクスに基づいての重み付けがされている．右，下へ

の伝搬経路ではギャップスコアに基づく遅延が発生するように、光遅延素子を設定する。斜め下への伝搬経路では、比較する文字が一致した場合のスコアに基づく遅延が発生するように光遅延素子を設定する。その後、光スイッチを通過する。この光スイッチでは、比較する文字列が一致した場合 ON、不一致の場合に OFF になるように設定する。光スイッチが ON の時、光伝搬信号は一致スコアに対応した遅延時間を付加されて斜め下のセルへと伝搬する。光スイッチが OFF の時、光伝搬信号は斜め下のセルに伝搬されない。これは無限大の遅延時間を付加された、と考えることができ、不一致スコアに対応した重み付けがされたとみなすことができる。また、文字列が一致し光スイッチが ON の場合、すなわち 3 方向に信号が伝搬される場合、次のセルへの出力のタイミングも同時になるように調整する必要がある。このセルを並べることで DNA 配列アラインメントスコア計算を対象とした光 Race Logic 回路の array を構築する。アレイに信号を入力してから出力されるまでの伝搬遅延が 2 つの文字列間の最適なアラインメントスコアに対応している。また、今回の提案で非同期型を選択したのは、光は蓄積が困難であるという点を考慮した結果である。

この提案回路について、次章で検証・評価を行う。

第 4 章

検証・評価

本章では提案した DNA 配列アラインメントスコア計算用の光 Race Logic 回路について機能検証と評価を行う。

4.1 検証

検証に用いたのは Optiwave 社が提供する Optisystem というシミュレータである。OptiSystem は光ネットワークのあらゆるタイプの広範囲のシステムの設計、評価、シミュレーションを行なうソフトウェアである。素子レベルからシステムレベルまでの物理レイヤー上の光通信システムの設計と解析を行うことができる。

文字列長 $N=2$ の提案回路について、光 Race Logic array の動作を確認した。今回のシミュレーションにおいて、各素子において光伝搬信号に影響を与える雑音やロスは考慮していない。また Optisystem の仕様上、遅延素子で付与された遅延時間のみが考慮され、素子や導波路の伝搬遅延については考慮されていない。その結果を図??に示す。図??の nsns の区間が文字列が完全一致の場合の出力、nsns の区間が文字列が一文字不一致の場合の出力、nsns の区間が文字列が完全不一致の場合の出力である。それぞれにおいて、出力から最初に観測される信号のタイミングは入力から ns 後、ns 後、ns 後である。図??を見ると、想定した動作をしているのが確認できる。

4.2 評価

第 5 章

おわりに

謝辞

本研究の進行および本論文執筆にあたりまして，懇切丁寧なご指導を頂きました井上弘士教授に心より感謝申し上げます．本論文執筆にあたり，多大なご指導を頂きました小野貴継助教に心より感謝申し上げます．本研究を行うにあたり，多大なご指導を賜りました日本電信電話株式会社 物性科学基礎研究所 ナノフォトニクスセンタ 主幹研究員 新家昭彦様に感謝の意を表するとともに厚く御礼申し上げます．本論文執筆にあたり多大な指導頂きました，井上研究室大学院博士2年川上哲志氏に深く感謝致します．

最後に，井上研究室の皆様の御意見，御厚意に感謝の意を表します．

参考文献

- [1] Advait Madhavan, Timothy Sherwood, and Dmitri Strukov. Race logic: A hardware acceleration for dynamic programming algorithms. *ACM SIGARCH Computer Architecture News*, Vol. 42, No. 3, pp. 517–528, 2014.
- [2] 浅井潔ほか. 配列情報と確立モデル (「分子生物情報学の新展開」). 人工知能学会誌, Vol. 15, No. 1, pp. 35–42, 2000.
- [3] 後藤修. マルチプルアラインメントは生体高分子情報の交差点. 生物物理, Vol. 38, No. 2, pp. 52–56, 1998.
- [4] 浅井潔ほか. バイオインフォマティクス: 7. ポストゲノム時代の配列情報解析. 情報処理, Vol. 46, No. 3, pp. 271–276, 2005.
- [5] 須戸里織, 吉見真聡, 三木光範. バイオインフォマティクスゲノム配列から機能解析へバイオインフォマティクスゲノム配列から機能解析へ, 2005. 電子情報通信学会技術研究報告. CPSY, コンピュータシステム, Vol. 111, No. 328, pp. 35–40, 2011.
- [6] Saul B Needleman and Christian D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, Vol. 48, No. 3, pp. 443–453, 1970.
- [7] Temple F Smith and Michael S Waterman. Identification of common molecular subsequences. *Journal of molecular biology*, Vol. 147, No. 1, pp. 195–197, 1981.
- [8] 須戸里織, 吉見真聡, 三木光範. Gpu を用いた 3 次元 smith-waterman 法の高速化手法の提案. 電子情報通信学会技術研究報告. CPSY, コンピュータシステム, Vol. 111, No. 328, pp. 35–40, 2011.

- [9] 宗川裕馬, 伊野文彦, 萩原兼一ほか. 統合開発環境 cuda を用いた gpu での配列アライメントの高速化手法. 情報処理学会研究報告計算機アーキテクチャ (ARC), Vol. 2008, No. 19 (2008-ARC-177), pp. 13–18, 2008.
- [10] Edans Flavius de O Sandes and Alba Cristina MA de Melo. Smith-waterman alignment of huge sequences with gpu in linear space. In *Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International*, pp. 1199–1211. IEEE, 2011.
- [11] Yu Liu, Yang Hong, Chun-Yuan Lin, and Che-Lun Hung. Accelerating smith-waterman alignment for protein database search using frequency distance filtration scheme based on cpu-gpu collaborative system. *International journal of genomics*, Vol. 2015, , 2015.
- [12] 伊野文彦, 小谷裕基, 萩原兼一ほか. Gpu グリッドによる高速な塩基配列アライメント. 情報処理学会研究報告ハイパフォーマンスコМПユーティング (HPC), Vol. 2007, No. 80 (2007-HPC-111), pp. 73–78, 2007.
- [13] Advait Madhavan, Timothy Sherwood, and Dmitri Strukov. A 4-mm 2 180-nm-cmos 15-giga-cell-updates-per-second dna sequence alignment engine based on asynchronous race conditions. In *Custom Integrated Circuits Conference (CICC), 2017 IEEE*, pp. 1–4. IEEE, 2017.
- [14] 大津元一, 小林潔. ナノフォトニクスの基礎. オーム社, 2006.
- [15] 三木哲也, 須藤昭一. 光通信技術ハンドブック. (株) オプトロニクス社, 2002.