

情報科学実験 1

2. フリップフロップと同期式回路

実験グループ: A 班

報告者: J2200071 齊藤 隆斗

共同実験者: J2200038 小淵 萌, J2200104 知久拓生, J2200158 矢島 琴恵

実験実施日: 2024 年 6 月 6 日

レポート提出日: 2024 年 6 月 13 日

提出期限: 2024 年 6 月 13 日

1. 実験の目的

論理回路から得られる出力が、その時点で与えられている入力のみならず、それまでに与えられてきた過去の入力にも依存する場合、そのような論理回路を「順序回路 (sequential circuit)」と呼ぶ。本実験では、この順序回路を構成する際に基本的な要素となるフリップフロップおよびレジスタについて詳しく取り上げる。具体的には、フリップフロップとレジスタの構成およびその基本的な動作原理について理解することを目的とする。さらに、同期式論理回路の一例として、レジスタと全加算器 (full adder) を用いた直列加算器の構成についても取り扱う。これにより、同期式論理回路がどのように機能するかを学ぶ。

2. 原理

2.1. フリップフロップ

論理値の 0 か 1 を安定状態として保持できる 1 ビットのメモリをフリップフロップという。フリップフロップが保持している論理値をそのフリップフロップの状態という。

2.1.1. SR フリップフロップ

SR-FF は、入力 S と R を持つ。動作原理は以下の通りとなる。

- S (Set) 入力が 1 で、R (Reset) 入力が 0 の場合、フリップフロップはセット状態になり、出力 Q は 1 に、 \bar{Q} は 0 になる。
- S 入力が 0 で、R 入力が 1 の場合、フリップフロップはリセット状態になり、出力 Q は 0 に、 \bar{Q} は 1 になる。
- S と R の両方が 0 の場合、出力は変化せず、前の状態が保持される。
- S と R の両方が 1 の場合は、通常、禁止されている操作であり、フリップフロップの動作は未定義（または特定の設計に依存）になる。

2.1.2. JK フリップフロップ

JK フリップフロップは、2 つの入力 (J と K) と 2 つの出力 (Q と \bar{Q}) を持つ。動作原理は以下の通りとなる。

- J=1、K=0 のとき、フリップフロップはセット状態になり、Q=1、 \bar{Q} =0 となる。
- J=0、K=1 のとき、フリップフロップはリセット状態になり、Q=0、 \bar{Q} =1 となる。
- J=0、K=0 のとき、フリップフロップの出力は変化せず、前の状態を保持する。
- J=1、K=1 のとき、フリップフロップはトグル状態になり、前の状態の反転 (Q が 1 なら 0 に、0 なら 1 に) を行う。

2.1.3. T フリップフロップ

入力信号 (T) とクロック信号に基づいて状態を変更する。T フリップフロップの動作原理は以下の通りである。

- T=1 のとき、クロックパルスが入力されるたびに出力が反転する。つまり、現在の出力が 0 の場合は 1 に、1 の場合は 0 に変わる。
- T=0 のとき、クロックパルスが入力されても出力は変わらず、現在の状態を維持する。

2.1.4. D フリップフロップ

入力データ (D) をクロック信号 (CLK) に基づいて出力に保持する。D フリップフロップの動作原理は以下の通りである。

- クロック信号の立上がりで $D=1$ ならば、出力 Q は 1 になる.
- クロック信号の立上がりで $D=0$ ならば、出力 Q は 0 になる.

このように、D フリップフロップはクロックエッジに同期してデータをキャプチャし、その後のクロックエッジまでそのデータを保持する.

2.2. レジスタ

n 個の FF を用いると、 n ビットのデータを記憶できる.これをレジスタという.シフトレジスタとは、外部からのクロックパルスに同期して、各 FF の内容を隣りの FF へ順次転送する機能を持ったレジスタである.

2.3. 直列加算器

通常一番下位の桁から一桁ずつ順に加算を行い、その桁での和と桁上げを計算する.次の桁では、その桁上がりを使って再び和と桁上げを計算することを繰り返す.この手順をハードウェアとして実現したものが直列加算器である.直列加算器とは、二つの n ビット 2 進数を下位の桁から 1 サイクルに一桁ずつ加えていく方法の加算器である.これは加算する 2 数を格納するシフトレジスタ X, Y , 加算結果を格納するシフトレジスタ S , 全加算器, D-FF で構成できる. X と Y の下位の方から一桁ずつ D-FF に保存された値を全加算器へ入力し、桁上げを再び D-FF へ保存する.

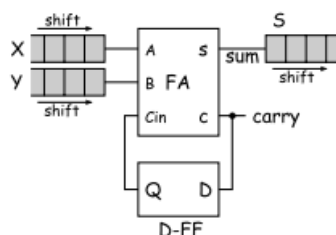


図 1: 直列加算器の構成

3. 実験方法

ロジックトレーナを用いてそれぞれのフリップフロップの回路を設計し、回路の挙動をタイミングチャートによって確認した.

4. 実験結果

4.1. フリップフロップの動作確認

1(a)

図 2 の回路を NAND ゲートと NOT ゲートを用いて構成した. この回路で観測したタイミングチャートは図 3 のようになる. 作成した回路が実験装置に予め備え付けられている SR-FF と同じ動作となることが確認できた.

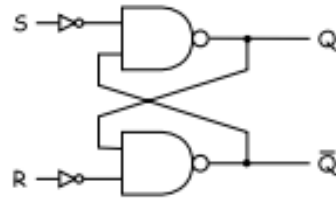


図 2: SR-FF

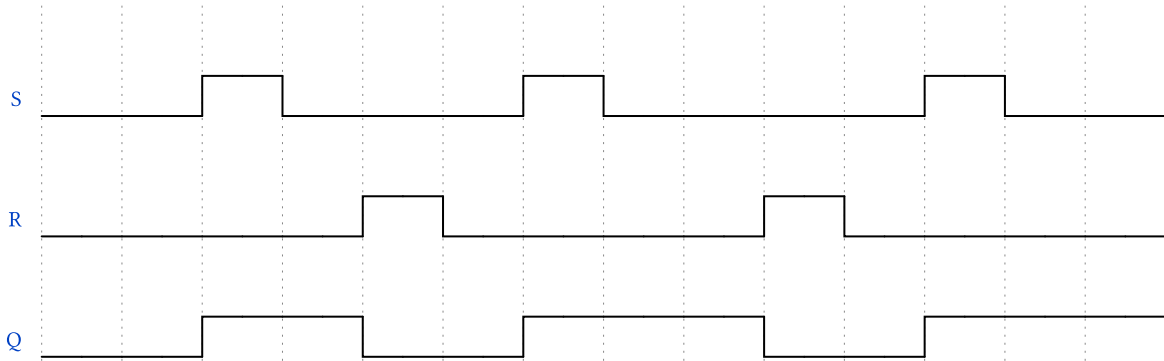


図 3: SR-FF のタイミングチャート

1(b)

図 4 のクロック付き SR-FF を NAND ゲートを用いて構成した. この回路で観測したタイミングチャートは図 5 のようになる. 作成した回路が実験装置に予め備え付けられている SR-FF と同じ動作となることが確認できた.

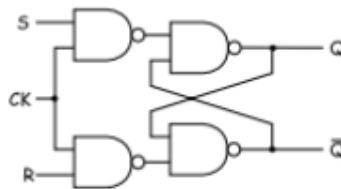


図 4: クロック付き SR-FF

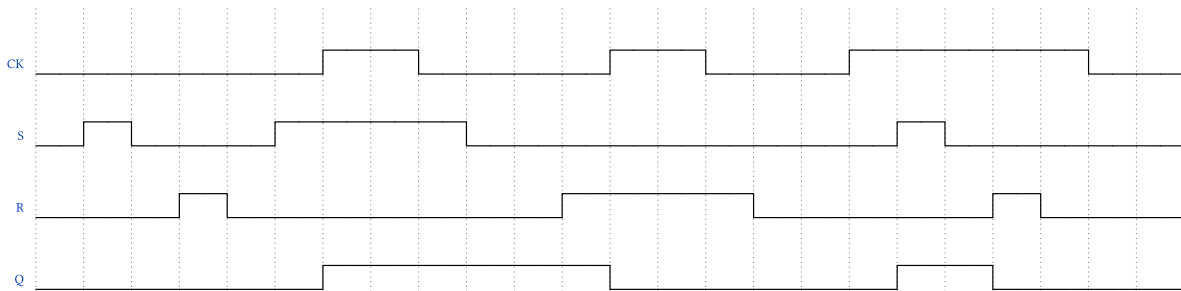


図 5: クロック付き SR-FF のタイミングチャート

1(c)

図 6 のマスタースレーブ式 SR-FF を作成した. この回路で観測したタイミングチャートは図 7 のようになる.

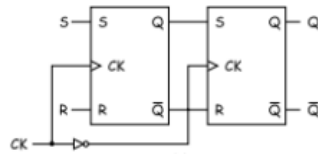


図 6: マスタースレーブ式 SR-FF

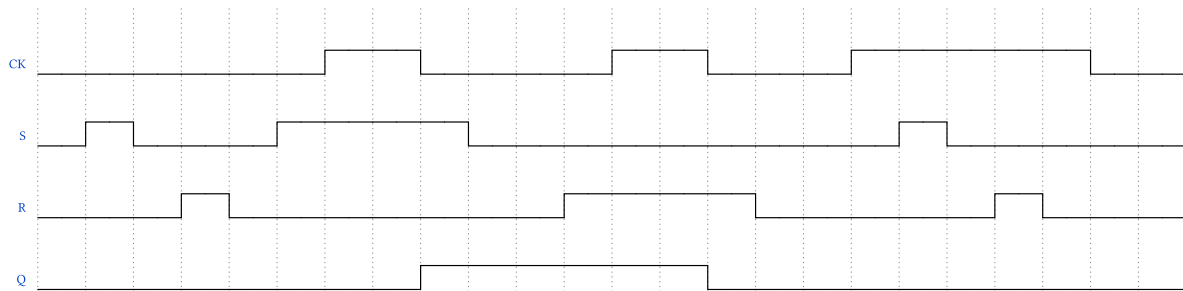


図 7: マスタースレーブ式 SR-FF のタイミングチャート

2(a)

図 8 のマスタースレーブ式 JK-FF をロジックトレーナ上のクロック付き SR-FF を使って作成した. この回路で観測したタイミングチャートは図 9 のようになる.

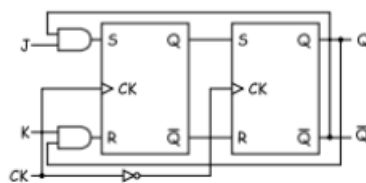


図 8: マスタースレーブ式 SR-FF

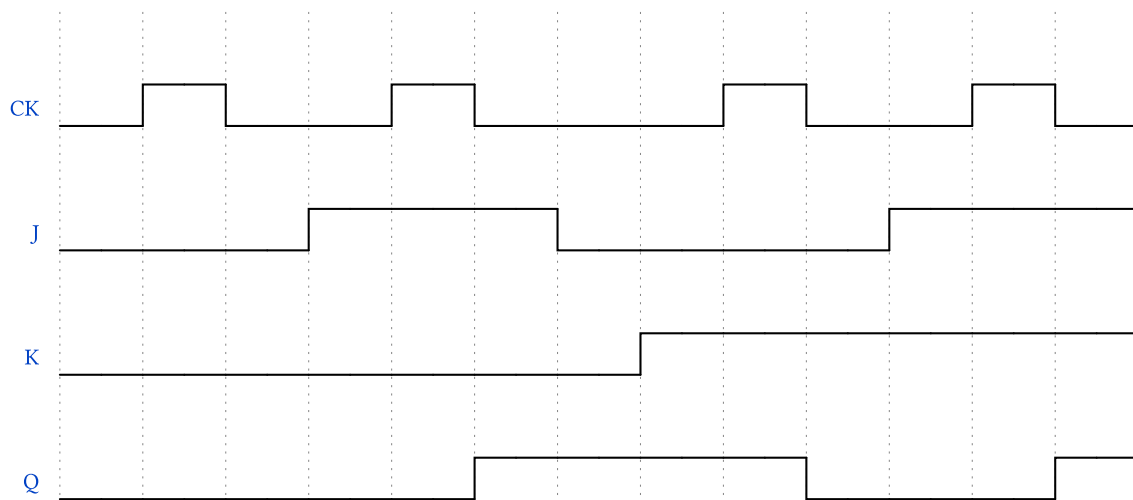


図 9: マスタースレーブ式 JK-FF のタイミングチャート

2(b)

マスタースレーブ式 JK-FF で $T = J = K$ とすることにより, T-FF を作成した. この回路で観測したタイミングチャートは図 10 のようになる.

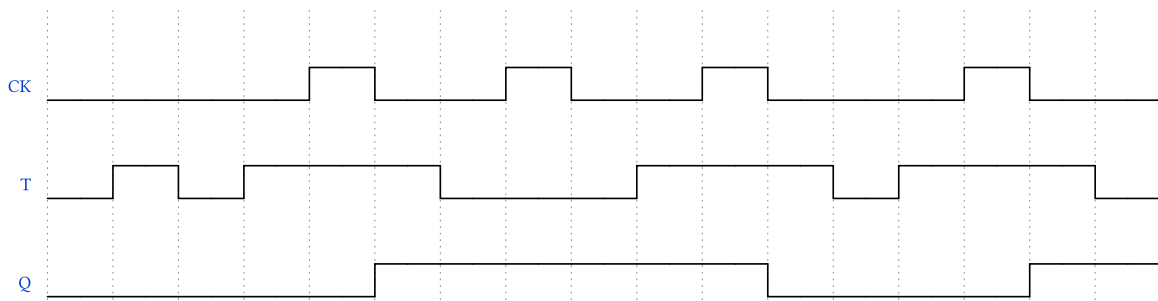


図 10: T-FF のタイミングチャート

3(a)

図 11 のようにクロック付き SR-FF で $D = S = R$ とすることで、レベルセンス式 D-FF を作成した。この回路で作成したタイミングチャートは 図 12 のようになる。

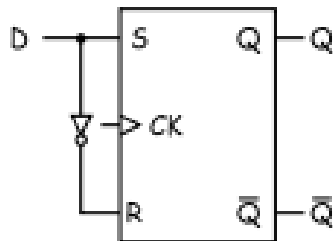


図 11: レベルセンス式 D-FF

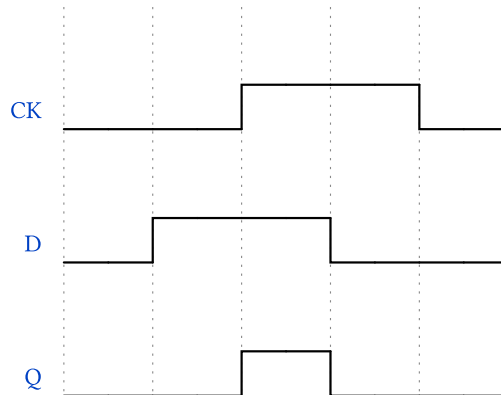


図 12: レベルセンス式 D-FF のタイミングチャート

3(b)

図 13 のようにマスタスレーブ式 D-FF を作成した。この回路で観測したタイミングチャートは 図 14 のようになる。これが、ロジックレーナ上のマスタスレーブ式 JK-FF で $D = J = K$ としたものと同じ動作であることを確認した。

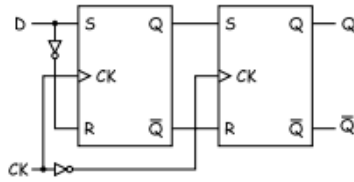


図 13: マスタースレーブ式 D-FF

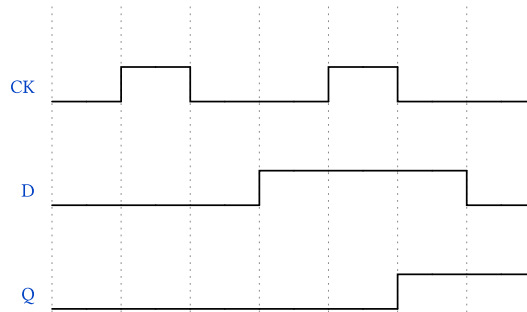


図 14: マスタースレーブ式 D-FF のタイミングチャート

4.2. レジスタ

1.

図 15 のようにレベルセンス式 SR-FF を使って、4 ビットシフトレジスタを構成し、その動作を確認した.

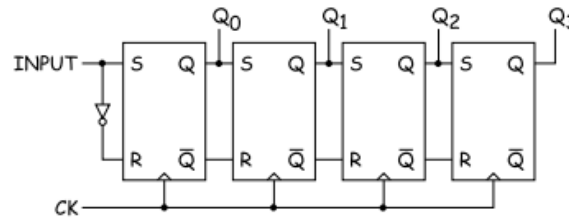


図 15: レベルセンス式 SR-FF を用いた 4 ビットシフトレジスタ

表 1: レベルセンス式 SR-FF を用いた 4 ビットシフトレジスタの状態遷移図

Input	$Q_0 Q_1 Q_2 Q_3$	$Q_0^+ Q_1^+ Q_2^+ Q_3^+$
1	0000	1000
0	1000	0100
0	0100	0010
1	0010	1001
1	1001	1100

2.

図 16 のようにロジックトレーナ上のマスタースレーブ式 JK-FF を使って、4 ビットシフトレジスタを構成し、その動作を確認した. レベルセンス式 SR-FF と同様の結果となることがわかった.

表 4: 直列加算器における 0111 + 0100 の計算の状態遷移図

XY	$Q_0Q_1Q_2Q_3Q_4Q_5$	$Q_0^+Q_1^+Q_2^+Q_3^+Q_4^+Q_5^+$
10	000000	100000
10	100000	101000
11	101000	111100
00	111100	000110
00	000110	001011

表 5: 直列加算器における 0110 + 0101 の計算の状態遷移図

XY	$Q_0Q_1Q_2Q_3Q_4Q_5$	$Q_0^+Q_1^+Q_2^+Q_3^+Q_4^+Q_5^+$
01	000000	010000
10	010000	101000
11	101000	111100
00	111110	000110
00	000110	001011

5. 検討課題

1. マスタースレーブ式のフリップフロップの動作について詳しく解説せよ.

マスタースレーブ式フリップフロップの動作について見ていく. 入力側の FF をマスター FF、出力側の FF をスレーブ FF と呼ぶ. $CK = 1$ となるとマスター FF が動作する. このとき、スレーブ FF はクロックは 0 となっているので出力を保持している. $CK = 0$ となるとスレーブ FF が動作する. このスレーブ FF の出力が、全体としての出力となる. また、このとき、マスター FF のクロックは 0 となっているので、マスター FF は、出力を保持する. このように $CK = 0$ となるようなタイミング、すなわちクロックの立ちさがりで出力が切り替わることで、 $CK = 1$ である時間内に複数回の状態遷移を防止することができる.

2. 図 18 の回路で $Input = 1$ かつ $CK = 1$ であるとき、出力 Q が不安定になる(振動する)理由を説明せよ. このフリップフロップをマスタースレーブ式 SR-FF に変更すると、クロック 1 個につき出力が高々 1 回しか変化しなくなり不安定な動作が解消される. その理由を説明せよ.

この回路において $CK = Input = Q = 0$ から、 $Input = 1$ とし、 $CK = 1$ となる場合について見ていく. この場合に状態が不安定となる. $Input = 1$ とすると、この時点では $Q = 0$ であるから排他的論理和をとると 1 となる. よって、入力 S に 1、入力 R に 0 が入る. この状態で $CK = 1$ とすると、 $Q = 1$ となる. ここで、 $Input = 1$ であるから、排他的論理和は 0 となる. よって、入力 S に 0、入力 R に 1 が入る. この状態で $CK = 1$ であるから、 $Q = 0$ となる. これらの処理が繰り返されることで状態が不安定となる.

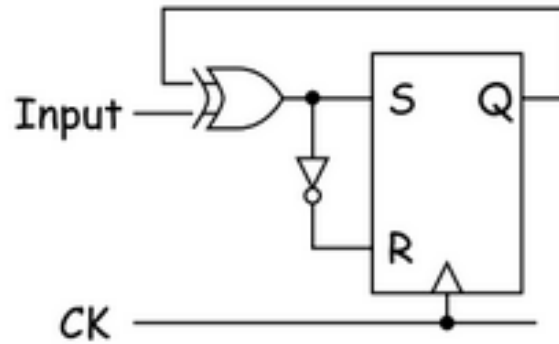


図 18: 状態が不安定になる順序回路

検討課題の1で検討したようにマスタスレーブ式フリップフロップでは、 $CK = 1$ である時間内に複数回の状態遷移を防止することができる。これによって、クロック1個につき、出力が高々1回しか変化せず、不安定な動作が改善される。

3. エッジトリガ式フリップフロップについて調べよ。

エッジトリガ式フリップフロップはクロックパルスに同期して動作するフリップフロップのことである。クロックパルスの立上がり(立下がり)に同期して状態が変化するフリップフロップをポジティブエッジ型、クロックパルスの立下がり(立上がり)に同期して状態が変化するフリップフロップをネガティブエッジ型と呼ぶ。エッジトリガは、次の立上がり(立下がり)エッジまでは出力は切り替わらず保持される。エッジトリガ式フリップフロップは、特定のクロック信号の立ち上がりや立ち下がりエッジでのみ動作するため、同期動作が可能である。これにより、信号の安定性と整合性が向上し、信号のタイミングを制御しやすくなる。

6. 考察

本実験を通して、フリップフロップやレジスタ、直列加算器などの順序回路の基本的な動作原理について調べた。これらの回路は、コンピュータのCPUやメモリなどの基本的な部品として不可欠な役割を果たしている。特に、フリップフロップは1ビットのデータを一時的に保持する重要な機能を持っている。マスタスレーブ方式のフリップフロップでは、クロックのタイミングを厳密に制御することで、安定した動作を実現している。エッジトリガ式フリップフロップではクロックエッジでのみ動作するため、ノイズに強く信頼性の高い回路を構成できる。また、レジスタは複数のフリップフロップを組み合わせることでデータを保持する機能を持ち、CPUのレジスタファイルやメモリなどに広く用いられていて、シフトレジスタは、データを順次シフトさせる機能がある。さらに、今回構成した直列加算器は、筆算で行うような2進数の加算をハードウェア的に実現したもので、加算結果をシフトレジスタへ格納した。直列加算器は通常の加算器に比べてわずかなハードウェア資源で実装できる。

7. 参考文献

- [1]情報科学実験Iテキスト 2024 年
- [2]中部大学, 5. フリップフロップ, <https://edu.isc.chubu.ac.jp/naga/dec/dec5.html>