

1 はじめに

近年は超小型化された電子素子（主にトランジスタなどの半導体）を用いた電気信号を、0 か 1 かのデジタル信号として扱う技術は身の回りや社会で幅広く見かける。したがってこの手の技術者にはデジタル信号、回路を扱えることが必須となってきた。

電子素子を用いた回路には集積回路 (IC) や大規模集積回路 (LSI) などがあり、これらを組み合わせることで大きな電気的システムを構成することができる。

一方、FPGA といわれるプログラミングにより自由な回路の書き換えを可能にするボードが存在する。今回はそれをもちいてパソコン上で発生、取得したデジタルデータを扱っていくものとする。

今回の実験内容は大きく分けて 7 つに分類することができて

1. AND,OR,XOR ゲートの論理演算の入出力動作確認
2. NOT ゲートの遅延測定
3. 全加算器の動作確認 (真理値表の作成)
4. RS_FF の動作確認 (真理値表の作成)
5. JKT_FF, JKTMS_FF のタイミングチャートの作成
6. T_FF, TMS_FF のタイミングチャートの作成
7. D_FF, DMS_FF のタイミングチャートの作成

の構成で、進めていく。

つまり今回扱う内容は次のようにまとめることができる。

- 組み合わせ論理回路 (AND,OR などの各論理ゲート)
- 順次論理回路 (フリップフロップによる記憶回路)

2 実験の目的と原理

デジタル信号というのは必ず 1 か 0、どちらかの値で表される。これと論理演算の真、偽関係を対応させて回路を組立てていくのがデジタル回路である。つまりこの論理回路に対応する AND, OR, XOR, NOT などのゲートの特性を調べていくことは、アナログ受信からデジタル信号として扱われるまでの過程の理解につながる。

2.1 組み合わせ論理回路

論理回路において、出力が一時点の入力に完全関数的であるとき、それを組み合わせ論理回路と呼ぶ。また組み合わせ論理回路は複数個の論理ゲートと呼ばれる単位演算子に分けられる。この実験で用いる重要ないくつかの論理ゲートとその真理値表を図 1 に示す。

各論理ゲートについて触れておきたい。

1. AND ゲート:

2 つの入力値が等しいときに 1、異なるときに 0 を出力する論理ゲート

2. OR ゲート:

2 つの入力のうち、少なくともどちらか 1 つが真 (1) であれば 1、どちらも偽 (0) であれば 0 を出力する論理ゲート

3. XOR ゲート:

2 つの入力が等しいとき 0、異なるとき 1 を出力する論理ゲート

4. NOT ゲート:

入力は 1 つで、出力値として入力の逆を返す論理ゲート

また複数桁の数値計算をするうえでも重要な半加算器・全加算器も論理ゲートの組み合わせで構成することができる。これについては実験内容の章で図とともに述べるが、全加算器は 2 つの入力データ、そして 1 つ前の桁上げ出力による入力に対して、それらの和の数をもとめ桁上げ、同桁として出力することができる論理回路である。

2.2 順次論理回路

上の組み合わせ論理回路に対して、順次論理回路とは、その出力が一時点での入力だけでなく今までの入力も関係してくる論理回路のことを指す。

最も主要な順次論理回路にフリップフロップ回路がある。これは前回の出力結果を保持、もしくは消去、あるいは前回の値には関わらず発散する、など様々な記録の操作を可能にする回路である。次にこの実験でも用いる重要なフリップフロップ回路を図 による回路図、真理値表とともに説明する。

2.2.1 RS フリップフロップ

まず RS フリップフロップについて説明する。回路図は図 1 に示す通りだが、入力が $(S,R)=(0,0)$ の場合には出力 Q は変化なし、入力が $(0,1)$ のときには出力 Q は 0、入力 $(1,0),(1,1)$ のとき出力 Q は 1 となるような回路である。

またここでは RST_FF の記述されるようなクロック入力付きのフリップフロップ回路の説明もする。例として RST_FF を図 2 に示す。ck(クロック入力) が 0 の場合、 $(S,ck),(R,ck)$ を入力とするような最初の NAND ゲートで出力は必ず 1 となり、RS における $(0,0)$ 入力の場合と、つまり出力 Q が変化しない場合と等価になる。一方 ck が 1 の場合、最初の NAND ゲートの出力は S,R それぞれの NOT 出力と、つまり ck によらず RS の回路と等価になる。

まとめると、クロック入力付きの場合は、

- ck=0 のときは、入力値によらず Q は変化なし。
- ck=1 のときは、クロック無し回路の動作にしたがう。

ということになる。つまり、ck によってフリップフロップ回路の動作の on,off を制御することができる。

2.2.2 JK フリップフロップ

つぎに JK フリップフロップ回路 (図) について説明する。この回路はおおよそその動作が入力 $(J,K)=(0,0)$ の場合には出力 Q は変化なし、入力が $(0,1)$ のときには出力 Q は 0、入力 $(1,0)$ のとき出力 Q は 1、として RS フリップフロップとにている。ただし、入力が $(1,1)$ の場合は出力 Q は以前の値と反転する。これが JK フリップフロップ回路の特徴で、真理値表が明瞭になる。

ここで、JKMS_FF と表記されるようなマスタースレーブ (MS) についての説明もしておく (図参照)。ck 入力による図 向かって左側の RS(マスター側)、向かって右側の RS(スレーブ側) の動作について考えよう。ck=1 のとき、マスター側は通常の出力をするがスレーブ側が ck=0 の入力を受けとることで、結局出力 Q の値は変わらない。一方 ck=0 のとき、マスター側が機能しないが、スレーブ側は ck=1 を受け取って通常の機能をする。したがって

- ck=0 のとき JK フリップフロップとして機能。
- ck=1 のとき JK フリップフロップの機能が停止 (入力によらない Q の保持)。

となり、通常の ck 挙動と反転する。

ただし、MS 使用の目的はそこではなく、RS が二つあることで、必ずどちらかは ck 入力によって off にされている。したがって実際の使用に際してもその境界がはっきりすることで動作が安定する。

2.2.3 T フリップフロップ

ここからは ck をのぞき入力が 1 つの特殊なフリップフロップ回路である。図 に示すものが T フリップフロップ回路である。図 を見た通り、JK の入力が T で統一されているため、Q の出力値は

- $T=0$ で、Q の変化無し。
- $T=1$ で、Q は反転。

となる。

2.2.4 D フリップフロップ

最後に D フリップフロップ (図) について。D は JK の入力を同じ値で統一したのと反対に、 $D=0$ なら $(J,K)=(0,1)$ 、 $D=1$ なら $(J,K)=(1,0)$ というように反転した入力 JK 回路に相当する。

つまり

- $D=0$ で、Q は必ず 0。
- $D=1$ で、Q は必ず 1。

となるような今までの入力に関係なく動作する順次論理回路である。

3 実験内容

今回の実験で使用したハード、ソフトウェア環境を記載する。

表 1 実験環境 (使用したハードウェア、ソフトウェア)

コンピュータ	DOS/V PC
OS	Windows 8.1
CAD(回路図用)	Quartus2 12.1(Altera 社)
FPGA ボード	DE2-70 ボード (EP2C70F896C6N 搭載)

これより実験は以下のような順次手順で進めた。

3.1 AND,OR,XOR ゲートの論理演算の入出力動作確認

実験回路は図 1 に示す。入力 A,B を on,off で切り替えて、X,Y,Z 等の出力挙動を観察。

3.2 NOT ゲートの遅延測定

図 2 に回路図を示す。

1. NOT ゲート 8 個に対して、その入力からの出力の遅延を観察。
2. こんどは NOT ゲートを 7 個にして、同様に出力の入力からの遅延を観察。

3.3 全加算器の動作確認 (真理値表の作成)

半加算器、全加算器の論理回路をそれぞれ図 , に示す。ここでは全加算器を半加算器を用いて組み、全加算器の真理値表を完成させた。

3.4 RS の動作確認 (真理値表の作成)

図 , にそれぞれ示した RS,RST 回路を用いる。

1. RS に対して、動作表を完成。
2. RS,RST 回路に次のテストベクトル (R,S の組) を入れ、タイミングチャートを作成。

000/010/110/010/000/000/001/101/001/000/000/011/111/011/111/011/010/110/010/000/00/000/001

3.5 JKT,JKTMS のタイミングチャートの作成

図 の JKT,JKTMS 回路に RS の実験と同様のテストベクトル (J,K の組) を入れて、タイミングチャートを作成。

3.6 T,TMS のタイミングチャートの作成

T、TMS 回路を図 に示した。次のテストベクトルを入力して、タイミングチャートを作成。

00/10/00/10/00/01/11/01/11/01/11/01/11/01

3.7 D,DMS のタイミングチャートの作成

図 に示すような D,DMS の論理回路に T と同様のテストベクトルを入力してタイミングチャートを作成。

4 実験結果

まず1の「AND,OR,XOR ゲートの論理演算の入出力動作確認」の表を表 1 に示す。これは準備課題で予測してきたものとまったく一致していた。

表 2 AND,OR,XOR の入出力動作の観察結果

SW0	SW1	P31	P33	A	B	X	P35	LEDR2	Y	P37	LEDR3	Z	P39	LEDR4
on	on	H	H	1	1	1	H	点	1	H	点	0	L	滅
on	off	H	L	1	0	0	L	滅	1	H	点	1	H	点
off	on	L	H	0	1	0	L	滅	1	H	点	1	H	点
off	off	L	L	0	1	0	L	滅	0	L	滅	0	L	滅

また同実験の AND,OR,XOR のデジタル出力波形を次ページの図 1 に、AND,OR ゲートのアナログ出力波形を図 2 に、XOR ゲートのアナログ出力波形を図 3 に示した。デジタル波形とアナログ波形の概形は等しいが、デジタル波形に対してアナログ波形には立ち上がり、立ち下がりの際の有限の勾配が確認できる。

この「立ち上がり、立ち下がりの際の遅延」を「NOT 回路の複数組み合わせによって測定した実験」のデータを前ページの図 1.10 に載せた。それぞれ NOT ゲート 8 本、NOT ゲート 7 本を直列につないだ出力として得られた立ち上がり、立ち下がりのアナログ波形である。

ここから得られた結果を次の表 1.11 にまとめる。

表 3 7 本、及び 8 本直列 NOT ゲートによる出力遅延測定

	NOT8 本	NOT7 本
立ち上がり遅延時間/ns	67	70
立ち下がり遅延時間/ns	68.5	77

次にフリップフロップ回路についての実験データを RS より順に載せる。RS のタイミングチャートは図 1.12 に示す。RS フリップフロップは少ない数の論理ゲートを組み合わせたものであるため、動作は安定している。タイミングチャートの形も準備課題どおりとなっている。

最後に次ページの図 , にそれぞれ示すのが T フリップフロップ、D フリップフロップの測定タイミングチャートである。これは先ほどの RS とは違い、T の Q_n や、D の Q, Q_n が不安定な出力をしていることが確認できる。

5 考察

5.1 AND,OR,XOR ゲートの論理演算の入出力動作確認

実験結果の表 にあるように、動作表の実験値は準備した予測値と一致した。この予測をするときには X は A,B の AND 出力、Y は A,B の OR 出力、Z は A,B の XOR 出力であることから実験値を求めた。したがってそれが正しく、安定して 0 か 1 かの値で返されていることが確認できた。

デジタル回路は実験結果の表 からわかるように非常に安定して、正確な値を出力している。図 を見てみると、大きなスケールに対して一瞬で高電圧、低電圧を変換しているように見える。ただしのちの NOT ゲートの遅延測定でもわかる通り、立ち上がり、立ち下がりは ns スケールで行われるため、閾値電圧を 1.8V と定めた場合、その出力が安定することはわかる。

5.2 NOT ゲートの遅延測定

ゲートの遅延は論理ゲート系と導線との間の金属接触の際の微小な電気容量に起因している。[1] しかし今回の実験結果を見てみると、NOT ゲートが 7 本の時の方が 8 本よりも遅延時間が長い。これは上の説明には矛盾している。つまりこれ以外の理由が私たちの実験結果にはあったはずである。個数を除いた 7 本と 8 本の NOT 直列回路の違いは、出力が反転されるか反転されないか、にある。

したがって今回の直列本数に関して 2 種類試した実験結果では、最終的に反転している出力の方がしていない出力よりも遅延時間が長くなりうる、という原因要素が上の金属接触の電気容量に加えて存在するものと判断した。

5.3 T,TMS のタイミングチャートの作成

5.4 D,DMS のタイミングチャートの作成

6 おわりに

今回は、自然物理的にはアナログである入力から、コンピュータ科学などの分野で使われるデジタルという概念への転換を見ていくことが私の主な目的であった。

しかし、論理ゲートによる遅延など、考察するのに幾分多くの実験データがないといけない、と言う点で止まってしまったのは理解が足りていない状況で実験に挑んでいたことを感じた。

一方、フリップフロップなどの順次論理回路からは、出力の安定化という実際の現場での必要性を考えることができ、その点、デジタル信号の操作に関して興味のわくきっかけとなった。

7 参考文献

[1] www.ai-l.jp/~LB5/Logic-delay-power.pdf