

2. コンピュータの基礎 (2) メモリ (記憶素子)

Memory

コンピュータはどのようにものを憶えているのでしょうか？磁気ディスクや光ディスクにデータを保存する使い方の場合に、電源を入れていない間どのように情報が保持されているかは想像しやすいでしょう。これらの記録装置は「補助記憶装置」と呼ばれます。昔は紙テープやパンチカード、磁気テープなどが補助記憶装置として使われていました。現在はフラッシュメモリと呼ばれる半導体記憶素子も補助記憶装置として用いられます。

データを補助記憶装置から読み込んで、何らかの演算処理をして、補助記憶装置に書き込むまでの間、データは電子回路に記憶されています。また、コンピュータの中の処理装置（プロセッサ）の動作を指定する一連の指令（コマンド）はコードと呼ばれ、これも電子回路に記憶されます。データやコードを記憶しておくための電子回路を記憶素子（メモリ）と呼びます。

原理的には、電子回路による記憶素子が無くても動作するコンピュータを作ることが可能ですが、一般的には補助記憶装置を使ったデータの読み書きより記憶素子を用いたデータの読み書きの方がずっと高速であり、実際上記憶素子はコンピュータにとって必要不可欠な部品です。

主に使われる記憶素子には2種類あります。

- (1) 静的記憶素子；スタティック メモリ Static memory
- (2) 動的記憶素子；ダイナミック メモリ Dynamic memory

これら2種類の記憶素子は全然違うしくみになっています。詳しいしくみは後述するとし、これらの記憶素子の特徴を比較すると [Table 2.1](#) のようになります。

Table 2.1 静的メモリと動的メモリの比較

素子の種類	読み書きの速さ	構造の複雑さ	高集積密度化	単価
静的	高速	やや複雑	やや困難	やや高価
動的	やや低速	単純	容易	安価

2-1 静的記憶素子 Static memory

2-1-1 6トランジスタの静的記憶素子

典型的な静的メモリは2つのP-チャンネルMOS-FETと4つのN-チャンネルMOS-FETの合わせて6つのトランジスタを組み合わせて実現されます。

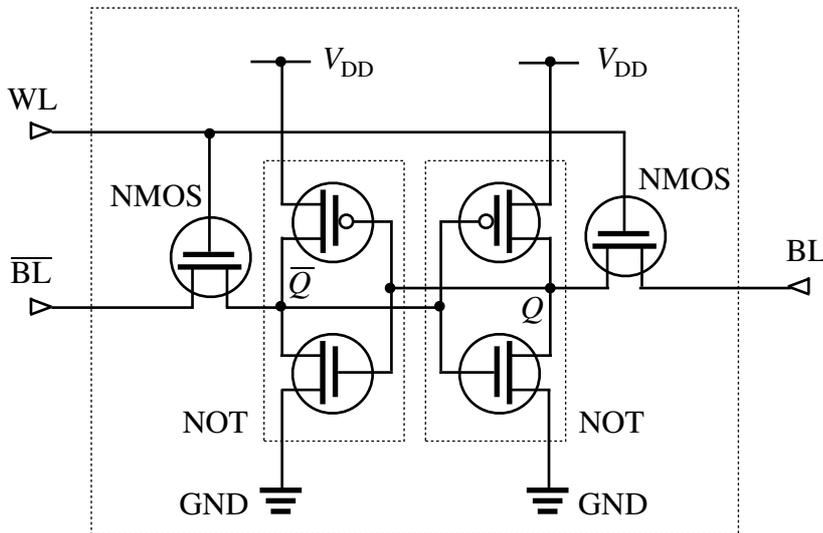


Figure 2.1.1.1 6トランジスタの静的記憶素子 (6T-SRAM)。 V_{DD} は電源電圧，GND は0 Vを表す。WL はワード線，BL はビット線， \overline{BL} は反転ビット線。

[Figure 2.1.1.1](#) のように、6トランジスタの静的記憶素子 (6T-SRAM) は、2つの否定回路と、2つのNMOSとを組み合わせて、3つの信号線を引き出したものです。2つの否定回路は、一方の出力を他方の入力と接続する組み合わせ方をしています。否定回路には入力信号を増幅する働きがある（入力抵抗が高く出力抵抗が低い）ので、このような接続のしかたは正帰還回路 (positive feedback circuit) とも呼ばれます。

3つの信号線のうち、ワード線 (word line) WL は2つのNMOSのソースとドレインの間の導通をオン・オフする働きをします。ビット線 (bit line) BL と反転ビット線 (inverted bit line) \overline{BL} は、1ビットのデータを書き込むためにも読み出すためにも（入力するためにも出力するためにも）用いられます。[Fig. 2.1.1.1](#) 中のQの位置の電圧をメモリ（記憶）とみなすことができます。

通常はワード線 WL はGNDレベルとして、2つのビット線，BLと \overline{BL} とから内部の回路を絶縁させた状態にします。

6T-SRAMに「ゼロ」を書き込む（リセットする）ためには、ビット線BLの電圧をGNDレベル ($BL = 0$)，反転ビット線 \overline{BL} の電圧を V_{DD} レベル ($\overline{BL} = 1$)としてから、ワード線WLの電圧を上げて ($WL = 0 \rightarrow WL = 1$)，2つのNMOSのソースとドレインの間

を導通状態に変化させます。このとき、[Figure 2.1.1.1](#) 中の Q の位置の電圧は GND レベル ($Q = 0$) となり、 \bar{Q} の位置の電圧は V_{DD} レベル ($\bar{Q} = 1$) になります。

この後に、ワード線 WL の電圧を GND レベルに落として ($WL = 1 \rightarrow WL = 0$)、2本のビット線 BL と \bar{BL} とから「内部の回路」を絶縁させた状態にすれば、この後にビット線 BL と反転ビット線 \bar{BL} の電圧が変化したとしても、 $Q = 0$ 、 $\bar{Q} = 1$ の状態は変化しません。この状態でこの電子回路（素子）は「ゼロ」を書き込まれた（リセットされた）ことを記憶していることとなります。

6T-SRAM に「1」を書き込む（セットする）ためには、BL を V_{DD} レベル ($BL = 1$)、 \bar{BL} を GND レベル ($\bar{BL} = 0$) としてから、ワード線 WL の電圧を上げて ($WL = 0 \rightarrow WL = 1$)、2つの NMOS のソースとドレインの間を導通状態に変化させます。このとき、[Figure 2.1.1.1](#) 中の Q の位置の電圧は V_{DD} レベル ($Q = 1$) となり、 \bar{Q} の位置の電圧は GND レベル ($\bar{Q} = 0$) になります。

この後に、ワード線 WL の電圧を GND レベルに落として ($WL = 1 \rightarrow WL = 0$)、ビット線 BL と反転ビット線 \bar{BL} とから内部の回路を絶縁させた状態にすれば、この後にビット線と反転ビット線の電圧が変化したとしても、 $Q = 1$ 、 $\bar{Q} = 0$ の状態は変化しません。この素子は「1」を書き込まれた（セットされた）ことを記憶していることとなります。

6T-SRAM が記憶した 1 ビットのデータを読み出すためには、ビット線 BL、 \bar{BL} のどちらか、あるいは両方を他の論理回路の入力に接続してから、ワード線 WL の電圧を上げ ($WL = 0 \rightarrow WL = 1$)、その電圧を下げるまでの間に値を読み出します。この後にワード線 WL の電圧を GND レベルに落としても、記憶はそのまま残されます。

2-1-2 SR NOR ラッチ SR NOR latch

静的記憶素子は論理演算素子の組み合わせで実現できます。一つの例は [Fig. 2.1.2.1](#) に示すように 2 つの否定論理和 NOR ゲートを組み合わせた回路です。このタイプの回路は エス・アール ノー ア SR NOR ラッチ (latch) と呼ばれます。

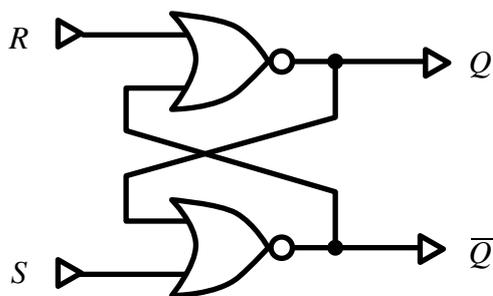


Fig. 2.1.2.1 SR NOR ラッチ回路

この SR NOR ラッチ回路の出力 Q と \bar{Q} は、2つの入力 R, S によってどのように変化するでしょうか？

NOR ゲートのはたらきから、以下の2つの論理式が成り立ちます。

$$\begin{cases} Q = \overline{R \vee \bar{Q}} \\ \bar{Q} = \overline{\bar{Q} \vee S} \end{cases}$$

この2番目の式を1番目の式に代入すると、

$$Q = \overline{R \vee \overline{\bar{Q} \vee S}} = \bar{R} \wedge (Q \vee S)$$

という関係が得られます。2番目の式の変形にはド・モルガンの法則を使いました。さらに、 R と S の値に応じて、 Q の値がどのような値を取るかを調べると、

以下のようになります。

$$R=0, S=0 \text{ のとき, } Q = 1 \wedge (Q \vee 0) = 1 \wedge Q = Q$$

$$R=0, S=1 \text{ のとき, } Q = 1 \wedge (Q \vee 1) = 1 \wedge 1 = 1$$

$$R=1, S=0 \text{ のとき, } Q = 0 \wedge (Q \vee 0) = 0 \wedge Q = 0$$

$$R=1, S=1 \text{ のとき, } Q = 0 \wedge (Q \vee 1) = 0 \wedge 1 = 0$$

つまり、

$$Q = \begin{cases} \text{uncertain} & [R = S = 0] \\ 1 & [R = 0, S = 1] \\ 0 & [R = 1, S = 0] \\ 0 & [R = S = 1] \end{cases}$$

となります。 $R=0, S=0$ のときに Q の値が不定 アンサーテン uncertain となることに注意してください。

この回路の Q の値をメモリ（記憶）とみなすことができます。 R と S はメモリを書き換えるための信号線で、 R をリセット reset 信号、 S をセット set 信号と呼びます。SR NOR ラッチの実際の動作のしかたについて以下に説明します。

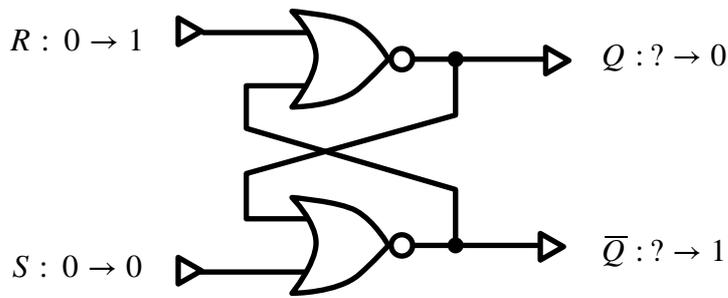


Figure 2.1.2.2 SR NORラッチ回路に「0」を書き込む。R=1 とすれば必ず $Q=0$ となる。

通常の状態では $R=0, S=0$ とします。はじめは Q の値は不定です。初期化のために「0」というデータを「書き込む」ためには、 $S=0$ に保ったまま $R=1$ とします。そうすると、[Figure 2.1.2.2](#) に示すように Q の値が0に変化します。

[Figure 2.1.2.2](#) の中に2つある NOR ゲートの上の方に注目してください。この NOR ゲートの入力の一方が1になれば、もう一つの入力の値によらず、出力 Q は必ず0になります。下の方の NOR ゲートは2つの入力が両方とも0になるので、出力 \bar{Q} が1になります。

この $R=1, S=0$ の状態から、はじめと同じ $R=0, S=0$ の状態に戻したらどうなるでしょうか？ [Figure 2.1.2.3](#) に示すように、出力は $Q=1$ のままで変化しません。上の方の NOR ゲートの入力のうちの1つ \bar{Q} が、 $\bar{Q}=1$ になっているので、もう一つの入力 R の値に関わらず、出力は $Q=0$ のままになるはずですが、一方で、下の方の NOR ゲートの2つの入力は両方とも0なので、その出力は $\bar{Q}=1$ でつじつまが揃っていることになります。

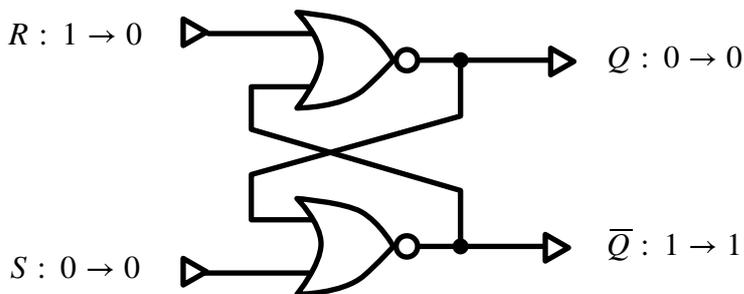


Figure 2.1.2.3 SR NORラッチ回路に「0」を書き込んだあと、 $R=1$ から $R=0$ に戻しても $Q=0$ のままである。（書き込まれた値を憶えている）

SR NOR ラッチに「1」という値を「書き込む」には、 $R=0$ に保ったまま $S=1$ とします。すると、[Figure 2.1.2.4](#) のようになります。下の NOR ゲートの入力の一方が $S=1$ なので、その出力は $\bar{Q}=0$ になります。上の NOR ゲートの入力は $R=0, \bar{Q}=0$ となるので、その出力は $Q=1$ となります。

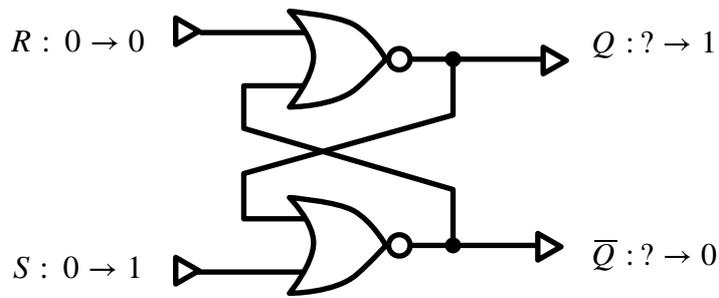


Figure 2.1.2.4 SR NORラッチ回路に「1」を書き込む。R=0のままS=1とすればQ=1となる。

この後S=1からS=0に戻しても、[Figure 2.1.2.5](#)のように、 $\bar{Q} = 0$ なのでNORゲートの出力はQ=1のままです。

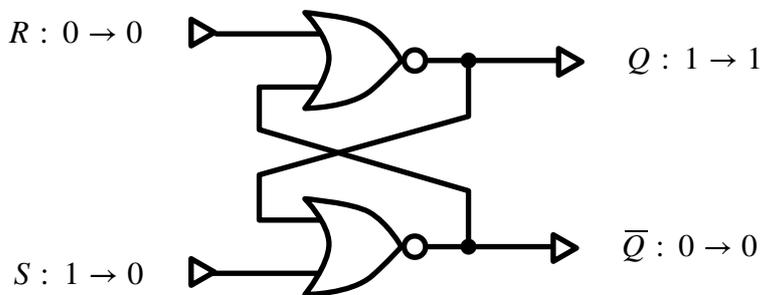


Figure 2.1.2.5 SR NORラッチ回路に「1」を書き込んだあと、S=1からS=0に戻してもQ=1のままである。（書き込まれた値を憶えている）

SR NOR ラッチの入力と出力の関係を [Table 2.1.2.1](#) に示します。

Table 2.1.2.1 SR NOR ラッチ回路の論理値表

S	R	Q	\bar{Q}	
0	0	Q	\bar{Q}	保持 <small>ホールド</small> hold
0	1	0	1	リセット reset
1	0	1	0	セット set
1	1	0	0	不許可 not allowed

S = R = 1 の場合には2つの出力信号が反転の関係にならないので、普通そのような使い方はしないこと（不許可）とされています。

2-1-3 反転 SR NAND ラッチ Inverted SR NAND latch

Figure 2.1.3.1 のように、2つの否定論理積 NAND 素子を組み合わせても静的メモリ素子を作ることができます、このタイプの回路は反転 SR NAND ラッチ (\overline{SR} NAND latch) と呼ばれます。

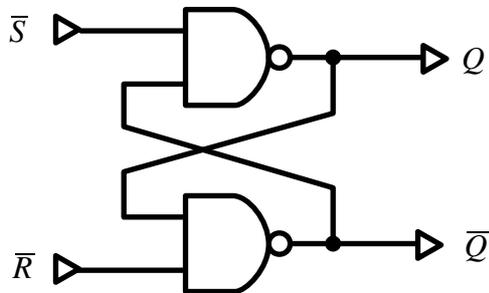


Figure 2.1.3.1 \overline{SR} NAND ラッチ回路

反転 SR NAND ラッチの入力と出力の関係を [Table 2.1.3.1](#) に示します

Table 2.1.3.1 反転 SR NAND ラッチ回路の論理値表

\overline{S}	\overline{R}	Q	\overline{Q}	
0	0	1	1	不許可 not allowed
0	1	0	1	反転セット inverted set
1	0	1	0	反転リセット inverted reset
1	1	Q	\overline{Q}	ホールド 保持 hold

通常は $\overline{S} = \overline{R} = 1$ の状態に保っておき、「0」を書き込む（リセットする）ときには $\overline{S} = 1$ のまま $\overline{R} = 0$ と変化させて $Q = 0$ としたあと $\overline{S} = \overline{R} = 1$ の状態に戻します。「1」を書き込む（セットする）場合には $\overline{S} = 1$ のまま $\overline{R} = 0$ に変化させて $Q = 1$ としたあと $\overline{S} = \overline{R} = 1$ の状態に戻します。 $\overline{S} = \overline{R} = 0$ の状態では2つの出力信号が反転の関係にならないので、普通そのような使い方はしないこと（不許可）とされています。

2-2 動的記憶素子 Dynamic memory

動的記憶素子 ダイナミック メモリ dynamic memory は、キャパシタ（コンデンサ）に静電気をためることでデータを記憶する電子回路です。（[補足 2.2.A](#)）キャパシタの構造は、[Figure 2.2.1](#) のように2枚の金属板を向い合せて近づけて配置したものです。

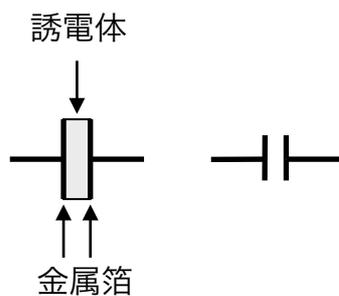


Figure 2.2.1 キャパシタ (コンデンサ) の構造 (左) と回路記号 (右)

キャパシタと NMOS を [Figure 2.2.2](#) のように組み合わせれば，動的メモリ素子として働きます。キャパシタに電荷がたまっているかないかで 1 か 0 の値が表されます。

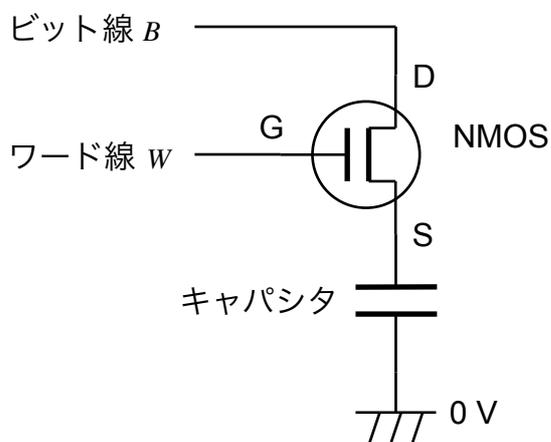


Figure 2.2.2 動的メモリ回路

実際のメモリ素子ではワード線とビット線はマトリックス (行列) 状に配線されます ([Figure 2.2.3](#))。たとえば，4本のワード線と4本のビット線を使って， $4 \times 4 = 16$ 個のメモリ素子を制御することができます。この一つ一つのメモリ素子のことは「メモリセル」と呼ばれます。

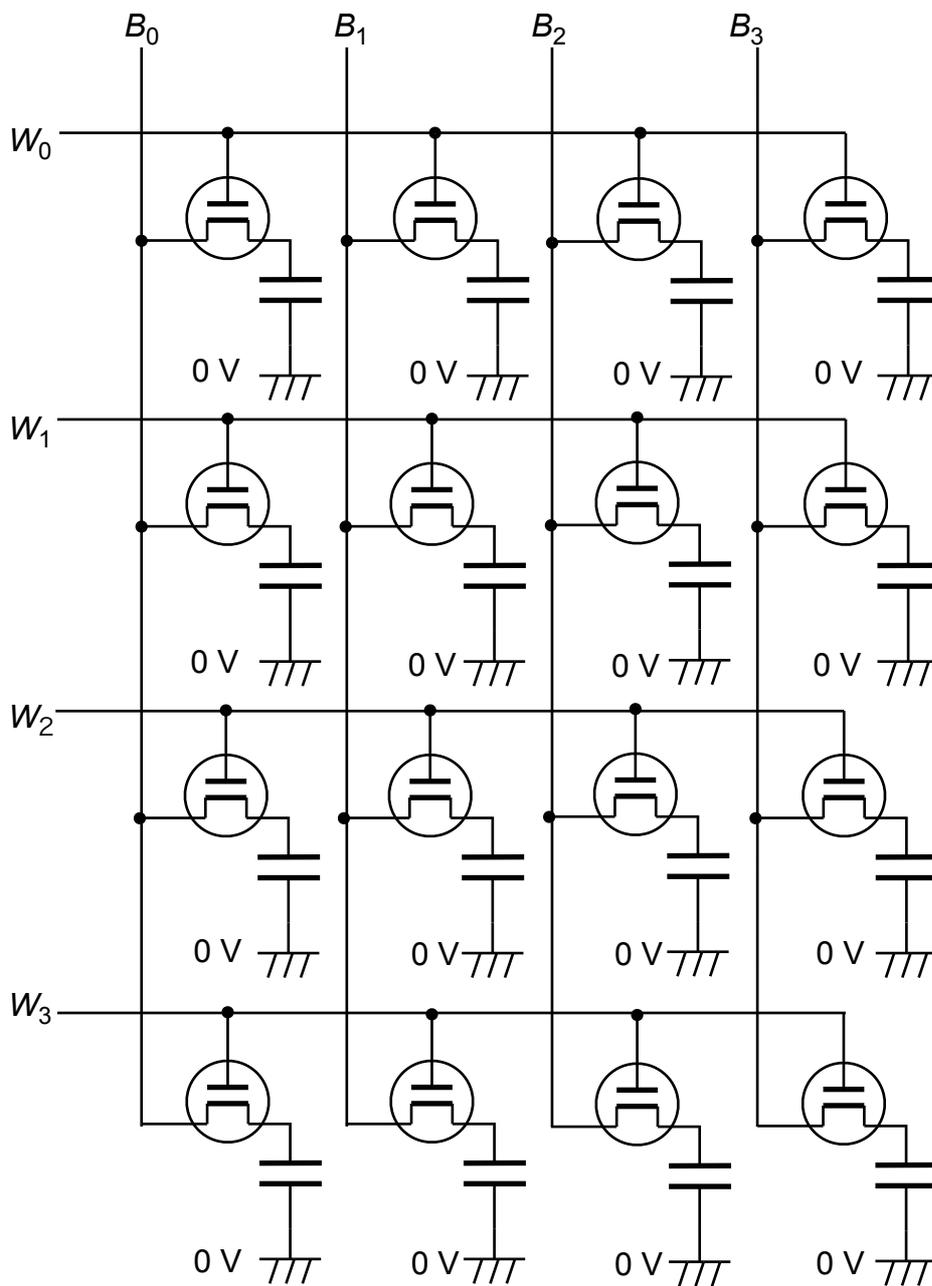


Figure 2.2.3 動的メモリのマトリックス状の配線

動的メモリに「0」という値を書き込むにはゲート電圧をかけた状態でビット線の電圧を下げてキャパシタから放電させます (Figure 2.2.4)。この後にワード（ゲート）電圧を下げればビット線からキャパシタへは電流が流れにくくなるので、ビット線の値にかかわらずキャパシタは放電したままです。この電子回路は、書き込んだ値をしばらくは憶えていることとなります (Figure 2.2.5)。

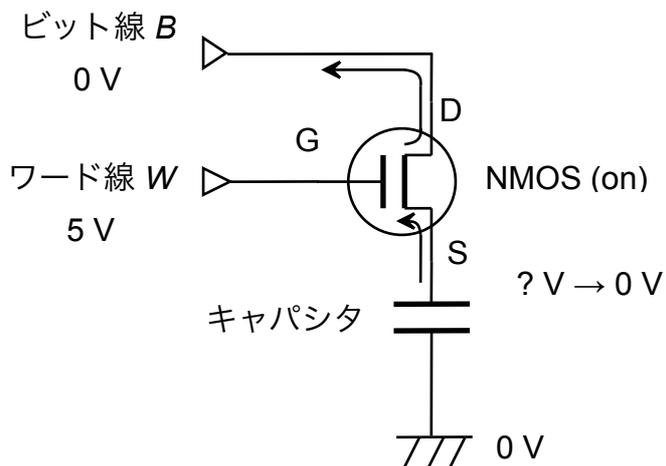


Figure 2.2.4 動的メモリ回路に「0」を書き込む。ワード線 W を 5 V とすれば NMOS が on 状態になり、ビット線 B を 0 V にすれば、矢印の向きに電流が流れてキャパシタが放電する。

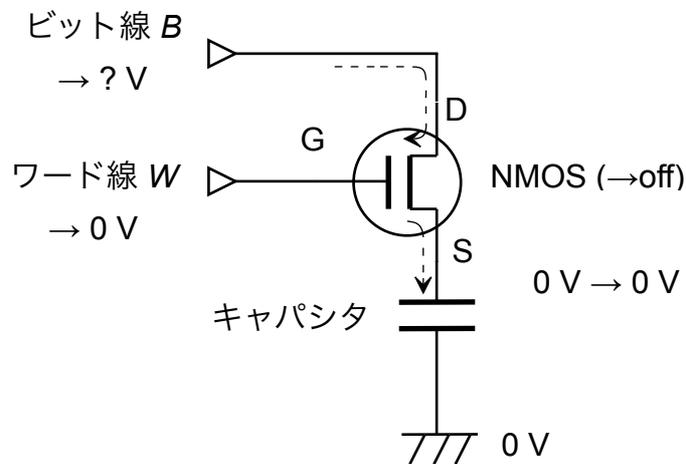


Figure 2.2.5 動的メモリ回路は「0」という値を憶えている。ワード線 W を 5 V から 0 V に戻せば、ビット線が 5 V になったとしても、破線矢印の向きには電流が流れにくいので、キャパシタが放電された状態がしばらく保たれる。

NMOS とキャパシタによる動的記憶素子に「1」という値を書き込むには、ワード線の電圧を上げてゲート電圧をかけた状態にし、ビット線の電圧の電圧を上げてキャパシタに充電します (Figure 2.2.6)。その後ゲート電圧を下げればキャパシタからビット線へは電流が流れにくくなり、書き込んだ値をしばらく憶えていることとなります (Figure 2.2.7)。

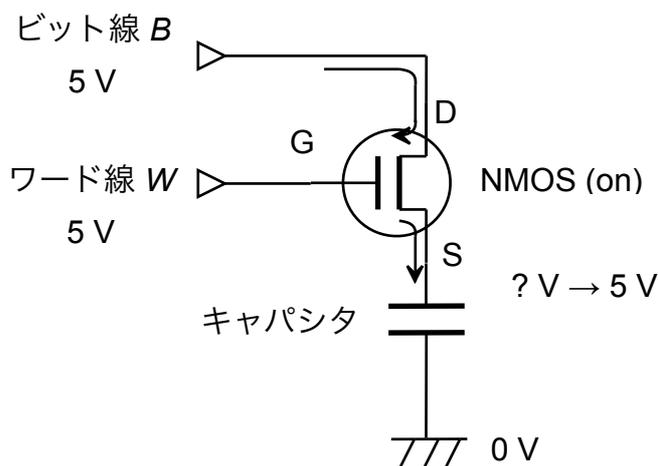


Figure 2.2.6 動的メモリ回路に「1」を書き込む。ワード線 W を 5V とすれば NMOS が on 状態になり、ビット線 B を 5V にすれば、矢印の向きに電流が流れてキャパシタが充電される。

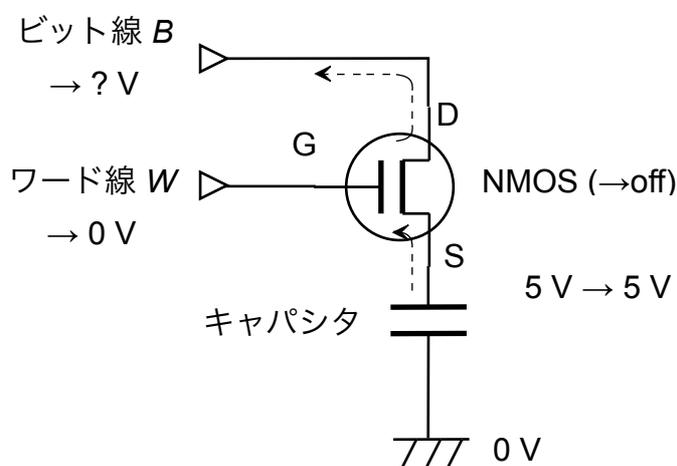


Figure 2.2.7 動的メモリ回路は「1」という値を憶えている。ワード線 W を 5V から 0V に戻してやれば、ビット線が 0V になったとしても、破線矢印の向きには電流が流れにくいので、キャパシタの充電された状態がしばらく保たれる。

[Figure 2.2.3](#) に示すようなマトリクス配線で動的メモリを用いる場合に、通常はワード線を 0V に保ち、ビット線は絶縁状態とします。特定のメモリセルに値を書き込むときには、該当するワード線の電圧を 5V に上げて、該当するビット線を 0V か 5V にします。

動的メモリ回路から値を読み出すためには、ワード線電圧を上げて、その間にビット線に流れ出る電流を検出します。キャパシタに電荷がたまっていれば（記憶が「1」であれば）ビット線に電流が流れ出し（[Figure 2.2.8](#)），キャパシタに電荷がたまっていなければ（記憶が「0」であれば）ビット線には電流が流れ出ません。電流が流れ出すかどうかで記録されたデータが「1」であったか「0」であったかがわかります。

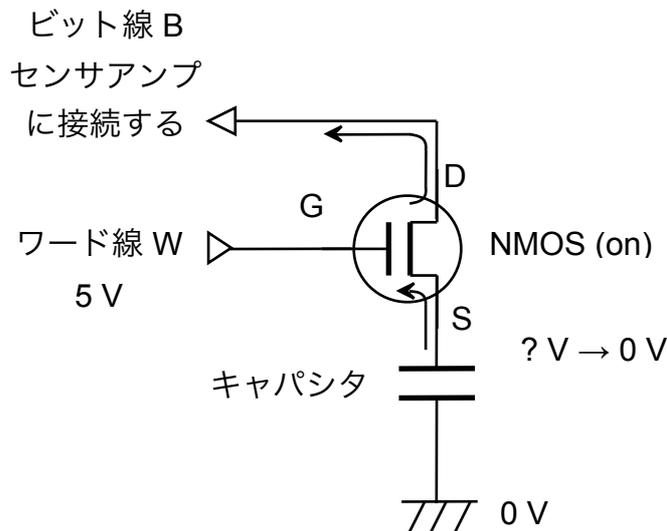


Figure 2.2.8 動的メモリ回路からデータを読み出す。ワード線 W を $5V$ としてビット線 B を検出回路に接続する。キャパシタからビット線に電荷が流れ出せば記憶が「1」だったということがわかる。

ただし、この方法では、素子ごとの特性のばらつきや、温度の変化などの影響を受けて結果が確実でなくなるので、実際には [Figure 2.2.9](#) のように記憶用キャパシタとは別に基準用のキャパシタを設置する方法が使われます。ワード線電圧を $0V$ として NMOS を開いた（絶縁）状態で、ビット線電圧をあげて、あらかじめ基準用のキャパシタを充電します。その後にワード線電圧をあげて NMOS を導通状態にしたときに、ビット線の電圧がどのように変化するかを調べます ([Figure 2.2.10](#))。「1」が記憶されていた場合にはビット線の電圧の変化が小さく、「0」が記憶されていた場合には電圧の変化が大きくなることから、記憶されていた情報を区別します。

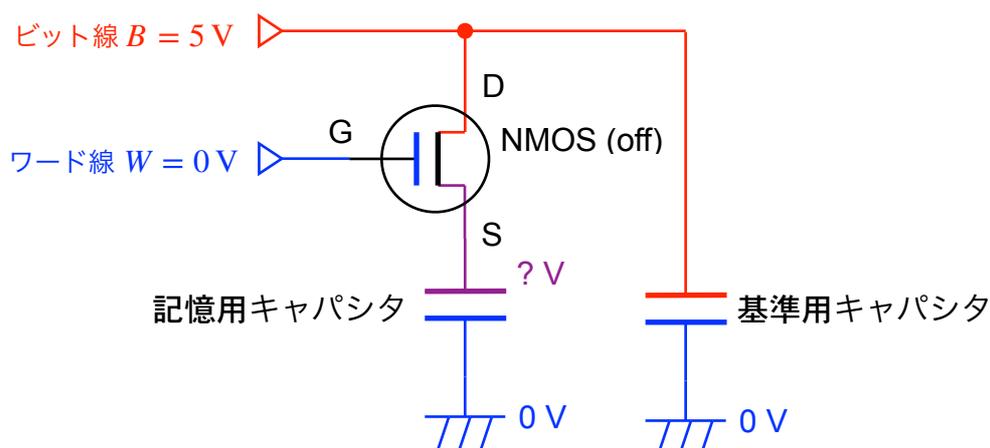


Figure 2.2.9 動的メモリ回路からデータを読み出す前のプリチャージ操作。ワード線 W を $0V$ のまま、ビット線 B を通して基準用キャパシタに充電をする。

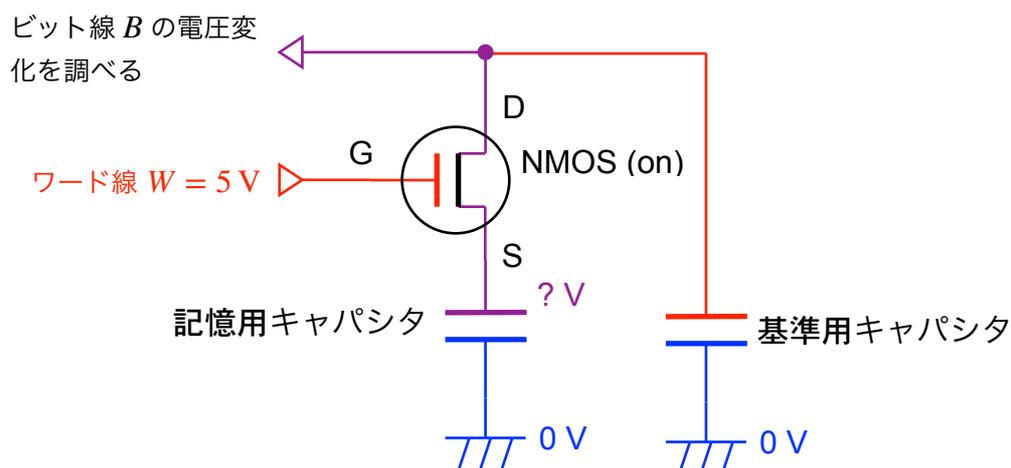


Figure 2.2.10 動的メモリ回路からデータを読み出す操作。ワード線 W を 5V にしたときのビット線 B の電圧の変化を調べる。

動的メモリからデータを読み出すときに、基準用キャパシタにあらかじめ充電をしておく操作のことを**プリチャージ precharge** と呼びます。

動的メモリからデータを読み出せば「記憶が破壊される」ことになるので、データを読み出すたびにそのデータを書き戻す操作が必要となります。動的メモリを使う場合、データを「書き込む」より、データを「読み出す」方に長めの時間がかかります。

また読み出しを行わない間にもキャパシタは徐々に放電して記憶が失われるので、データを書き込んで一定の時間が経過したら、データを一度読み出して、読み出したデータを書き込み直す操作が必要になります。この操作のことを**リフレッシュ refresh** と呼びます。通常は 1 秒間に数万回 (数十 μs に一回) ていどりフレッシュ動作が行われます。

(補足 2.2.A) コンデンサとキャパシタ (↔)

静電気を蓄えるための部品 (蓄電器) を日本の高校物理では「コンデンサー」として習うが、英語圏では キャパシタ capacitor と呼ばれる。一方で、水冷などの方法で高温の蒸気を凝縮させて液体として取り出す操作 (蒸留 ; 蒸餾) に用いられる器具 (冷却器) のことが コンデンサ condenser と呼ばれる (Figure 2.2.A.1)。

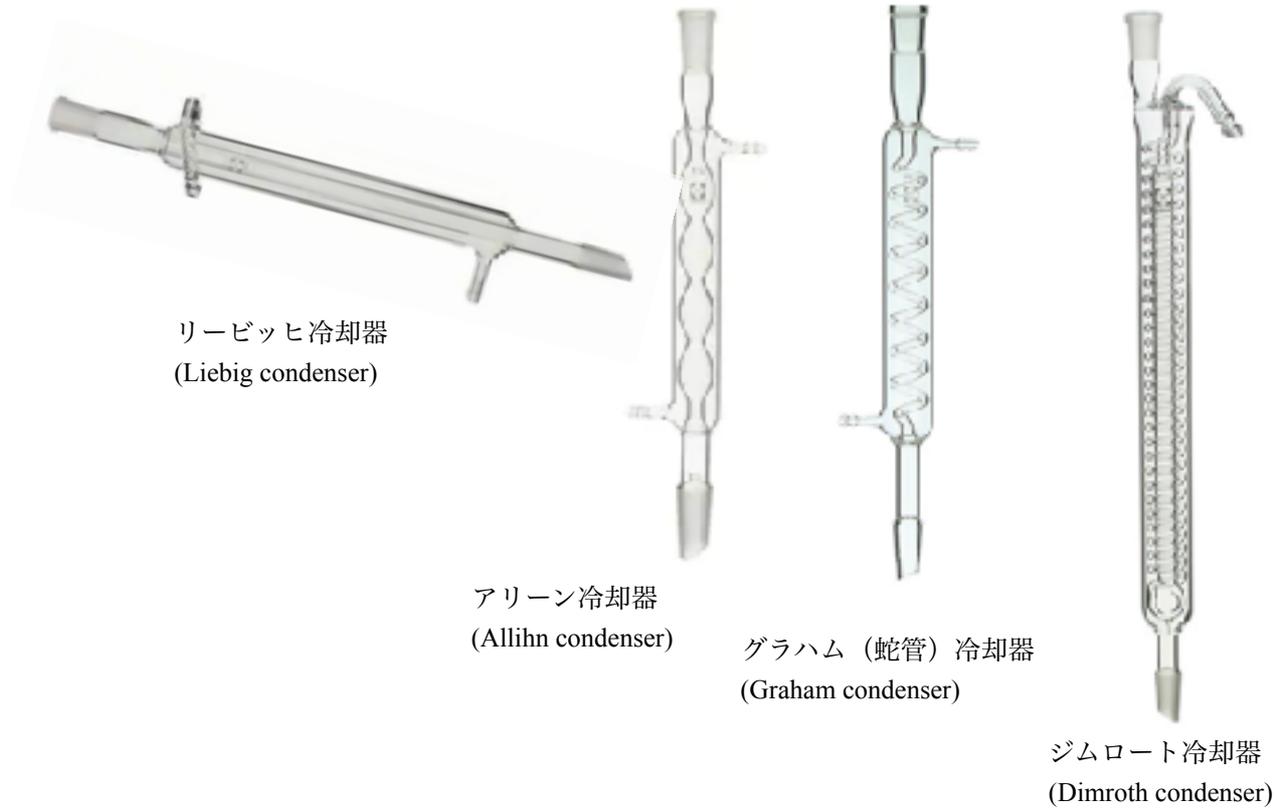


Figure 2.2.A.1 リービッヒ冷却器とアリーン冷却器, グラハム冷却器, ジムロート冷却器

(↺)