CPUはここさえ分かれば自作できる

Kariya Micro Maker Faire 2024

みやこ電子工房

目次

みやこ電子工房
MIYAKO DENSHI KOBO

- はじめに
  - 自己紹介
  - 今日の目標
  - CPUとは？
  - CPUが作れるまでのロードマップ
  - 今日作る目標のCPU紹介
- 概念の学習
  - 組み合わせ回路
  - レジスタ
  - データのコピー
  - 命令とアドレス
- CPUを作る
  - 回路をループさせる
  - プログラムカウンタ
  - 命令デコーダ
  - 使用ソフト Logisim 使い方
  - Logisim上で作成

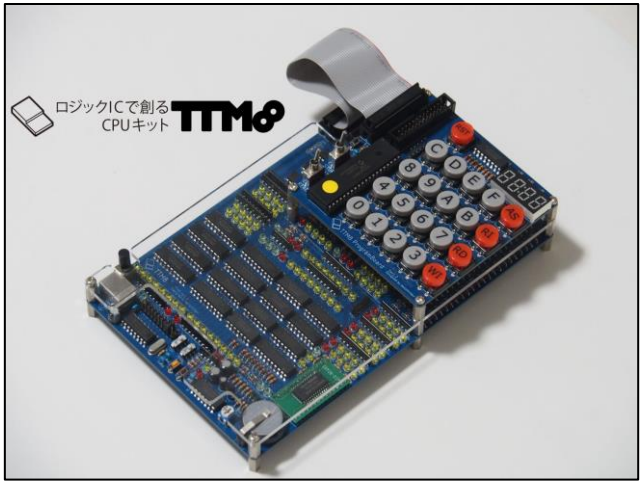
はじめに

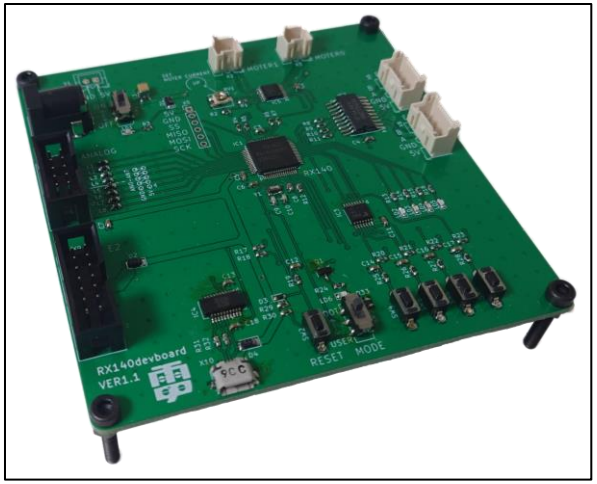
自己紹介

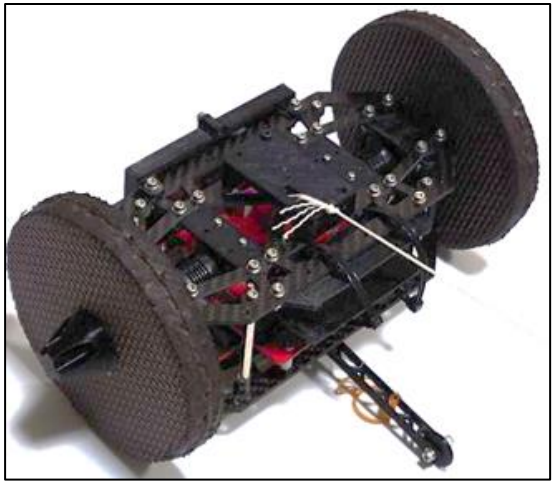
みやこ電子工房
MIYAKO DENSHI KOBO

みやこ電子工房
MIYAKO DENSHI KOBO

電子工作中心のモノづくりのサークル
自作CPUやマイコンボードを製作







今日の目標

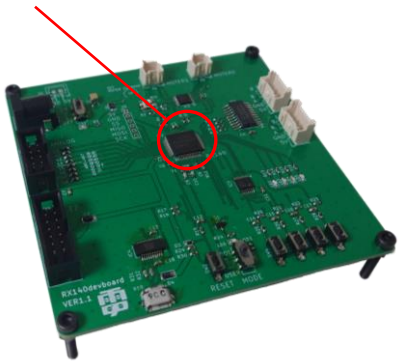
みやこ電子工房
MIYAKO DENSHI KOBO

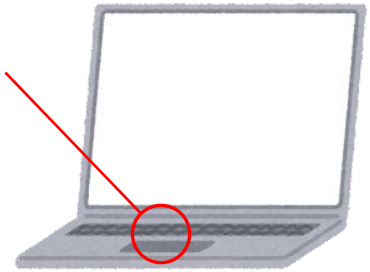
CPUを作れるようになる

CPUとは

CPU(中央演算処理装置)
Central Processing Unit

・プログラムを解釈して実行できる





CPUが作れるまでのロードマップ

みやこ電子工房
MIYAKO DENSHI KOBO

CPUを構成する
概念の理解

- レジスタ
- データのコピー
- 組み合わせ回路
- 命令とアドレス
- クロック
- リセット

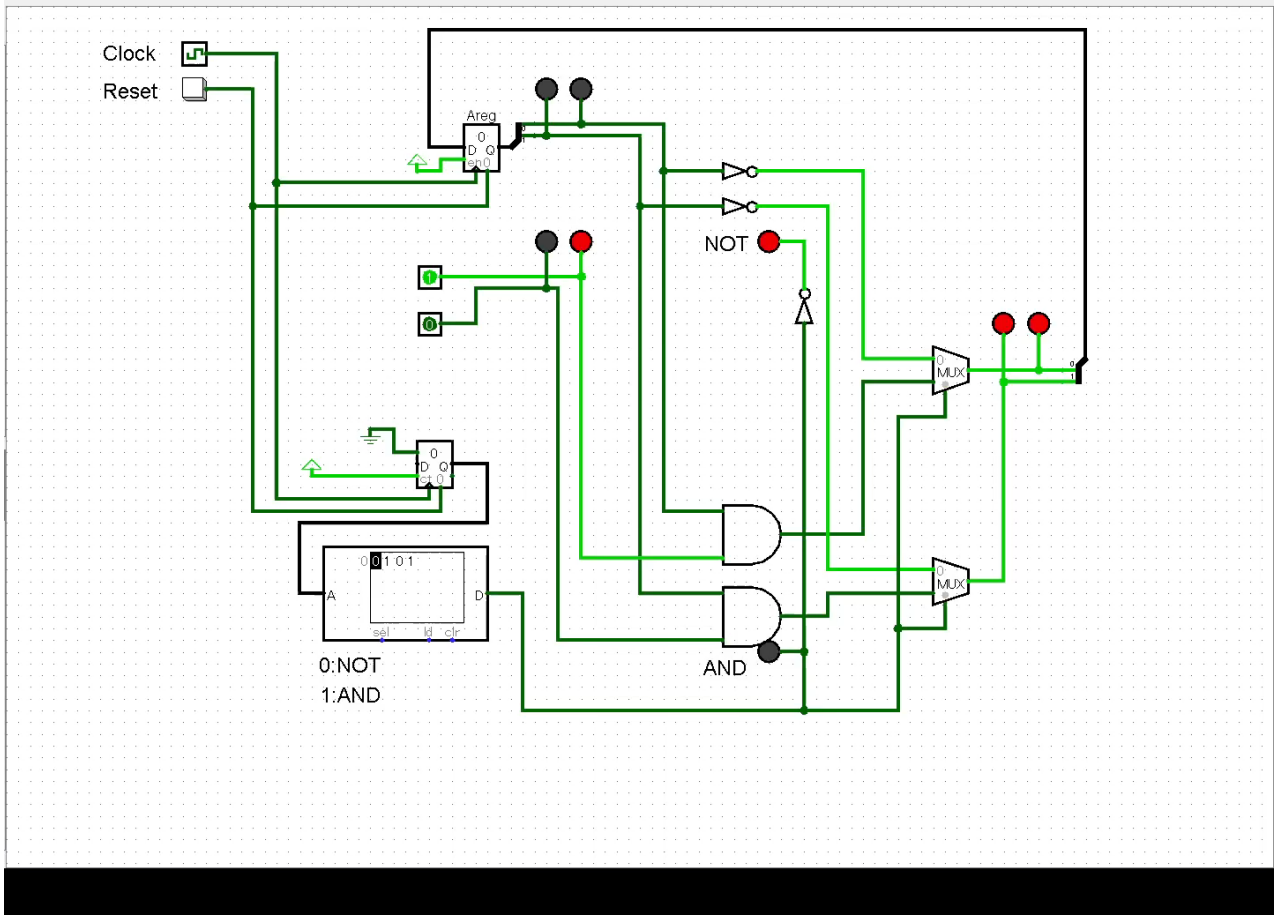
CPUのつくりかたを理解

- 回路をループさせる
- プログラムカウンタ
- 命令デコーダ
- シミュレータ Logisim
- Logisim上で作成

今日つくるCPUの紹介

みやこ電子工房
MIYAKO DENSHI KOBO

名前	2bit-MEW
bit数	2bit
レジスタ数	2
入出力	入力2bit
命令数	2
アドレス幅	2bit



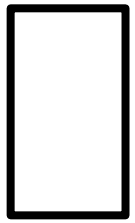
2bitCPUをつくっていきます

概念の学習

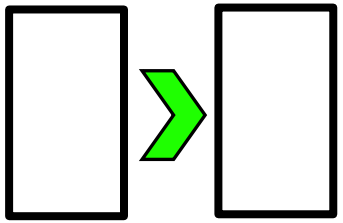
CPUを構成する概念

みやこ電子工房
MIYAKO DENSHI KOBO

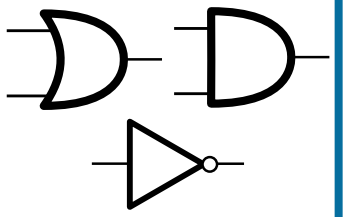
レジスタ



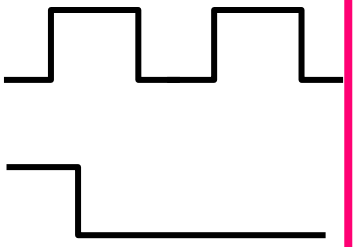
データのコピー



組み合わせ回路



クロックとリセット



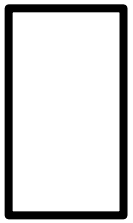
命令とアドレス

アドレス	命令
00	命令1
01	命令2
02	命令3
03	命令4

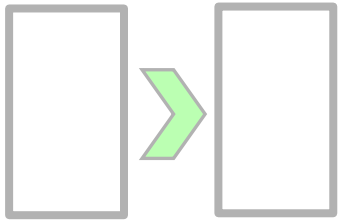
CPUを構成する概念

みやこ電子工房
MIYAKO DENSHI KOBO

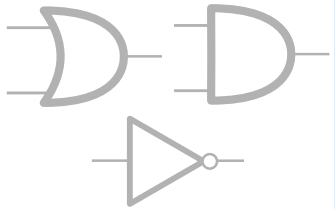
レジスタ



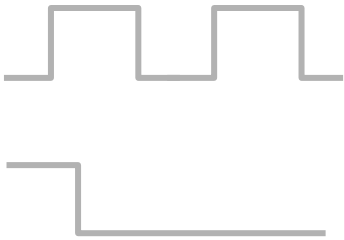
データのコピー



組み合わせ回路



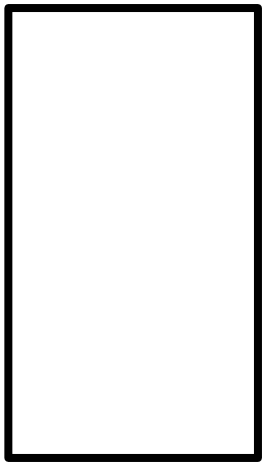
クロックとリセット



命令とアドレス

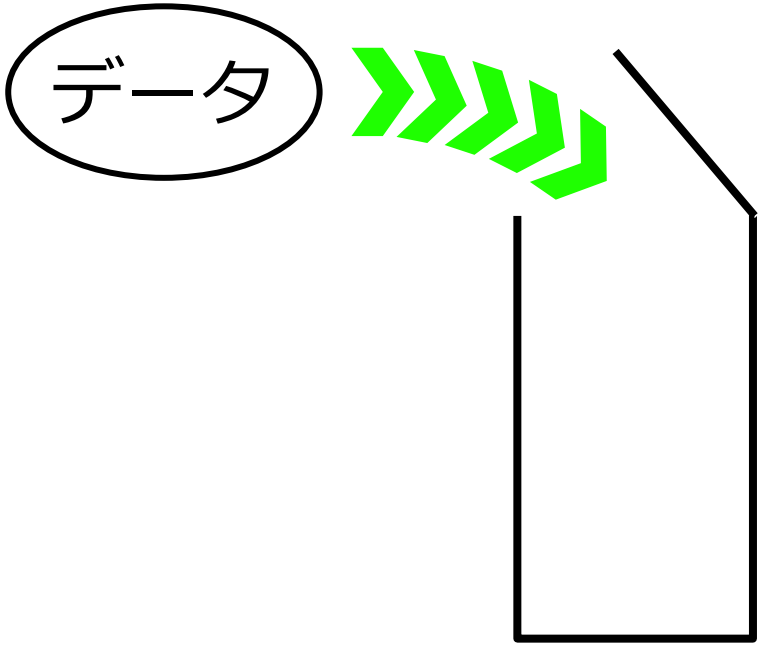
アドレス	命令
00	命令1
01	命令2
02	命令3
03	命令4

レジスタ



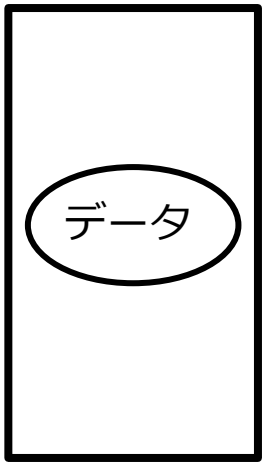
レジスタとはデータを入れる箱である

レジスタ



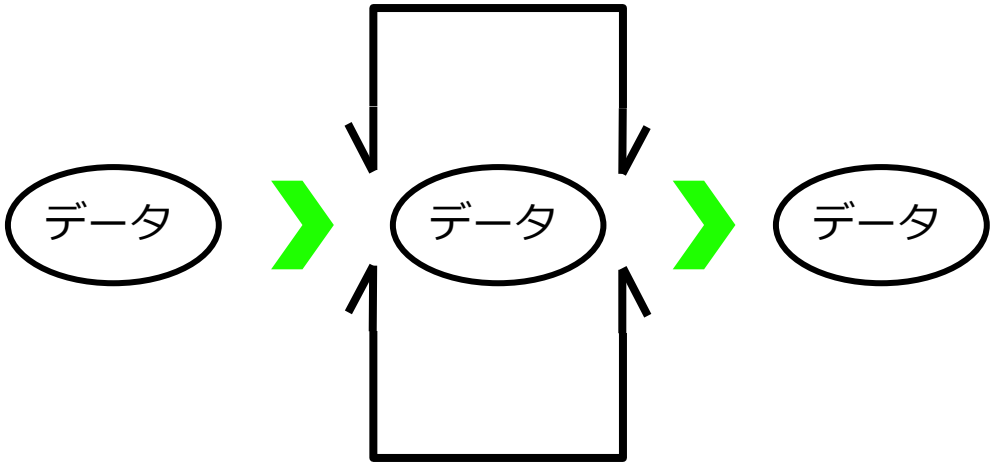
レジスタ ＝ データを入れる箱

レジスタ



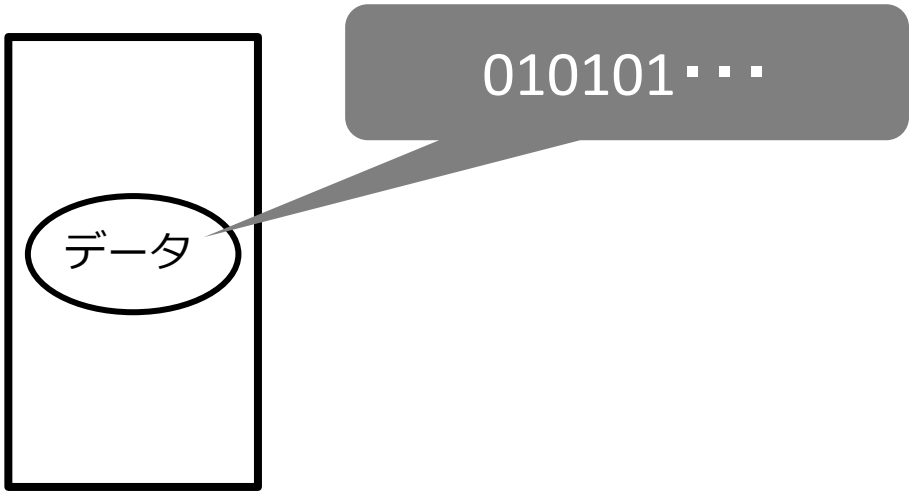
レジスタの中にはデータが入っている

レジスタ



レジスタにはデータが入る入り口と
データが出て行く出口がある

レジスタ



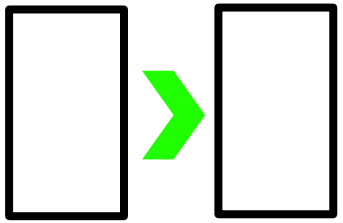
データは2進数

CPUを構成する概念

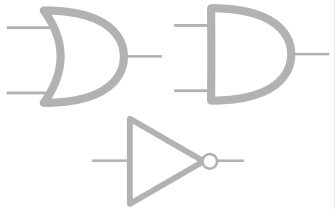
みやこ電子工房
MIYAKO DENSHI KOBO

レジスタ

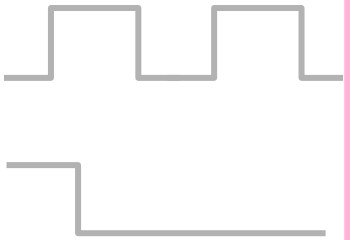
データのコピー



組み合わせ回路



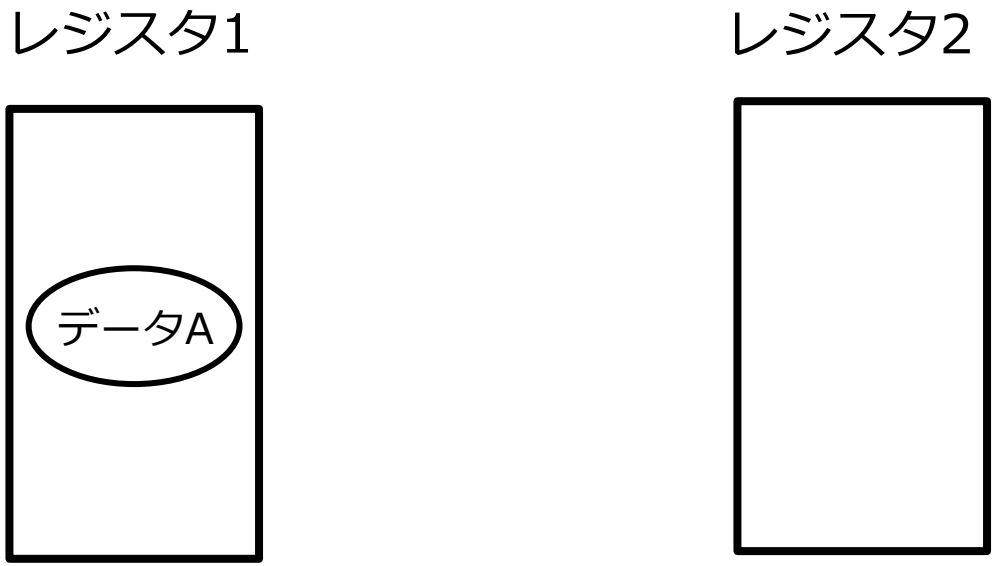
クロックとリセット



命令とアドレス

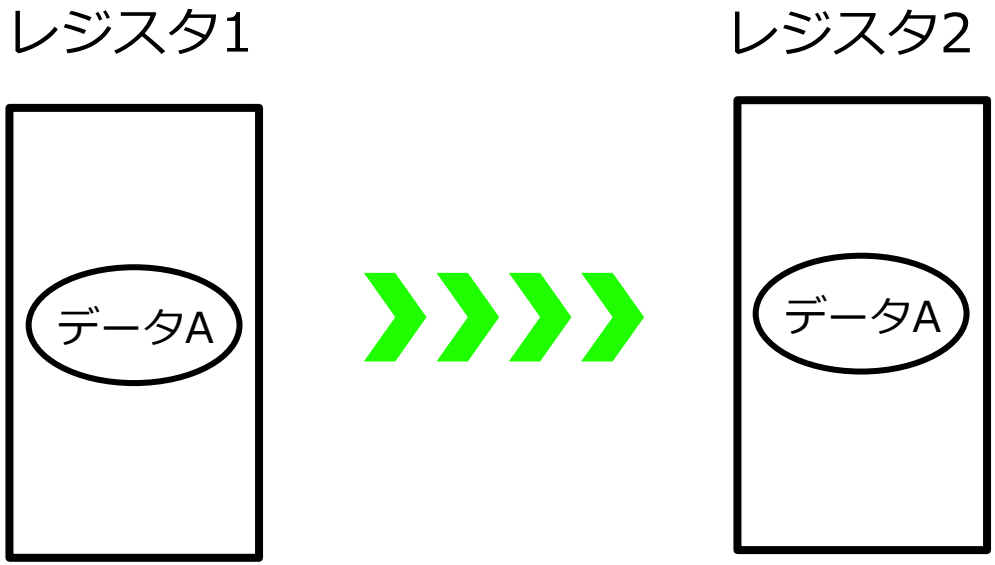
アドレス	命令
00	命令1
01	命令2
02	命令3
03	命令4

データのコピー



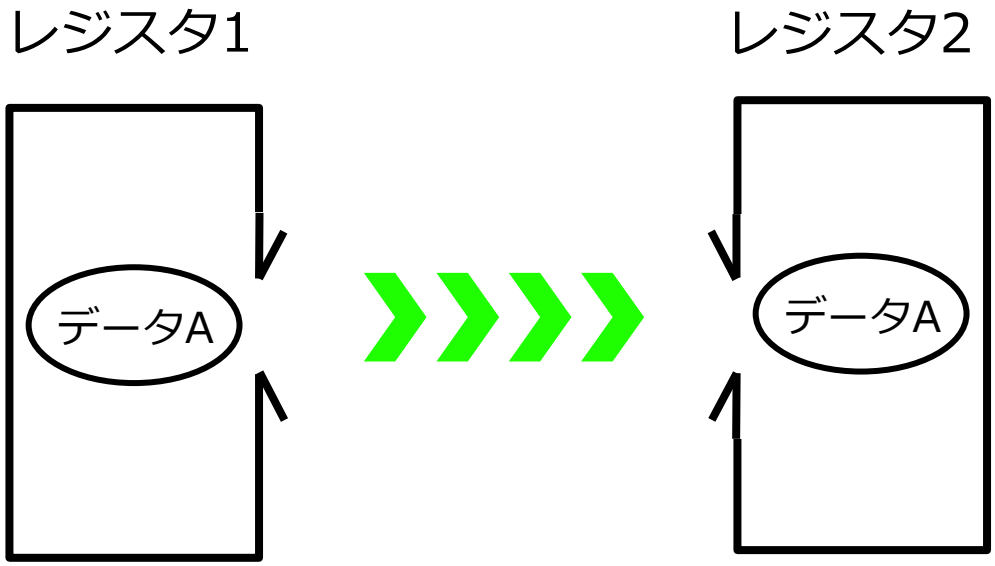
レジスタ2へデータAを格納したい

データのコピー



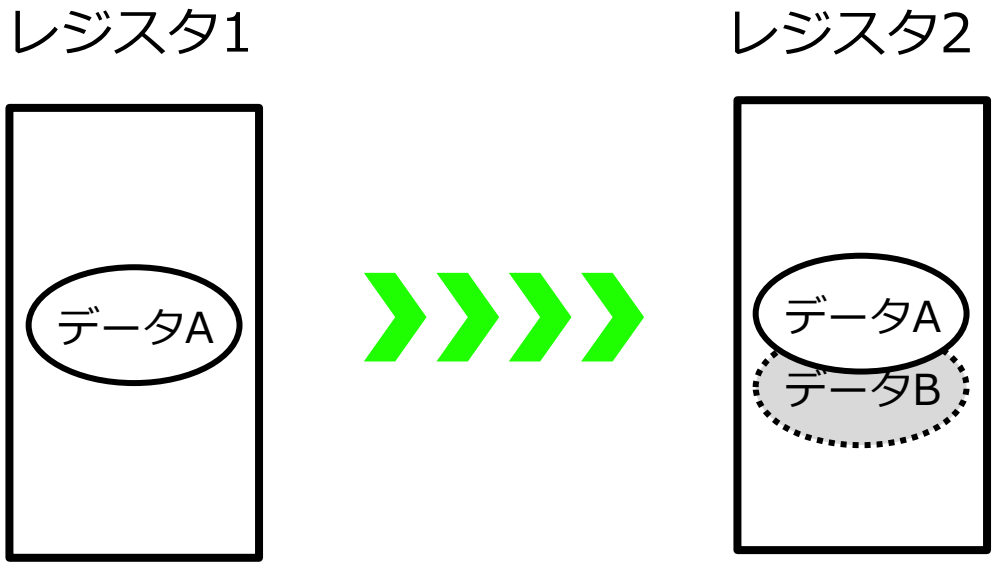
レジスタ1からレジスタ2へコピーする

データのコピー



コピーするときは出入り口を開ける

データのコピー



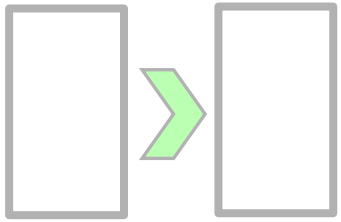
レジスタ2は上書きされる

CPUを構成する概念

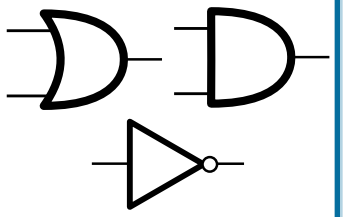
みやこ電子工房
MIYAKO DENSHI KOBO

レジスタ

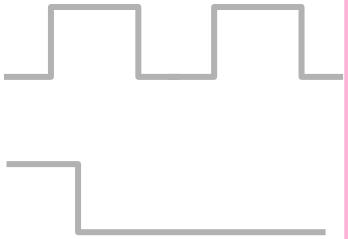
データのコピー



組み合わせ回路



クロックとリセット



命令とアドレス

アドレス	命令
00	命令1
01	命令2
02	命令3
03	命令4

組み合わせ回路

組み合わせ回路はデータを変換する

組み合わせ回路

変換元が2つのデータの時もある

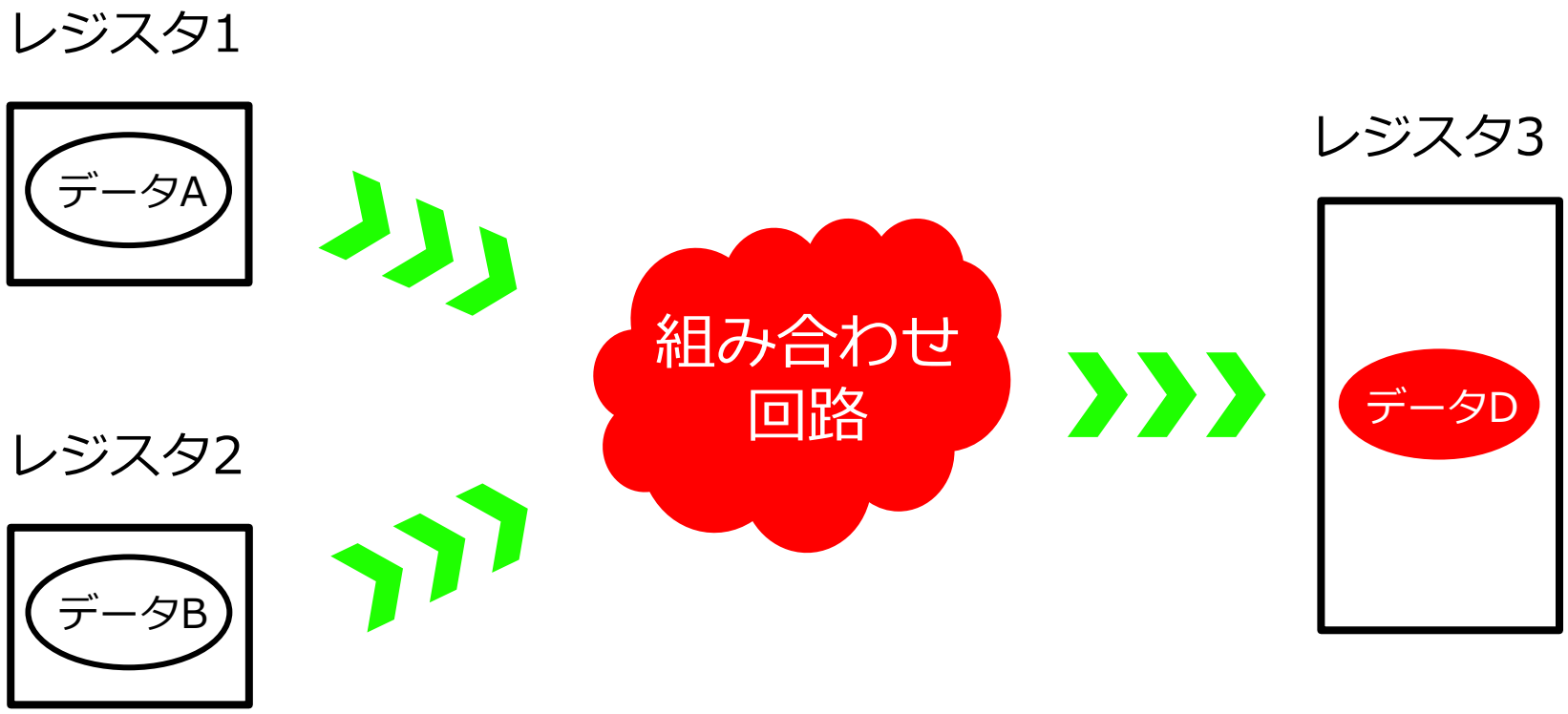
組み合わせ回路

データをコピーする途中に
組み合わせ回路を介して変換

組み合わせ回路

変換元が2つのデータの時

組み合わせ回路



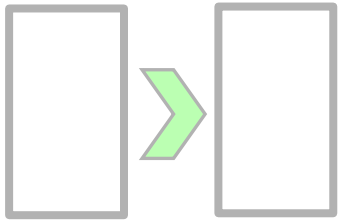
組み合わせ回路を変えると
変換の仕方も変わる

CPUを構成する概念

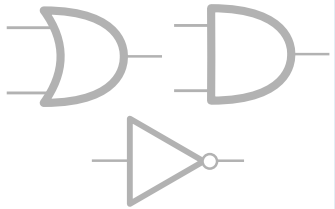
みやこ電子工房
MIYAKO DENSHI KOBO

レジスタ

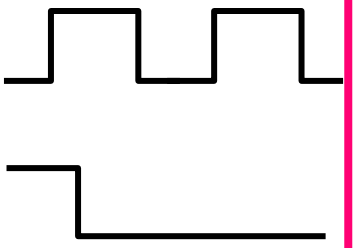
データのコピー



組み合わせ回路



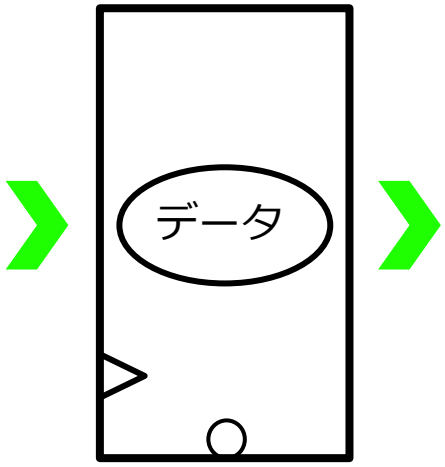
クロックとリセット



命令とアドレス

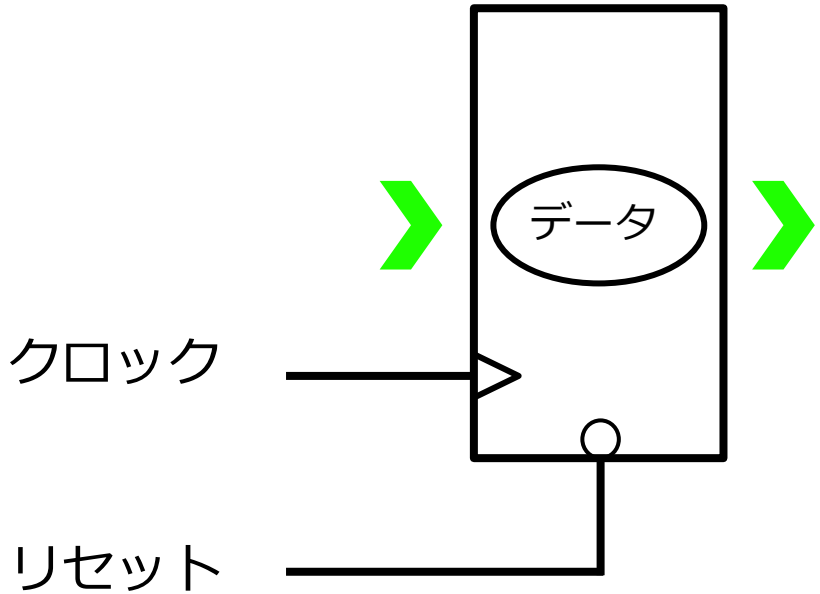
アドレス	命令
00	命令1
01	命令2
02	命令3
03	命令4

クロックとリセット



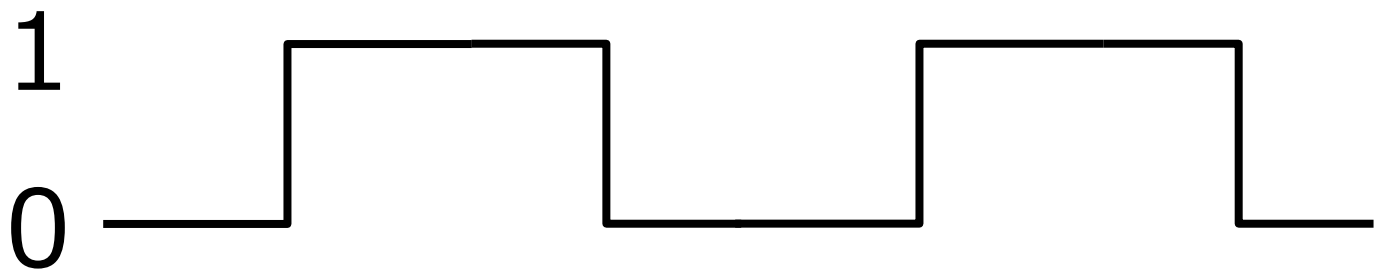
レジスタにはデータの出入り口の他に
レジスタを制御する口がある

クロックとリセット



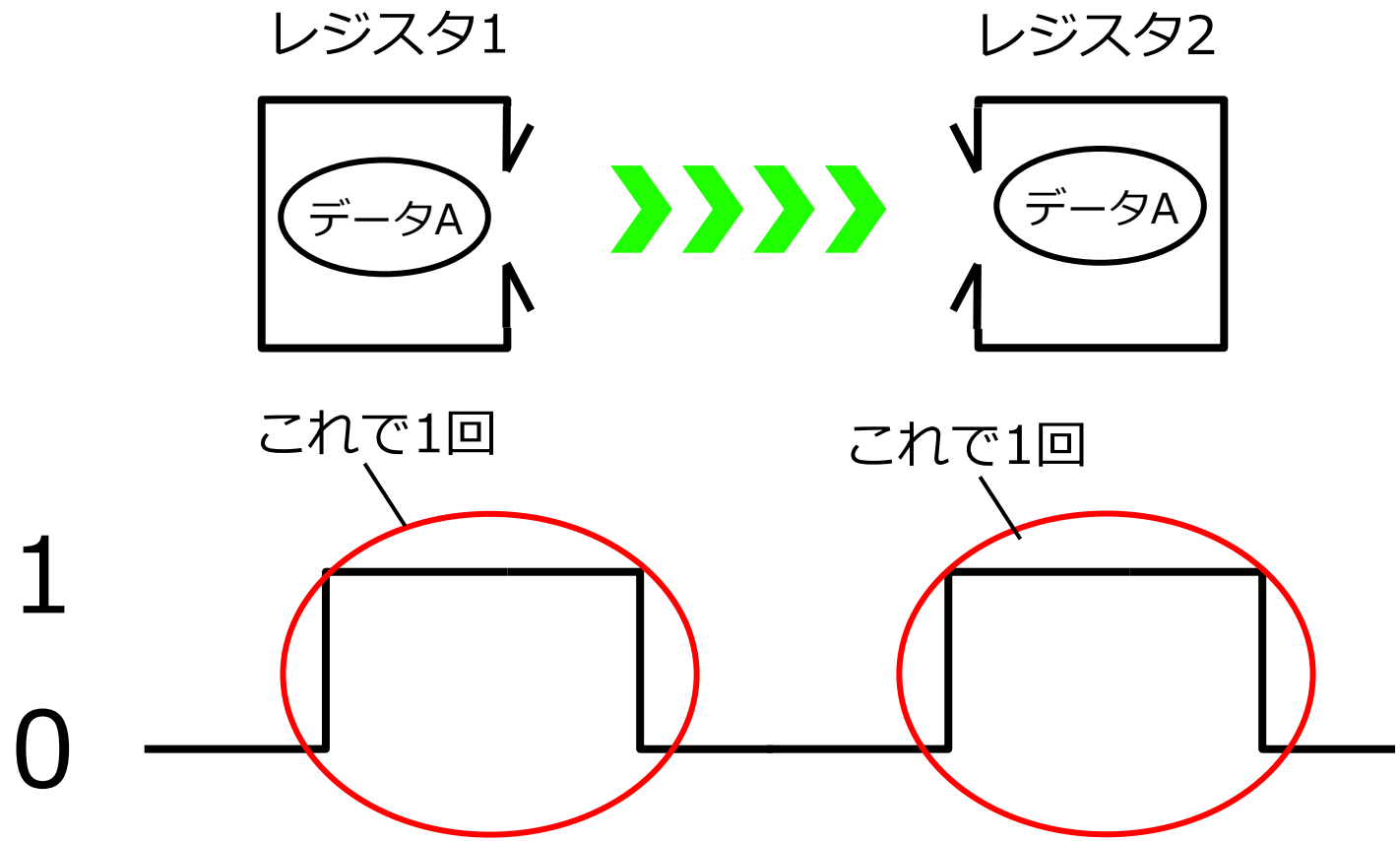
クロックは動きを制御する
リセットはデータをリセットする

クロックとリセット



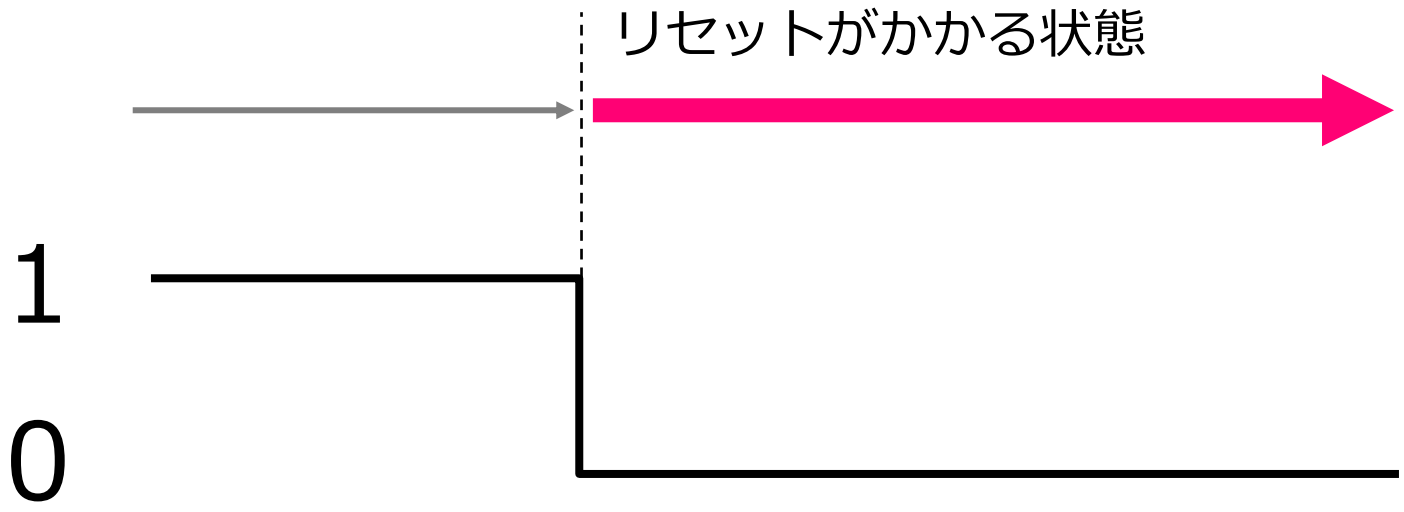
クロックは0と1の繰り返しの信号

クロックとリセット



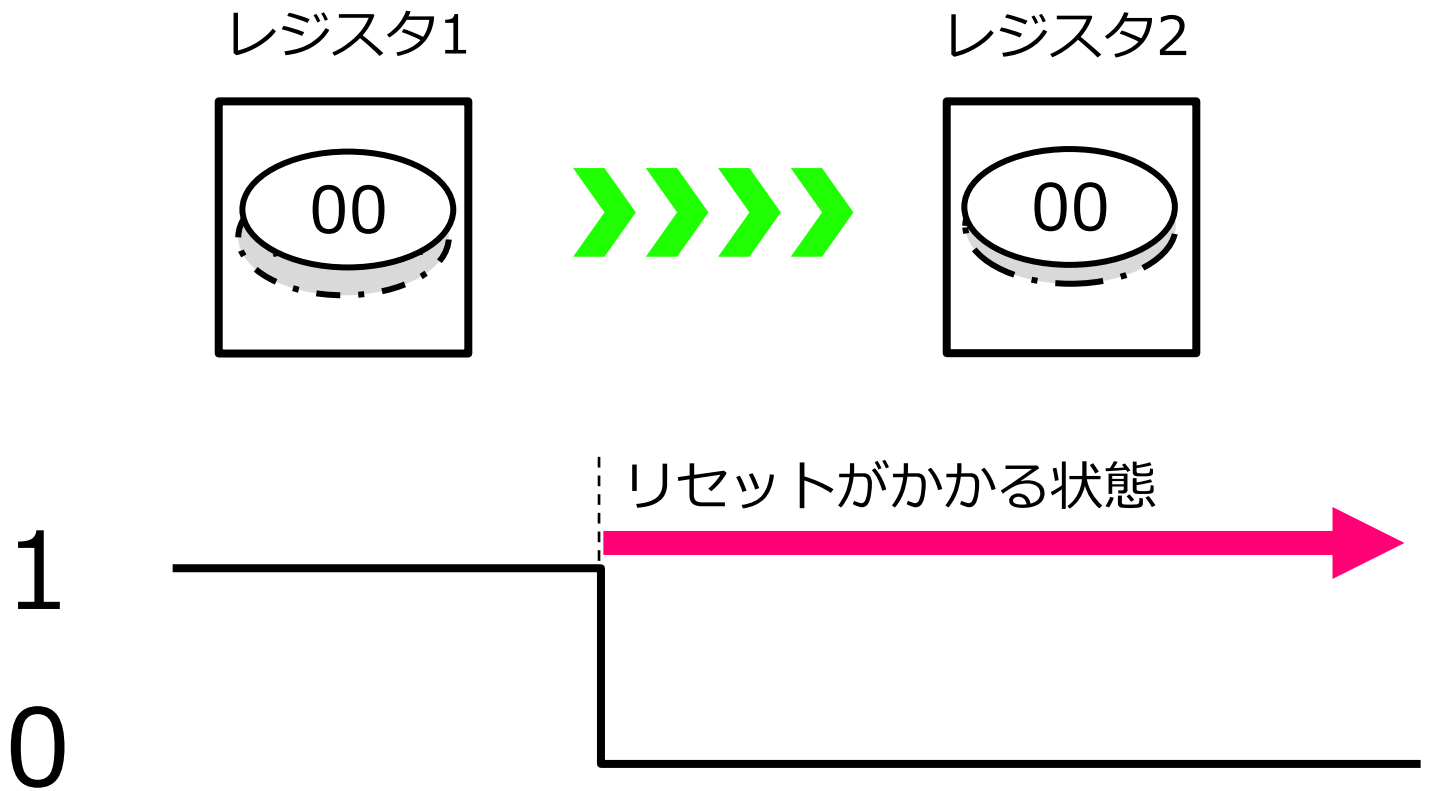
クロック1回でコピーが1回

クロックとリセット



リセットは普段ずっと1
リセットをかける時に0になる

クロックとリセット



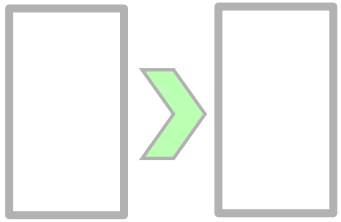
リセットがかかるとデータが
0に上書きされる

CPUを構成する概念

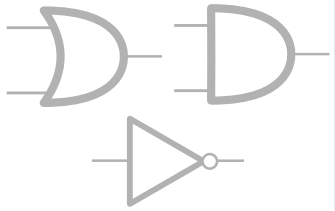
みやこ電子工房
MIYAKO DENSHI KOBO

レジスタ

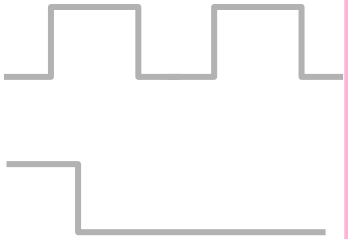
データのコピー



組み合わせ回路



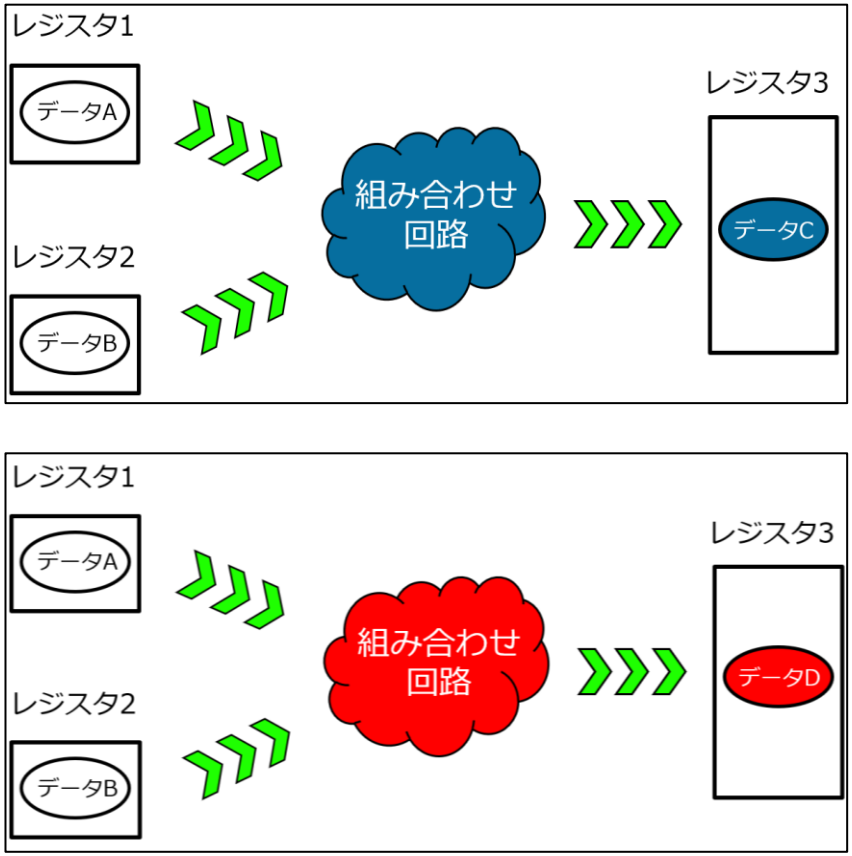
クロックとリセット



命令とアドレス

アドレス	命令
00	命令1
01	命令2
02	命令3
03	命令4

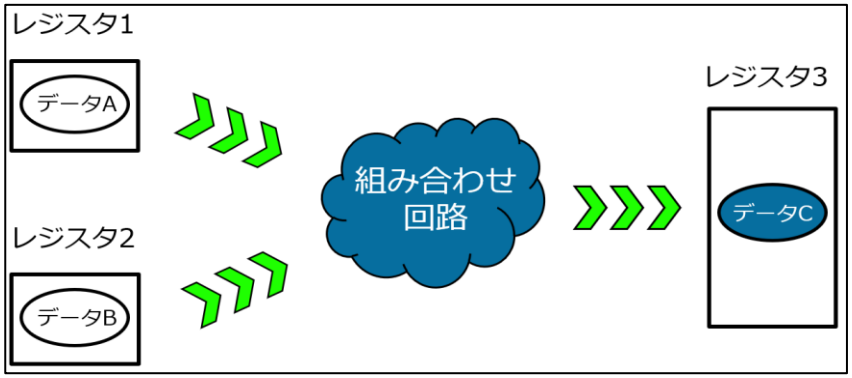
命令とアドレス



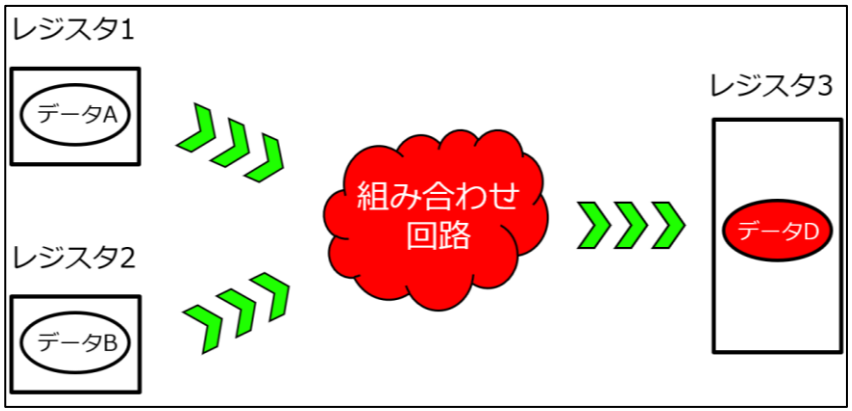
使用する組み合わせ回路によって
変換結果が変わる

命令とアドレス

命令0

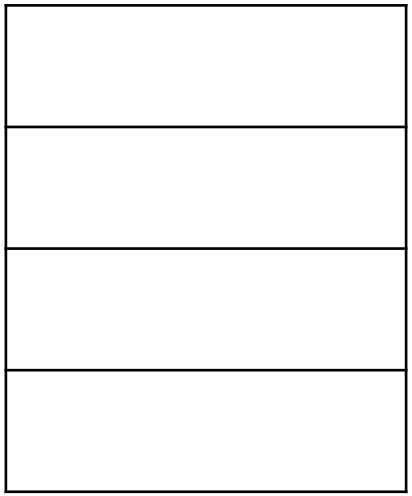


命令1



使用する組み合わせ回路に
名前を付ける。→命令名

命令とアドレス

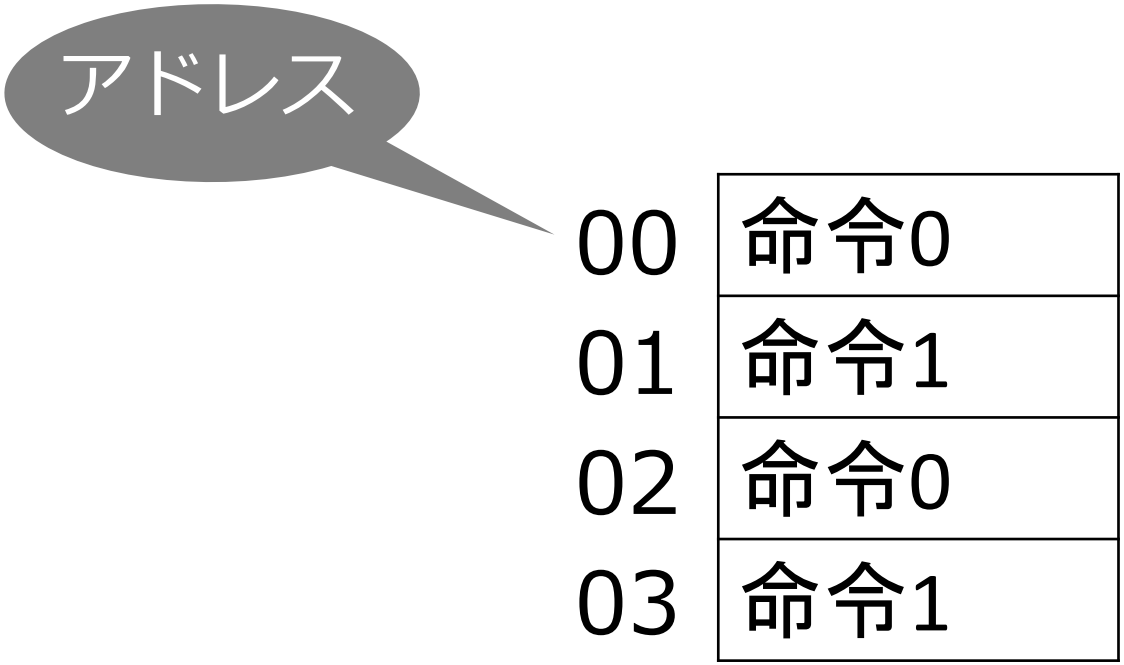


命令を入れる箱をたくさん用意する

命令とアドレス

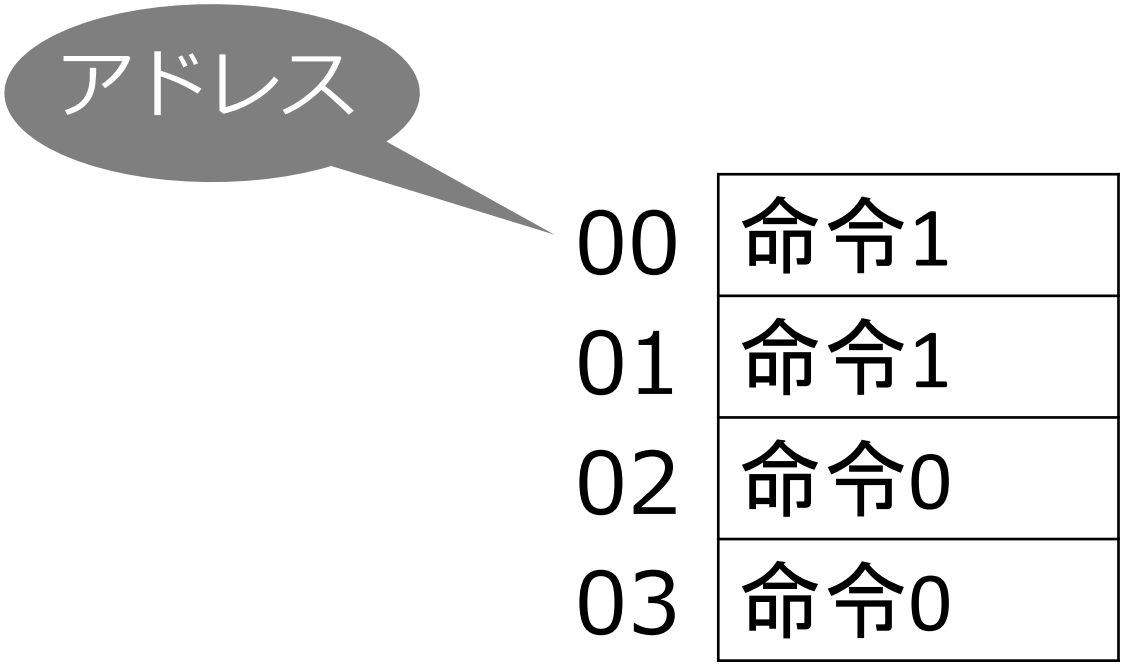
箱に実行したい命令を入れておく
複数並んだ命令＝プログラム

命令とアドレス



箱には番号がついている
これがアドレス

命令とアドレス



実行したい順番に命令を入れる

命令とアドレス

みやこ電子工房
MIYAKO DENSHI KOBO

メモリ

00	命令1
01	命令1
02	命令0
03	命令0

いっぱいの箱はメモリという

CPUを構成する概念

みやこ電子工房
MIYAKO DENSHI KOBO

レジスタ	データを入れる箱
データのコピー	データをレジスタからレジスタへコピー
組み合わせ回路	コピーする途中でデータを変換
クロックとリセット	クロックでコピーが進む リセットで全部0になる
命令とアドレス	どの組み合わせ回路を使うか指示する

CPUをつくる

CPUをつくる

みやこ電子工房
MIYAKO DENSHI KOBO

- 回路をループさせる
- プログラムカウンタ
- 命令デコーダ

- シミュレータ　Logisim
- Logisim上で作成

回路をループさせる

コピーの回路をもってくる

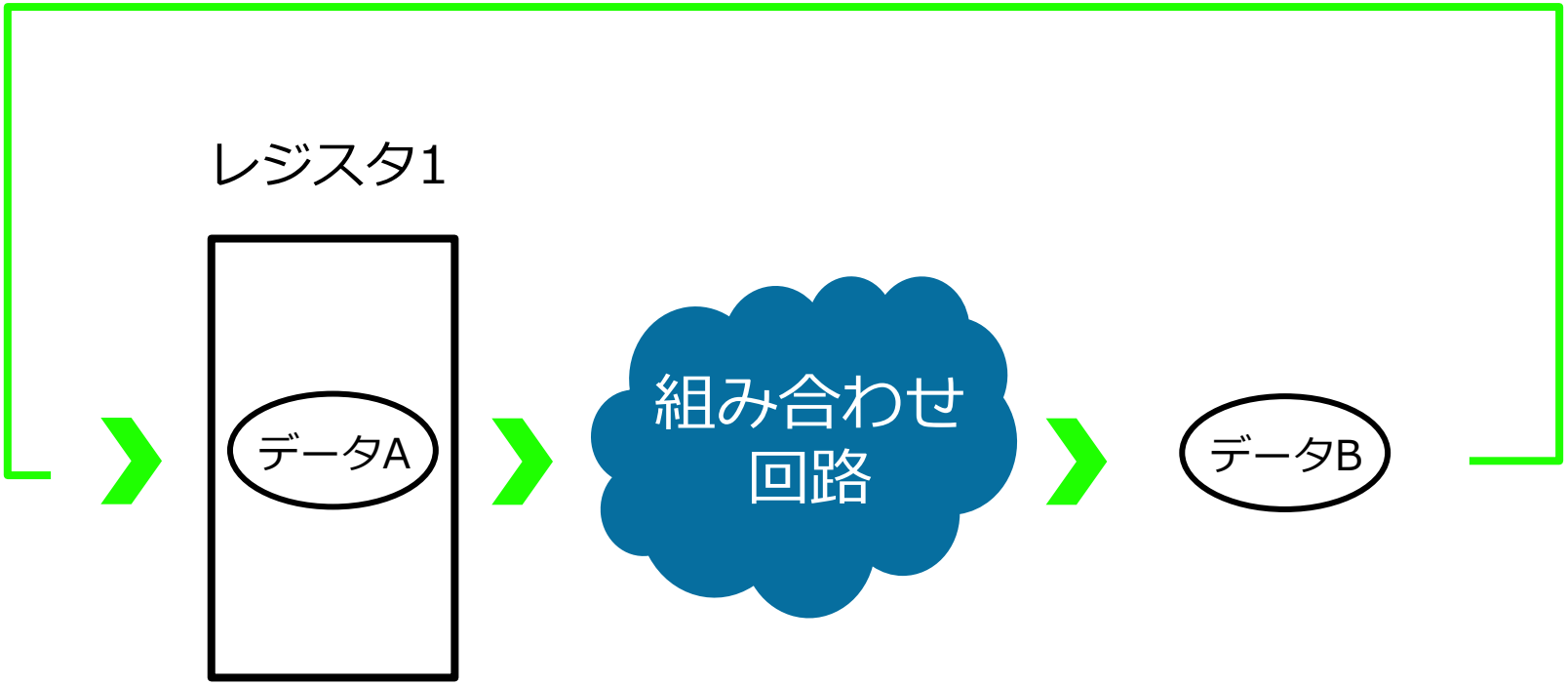
回路をループさせる

コピーは一度実行すると止まってしまう
次々に命令を実行させたい

回路をループさせる

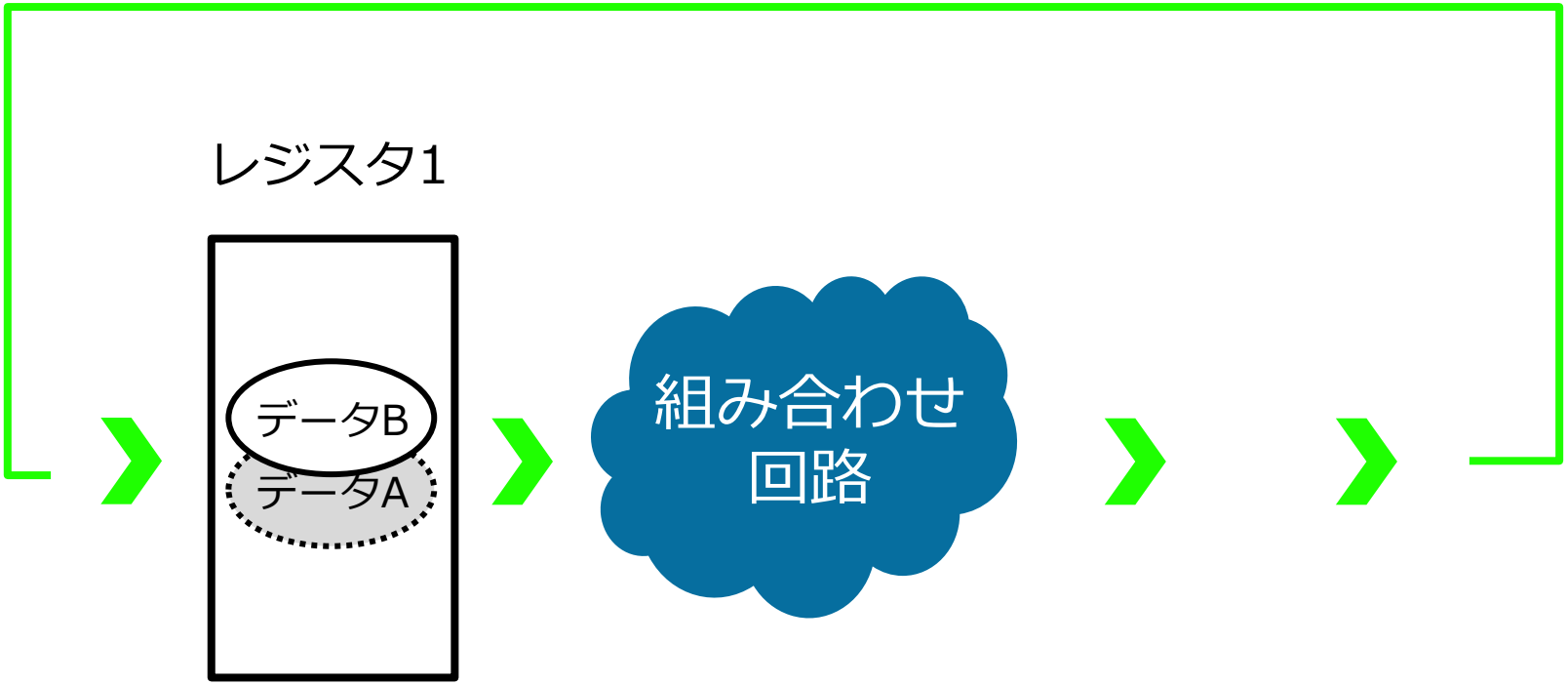
レジスタ2を削除する

回路をループさせる



ループさせてレジスタ1につなぐ

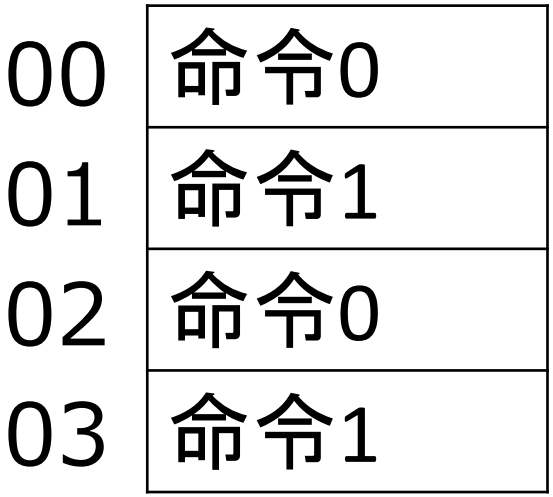
回路をループさせる



データAがデータBに変換され上書き

プログラムカウンタ

メモリ



メモリをもってくる

プログラムカウンタ

メモリ

00	命令0
01	命令1
02	命令0
03	命令1

命令を順番に実行するには
00→01→02→03と数える必要がある

プログラムカウンタ

プログラムカウンタ

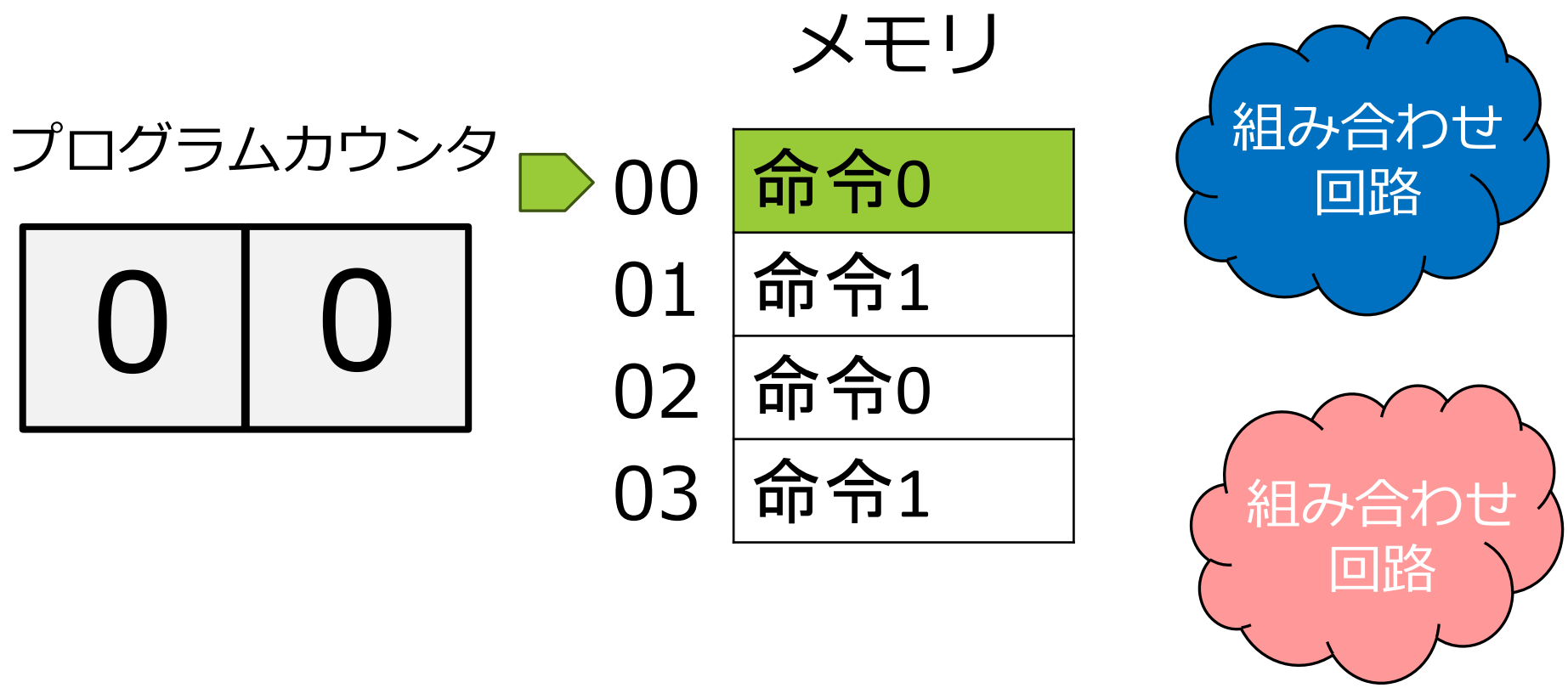
0	0

メモリ

00	命令0
01	命令1
02	命令0
03	命令1

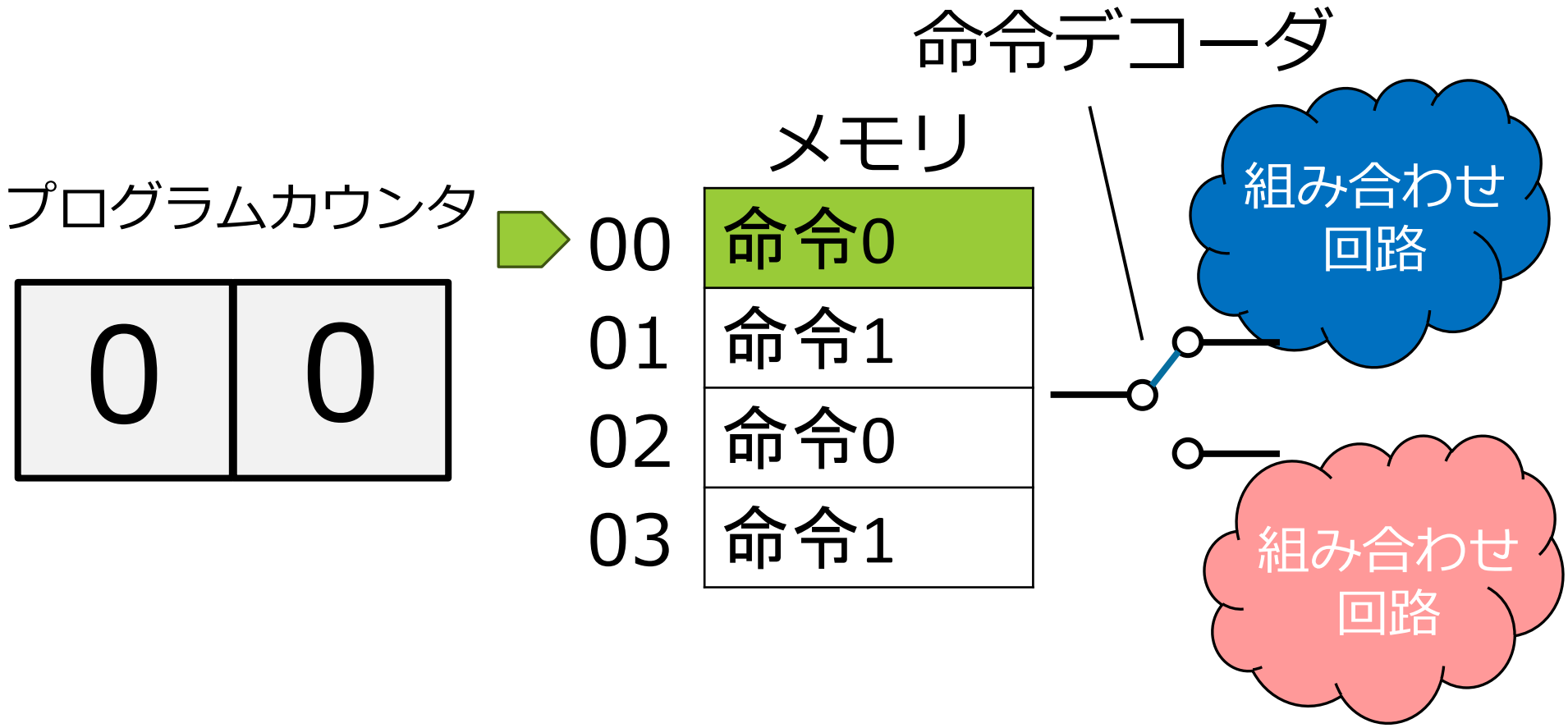
命令を順番に実行するには
00→01→02→03と数える必要がある

命令デコーダ



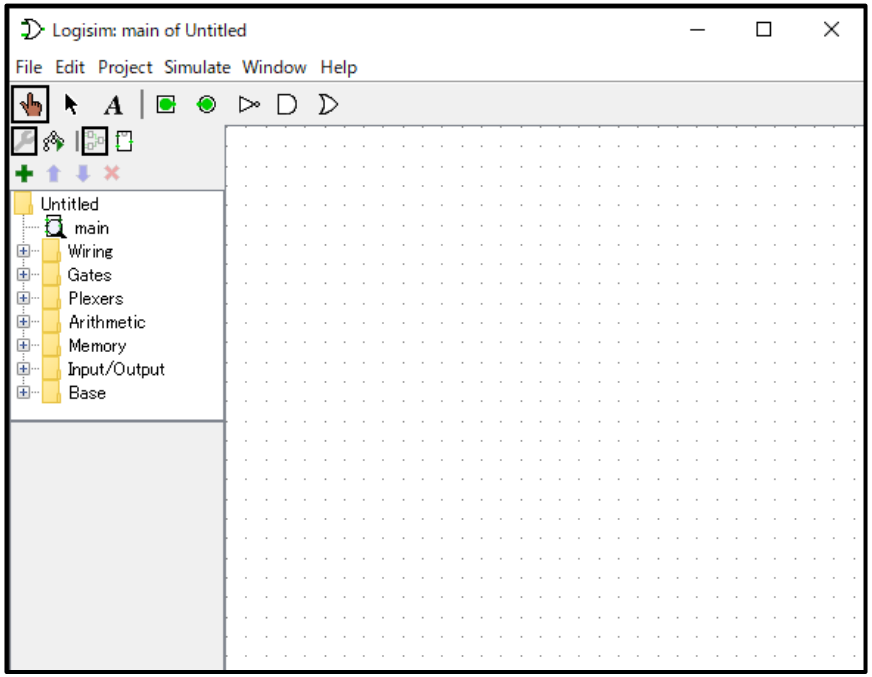
選択された命令により組み合わせ回路が選択される

命令デコーダ



メモリと組み合わせ回路の間に
命令デコーダがあり選択される

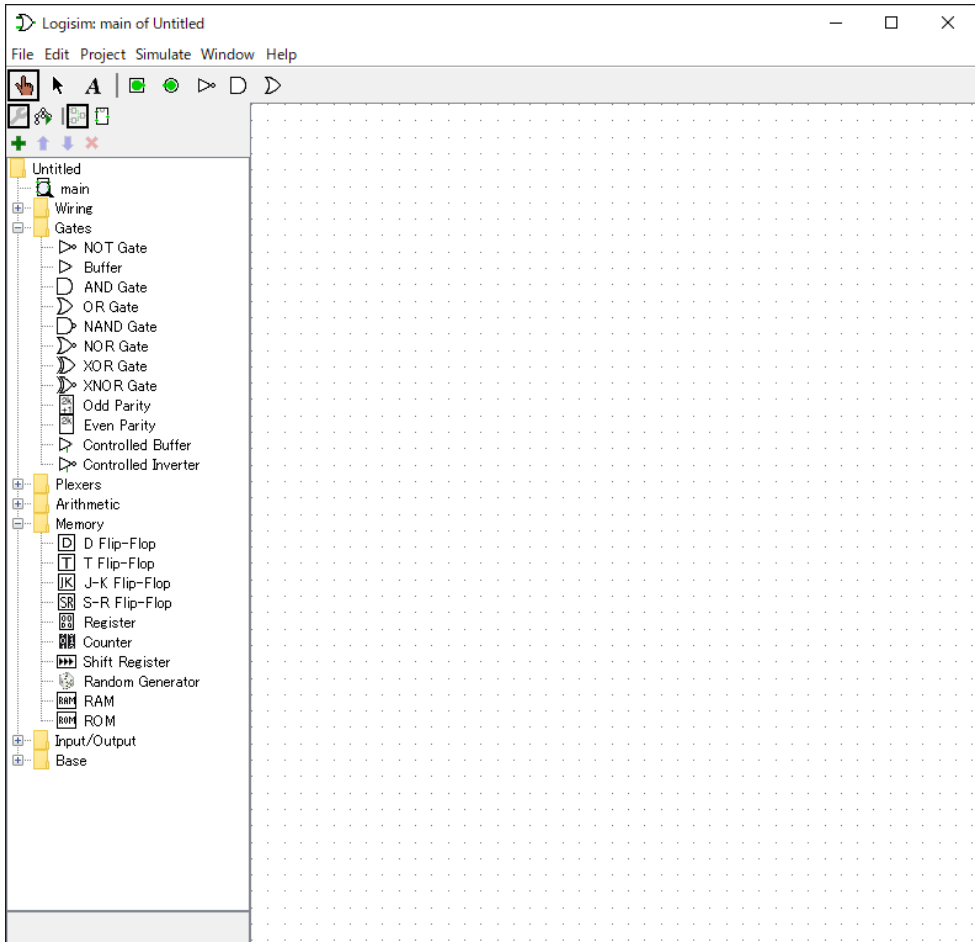
シミュレータLogisim



http://www.cburch.com/logisim/

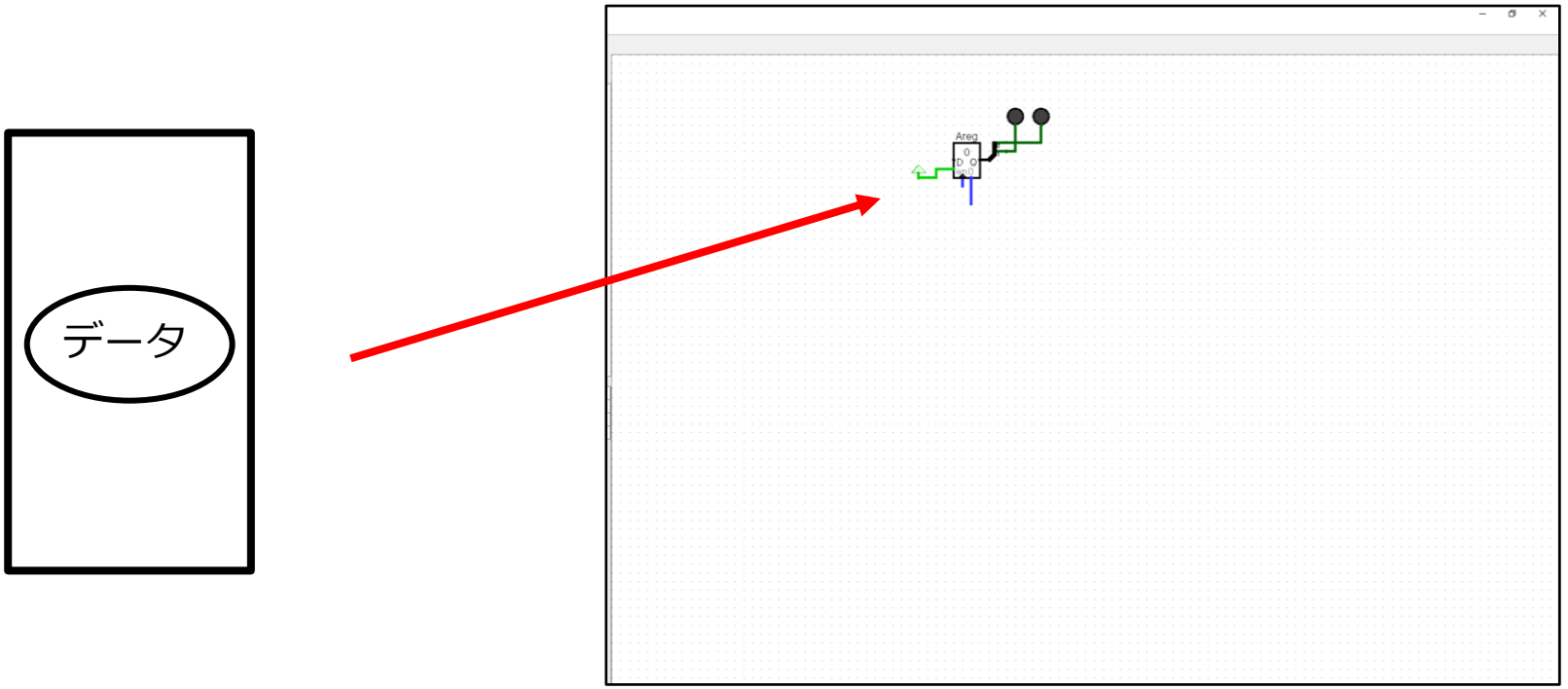
フリーの論理回路シミュレータです

シミュレータLogisim



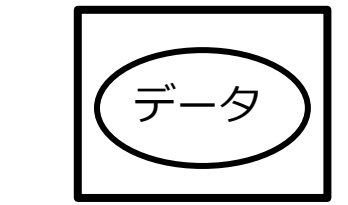
部品リストから部品を選択して配置する
詳細な使い方は割愛

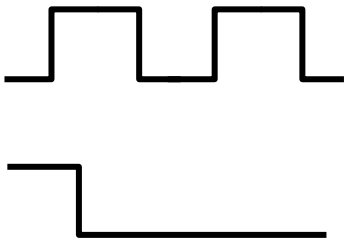
Logisim上で作成

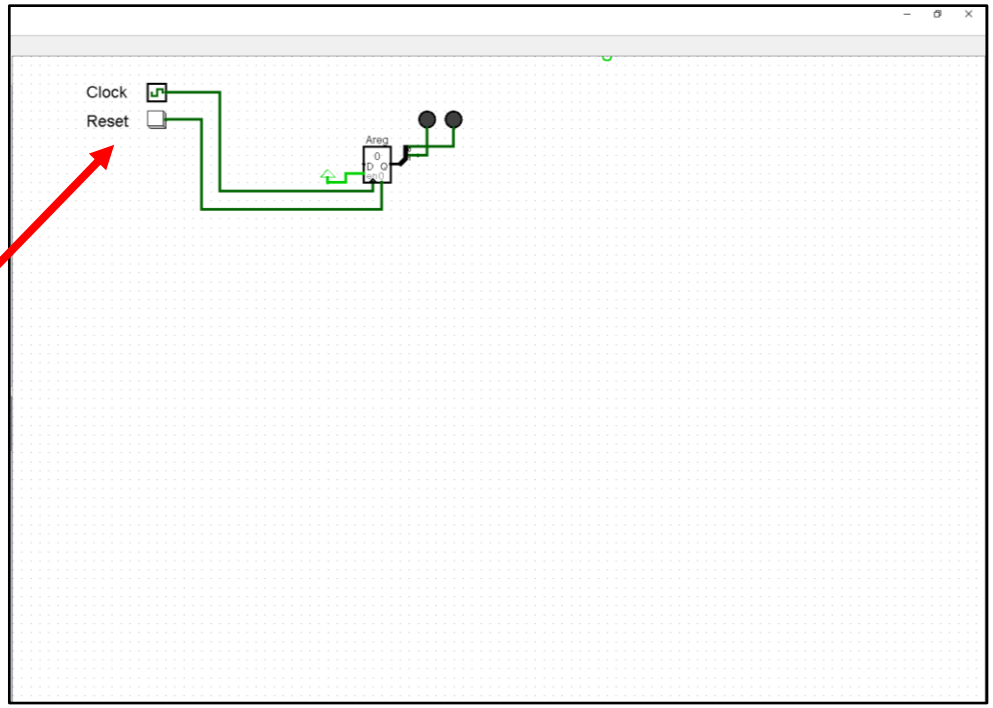


レジスタを1個配置する

Logisim上で作成

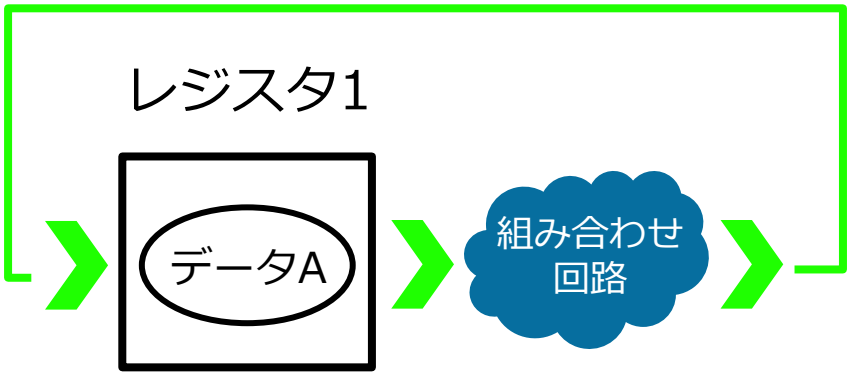


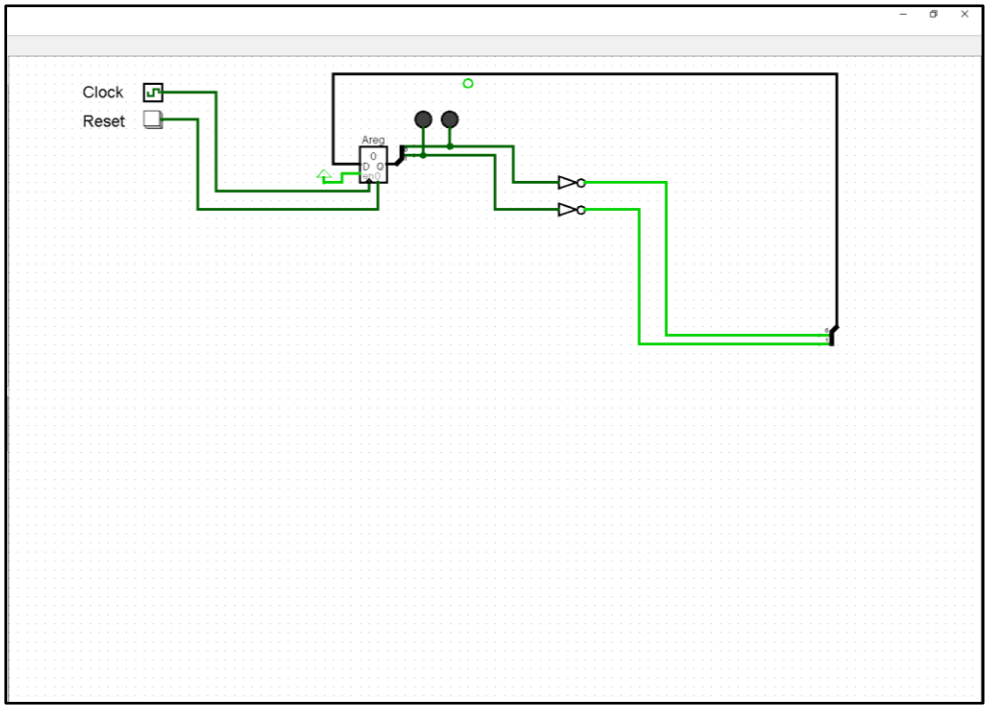




レジスタにクロックとリセットを追加

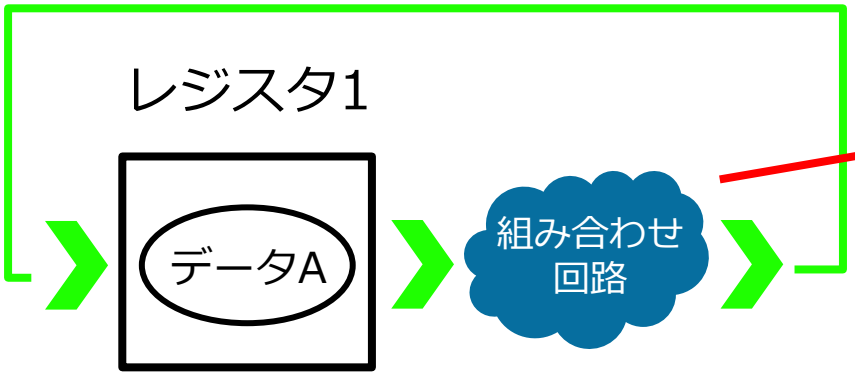
Logisim上で作成

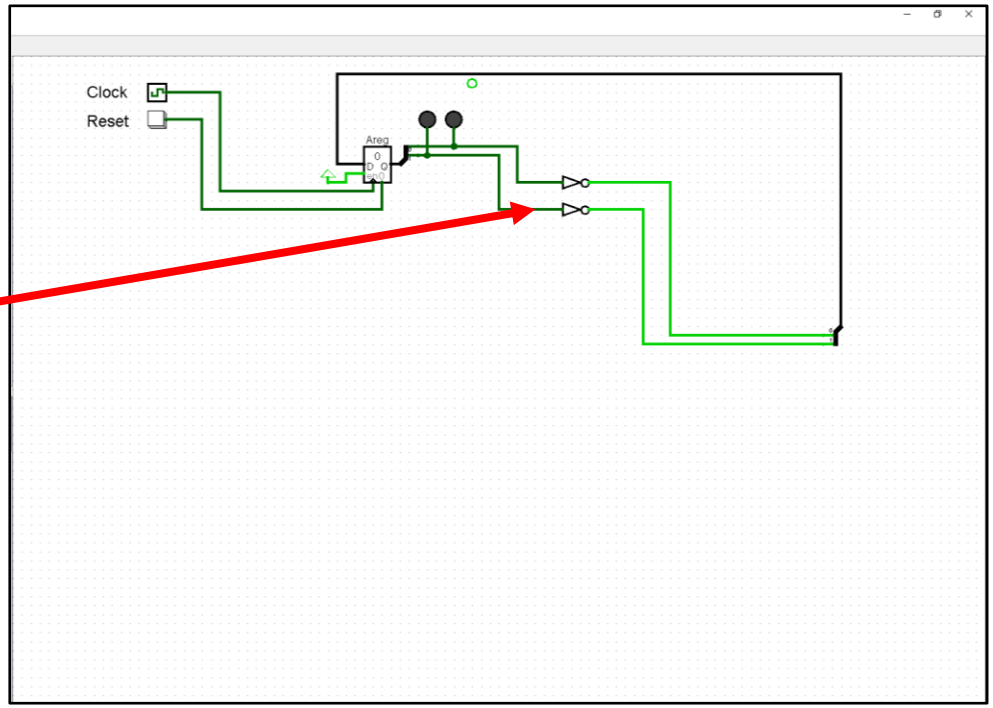




回路をループさせる

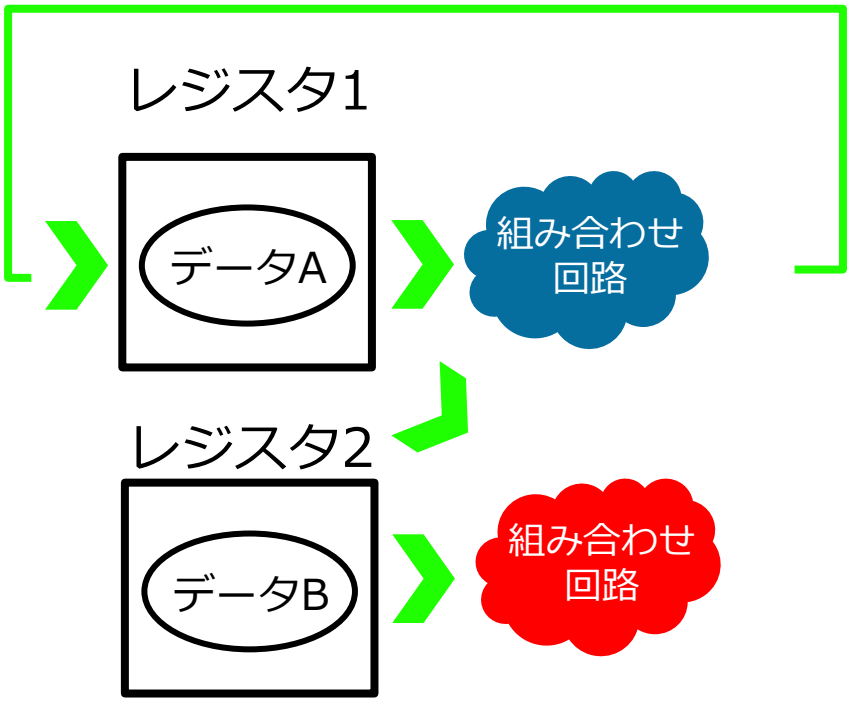
Logisim上で作成

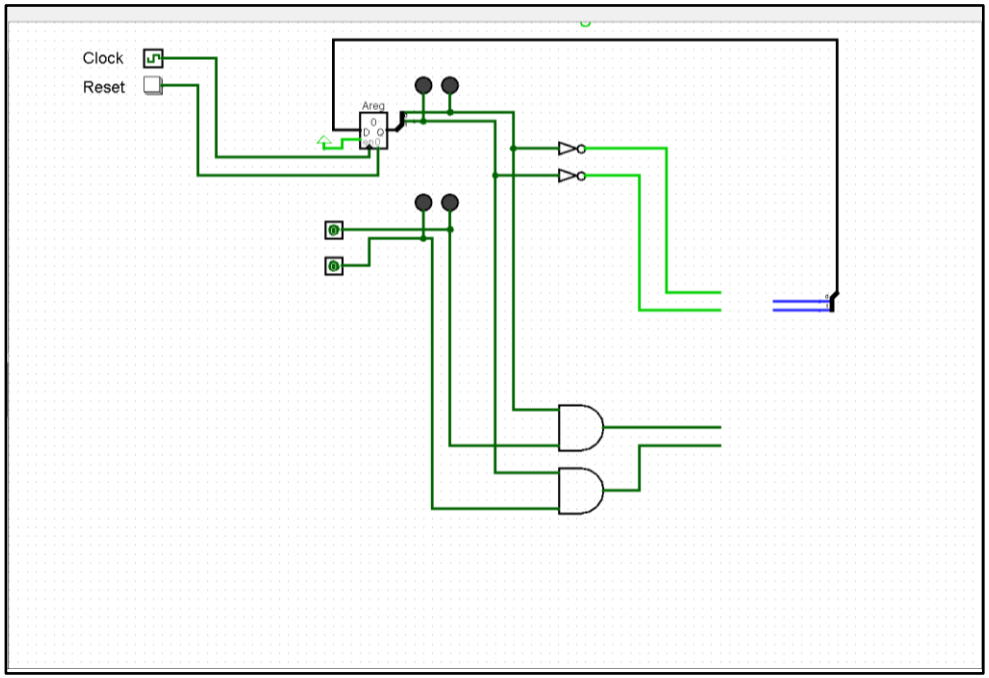




組み合わせ回路はNOT回路

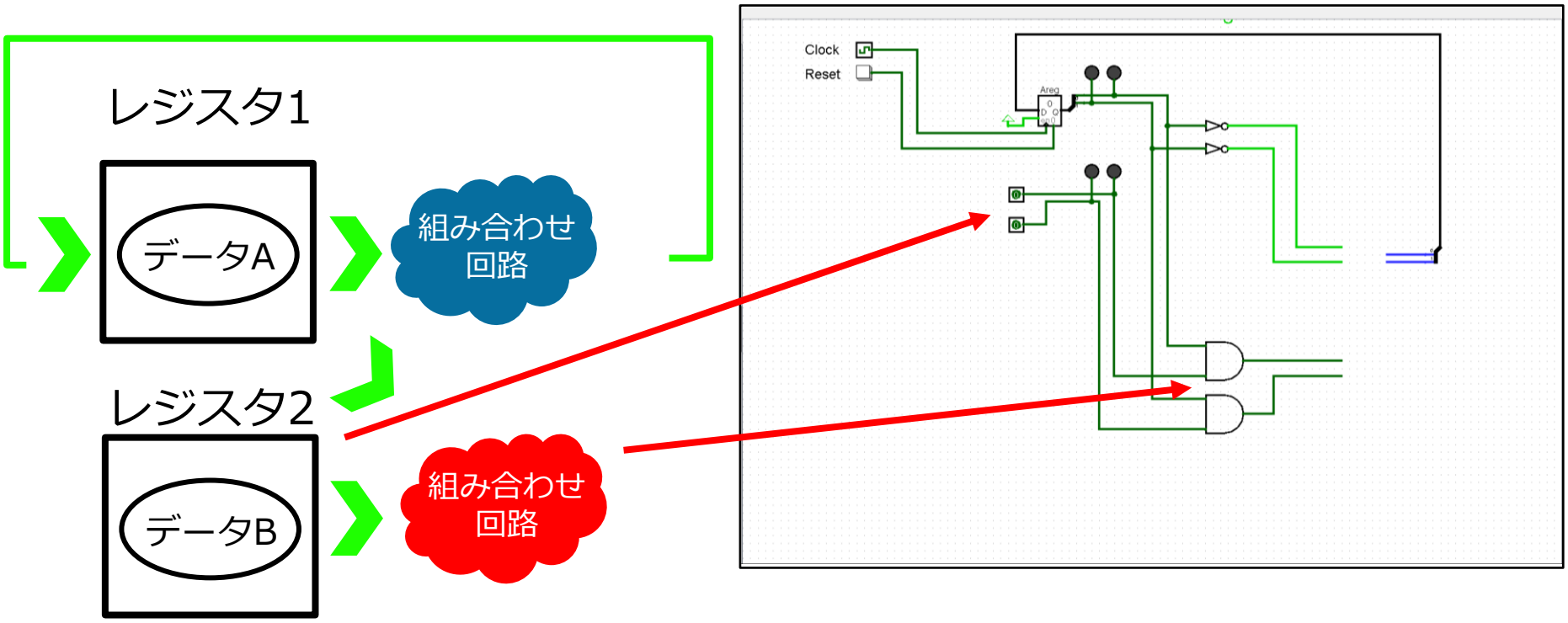
Logisim上で作成





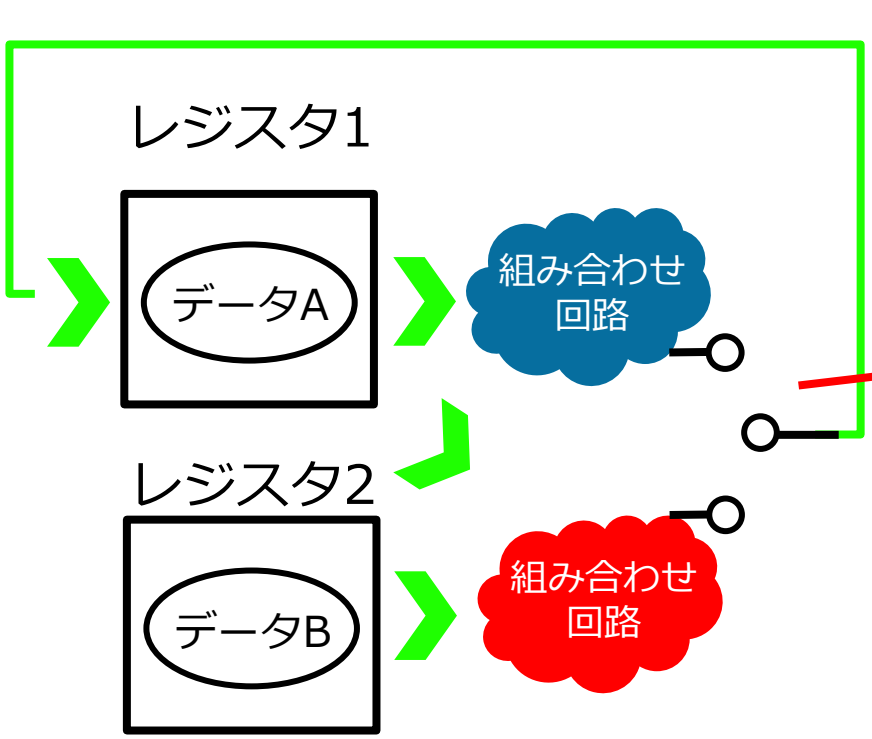
レジスタ2と組み合わせ回路を追加

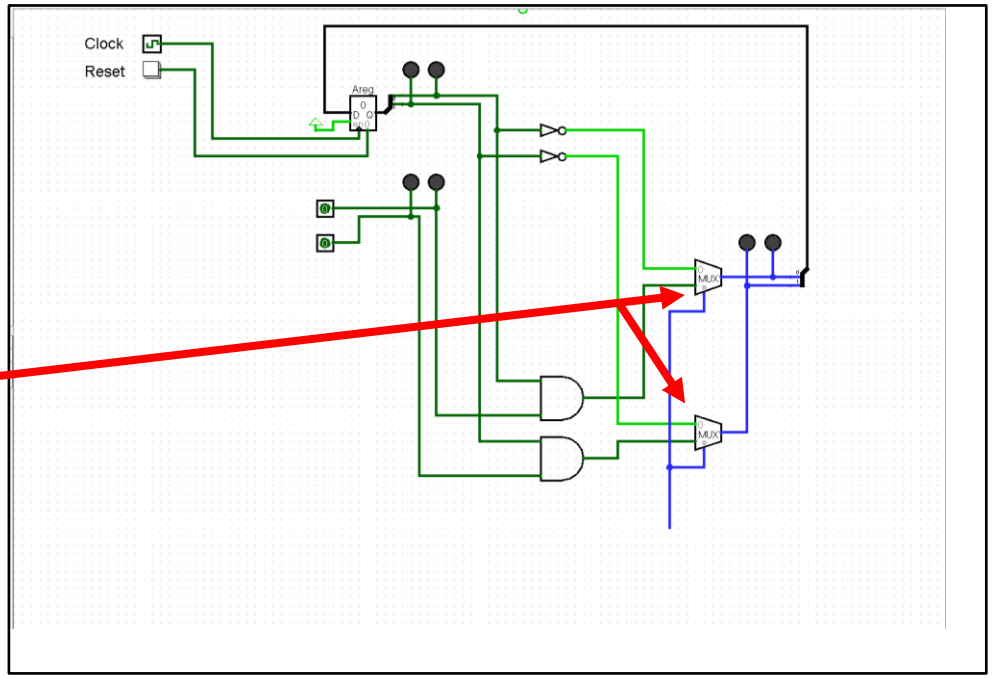
Logisim上で作成



レジスタ2と組み合わせ回路を追加
AND回路は変換元が2つの組み合わせ回路

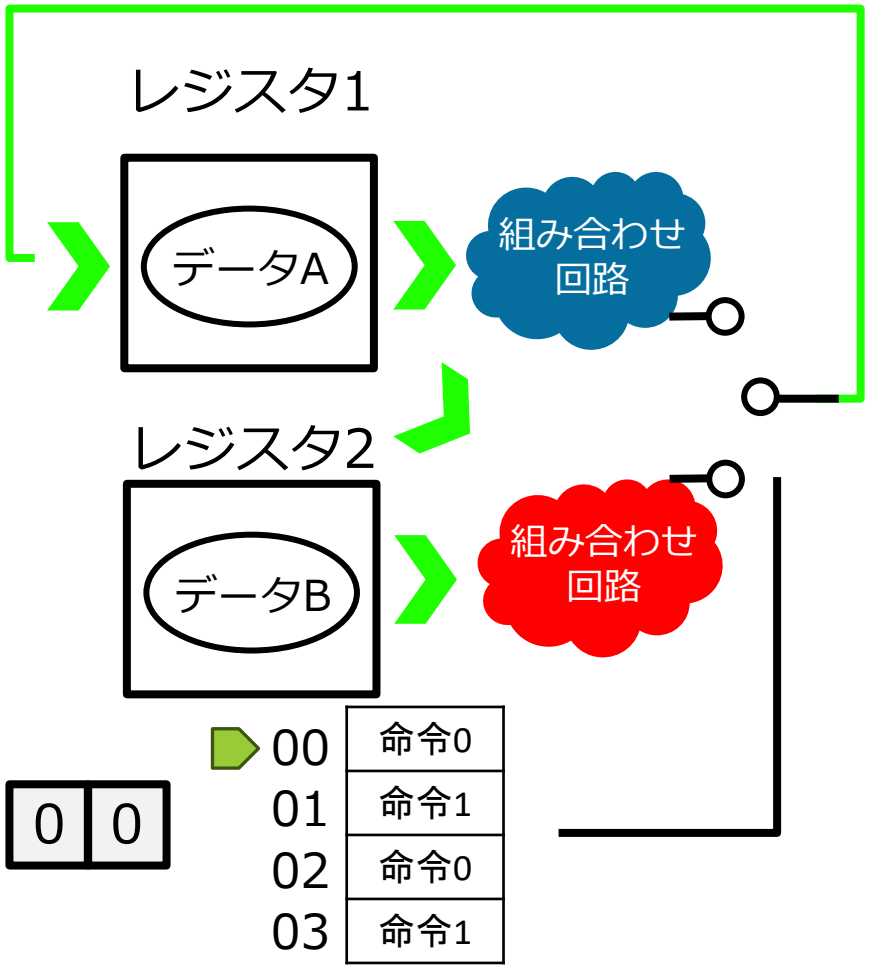
Logisim上で作成

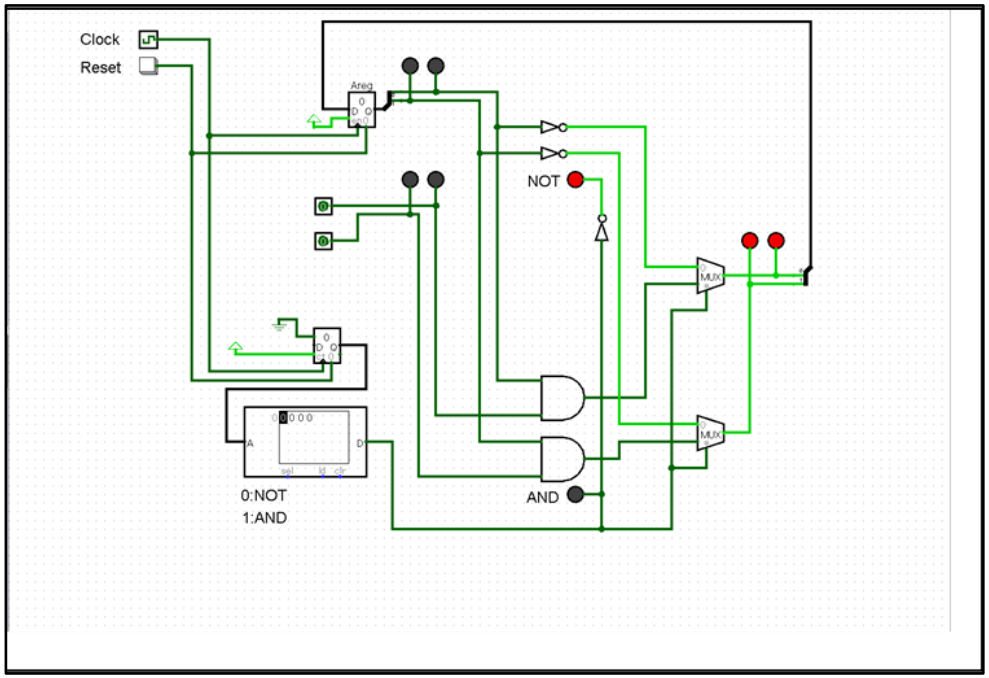




命令デコーダを追加

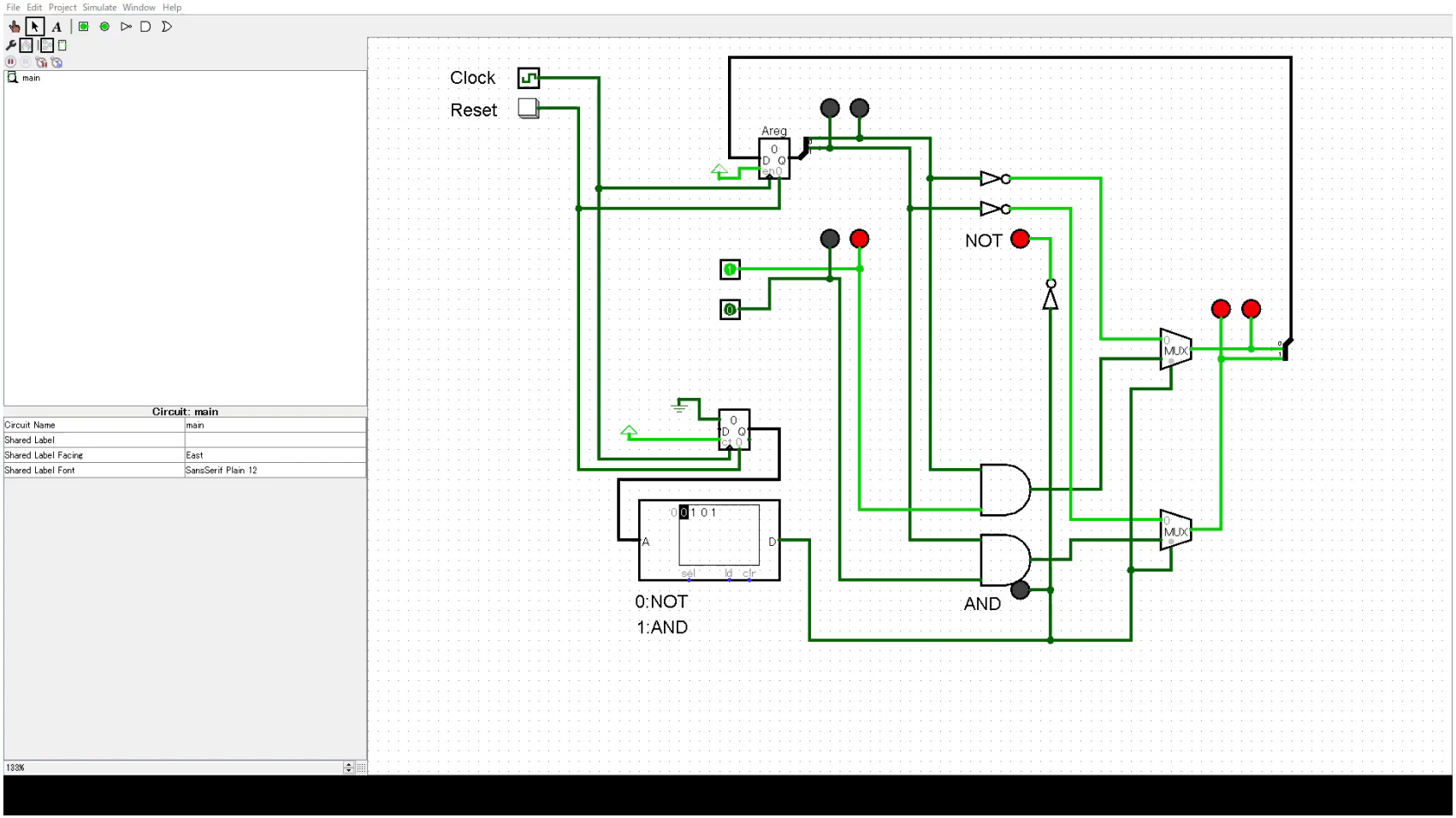
Logisim上で作成





プラグラムカウンタを追加
メモリを追加

Logisim上で作成



完成!!

みやこ電子工房
MIYAKO DENSHI KOBO