

# コンピュータのしくみを知ろう！！



移動ロボット

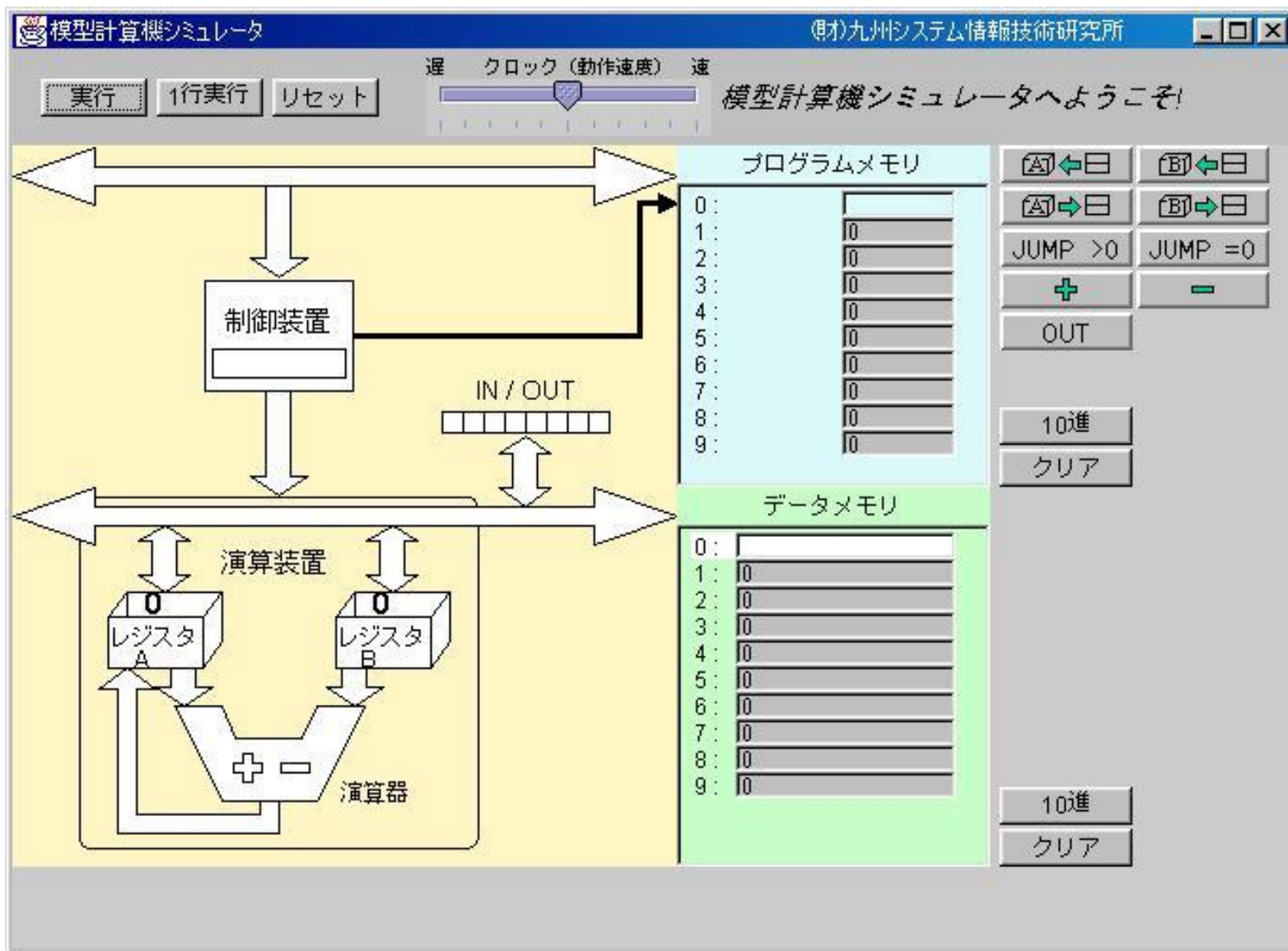


ノートパソコン

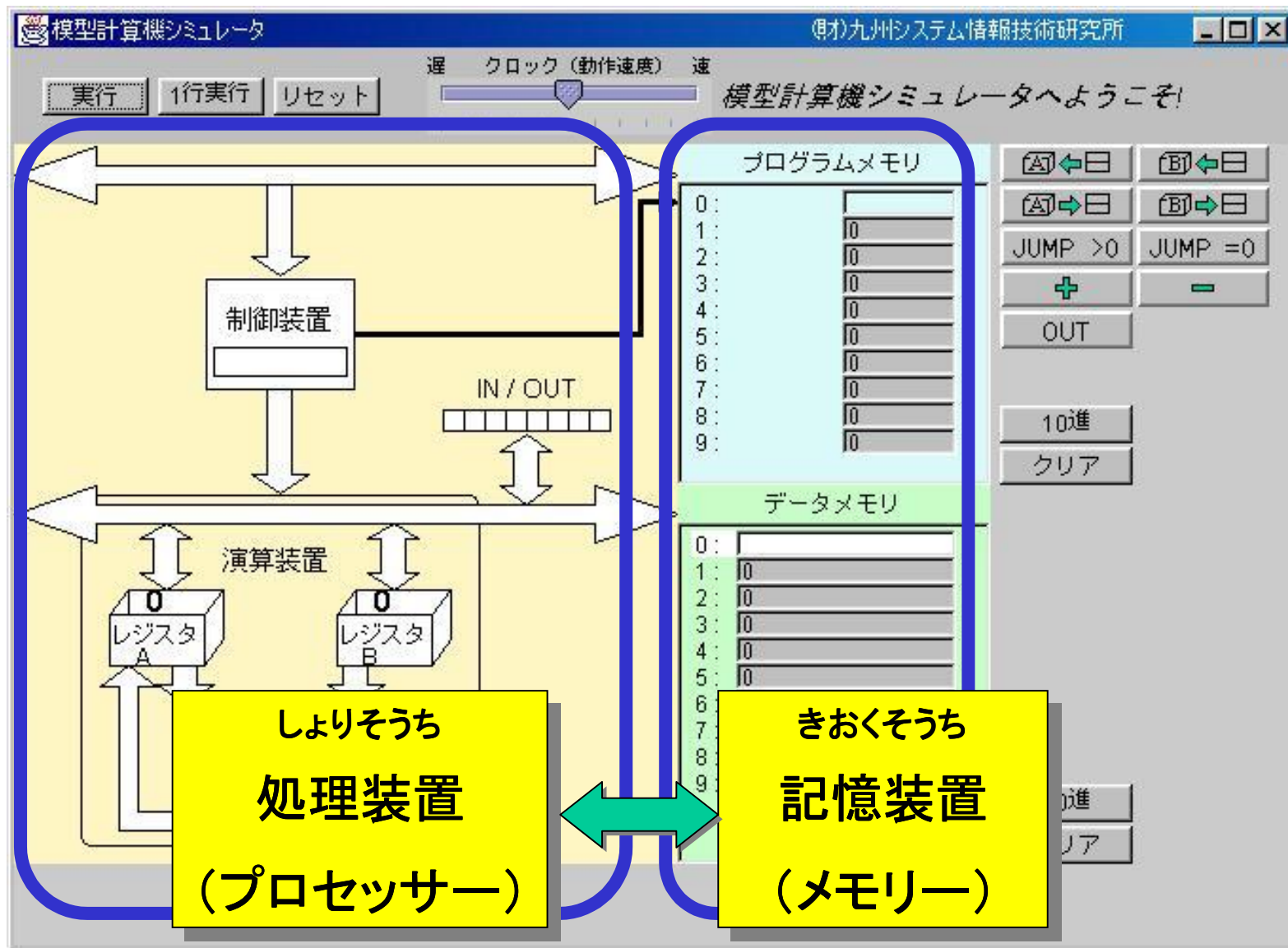


でも、本当に同じコンピュータなのかなあ？

# コンピュータのしくみ シミュレータをうごかしてみる



# コンピュータのしくみ 「五大装置」は知っているかな？



# コンピュータは計算機だ！ たし算をさせてみよう。

このボタンをおすと、うごきはじめるよ。

The image shows a software interface for a computer simulator. At the top, the title bar reads '模型計算機シミュレータ' (Model Computer Simulator) and '(株)九州システム情報技術研究所' (Kyushu System Information Technology Research Institute). Below the title bar are control buttons: '実行' (Execute), '1行実行' (Execute 1 line), and 'リセット' (Reset). A slider for '遅 クロック (動作速度) 速' (Slow Clock (Operation Speed) Fast) is also present.

The main area is divided into two parts. On the left is a schematic diagram of a computer system. It includes a '制御装置' (Control Unit) at the top, an 'IN / OUT' bus in the middle, and an '演算装置' (Arithmetic Unit) at the bottom. The arithmetic unit contains two 'レジスタ A' and 'レジスタ B' (Registers A and B) and an '演算器' (Arithmetic Logic Unit) with '+' and '-' symbols. Arrows indicate the flow of data between these components.

On the right is a control panel with several sections. The 'プログラムメモリ' (Program Memory) section is highlighted with a blue border and contains a list of instructions:

Address	Instruction	Value
0:	[A] ← [ ]	10
1:	[B] ← [ ]	1
2:	+	10
3:	[A] → [ ]	2
4:		1
5:		10
6:		10
7:		10
8:		10
9:		10

The 'データメモリ' (Data Memory) section is highlighted with a green border and contains a list of data values:

0:	7
1:	4
2:	
3:	10
4:	10
5:	10
6:	10
7:	10
8:	10
9:	10

Other controls on the right include buttons for '10進' (Decimal) and 'クリア' (Clear) for both memory sections, and a set of operation buttons: '[A] ← [ ]', '[B] ← [ ]', '[A] → [ ]', '[B] → [ ]', 'JUMP >0', 'JUMP =0', '+', '-', and 'OUT'.

# コンピュータは計算機だ！ ひき算をさせてみよう。

模型計算機シミュレータ (九州システム情報技術研究所)

実行 1行実行 リセット

遅 クロック (動作速度) 速

実行しました。

制御時  
ストア

プログラムをちょっとかえるだけで...

プログラムメモリ

0:	←	←	←	←	←	←	←	←	←
1:	←	←	←	←	←	←	←	←	←
2:	-	←	←	←	←	←	←	←	←
3:	←	←	←	←	←	←	←	←	←
4:	←	←	←	←	←	←	←	←	←
5:	←	←	←	←	←	←	←	←	←
6:	←	←	←	←	←	←	←	←	←
7:	←	←	←	←	←	←	←	←	←
8:	←	←	←	←	←	←	←	←	←
9:	←	←	←	←	←	←	←	←	←

データメモリ

0:	17
1:	4
2:	11
3:	0
4:	0
5:	0
6:	0
7:	0
8:	0
9:	0

レジスタ A: 11

レジスタ B: 4

演算器: + -

OUT

10進

クリア

# コンピュータは計算機だ！ だけど電卓とは、ちょっとちがう？

The image shows a screenshot of a computer simulator window titled "模型計算機シミュレータ". The interface includes a control panel at the top with buttons for "実行" (Execute), "1行実行" (Execute 1 line), and "リセット" (Reset), along with a "遅 クロック (動作速度)" (Slow Clock (Operation Speed)) slider. A callout bubble points to the slider with the text "ちょっとスピードをはやくしてみよう" (Let's try to speed it up a little). Below the control panel is a diagram of the computer's internal components, including a "制御装置" (Control Unit), "演算装置" (Arithmetic Unit) with "レジスタ A" and "レジスタ B" (Registers A and B), and an "演算器" (Arithmetic Logic Unit) with "+" and "-" signs. A callout bubble points to the "実行" button with the text "何回も実行してみよう" (Let's try to execute many times). To the right of the diagram is a "プログラムメモリ" (Program Memory) table and a "データメモリ" (Data Memory) table. The Program Memory table contains instructions: 0: A ← B, 1: B ← A, 2: +, 3: A → B, 4: (blank), 5: (blank), 6: (blank), 7: (blank), 8: (blank), 9: (blank). The Data Memory table contains values: 0: 10, 1: 1, 2: (blank), 3: 10, 4: 10, 5: 10, 6: 10, 7: 10, 8: 10, 9: 10. A callout bubble points to the Data Memory table with the text "前の、たし算のプログラムとどこがちがうか、わかりますか？" (Does anyone know where the previous addition program differs?). On the far right, there are control buttons for "A ← B", "B ← A", "A → B", "B → A", "JUMP >0", "JUMP =0", "+", "-", "OUT", "10進" (Decimal), and "クリア" (Clear).

# コンピュータは計算機だ！ だけど電卓とは、すごくちがう？

The image shows a screenshot of a computer simulator window titled "模型計算機シミュレータ" (Model Computer Simulator) from the "九州システム情報技術研究所" (Kyushu System Information Technology Research Institute). The interface includes a control panel at the top with buttons for "実行" (Execute), "1行実行" (Execute 1 line), and "リセット" (Reset), along with a clock speed slider and a "リセットしました。" (Reset) message. The main area is divided into several sections: a central control unit (制御装置), a program memory (プログラムメモリ) table, a data memory (データメモリ) table, and an arithmetic unit (演算装置) with two registers (レジスタ A and B) and an arithmetic logic unit (演算器). A callout box with a speech bubble points to the program memory table, containing the text "ジャンプ(JUMP)って、どうなるんだろう?" (What happens with a JUMP?). The program memory table shows instructions at addresses 0 through 9, with the instruction at address 4, "JUMP >0", highlighted by a blue box. The data memory table shows values 0 through 9, with the value at address 9 being 10. The arithmetic unit shows two registers, A and B, both containing the value 0, and an arithmetic unit with plus and minus signs.

模型計算機シミュレータ (九州システム情報技術研究所)

遅 クロック (動作速度) 速

実行 1行実行 リセット

リセットしました。

制御装置

プログラムメモリ

0:	[A] ← [B]	10
1:	[A] ← [B]	1
2:	+	10
3:	[A] → [B]	10
4:	JUMP >0	10
5:		10
6:		10
7:		10
8:		10
9:		10

データメモリ

0:	10
1:	1
2:	
3:	10
4:	10
5:	10
6:	10
7:	10
8:	10
9:	10

レジスタ A 0

レジスタ B 0

演算器 + -

ジャンプ(JUMP)って、どうなるんだろう?

JUMP >0 JUMP =0

OUT

10進 クリア

10進 クリア

# コンピュータのしくみ もう君たちは、コンピュータのエキスパート

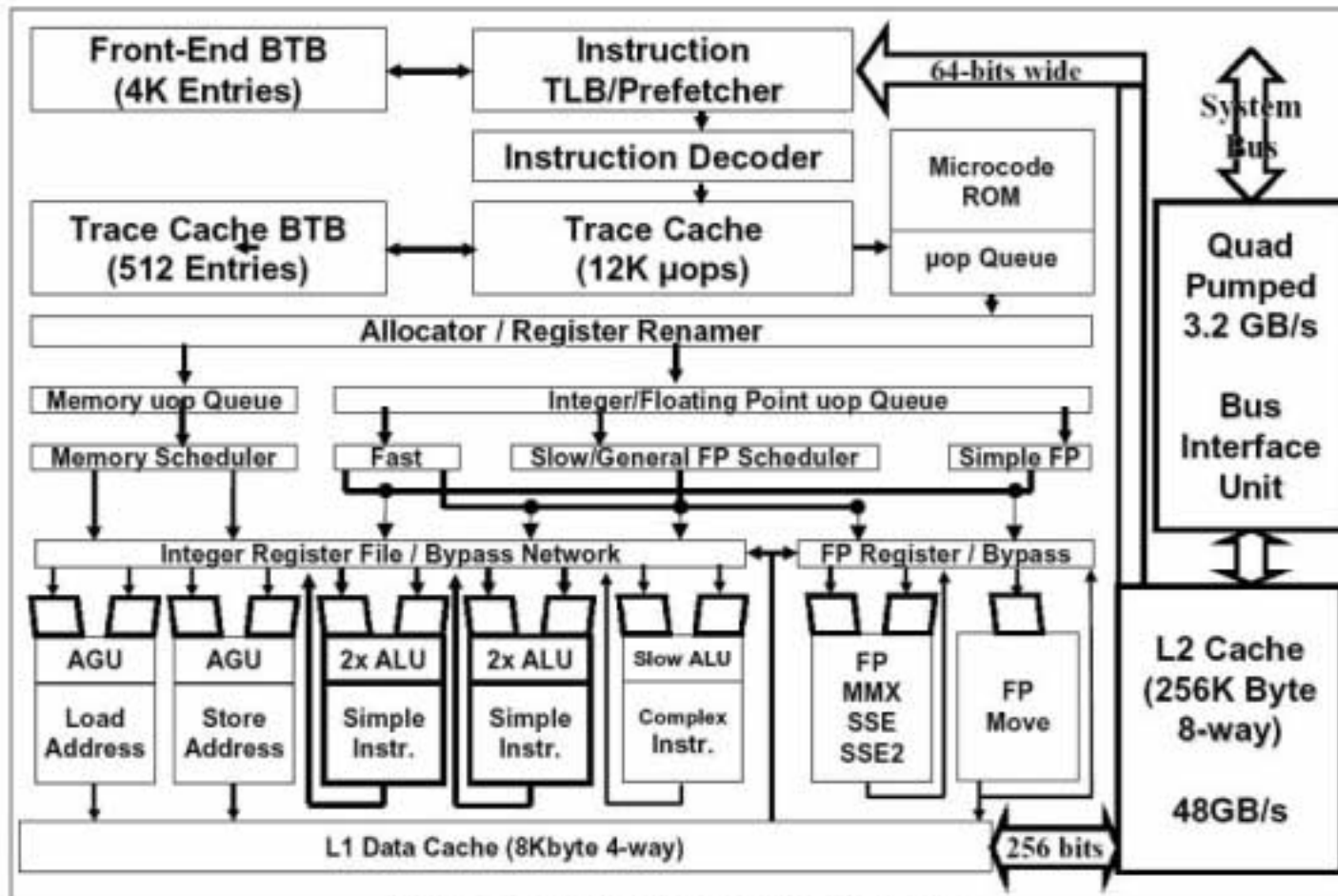


Figure 4: Pentium<sup>®</sup> 4 processor microarchitecture

[http://developer.intel.com/technology/itj/q12001/pdf/art\\_2.pdf](http://developer.intel.com/technology/itj/q12001/pdf/art_2.pdf) より引用

## ペンティアム4プロセッサの構造



# コンピュータのしくみ もう君たちは、コンピュータのエキスパート

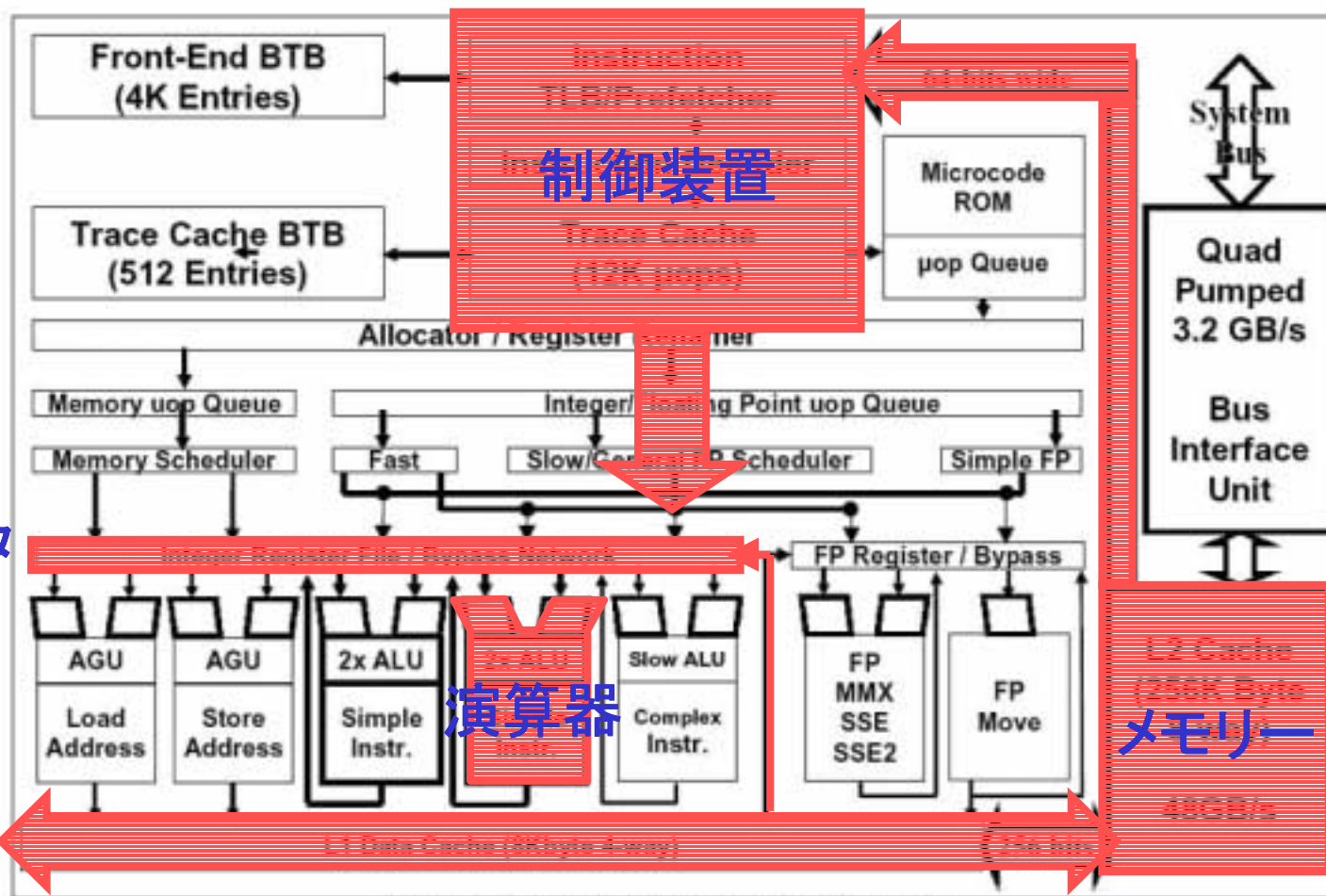


Figure 4: Pentium® 4 processor microarchitecture

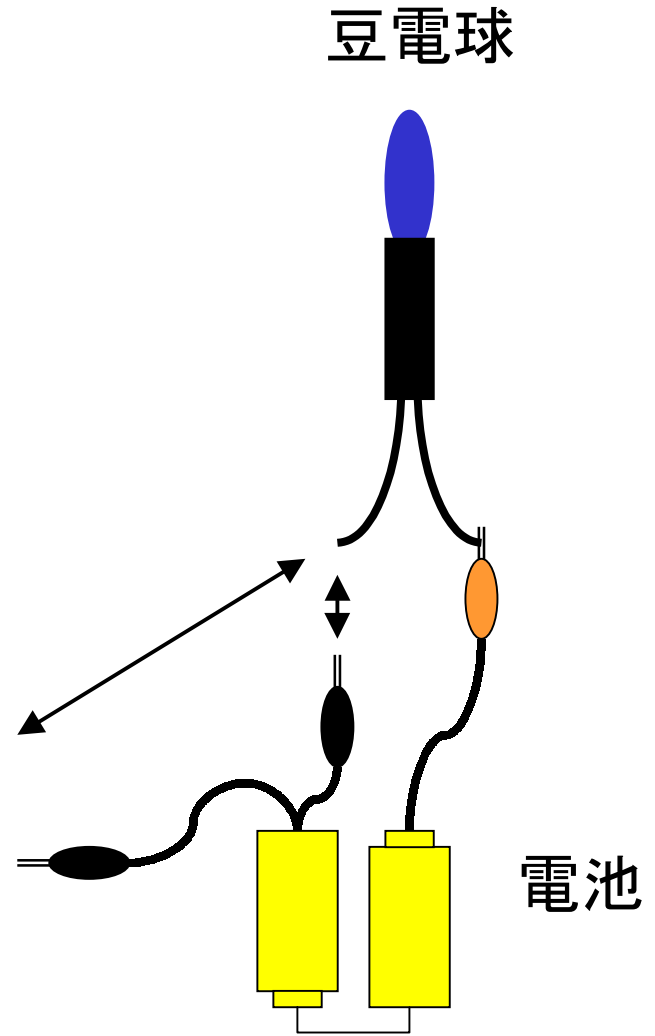
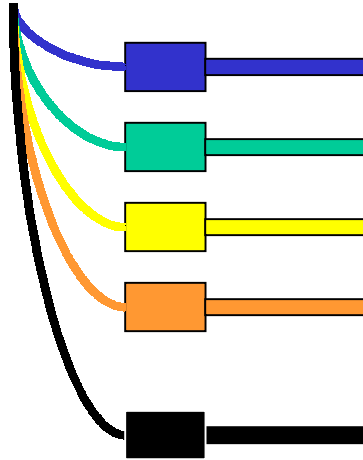
[http://developer.intel.com/technology/itj/q12001/pdf/art\\_2.pdf](http://developer.intel.com/technology/itj/q12001/pdf/art_2.pdf) より引用

## ペンティアム4プロセッサの構造

コンピュータは計算機だ！ でも、計算するだけじゃない！ その1



青(4)  
緑(3)  
黄(2)  
赤(1)  
黒(一)



# コンピュータは計算機だ！ でも、計算するだけじゃない！ その1

The screenshot shows a software interface for a computer simulation. At the top, there are buttons for '実行' (Execute), '1行実行' (Execute 1 line), and 'リセット' (Reset), along with a '遅' (Slow) / '速' (Fast) speed slider. The main area is divided into several sections:

- Control Unit (制御装置):** A central box that manages the flow of data and instructions.
- Program Memory (プログラムメモリ):** A list of memory addresses (0-9) containing instructions. Address 0 contains 'OUT', and others contain '0'. To its right are control buttons: 'JUMP >0', 'JUMP =0', '+', '-', 'OUT', '10進' (Decimal), and 'クリア' (Clear).
- Arithmetic Unit (演算装置):** Contains two registers, 'レジスタ A' and 'レジスタ B', both showing '0'. Below them is an '演算器' (Arithmetic Logic Unit) with '+' and '-' symbols. It is connected to an 'IN / OUT' bus.
- Data Memory (データメモリ):** A list of memory addresses (0-9) containing data. Address 0 contains '11', and others contain '0'. To its right are buttons: '10進' (Decimal) and 'クリア' (Clear).

Two callout boxes provide commentary:

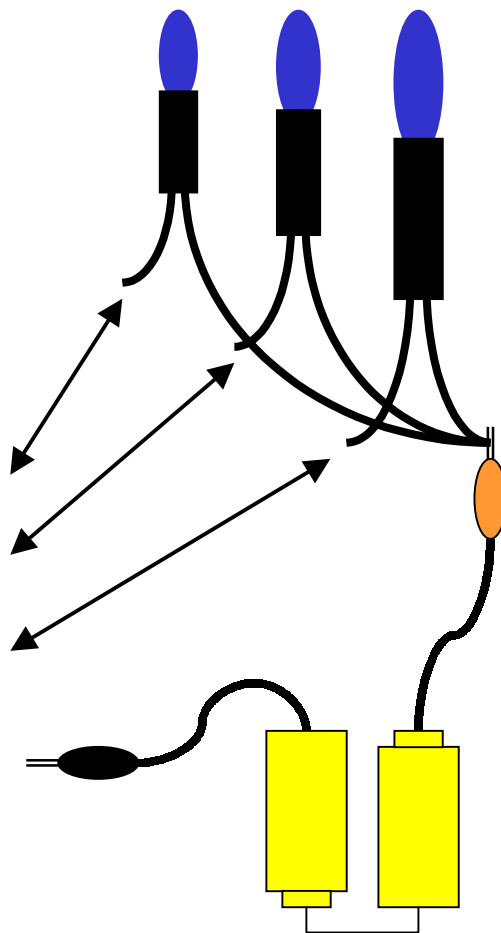
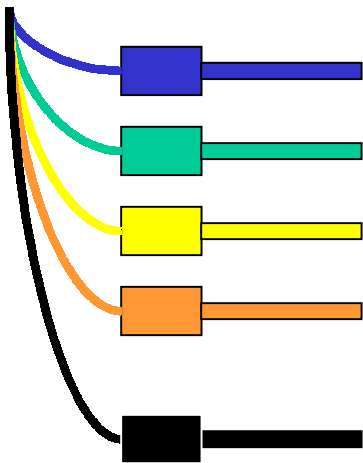
- Top right: 'アウト(OUT)って、どうなるんだろう？' (What happens with OUT?)
- Bottom right: '0や1, それ以外の数字を入れてみると...' (0 or 1, try entering other numbers...)

# コンピュータは計算機だ！ でも、計算するだけじゃない！ その2

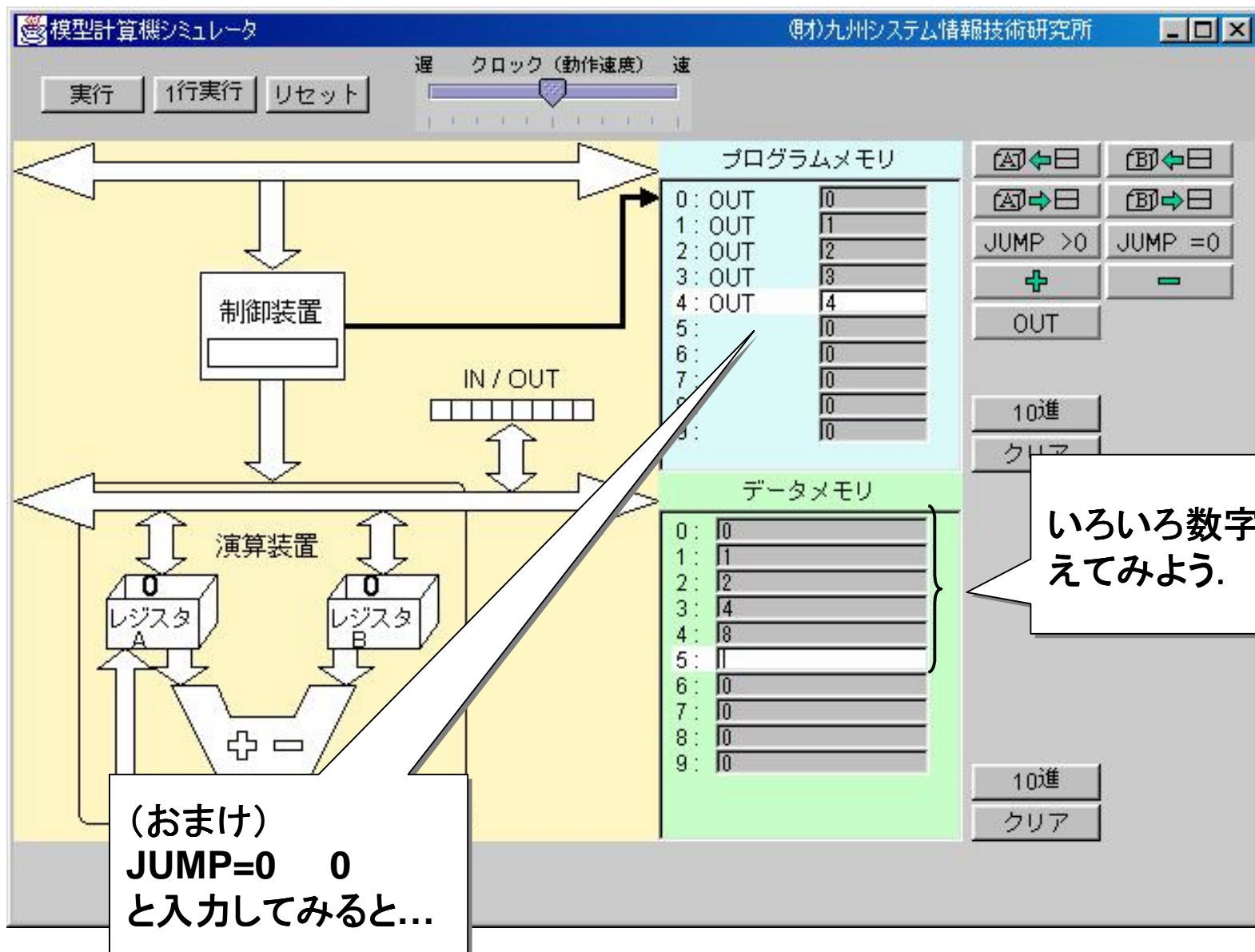


豆電球, 豆電球, 豆電球...

- 青 (4)
- 緑 (3)
- 黄 (2)
- 赤 (1)
- 黒 (一)



# コンピュータはコントローラ. 自動制御ってこんなこと.



コンピュータはコントローラ. ロボットをうごかしているのも...



ラジコンカー



送信機

# コンピュータはコントローラ. ロボットをうごかしているのも...



九州システム情報技術研究所

遅 クロック (動作速度) 速

リセット

プログラムメモリ

0:	OUT	10
1:	OUT	11
2:	OUT	12
3:	OUT	13
4:	OUT	14
5:		10
6:		10
7:		10
8:		10
9:		10

データメモリ

0:	10
1:	11
2:	12
3:	14
4:	18
5:	11
6:	10
7:	10
8:	10
9:	10

コントローラボタン: A ←, B ←, A →, B →, JUMP >0, JUMP =0, +, -, OUT, 10進, クリア

ロボット車

ロボット車本体

~~考えてみよう~~

右ハンドル	00000001	1
左ハンドル	00000010	2
前進命令	00000100	4
バック命令	00001000	8
右回転命令	0000	
左回転命令	0000	

Uターンさせるプログラムは?  
8の字に走らせるプログラムは?  
おもしろい動きを考えてみよう!

コンピュータのしくみ 時間があったら、ためしてみよう.



これまで改造してきたラジコンカー

ラジコンの改造方法は、ココ↓のウェブページにあります.

[http://www.isit.or.jp/~kimuro/rcupj/rcupj\\_hw2.htm](http://www.isit.or.jp/~kimuro/rcupj/rcupj_hw2.htm)