

教科書p.48~

コンピュータ学習をはじめる前に、次のことに注意してください。

勝手に設定を変えないでください。

あくまでも学習ですので、ゲームソフトなどは禁止します。

利用するノートパソコンは壊れやすいので注意してください。特に液晶を鉛筆の先などで突っついたりしてはいけません。故意に破損した場合は相当の責任を負ってもらいます。

ノートパソコンは班に1台です。決められた番号のパソコンを使ってください。

1. マイコンの構成

コンピュータを構成する種々の機器の総称..... (ハードウェア )

コンピュータを利用する技術・データの総称..... (ソフトウェア )

(1) ソフトウェアの種類

アプリケーションプログラム.....目的の仕事ですぐに実行できるようにつくられたプログラム。ワープロ・ゲームなど。 末端利用者はこれだけで充分パソコンを活用できる。

サービスプログラム.....ユーティリティなどパソコンを運用するのに便利なプログラム。ファイル変換、ディスク管理など。

(言語処理 ) プログラム.....一定の書式で書かれたプログラムから実行可能なファイルをつくるプログラム。ユーザーが自作プログラムをつくるために必要。 BASIC, FORTRAN, COBOL, C など。

(基本プログラム ) .....キー入力やディスプレイへの表示といった基本的なコンピュータ操作をするソフトウェア

Operating System (OS) ともいう。

OSがないと、コンピュータはただの箱。

(2) パソコンですぐに目につくハードウェア

パソコン ( 本体 ) ..... CPU とメモリを合わせていう。関連する電源部などは省略。

その他の機器: キーボード, CRTディスプレイ, プリンタ, マウス

フロッピーディスク・ドライブ装置, CD-ROMドライブ装置

( 本体に内蔵されていても, 分類上は本体ではない)

MO ( 光磁気ディスク) ドライブ装置

2. ハードウェア

(1) (中央処理装置 ) Central Processing Unit , 略して ( CPU )

コンピュータの中心部で, 加減算などの ( 計算 ) とコンピュータ全体の ( 制御 ) を司る。

マイクロコンピュータ..... CPU が ( マイクロプロセッサ ) と呼ぶLSI からできているコンピュータ。略してマイコン。

マイコンが個人で購入できることから 個人用 ( パーソナル ) コンピュータ略してパソコン

最近では, 掃除機, 洗濯機, 電気炊飯器などにもマイクロプロセッサが使われている。

マイクロプロセッサのしくみ

例 ( 4ビット)

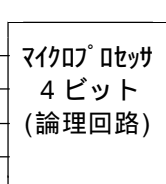
入力例

0

1

1

0



出力例: 入力に対して

1

1

0

0

決まる。

CPUが一度に扱える並列データ ( 2進数) の桁数 ( ビット数)

= CPU内部の主なレジスタ ( メモリのようなもの) のビット数

= コンピュータのビット数

(2) 主記憶装置 ( メモリ )

CPUの処理手順 ( プログラム ) やその対象となるデータを記憶するもの。

CPUはメモリの番地 ( アドレス ) を指定することで, メモリにアクセスする。

メモリの1つの番地は 8ビット = 1 ( バイト ) の容量をもつ。

<メモリチップの種類>

読み出し専用メモリ Read Only Memory, 略して ( ROM )

随時読み書きメモリ Random Access Memory, 略して ( RAM )

ROMは電気を切っても覚えています。RAMは電気を切るとすぐ忘れる。

<記憶容量の単位>

2進数1桁 = 1 ( ビット ) , 8ビット = 1 バイト,

1キロ

2<sup>10</sup> バイト = 1 K バイト,

( 1024 )

1メガ

2<sup>20</sup> バイト = 1 M バイト

( 1048576 )

注意Kは大文字

問. 1 Mバイトは, 何Kバイトか。

(3) I/O機器..... CPU + メモリ = 本体 以外の全ての機器。

I/O機器.....入力input装置と出力output装置。外部機器とか ( 周辺 ) 装置ともいう。

例. 入力装置: キーボード, マウス

出力装置: CRTディスプレイ, プリンタ

入出力装置: フロッピーディスク&ドライブ装置, モデム

( I/Oインターフェイス ) ..... I/O機器とCPU・メモリとがデータのやり取りをするとき, 間に入って信号の形式を変換したり, データ読み書きのタイミングをはかったりする。専用のLSIが担当する。

例. RS232Cインターフェイス..... 8251 ( モデム用)

(4) クロック Clock

CPUの動作の同期信号源。その周波数を ( クロック) 周波数いう。

最近のパソコンに使われているCPUのクロック周波数は 1000MHz を超えている。

(5) フロッピーディスク

補助記憶装置の代表的なもの。

5インチ角の ( ミニ ) サイズ, 3.5インチ角の ( マイクロ) サイズ

ディスク本体は, 厚さ0.08mm程度のポリエステルフィルムに ( 磁性 ) 粉末を塗布。

フロッピーディスクにデータを書き込むには ( フォーマット )

と呼ぶOS指定の様式を設定する必要がある。フォーマットとは同心円状の ( トラック ) と扇形の小さな区画 ( セクター ) を, 一定の様式に設定すること。

<性能の良いパソコン・システムの条件とは>

- (1) 処理速度が速い..... CPUの(ビット)数が多いほど、一度に多量のデータを処理できる。  
(クロック)周波数が高いほどすなわちCPUの同期信号が速いほど、  
処理が速い。  
I/Oインターフェース用LSIおよびI/O機器の処理速度が速い。
- (2) 処理できる量が..... (メモリー)の記憶容量が大きい。  
多い (ハードディスク)など補助記憶装置の容量が大きい。  
また、そのドライブ装置がたくさんある。  
(ハード)ディスクでは、500Gバイトの記憶容量のものが市販され  
ている。キガ(10<sup>9</sup>)

3. 文字コード

- (1) 1バイト系文字(アルファベット数字カナ).....(8)ビットコード  
マイコンでよく用いられているコード: アスキー  
情報交換用アメリカ標準コード 略して(ASCII)  
American Standard Code for Information Interchange  
日本版 JIS C6220  
キャラクタコードともいう。

(2) 2バイト系文字

漢字・ひらがな・ギリシャ文字・ロシア文字・記号など。  
1バイト(8ビット)のコードでは充足することができないため、2バイトのコードが考えられた。  
JIS C6226 ..... 通称(JIS漢字)コード  
使用頻度の高いもの(第1)水準  
高くない(第2)水準.....難解な漢字に多い。

1バイト=8ビットの数.....2進数8桁, 16進数2桁  
2バイト=16ビット ..... 16 4

(参) 16進数 0~9 A(10) B(11) C(12) D(13) E(14) F(15) の16個の数字で表す。

$$3EA_{16}H = 3 \times 16^3 + 14 \times 16^2 + 10 \times 16 + 11$$

例. P.54の表を参考にして、次の文字のキャラクタコードまたはJISコードを16進数と2進数で  
記せ。

答. d 胃  
64<sub>H</sub> = 0110 0111<sub>(2)</sub> 305F<sub>H</sub> = 0011 0000 0101 1111<sub>(2)</sub>

練. P.54の表を参考にして、次の文字のキャラクタコードまたはJISコードを16進数と2進数で  
記せ。

ハ 雨

コンピュータの歴史

1. 次の視点でビデオを見てください。「ビデオファイル "01エニアック誕生.mpg" 3分55秒」

- (1) 最初のコンピュータが開発された  
何時 第2次世界大戦中  
何のために 弾道計算
- (2) コンピュータが開発される前は  
何で計算したか。 微分解析器  
一つの計算にどのぐらいの時間がかかったか 一月以上
- (3) 開発された電気式計算機は  
何個の真空管を使ったか 1,8000本  
スイッチは何本を使ったか 6000個  
重量は 30トン  
世界最初の実用的なコンピュータの名前は エニアック ENIAC  
1秒間に何回の計算ができるか 1200回  
何人分の計算をこなすと言われたか 20000人

2. 次の視点でビデオを見てください。「ビデオファイル"02エドサックノイソ方式電子計算機.mpg" 3分50秒」

- (1) エニアックはプログラムをどのような方法で行っていたか。 手作業で線をつなぎ替える
- (2) プログラムを記憶できる最初のコンピュータについて  
何という名前のコンピュータか ID'サック EDSAC  
何にプログラムを記憶したか 水銀遅延線  
このコンピュータがはじめて動作したのは西暦何年か 1949年5月  
はじめにどのような計算をしたか 整数を自乗する

3. 次の視点でビデオを見てください。「ビデオファイル"10パソコンとBASIC.mpg"7分39秒」

- (1) intelが開発したマイクロプロセッサについて  
コンピュータの主要部分が1つのICになった。値段はいくらぐらいであったか 10万円  
世界初のパソコンの名前は アルテア  
何でプログラムを入力したか スイッチ  
計算結果は何に表示されたか ランプ(LED)  
ポール・アレンが紙テープで持ち込んだプログラムはなにか。 BASIC言語  
ポール・アレンと共にこのプログラムを開発した人は誰か ビル・ゲイツ  
持ち込んだプログラムで作成したプログラムは何か 月着陸船ゲーム  
個人でコンピュータを持った人たちがクラブを作り始めた何の目的があったか ソフトの入手  
日本ではじめて創刊されたパソコン雑誌は何か I/O  
日本ではじめて発売されたマイコンは何か TK80  
マイクロソフトが最初に発売したソフトは何か BASIC言語

4. 次の視点でビデオを見てください。「ビデオファイル"11OS\_Windows\_Linux.mpg"6分31秒」

- OS誕生以前のコンピュータはどのようにしてコンピュータに入れていたか。 人間がスイッチをON/OFFして入れた。  
最初の本格的なOSを開発した会社は何か IBM  
1981年IBM製パソコンのOSを開発した人は誰か ビル・ゲイツ  
Windowsを開発した会社はどこか マイクロソフト  
インターネットに公開した趣味で作ったOSの名称は何か リナックス (Linux)  
ハリウッドでリナックスを使ったパソコンで画像処理した映画は何か タイタニック

取り扱い上の注意

1. 壊れやすいので衝撃を与えたりしない。
2. 特に液晶画面は弱いので、鉛筆で突っついたりしない。
3. フタなどは無理にあけたりしない。(先生の指示に従う)



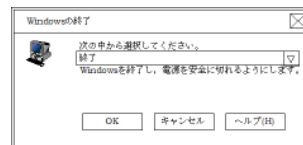
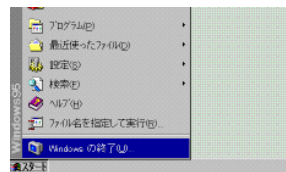
準備

1. 電源コネクタにA Cアダプターを接続する。
2. USBコネクタにマウスを接続する。
3. RS-232Cコネクタにケーブルを接続する。

起動の仕方

1. 電源スイッチを押す
2. しばらくするとWindows Me が起動する。

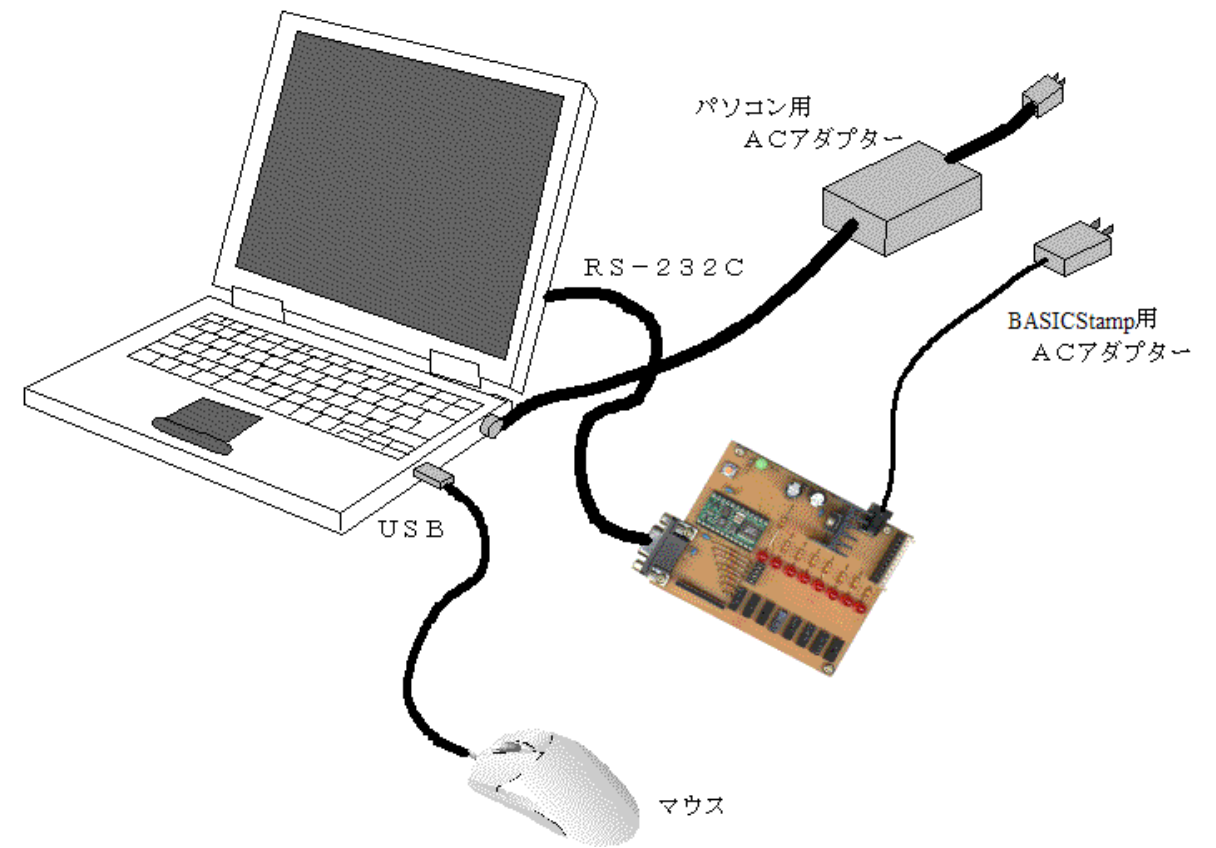
終了の仕方



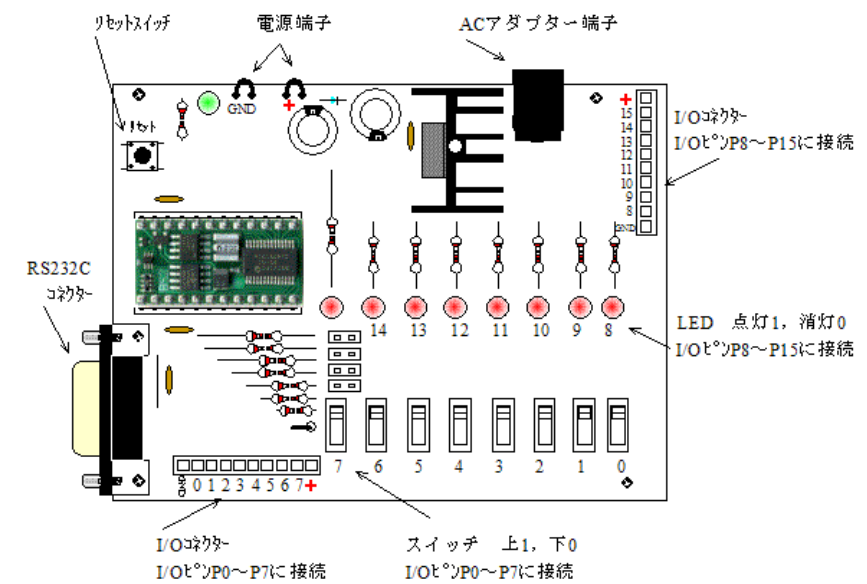
その他

1. 電源スイッチを押しても起動しないときは、主電源SWがONになっていないからである。

ノートパソコンとBASICStamp接続図




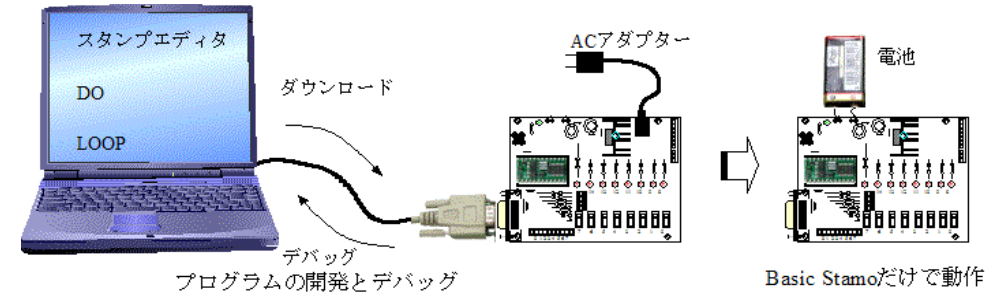
BasicStamp学習基板の各部の名称



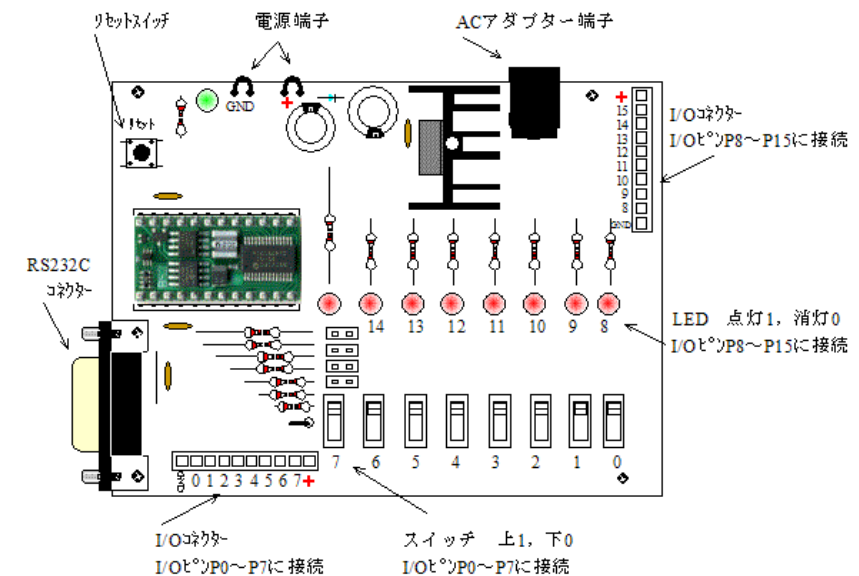
**(4)BasicStamp学習基板の使い方**

ノートパソコンと BasicStamp 学習基板をケーブル (RS232C ケーブル) で接続します。ノートパソコンではスタンプエディタというソフトを起動します。スタンプエディタで PBASIC の文法に従ってプログラムを作成します。

完成したプログラムは、スタンプエディタにある  ボタンを押すと、文法上の間違いがなければ、ケーブルを通じて BasicStamp 学習基板に転送 (ダウンロード) されて、実行されます。プログラムが意図した動作をしない場合は、プログラムを修正 (デバッグ) して再度転送します。



転送したプログラムは、電源を切っても BasicStamp のメモリーの中に残っているので、電源をつなぐだけで動作し始めます。

**(5) BasicStamp学習基板の各部の名称**

BasicStamp には 16 本の信号線が外部に出ています。それらは P0 ~ P15 の名前がつけられています。P0 ~ P7 の信号線にはスイッチが接続されています。スイッチが上で 1, 下で 0 が入ります。間違ってスイッチがつながっている信号線に出力しても壊れることはありません。P8 ~ P15 の信号線には LED が接続されています。出力 1 で点灯, 出力 0 で消灯します。電源は電源端子または AC アダプター端子のどちらかにつながります。両方につないでも壊れません。電圧の高い電源から電流は供給されます。

**(1)BASICによるプログラミング**

コンピュータを動作させるためには、コンピュータに動作の指示をしなければなりません。この指示を命令といいます。コンピュータが命令として理解できる言葉は、2 進数です。しかし 2 進数で命令を記述するのは大変煩雑です。そこで人間の言葉に近い表現で命令を記述して、コンピュータの力を借りて 2 進数に変換する手続きをとります。

人間の言葉に近い命令には、何種類かの言葉があり、コンピュータにさせる仕事に最適な言葉を選択して利用します。この色々ある言葉をコンピュータ言語といいます。

コンピュータ言語には、BASIC, FOTRAN, Pascal, C, COBOL などがあります。この中で BASIC 言語をこれから利用します。

**(2)BASIC言語**

Beginners' All-purpose Symbolic Instruction Code の略で、1964 年、ダートマス大学の John G. Kemeny, Thomas E. Kurtz によって開発されました。直訳すると「初心者向けで汎用の記号的命令コード」で文法が平易で扱いやすく、以前パーソナルコンピュータで広く採用されたことがあり、BASIC からプログラミングを覚えたプログラマもたくさんいます。

BASIC にはパソコンにあわせて機能を拡張した独自の BASIC が多数存在します。この結果 BASIC には多くの方言が生まれました。現在パソコンソフトで圧倒的なシェアを持つマイクロソフトも最初は BASIC 言語を開発販売することからスタートしています。現在 VisualBASIC を販売していますが、エクセルやワードの中にマクロ命令として BASIC 言語の方言が使われています。

**(3)PBASIC**

BasicStamp という切手大の小型コンピュータ用に作られた PBASIC という BASIC の言語があります。BasicStamp とは制御用のコンピュータで、マイクロコントローラとも呼ばれています。PBASIC は制御に適した機能の拡張が行われています。

PBASIC (BasicStamp) で簡単にできること

- ・ LED の点滅
- ・ スイッチが ON か OFF かの読みとり
- ・ モーターの回転数の制御
- ・ 正弦波の発生
- ・ DTMF の発生 (プッシュホンのボタンを押したときに発する音)
- ・ パルス信号の発生
- ・ パルスの幅の測定
- ・ 周波数の測定

PBASIC (BasicStamp) が苦手とすること

- ・ 算術計算
- ・ 大量のデータ処理

理由)  $1 \div 3$  は 0.333... ですが、PBASIC は整数しか扱えません。PBASIC では  $1 \div 3$  は 0 になります。変数の数も 26 バイトしかないので大量のデータは扱えません。



**(1)BASICによるプログラミング**

問題1 コンピュータが命令として理解できる言葉は何か

問題2 人間の言葉に近い表現でコンピュータの命令を記述して、コンピュータが理解できる2進数に変換するソフトを言語処理プログラムという。人間の言葉に近い表現には色々な表現がありコンピュータ言語という。どのようなコンピュータ言語があるか5つ書きなさい。

**(2)BASIC言語**

問題1 BASIC とは何の略か。

英語で答えなさい。

を日本語に訳すとどんな意味になるか。

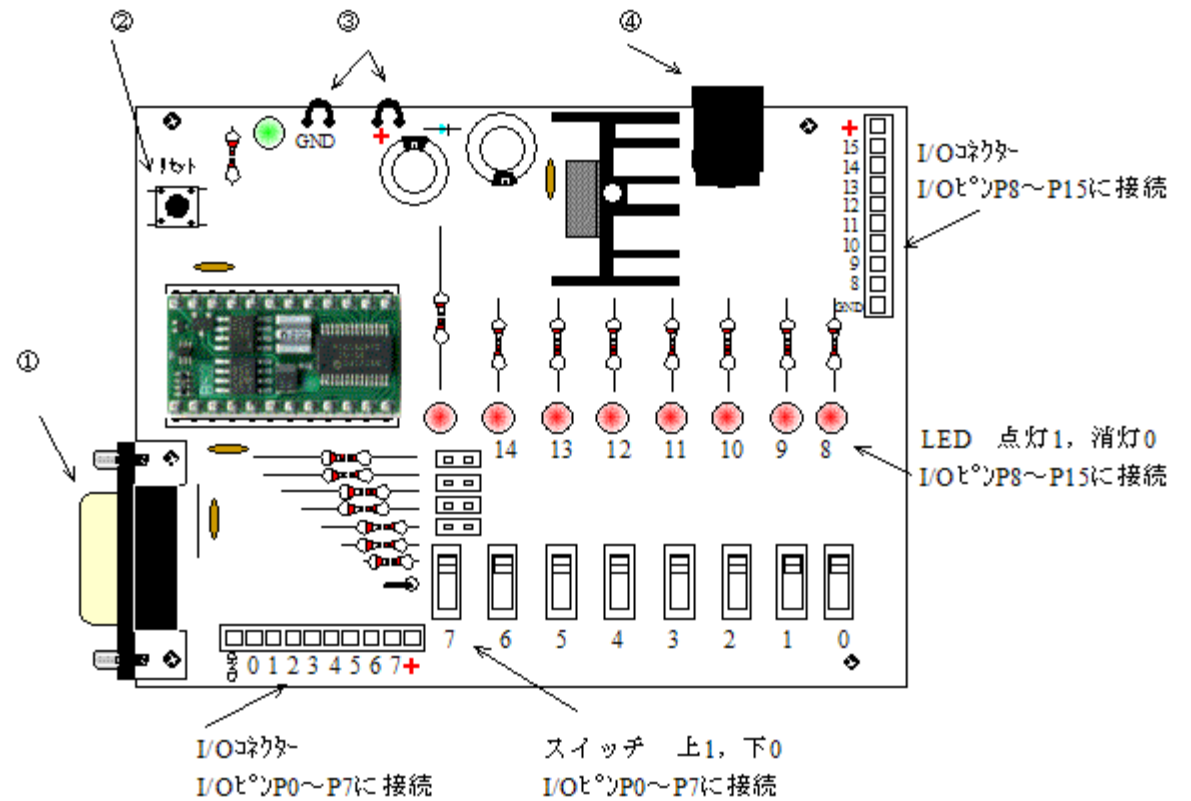
**(3)PBASIC**

BasicStamp という切手大の小型コンピュータ用に作られた PBASIC という BASIC の言語があります。BasicStamp とは制御用のコンピュータで、マイクロコントローラとも呼ばれています。PBASIC は制御に適した機能の拡張が行われています。

問題1 PBASIC は制御に適した機能の拡張が行われています。どのようなことが簡単にできるか箇条書きで3つ説明しなさい。

問題2 PBASIC が苦手とすることはどんなことですか。箇条書きで2つ書きなさい。

**(4) BasicStamp学習基板の各部の名称**



問題1 上の図の ① ~ ④ の名称は何か。

問題2 ①の先には何をつなぎ、何をするのが説明しなさい。

問題3 ②に何を接続するか。またこの時の動作を説明しなさい。

## PBASIC の使い方 NO.6

### (1)PBASICの文法

PBASIC は BasicStamp というコンピュータ用に作られた BASIC 言語です。BasicStamp は計算が苦手で、機械の制御を得意とするので、PBASIC も機械の制御に重点をおいた命令がたくさんあります。

機械の制御とは、例えばコンピュータによってスイッチを入れたり切ったりすることだと考えてもそれほど間違いはありません。PBASIC は 16 本ある P0 ~ P15 の入出力線に対して信号を出したり、信号の周期を測定したりする色々な命令があります。

プログラムを作成する上で押さえておきたい事実を 2 つまず示しておきます。

命令は、上から順に下に向かって実行されます。

PBASIC の命令の多くは P0 ~ P15 までの信号線に対して色々な操作を行います。

### (2)PBASICの命令 その1

P0 ~ P15 の信号線の電圧を 5 V , 0 V に変化させる命令 HIGH , LOW

#### HIGH

機能	1つのI/Oピンを出力に設定し、1を出力する
書式	HIGH Pin Pin は 0 ~ 15 の値を指定する。0 ~ 15 は定数/変数/式で指定できる。
説明	HIGH 7 Pin 7 の状態を 1(HIGH, 5V)にする。

#### LOW

機能	1つのI/Oピンを出力に設定し、0を出力する
書式	LOW Pin Pin は 0 ~ 15 の値を指定する。0 ~ 15 は定数/変数/式で指定できる。
説明	LOW 7 Pin 7 の状態を 0(LOW, 0V)にする。

プログラムの実行を指定した時間だけ停止させる命令 PAUSE

#### PAUSE

機能	指定された期間プログラムを(何もしないで)一時停止(ポーズ)する。
書式	PAUSE Period Period の値の範囲は 0 ~ 65535 単位は m 秒
説明	最大 65535m 秒(65.535 秒)一時停止する

プログラムの実行順を変える命令。

#### GOTO

機能	プログラムの新しいポイントに行く。ポイントにはラベルが書かれている。
書式	GOTO ラベル
説明	GOTO TOP TOP : と書かれたラベルに実行を移す。

### 確認事項

BasicStamp 学習基板では I/O ピン P8 ~ P15 が LED に接続されています。

I/O を HIGH ( 5V ) にすると LED は点灯します。

プログラミングの手順



- BasicStamp 学習ボードとノートパソコンを RS232C ケーブルで接続します。
- BasicStamp 学習ボードに AC アダプターを接続する 緑の LED が点灯します。
- このアイコンをクリックして StampEditor を起動させます。

### (3)プログラミングの手順

このアイコンをクリックして、BasicStamp 学習基板がどの RS232C ポートにつながれているかを調べます。

BASICStamp2 が表示されているポートを確認します。

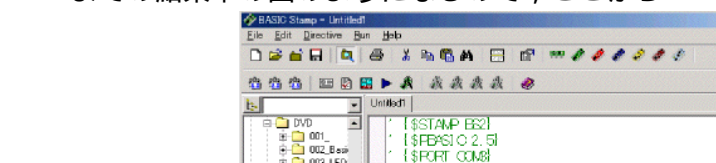
この図では COM3:になっています。

Directive Stamp BS2 を選びます。

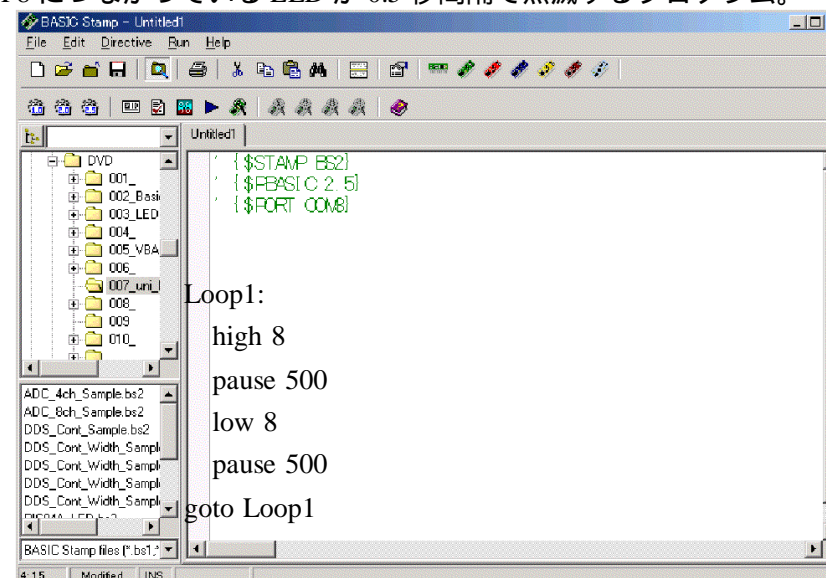
Directive PBASIC Version2.5 を選びます。

Directive Port COM3 を選びます。なお、この COM3 は で接続が確認できたポートです。

までの結果下の図のようになるので、ここから PBASIC の命令を記述します。



例題 P8 につながっている LED が 0.5 秒間隔で点滅するプログラム。



上の図のようにプログラムを入力して、 をクリックすると動作します。

- (1) プログラムを作成する上で大切なポイント。  
 命令は、上から順に下に向かって実行されます。  
 BasicStamp 学習基板では I/O ピン P8 ~ P15 が LED に接続されています。  
 I/O を HIGH (5V) にすると LED は点灯します。

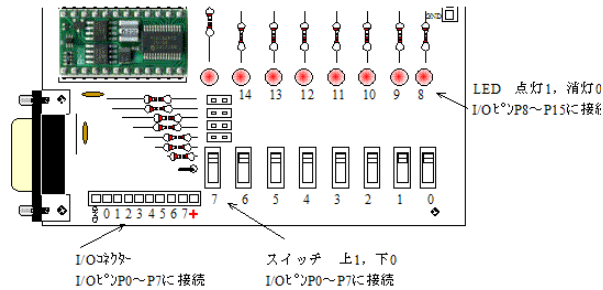
- (2) プログラミングの手順  
 BasicStamp 学習ボードとノートパソコンを RS232C ケーブルで接続します。  
 BasicStamp 学習ボードに AC アダプターを接続する 緑の LED が点灯します。  
 アイコンをクリックして StampEditor を起動させます。



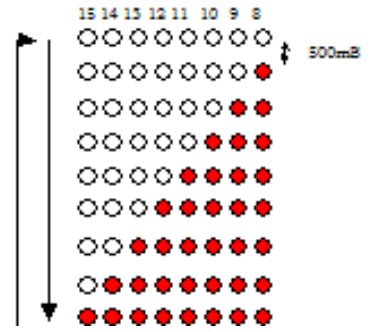
問題 1 次の 4 つの命令の意味を書きなさい。

- HIGH
- LOW
- PASUSE
- GOTO

問題 2 I/O ピン 8 と 9 に接続されている LED を 0.5 秒 (500ms) に点滅するプログラムを作りなさい。



問題 3 右の図のように LED が 500 m S 毎に 1 こずつ点灯する数を増やして、全て点灯した次に全てを消灯させ、以後この動作を繰り返すプログラムを書きなさい



```
' {$STAMP BS2}
' {$PBASIC 2.5}
' {$PORT COM1}
```

```
' {$STAMP BS2}
' {$PBASIC 2.5}
' {$PORT COM1}
```

PBASIC の命令の要約

BPASIC の全ての命令の要約です。BASICStampIC にはいくつかのタイプがあります。本校で利用する IC は BS2 です。各命令の が該当しています。

全ての命令を覚える必要はありません。

NO	グループ	命令	説明	BASICスタンプの種類			
				BS2	BS2e	BS2sx	BS2p
1	分岐	IF THEN	比較と条件分岐				
2		IF THEN ELSE	比較と条件分岐				
3		SELECT CASE	条件分岐				
4		BRANCH	アドレス・リストの一つに分岐する。				
5		ON GOSUB	アドレス・リストの一つを実行する。				
6		ON GOTO	アドレス・リストの一つに分岐する。				
7		GOTO	アドレスへジャンプする。				
8	GOSUB		サブルーチンを呼ぶ。 ネスティングの深さは4レベルまで。 一つのプログラムの中でGOSUBは255個まで使える。				
9		RETURN	サブルーチンから復帰する。				
10		RUN	実行を他のプログラムに切り換える。	x			
11	POLLRUN			x	x	x	
12	ループ	FOR NEXT	FOR - NEXTループを開設する。				
13		DO LOOP	ループ				
14		EXIT	ループから脱出する。				
15	EEPROM アクセス	DATA	プログラムのダウンロード時にEEPROMにデータを書き込む。				
16		READ	EEPROMからバイト値を読み出す。				
17		WRITE	EEPROMにバイト値を書き込む。				
18	RAM アクセス	STORE	READ / WRITEが操作するプログラム・スロットに切り換える。	x	x	x	
19		GET	スタラッチパッドRAMからバイト値を読み、変数に保存する。				
20		PUT	スタラッチパッドRAMにバイト値を書き込む。	x			
21	デジタル I/O	LOOKUP	ターゲットとリストを比較し、マッチしたインデックスを返す。				
22		LOOKDOWN	インデックスで指定した値をリストから取り出す。				
23		RANDOM	疑似乱数を発生する。				
24		INPUT	一つの /0ピンを入力に設定する。				
25		OUTPUT	一つの /0ピンを出力に設定する。				
26		REVERSE	一つの /0ピンの入出力方向を逆(入力を出力, 出力を入力)にする。				
27		LOW	一つの /0ピンを出力にし, 出力をLOにする。				
28	HIGH	一つの /0ピンを出力にし, 出力をHIにする。					
29	TOGGLE	一つの /0ピンを出力にし, 状態を反転(HIをLOに, LOをHIに)する。					
30	PULSIN	入力パルスの幅を測定する。					
31	PULSOUT	一つのピンを出力にし, 指定時間だけ出力を反転してパルスを発生する。					
32	BUTTON	ボタン(スイッチ)のバウンスを抑制し, オートリートをを行い, ボタンがターゲットの状態ならアドレスに分岐する。					

NO	グループ	命令	説明	BASICスタンプの種類				
				BS2	BS2e	x	BS2p	
33	デジタル I/O	COUNT	指定した期間中の信号サイクルを数える。					
34		XOUT	10パワーステップ制御コードを発生する。TW523またはTW513パワーステップ・インターフェース・モジュールに使用する。					
35		AUXIO	メイン /0ピンから補助(AUX) /0ピンに制御を移す。	x	x	x		
36		MAINIO	補助(AUX) /0ピンからメインI/Oピンに制御を移す。	x	x	x		
37		IOTERM	指定した /0ピン・グループ(メイン/補助)に制御を移す。	x	x	x		
38		POLLIN	ポールド・インタラプトするピンと状態を指定する。	x	x	x		
39		POLLOUT	ポールド・インタラプトにもとづいてピンと状態を指定する。	x	x	x		
40		POLLMODE	ポールド・コマンド・モードを設定する。	x	x	x		
41		非同期 シリアル I/O	SERIN	シリアル入力。 もしクオリアファイアが与えられると, 命令は変数を満たすまたは次の命令に行く前にそれらを受信する。 もしタイムアウトが与えられると, 時間内に何も受信できなければ命令は中断される。 ボーレート300 - 50,000(フロー制御ありでは0 - 19,200)が可能。 データはN81(パリティなし, 8ビット・データ, 1ストップ・ビット)またはE71(偶パリティ, 7データ・ビット, 1ストップ・ビット)で受信される。				
42			SEROUT	シリアル出力。 もしペースが与えられると, 命令は各バイト転送の間に指定された遅延を挿入する(フロー制御時にはペースは利用できない)。 ボーレート300-50,000(フロー制御ありでは0-19,200)が可能。 データはN81(パリティなし, 8ビット・データ, 1ストップ・ビット)またはE71(偶パリティ, 7データ・ビット, 1ストップ・ビット)で送信される。				
43		OWIN	1 - Wireプロトコルを使ってデバイスからデータを受け取る。	x	x	x		
44		OWOUT	1 - Wireプロトコルを使ってデバイスにデータを送る。	x	x	x		
45	同期 シリアル I/O	SHIFTIN	シフト・ビットを入力する(シリアル・パラレル変換)。					
46		SHIFTOUT	シフト・ビットを出力する(パラレル・シリアル変換)。					
47		2CIN	2Cプロトコルを使ってデバイスからデータを受信する。	x	x	x		
48		2COUT	2Cプロトコルを使ってデバイスにデータを送信する。	x	x	x		
49	パラレル I/O	LCDCMD	LCD表示器にコマンドを送る。	x	x	x		
50		LCDIN	LCD表示器からデータを受け取る。	x	x	x		
51		LCDOUT	LCD表示器にデータを送る。	x	x	x		
52	アナログ I/O	PWM	パルス幅変調信号を出力した後, ピンを入力に戻す。 コンデンサと抵抗器の平滑回路を使ってアナログ電圧(0 - 5V)を出力できる。					
53		RCTIME	ピンがHまたはLである期間(主にRC充放電時間)を測定する。					
54	時間	PAUSE	0 - 65535ミリ秒の間実行を中断する。					
55		POLLWAIT	ポールドされたピンが所定の状態になるまでプログラムを一時停止する。	x	x	x		
56	音	FREQOUT	一つまたは二つの正弦波を発生する(周波数はそれぞれ0-32767Hz)。					
57		DTMFOUT	DTMF電話トーンを発生する。					
58	電源制御	NAP	短時間の巣ループ・モードに入る。					
59		SLEEP	1 - 65535秒の間スリープする。電力消費は低減する。					
60		END	プログラムを終了し, ハードウェアリセットされるまでスリープする。電力消費は低減する。					
61		STOP	プログラムの実行を停止する。					
62	デバッグ	DEBUG	変数を調べるためにホストPCの画面に表示する。					
63		DEBUGIN	デバッグ・ターミナルからのユーザ入力。					



# フローチャート NO.9

フローチャート (flow chart)とは、別名「流れ図」ともいい、仕事や行動の過程を図に表す方法のひとつです。

コンピュータに仕事をさせるときの手順書(プログラム)の作製の補助として用いるものですが、それ以外にも利用法はあります。

簡単なプログラムでは、特にフローチャートは必要ありませんが、複雑な処理をするプログラムを作成する場合、フローチャートを作ることによって、有効に作成・修正が行えるようになります。

## 1. フローチャートの記号

フローチャートは、仕事(処理)を1つ1つの手順に分けて整理し、その順序を明確に表現した図記号と線で表したものです。次に、フローチャートで使用する主なフローチャート記号を示します。

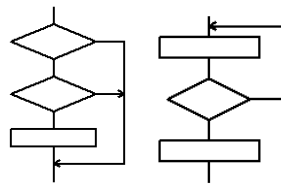
	流れ線	処理の順序を表す。順序を明確にするために、矢印を付けることもある。
	端子	処理の開始・終了を表す。
	処理	判断などの処理以外の処理を表す。
	入出力	データの入出力を表す。
	判断	条件により、流れが二つ以上に分岐する処理を表す。
	ループ端	繰り返しの開始と終了を表す。
	書類	プリンタなどへのデータ出力を表す。
	定義済み処理	別に用意した処理を利用することを表す。

## 2. フローチャート作成上の約束

処理の流れの方向は「左から右へ」、「上から下へ」とします。これ以外の場合は、矢印を使用します。

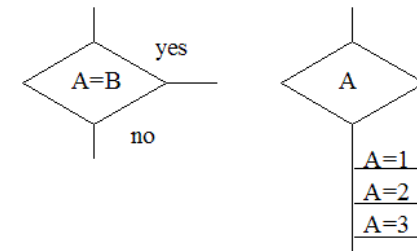
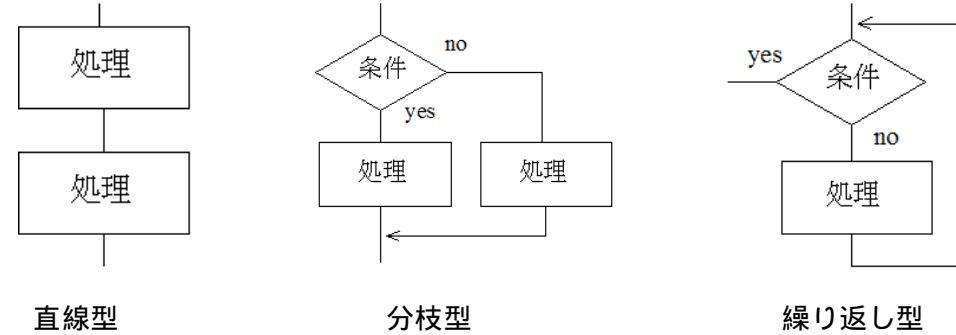
流れ線は互いに交差してもかまいません。

2つ以上の流れ線を集めて、1つにしてもかまいません。



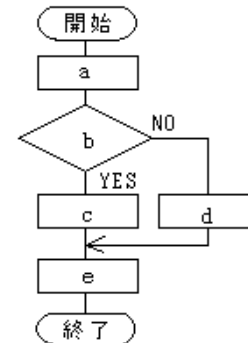
## 3. フローチャートの基本形

アルゴリズム(算法)とは、問題を解決するための手順を示したものです。その手順の示し方には、一般的にフローチャート(流れ図)が使用されます。そのフローチャートにはいくつかの基本形があります。

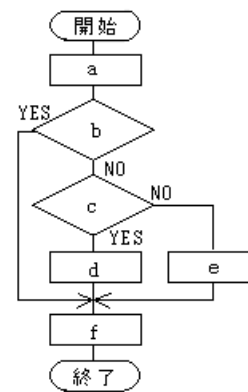


### 条件判断型

問題 次の文章を読んでフローチャートを完成しなさい。



- (1) 「チョコレートを買いにスーパーへ行き、A社とB社のチョコレートのうち安い方を買って家に帰る。」
- A社のチョコレートを買う。 a [ ]
  - B社のチョコレートを買う。 b [ ]
  - 家に帰る。 c [ ]
  - スーパーに行く。 d [ ]
  - A社のチョコレートの方が安い。 e [ ]



- (2) 「果物を買いにスーパーに行く。りんごかみかんのうち安い方を買うことにする。ただし、スーパーが休みなら家に帰る。」
- りんごを買う。 a [ ]
  - みかんを買う。 b [ ]
  - りんごの方が安い。 c [ ]
  - スーパーが休み。 d [ ]
  - 家に帰る。 e [ ]
  - スーパーに行く。

フローチャートの上から下へ処理の順序が一直線に進む(流れる)形の処理です。基本的には、データ入力(データの記憶) 計算 結果出力、の順となります。

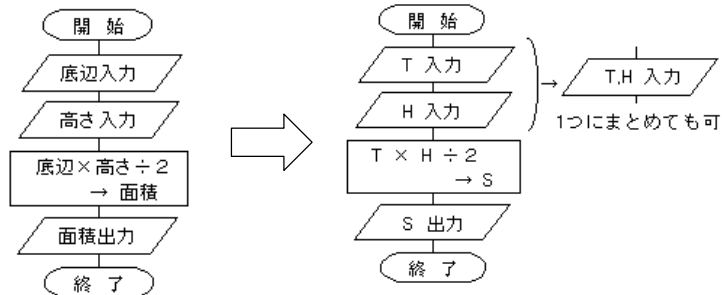
例えば、三角形の面積を求めるには、「底辺×高さ÷2」という公式を使って計算しますが、この計算を行うには底辺と高さのデータが必要であることがわかります。したがって、処理の流れは次のように考えることができます。

底辺のデータ入力、高さのデータ入力

公式により面積を計算

計算結果(面積)を出力

これをフローチャートで表すと次のようになります。



例題1 三角形の面積を求めるプログラムをPBASICで作成しなさい。

答え

<pre>{ \$STAMP BS2 } { \$PBASIC 2.5 } { \$PORT COM1 }  t var word h var word s var word  Debugin dec t Debugin dec h  s=(t * h)/2  Debug dec s,cr  End</pre>	<p>} スタンプエディターに対する指令です。直接プログラムには関係しません。</p> <p>} プログラムで利用する 変数 を VAR 指令で宣言しスタンプエディターに知らせます。 word : 16ビット byte : 8ビット bit : 1ビット で変数のビット数を指定します。</p> <p>パソコンから 10進数(dec)として t の値を読みます。 " " h "</p> <p>三角形の面積を計算します。</p> <p>面積 s を 10進数(dec)としてパソコンに出力します。 cr は出力ご改行させます。</p> <p>終わり</p>
--	--

実習1

例題1をスタンプエディターを使ってBASICStampにプログラミングしなさい。次に次の値を入力して答えを得なさい。BASICStamp 学習基板のリセットを押してからデータを入力すること

- (1)t=100, h=200 S= (2)t=128 h=256 S=  
 (3)t=1000, h=2000 S=

問題 1~Nまでの整数の和を求める計算の

フローチャートを描きなさい。

PBASICのプログラムを書きなさい。

スタンプエディターを使ってBASICStampにプログラミングしなさい。

公式  $S = \frac{(1+N) \times N}{2}$  を使って計算しなさい。

フローチャート

PBASICプログラム

	<pre>{ \$STAMP BS2 } { \$PBASIC 2.5 } { \$PORT COM1 }</pre>
--	---

実習2

問題をスタンプエディターを使ってBASICStampにプログラミングし、次に次の値を入力して答えを得なさい。BASICStamp 学習基板のリセットを押してからデータを入力すること

- (1)N=10 S= (2)N=100 S=  
 (3)N=200 S= (4)N=300 S=

数の表現 (記数法 Number System) NO.11

1年 組 番 氏名 \_\_\_\_\_

1. 10進数を2進数に変換する。(10進数の55を2進数に変換する)

55 を 2 で次々割っていきます

つまり

2	$\overline{32 + 16 + 4 + 2 + 1}$	
2	$\overline{16 + 8 + 2 + 1 + \cancel{0}}$	あまり 1 (1)
2	$\overline{8 + 4 + 1 + \cancel{0}}$	あまり 1 (2)
2	$\overline{4 + 2 + \cancel{0}}$	あまり 1 (4)
2	$\overline{2 + 1}$	あまり 0 (8)
2	$\overline{1 + \cancel{0}}$	あまり 1 (16)
2	$\overline{\cancel{0}}$	あまり 1 (32)

かんたんに書くと

2	$\overline{55}$	
2	$\overline{27}$	… 1
2	$\overline{13}$	… 1
2	$\overline{6}$	… 1
2	$\overline{3}$	… 0
2	$\overline{1}$	… 1
2	0	… 1

「あまり」を下の方から並べて、  
55<sub>(10)</sub> = 110111<sub>(2)</sub>

「あまり」を下の方から並べて、  
55<sub>(10)</sub> = 110111<sub>(2)</sub>

練習問題1 次の10進数を2進数に変換しなさい。

(1) 67	(2) 185	(3) 297
--------	---------	---------

問題 255 511 1023を2進数に変換しなさい。(すぐにできます)

2. 小数の10進数を2進数に変換する。(10進数の0.8125を2進数に変換する)

10進数の整数を2進数に変換するときは2で割って行って、あまりを2進数の対応する桁としましたが、少数の場合は2をかけて行って、整数部の値を対応する2進数の桁にします。小数部が0になるまで続けます。

次々に 2 をかけていきます

2	$\overline{.8125}$	
2	$\overline{\cancel{1}.6250}$	整数部 1 1/2
2	$\overline{\cancel{1}.250}$	整数部 1 1/4
2	$\overline{.50}$	整数部 0 1/8
2	$\overline{\cancel{1}.0}$	整数部 1 1/16

2	$\overline{.8125}$	
2	$\overline{1.6250}$	1
2	$\overline{1.250}$	1
2	$\overline{0.50}$	0
2	$\overline{1.0}$	1

0になったので終わり

上から順にならべて、  
0.8125<sub>(10)</sub> = 0.1101<sub>(2)</sub>

0.8125<sub>(10)</sub> = 0.1101<sub>(2)</sub>

練習問題1 次の10進数を2進数に変換しなさい。

(1) 0.75	(2) 0.3125	(3) 0.84375
----------	------------	-------------

問題 0.4を2進数に変換しなさい。(適当な桁で計算をやめて結構です!?)

3. 2進数の計算 足し算・かけ算

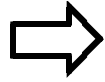
(1) 足し算

たし算

4つを覚える

$$\begin{array}{r} 0 \quad 0 \quad 1 \quad 1 \\ +0 \quad +1 \quad +0 \quad +1 \\ \hline 0 \quad 1 \quad 1 \quad 10 \end{array}$$

2は使えないから



たし算 右側から順にたしていき

(例1)  $\begin{array}{r} 101 \\ + 11 \\ \hline 1000 \end{array}$  (例2)  $\begin{array}{r} 1001 \\ + 101 \\ \hline 1110 \end{array}$

点のところは1を送る

問題1 次の2進数の足し算を計算しなさい。

(1)  $\begin{array}{r} 110 \\ + 11 \\ \hline \end{array}$  (2)  $\begin{array}{r} 1100 \\ + 110 \\ \hline \end{array}$

かけ算

ここも4つ覚えるだけ

$$\begin{array}{r} 0 \quad 0 \quad 1 \quad 1 \\ \times 0 \quad \times 1 \quad \times 0 \quad \times 1 \\ \hline 0 \quad 0 \quad 0 \quad 1 \end{array}$$

0に何かけても0です

かけ算

(例3)  $\begin{array}{r} 101 \\ \times 11 \\ \hline 101 \\ 101 \\ \hline 1111 \end{array}$  (例4)  $\begin{array}{r} 1001 \\ \times 101 \\ \hline 1001 \\ 0000 \\ 1001 \\ \hline 101101 \end{array}$

問題2 次の2進数のかけ算を計算しなさい。

(1)  $\begin{array}{r} 110 \\ \times 11 \\ \hline \end{array}$  (2)  $\begin{array}{r} 1100 \\ \times 110 \\ \hline \end{array}$

4. 2進数の論理演算

代表的な論理演算

AND

	2番目	1
1番目	0	0
1	0	1

and アンド 論理積  
1,2番ともに1のときのみ  
1を返します

OR

	2番目	1
1番目	0	1
1	1	1

or オア 論理和  
1,2番の少なくとも一方が  
1のとき1を返す

XOR

	0	1
0	0	1
1	1	0

xor エクスクルーシブオア  
論理的排他和  
1,2番のどちらか一方だけが  
1のとき1を返す  
つまり 同じとき0、異なるとき  
1を返す

NOT

0	1
1	0

not ノット 論理否定  
反対(逆)を返す

例題 8ビットの2進数で論理演算を行います。

$$\begin{array}{r} 0010 \quad 1011 \\ \text{AND } 0000 \quad 1111 \\ \hline 0000 \quad 1011 \end{array} \quad \begin{array}{r} 0010 \quad 1011 \\ \text{OR } 0000 \quad 1111 \\ \hline 0010 \quad 1111 \end{array}$$

$$\begin{array}{r} 0010 \quad 1011 \\ \text{XOR } 0000 \quad 1111 \\ \hline 0010 \quad 0100 \end{array} \quad \begin{array}{r} 0010 \quad 1011 \\ \text{NOT } 0010 \quad 1011 \\ \hline 1101 \quad 0100 \end{array}$$

問題 次の8ビットの論理演算を行いなさい。また論理演算の結果を10進数でも示しなさい。

(1)  $\begin{array}{r} 1111 \quad 1111 \\ \text{AND } 1010 \quad 1010 \\ \hline \end{array}$  255 AND 170 = [ ] 10進数で

(2)  $\begin{array}{r} 1111 \quad 1111 \\ \text{XOR } 1010 \quad 1010 \\ \hline \end{array}$  255 XOR 170 = [ ] 10進数で



答え

三角形の面積

```
' {$STAMP BS2}
' {$PBASIC 2.5}
' {$PORT COM1}
```

```
t  VAR  word
h  VAR  word
s  VAR  word
```

```
debugin  dec t
debugin  dec h
```

```
s=(t*h)/2
```

```
debug  dec s,cr
```

```
end
```

1 ~ Nの和

```
' {$STAMP BS2}
' {$PBASIC 2.5}
' {$PORT COM3}
```

```
n  VAR  word
s  VAR  word
```

```
debugin  dec n
```

```
s=(1+n)*n/2
```

```
debug  dec s,cr
```

```
end
```

論理演算

```
' {$STAMP BS2}
' {$PBASIC 2.5}
' {$PORT COM3}
```

```
t  VAR  word
h  VAR  word
s  VAR  word
```

```
debugin  dec t
debugin  dec h
```

```
debug  " t =",bin8 t,cr
debug  " h =",bin8 h,cr
```

```
s=t & h
```

```
debug  "AND=",bin8 s,cr
debug  "AND=",dec s,cr,cr
```

```
debug  " t =",bin8 t,cr
debug  " h =",bin8 h,cr
```

```
s=t ^ h
```

```
debug  "XOR=",bin8 s,cr
debug  "XOR=",dec s,cr,cr
```

```
debug  " t =",bin8 t,cr
debug  " h =",bin8 h,cr
```

```
s=t | h
```

```
debug  "OR=",bin8 s,cr
debug  "OR=",dec s,cr,cr
```

```
end
```