

PLCopenの活動紹介と PLC関連規格の最新動向

PLCopen Japan
チェアマン 宮澤 以鋼

PLCopenの活動紹介

- Wal氏のプレゼンの要約
- 詳細内容は技術委員会で紹介

PLC関連規格の最新動向: JIS

- 2012年6月20日 最新版を発行
- IEC 61131-3 Ed.2.0 + Errata

規格番号	JISB3503		
規格名称	プログラマブルコントローラープログラム言語		
英文名称	Programmable controllers-Programming languages		
主務大臣	経済産業	主務大臣	
部会名	標準部会	(部会長名)	稲葉 敦
専門委員会名	産業オートメーション技術	(委員長名)	高田 祥三
WG名		(WG主査名)	
制定年月日	1997/01/20		
最新改正年月日	2012/06/20	最新確認年月日	2008/03/20
原案作成団体	社団法人 日本電気制御機器工業会	原案作成団体(共同提案者)	一般財団法人 日本規格協会
関係団体名		関係団体名	

JIS B 3500シリーズの審議体制

- IEC TC65国内委員会 (JEMIMA)
一般社団法人日本電気計測器工業会
- SC65B/WG7 (PLC) 国際規格の審議

JIS化依頼

- PLCハードウェアのJIS化 (JEMA)
一般社団法人日本電機工業会
- Part 1、2、6のJIS化担当

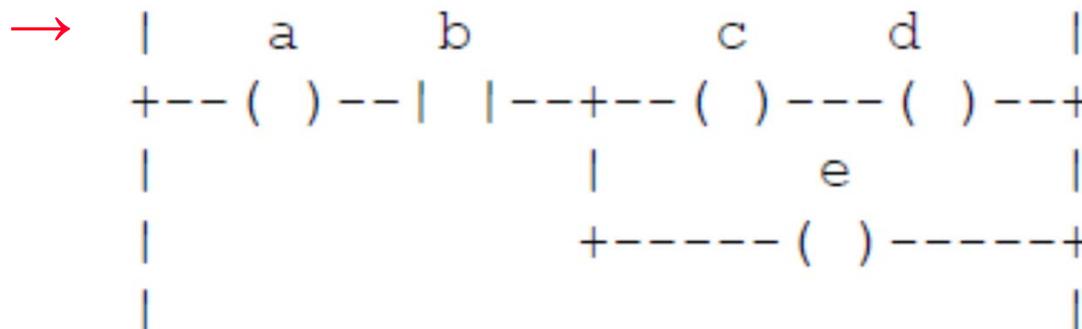
- PLCソフトウェアのJIS化 (NECA)
一般社団法人日本電気制御機器工業会
- Part 3のJIS化担当

PLCopen Japan参画

- 文字集合の拡張
 - **2バイトの文字列**が使用可能(日本語などの表現に必要)
- データ型の拡張
 - 2バイトの文字列のために、**WSTRING型**を新規定義
- 変数宣言に新制約(実質的に拡張)
 - **ワイルドカード(*)**の使用
 - **VAR_TEMP**(実行中のみ存在する変数)の追加
 - **NON_RETAIN**(電源OFFすると保持しない)の**明示的**定義
 - ファンクション呼び出しに**出力変数と入出力変数**の追加
 - **ファンクションブロックの機能拡張**(呼び出し方法など柔軟に)
- POUにおいては、**EN**と**ENO**の記述及び例を強化
 - **テキスト言語**で**EN/ENO**の使用方法の規範化
 - **FBD言語**でも明確に**EN/ENO**の使用方法の規範化
- 表34から**セマフォ**を削除

JIS B 3503改正要点(2)

- 表36にあるアップダウンカウンタのボディの記述の間違いの訂正
- 表45にあるSFCのAQにP1(立上り)とP0(立下り)の追加
→ 従来のPだけの記述より、詳細な記述可能
- SFCのアクション制御ACTION_CONTROLブロック
→ 実装が選択に
→ ファイナルスキャンも選択に
- 各プログラミング言語の説明や用例の記述を強化
→ 規格を参照
- ラダーダイアグラムのコイルの位置による取るべき値の例示



IEC 61131-3 Ed.3の改訂作業

- **2008年10月 米国クリーブランド会議**
 - 日本は準備不足で、代表派遣できず
 - PLCopenがDリエゾンで代表派遣
 - Ed.3に関する改定内容が議論
- **2009年 5月 スペインビゴ会議**
 - 日本は出席、日本のコメントを提示
 - オブジェクト指向の導入を議論、了承
- **2010年 6月 ドイツヴェルツブルク会議**
 - 日本は出席、日本のコメントを最初に議論(半日)、ほぼ受け入れ
 - オブジェクト指向の表現等表現に紛糾(ドイツ国内の意見が集約できず)
- **2011年 3月 オーストリアクレムス会議**
 - 日本は準備不足で、代表派遣中止(地震直後で、派遣しても出発できず)
 - 受け入れた日本コメントの一部は不明瞭の形で覆され
- **2012年 2月 ドイツケンプテン会議**
 - 日本は出席、FDIS作成 → **いつFDISが回付されるか？**

規格準拠宣言の詳細化(必須)

IEC 61131-3 “PLC Programming Languages”

Implementer: Company name, address, etc.

Product: Product name, version, etc. Controller type specific subset, etc.

Date: 2012-05-01

This Product complies with the requirements of the standard for the following language features:

Feature No.	Table Number and Title / Feature Description	Compliantly implemented in the language (✓)				Implementer's note
		LD	FB D	ST	IL	
	Table 1 – Character set					
1	ISO/IEC 10646 2011, Universal Coded Character Set (UCS)	✓	✓	✓	✓	
2a	Lower case characters a: a, b, c, ...	✓	✓	✓		No “ß, ü, ä, ö”

他言語コメント表記の追加

No.	Description	Examples
1	Single-line comment with // ...	<pre>X := 13; // comment for one line // a single line comments can start at // the first character position.</pre>
2a	Multi-line comment with (* ... *)	<pre>(* comment *) (***** A framed comment on three line *****)</pre>
2b	Multi-line comment with /* ... */	<pre>/* comment in one or more lines */</pre>
3a	Nested comment with (* .. (* .. *) .. *)	<pre>(* (* NESTED *) *)</pre>
3b	Nested comment with /* .. /* .. */ .. */	<pre>/* /* NESTED */ */</pre>

2バイト文字列表記の強化

	Double-byte typed string literals with # (NOTE)	
4a	Typed double-byte string (using “double quote” character)	WSTRING# "OK"
4b	Typed double-byte character (using “double quote” character)	WCHAR# "X"
4c	Typed double-byte string (using “single quote” character)	WSTRING# 'OK'
4d	Typed double-byte character (using “single quote” character)	WCHAR# 'X'

マイクロ秒の表記の追加

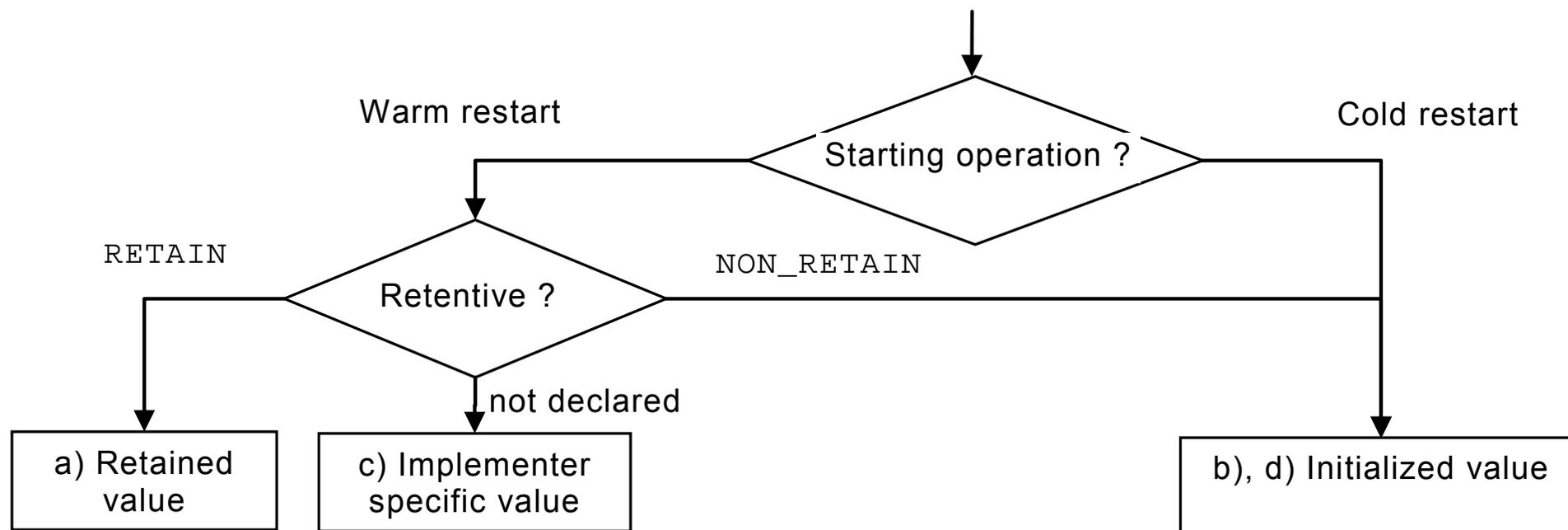
	Duration abbreviations	
1a	d	Day
1b	h	Hour
1c	m	Minute
1d	s	Second
1e	ms	Millisecond
1f	us (no μ available.)	Microsecond

Generic data types	Generic data types	Groups of elementary data types
ANY		
ANY_DERIVED		
ANY_ELEMENTARY		
ANY_MAGNITUDE		
ANY_NUM		
ANY_REAL		REAL, LREAL
ANY_INT	ANY_UNSIGNED	USINT, UINT, UDINT, ULINT
	ANY_SIGNED	SINT, INT, DINT, LINT
ANY_DURATION		TIME, LTIME
ANY_BIT		BOOL, BYTE, WORD, DWORD, LWORD
ANY_CHARS		
ANY_STRING		STRING, WSTRING
ANY_CHAR		CHAR, WCHAR
ANY_DATE		DATE_AND_TIME, LDT, DATE, TIME_OF_DAY, LTOD

データ型分類の規範化

- 基本データ型
- 総称データ型
- ユーザ定義データ型
 - 列挙データ型
 - 部分範囲データ型
 - 配列データ型
 - 構造化データ型
 - AT表現
 - 直接表現
 - 派生データ型
 - 参照

RETAINのスキープの明示



ANY_BIT変数への部分アクセス

No.	Description	Data Type	Example and Syntax (NOTE 2)
	Data Type - Access to		myVAR_12.%X1; yourVAR1.%W3;
1a	BYTE - bit VB2.%X0	BOOL	<variable_name>.%X0 to <variable_name>.%X7
1b	WORD - bit VW3.%X15	BOOL	<variable_name>.%X0 to <variable_name>.%X15
1c	DWORD - bit	BOOL	<variable_name>.%X0 to <variable_name>.%X31
1d	LWORD - bit	BOOL	<variable_name>.%X0 to <variable_name>.%X63
2a	WORD - byte VW4.%B0	BYTE	<variable_name>.%B0 to <variable_name>.%B1
2b	DWORD - byte	BYTE	<variable_name>.%B0 to <variable_name>.%B3
2c	LWORD - byte	BYTE	<variable_name>.%B0 to <variable_name>.%B7
3a	DWORD - word	WORD	<variable_name>.%W0 to <variable_name>.%W1
3b	LWORD - word	WORD	<variable_name>.%W0 to <variable_name>.%W3
4	LWORD - dword VL5.%D1	DWORD	<variable_name>.%D0 to <variable_name>.%D1
<p>The bit access prefix %X may be omitted according エラー! 参照元が見つかりません。 , e.g. By1.%X7 is equivalent to By1.7.</p> <p>Partial access shall not be used with a direct variable e.g. %IB10.</p>			

データ型変換の暗黙と明示表記

Source Data Type		Target Data Type																										
		real		integer				unsigned				bit				date & times						char						
		LREAL	REAL	LINT	DINT	INT	SINT	ULINT	UDINT	UINT	USINT	LWORD	DWORD	WORD	BYTE	BOOL	LTIME	TIME	LDT	DT	LDATE	DATE	LTOD	TOD	WSTRING	STRING	WCHAR	CHAR
real	LREAL	e	e	e	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	REAL	i	e	e	e	e	e	e	e	e	-	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
integer	LINT	e	e	e	e	e	e	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	DINT	i	e	i	e	e	e	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	INT	i	i	i	i	e	e	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	SINT	i	i	i	i	e	e	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
unsigned	ULINT	e	e	e	e	e	e	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	UDINT	i	e	i	e	e	i	e	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	UINT	i	i	i	i	e	i	i	e	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	USINT	i	i	i	i	e	i	i	i	e	e	e	e	e	-	-	-	-	-	-	-	-	-	-	-	-	-	-

ファンクションの戻り値の有無

- 日本は反対したが、結果的に残った。
- VOIDという表記だけはなくなった。
- 副作用については依然議論が必要である。
- 戻り値なしはファンクションブロックで実装すべきである。
- ファンクションブロックの戻り値は???

No.	Description	Example
1a	Without result FUNCTION ... END_FUNCTION	FUNCTION myFC ... END_FUNCTION
1b	With result FUNCTION <name> : <data type> END _FUNCTION	FUNCTION myFC: INT ... END_FUNCTION

EXAMPLE 1 Endianness

```
TYPE D: DWORD := 16#1234_5678; END_TYPE;
```

Memory layout

for big endian: 16#12, 16#34, 16#56, 16#78

for little endian: 16#78, 16#56, 16#34, 16#12.

EXAMPLE 2 Endianness

```
TYPE L: ULINT := 16#1234_5678_9ABC_DEF0; END_TYPE;
```

Memory layout

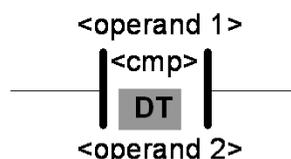
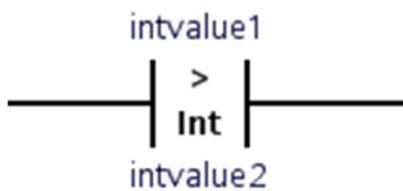
for big endian: 16#12, 16#34, 16#56, 16#78, 16#9A, 16#BC, 16#DE, 16#F0

for little endian: 16#F0, 16#DE, 16#BC, 16#9A, 16#78, 16#56, 16#34, 16#12.

正当性判定ファンクションの導入

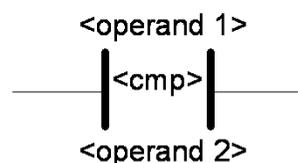
No.	Function	Graphical form	Example
1	IS_VALID	<pre> +-----+ IS_VALID ANY_REAL-- IN --BOOL +-----+</pre>	Validity of a REAL VAR R: REAL; END_VAR IF IS_VALID(R) THEN ...
2	IS_VALID_BCD	<pre> +-----+ IS_VALID_BCD -ANY_BIT-- IN --BOOL +-----+</pre>	Validity test for a BCD word VAR W: WORD; END_VAR IF IS_VALID_BCD(W) THEN ...

LDに比較接点表記の追加(1)

5a	Compare contact (typed)	 <p>The state of the right link is ON from one evaluation of this element to the next when the left link is ON and the <cmp> result of the operands 1 and 2 is true.</p> <p>The state of the right link shall be OFF otherwise.</p> <p>< cmp> may be substituted by one of the compare functions that are valid for the given data type.</p> <p>DT is the data type of both given operands.</p> <p>Example:</p>  <p>If the left link is ON and (intvalue1 > intvalue2) the right link switches to ON. Both intvalue1 and intvalue2 are of the data type INT</p>
----	-------------------------	---

LDに比較接点表記の追加(2)

5b Compare contact, (overloaded)

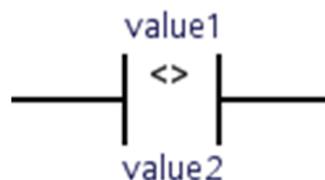


The state of the right link is ON from one evaluation of this element to the next when the left link is ON and the <cmp> result of the operands 1 and 2 is true.

The state of the right link shall be OFF otherwise.

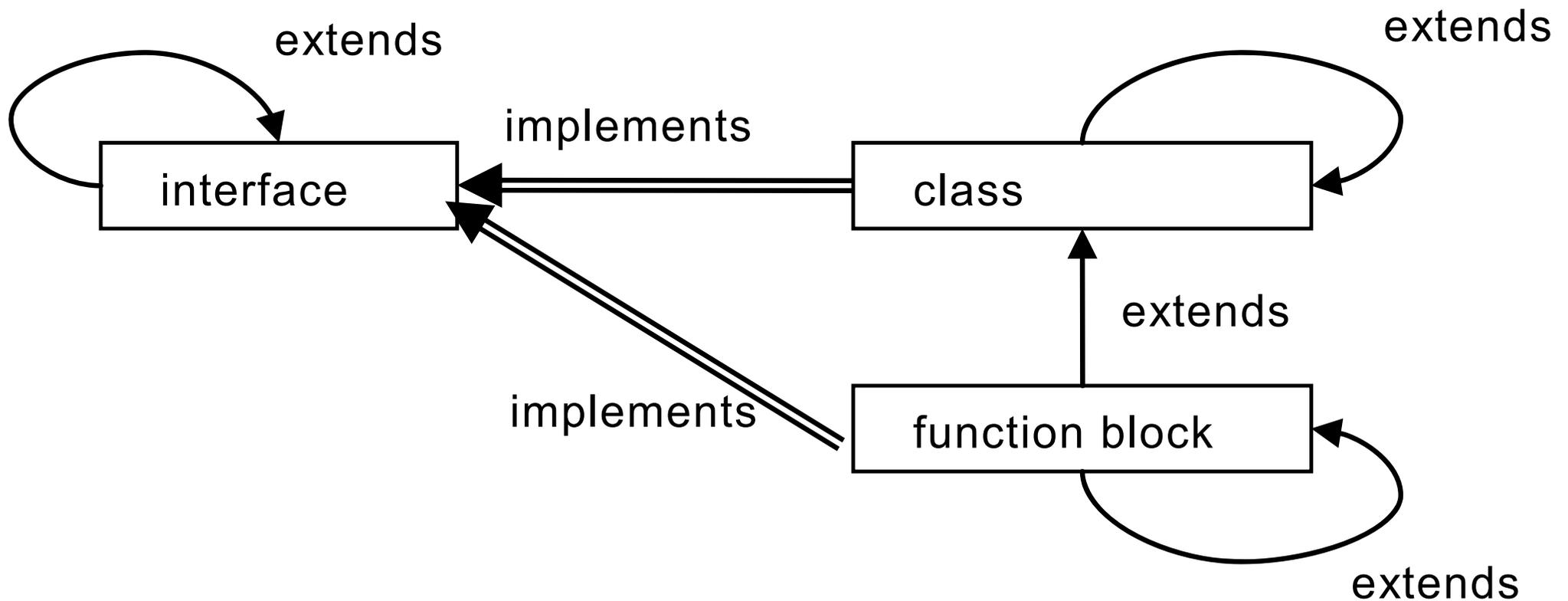
<cmp> may be substituted by one of the compare functions that are valid for the operands data type. The rules defined in エラー! 参照元が見つかりません。 shall apply.

Example:



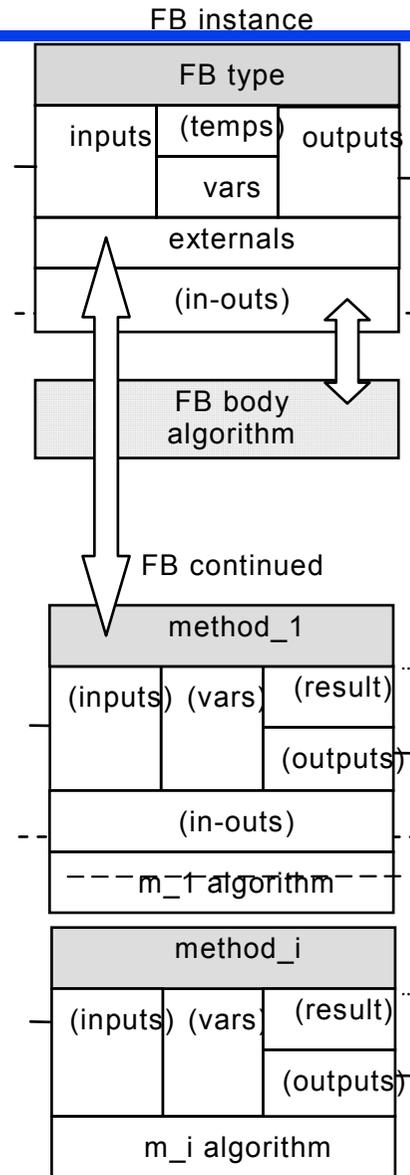
If the left link is ON and (value1 <> value2) the right link switches to ON.

クラスの概念の導入



オブジェクト指向概念の導入(2)

ファンクション
ブロックの
概念の拡張



ご清聴ありがとうございました。

