

Struktur memori pada SoMachine Basic untuk PLC M221

Pada SoMachine Basic ada beberapa jenis object representasi dari sebuah memori

1. Memori Bit Object

2. Word Object

3. Floating Point dan Double Word Object

1. Memori Bit Object

Pada SoMachine Basic untuk membuat program dengan model bit ada beberapa tipe adress yang dapat dipakai , Kita bisa menggunakan ,%M ,%S,%X

%M adalah bit yang berdasar alamat Memori yang akan berjalan ketika program Run

contohnya adalah %M0,%M20,%M50 ,nilai di belakang %M menunjukan nomor dari memori tersebut dibawah ini adalah contoh program menggunakan %M



System Bit

System Bit adalah bit yang akan langsung on jika PLC Run ,pada SoMachine Basic untuk bit ini disymbolkan dengan %Sx (x disini ada nomor dari System bit)

di SoMachine terdapat bit system number sampai %S123 dengan berbagai macam fungsi ,Contoh yang biasa digunakan yaitu:

%S4 = ini untuk clock 10 ms

%S5 = ini untuk clock 100 ms

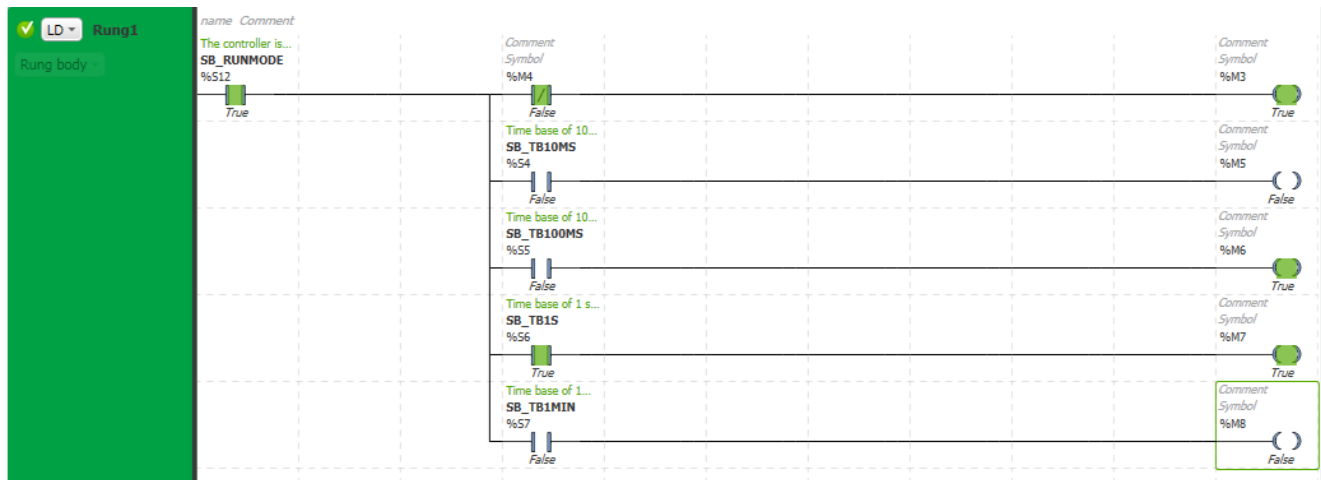
%S6 = untuk clock 1 s

%S7 = untuk clock 1m

%S12 = untuk always On flag

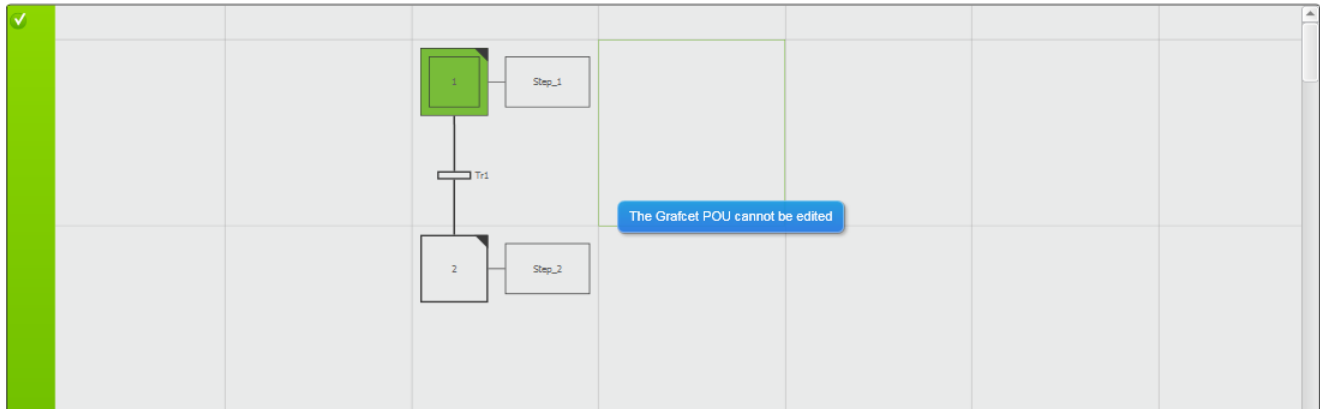
%S13 = untuk First scan Flag

dibawah ini adalah contoh program menggunakan system bit



Grafcet steps Bit

adalah sebuah bit yang terhubung dengan Grafcet(SFC) pada SoMachine ditandai dengan %X, bit ini akan On jika step aktif dan akan off jika step nonaktif.



2.Word Object

Word object dengan Structure 16-bit words pada SoMachine Basic terdapat beberapa jenis yaitu

Memory words(%MW)

Constant words(%KW)

System words(%SW)

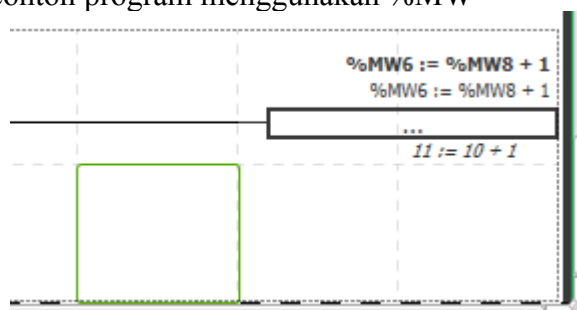
I/O exchange words(%IW,%QW,%IWS,%QWS)

Function Blocks

Memory Word(%MW)

Sama seperti halnya Memory Bit,pada SoMachine terdapat juga Memory Word .Memory Word ini mempunyai panjang data 16 bit dapat mengisi bilangan type INT ,disymbolkan dengan %Mwx(x adalah nomor dari Memory Words tersebut) pada SoMachine basic %MW4 dan %MW5 dapat digunakan bersama.

dibawah ini adalah contoh program menggunakan %MW



pada gambar diatas digunakan %MW6 dan %MW8

Memory word properties				%MW	%MD	%MF
Used	Equ Used	Address	Symbol			
<input type="checkbox"/>	<input type="checkbox"/>	%MW3				
<input type="checkbox"/>	<input type="checkbox"/>	%MW4				
<input type="checkbox"/>	<input type="checkbox"/>	%MW5				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW6				
<input type="checkbox"/>	<input type="checkbox"/>	%MW7				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW8				
<input type="checkbox"/>	<input type="checkbox"/>	%MW9				

pada SoMachine %MW7 dan %MW9 bisa digunakan.

Constant Words(%KW)

Constant words memudahkan kita untuk memasukan nilai string/text.dan perbedaan lainnya untuk %KW ini kita bisa masukan nilai sebelum PLC Run ke dalamnya bisa dalam bentuk decimal ,binary,hexadecimal ataupun ascii

Constant word properties

%KW

%KD

%KF

Import

Export

Quick Address Search

Allocation

Automatic

No. of Objects

0

Allocated: 0, Available: 512

Used	Equ Used	Address	Symbol	Decimal	Binary	Hexadecimal	ASCII	Configuration
<input type="checkbox"/>	<input type="checkbox"/>	%KW0		0	2#0000000000000000	16#0000	no meaning	<div>...</div>
<input type="checkbox"/>	<input type="checkbox"/>	%KW1		0	2#0000000000000000	16#0000	no meaning	<div>...</div>
<input type="checkbox"/>	<input type="checkbox"/>	%KW2		0	2#0000000000000000	16#0000	no meaning	<div>...</div>
<input type="checkbox"/>	<input type="checkbox"/>	%KW3		0	2#0000000000000000	16#0000	no meaning	<div>...</div>
<input type="checkbox"/>	<input type="checkbox"/>	%KW4		0	2#0000000000000000	16#0000	no meaning	<div>...</div>
<input type="checkbox"/>	<input type="checkbox"/>	%KW5		0	2#0000000000000000	16#0000	no meaning	<div>...</div>
<input type="checkbox"/>	<input type="checkbox"/>	%KW6		0	2#0000000000000000	16#0000	no meaning	<div>...</div>

Kita akan coba masukan nilai ascii ke dalam %KW0

Constant string Assistant ✕

Enter your text:

Hallo

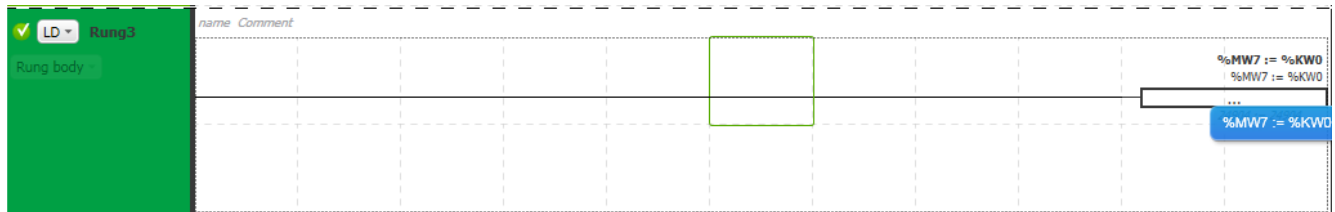
Range of constants required: from **0** to **2**

Apply Cancel

Klik apply dan Cancel maka hasilnya akan seperti ini

Used	Equ Used	Address	Symbol	Decimal	Binary	Hexadecimal	ASCII	Configuration	Cc
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%KW0		24904	2#0110000101001000	16#6148	aH	...	
<input type="checkbox"/>	<input type="checkbox"/>	%KW1		27756	2#0110110001101100	16#6C6C	ll	...	
<input type="checkbox"/>	<input type="checkbox"/>	%KW2		3439	2#0000110101101111	16#0D6F	o	...	

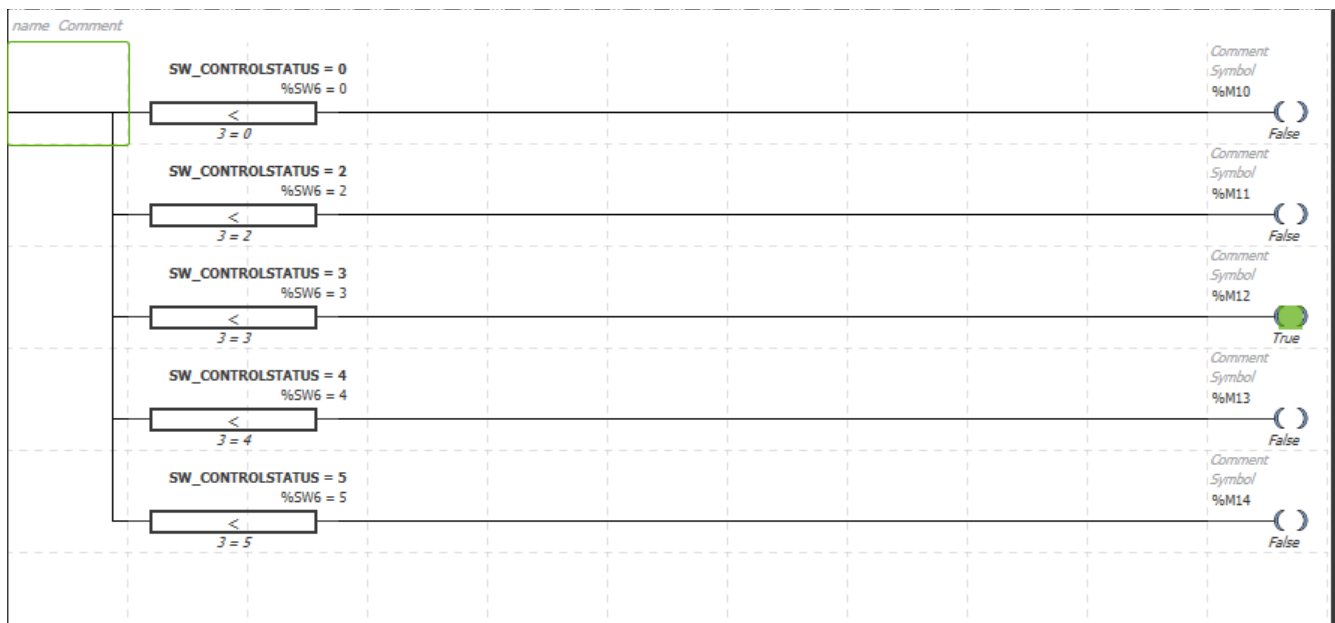
%KW dapat mentransfer nilai decimalnya ke memory type word, dan juga dapat digunakan untuk convert suatu nilai decimal ke dalam bentuk string atau sebaliknya.



System Word (%SW)

Sama seperti System Bit, Pada SoMachine terdapat juga System Word. Ini digunakan untuk menampilkan suatu keadaan PLC dan serta hal hal rinci seputar PLC yang dipakai.

Terdapat 212 nomor Word yang bisa digunakan. Contohnya adalah %SW6 untuk menampilkan kondisi state pada PLC



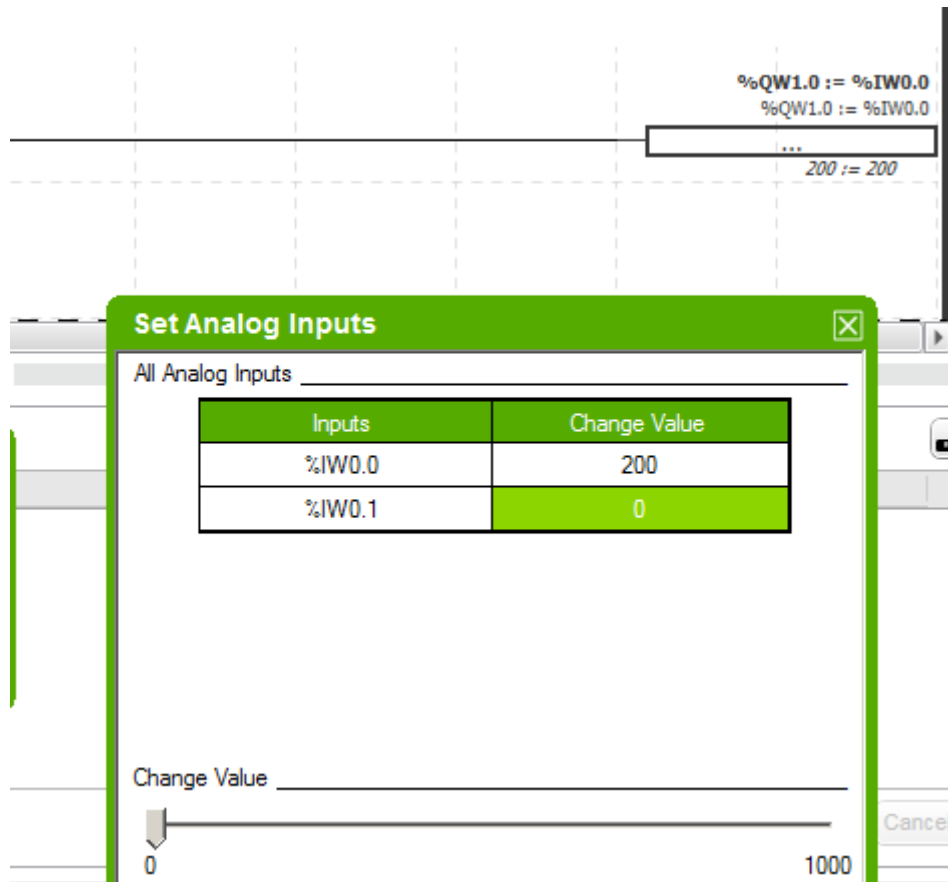
dibawah ini adalah contoh dari %SW / System Word
ini adalah keterangannya

%SW6	Controller state %MW60012	Controller state: 0 = EMPTY 2 = STOPPED 3 = RUNNING 4 = HALTED 5 = POWERLESS
-------------	------------------------------	---

I/O exchange words(%IW,%QW,%IWS,%QWS)

Ini adalah object word untuk analog input (%IW) dan analog output(%qw)
%IWS dan %QWS adalah word status untuk analog input dan analog output

dibawah ini adalah contoh move nilai dari %IW0.0 ke %QW1.0



3. Floating Point dan Double Words Object

Pada SoMachine terdapat object 32-bit diantaranya disimbolkan dengan %MF,%MD,%KF,%KD

%MF = Memory Floating

%MD = Memory Double

%KF = Konstanta Float

%KD = Konstanta Double

untuk object 32-bit ini terdapat overlap memory susunannya pada gambar di bawah ini jadi jika kita menggunakan %MF1 maka kita tidak bisa lagi menggunakan %MW1 dan %MW2 karena akan bertabrakan nilai datanya begitu juga pada %MD,%KD,%KF

Floating and Double	Odd Address	Memory Words
%MF0 / %MD0		%MW0
	%MF1 / %MD1	%MW1
%MF2 / %MD2		%MW2
	%MF3 / %MD3	%MW3
%MF4 / %MD4		%MW4
	...	%MW5
...		...
	%MF _i / %MD _i	%MW _i
%MF _i +1 / %MD _i +1		%MW _i +1

This table shows how floating and double constants overlap:

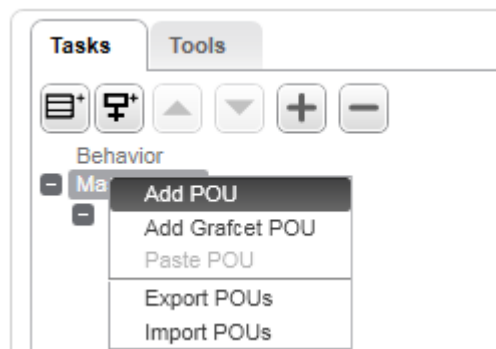
Floating and Double	Odd Address	Memory Words
%KF0 / %KD0		%KW0
	%KF1 / %KD1	%KW1
%KF2 / %KD2		%KW2
	%KF3 / %KD3	%KW3
%KF4 / %KD4		%KW4
	...	%KW5
...		...
		...

Task

Task pada SoMachine basic terdiri dari :

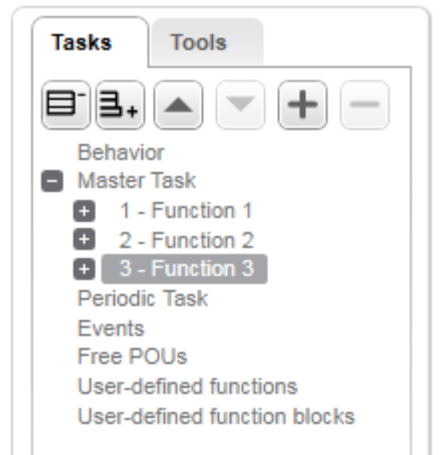
1.Master Task

Master Task adalah Organization Block (jika pada siemens) disini kita dapat membuat program dengan menambahkan SubRoutine/Function(siemens) pada MasterTask untuk menambahkannya langkahnya adalah sebagai berikut :



POU adalah Function versi SoMachine Basic disini dapat ditambahkan banyak function di Master Task,di dalam function kita bisa membuat sebuah program dengan nenambahkan rung/baris.

dibawah ini adalah contoh SubRoutine/Function pada SoMachine Basic

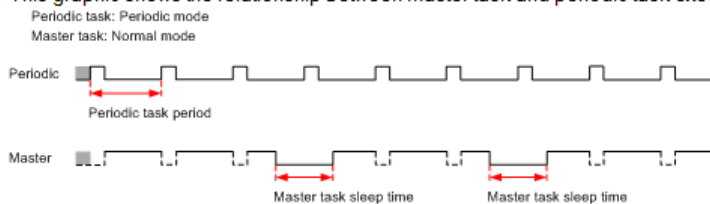


2.Periodical Task

Periodical Task sama seperti halnya Master Task yang membedakannya adalah jika pada cyclenya terhadap program.

Master Task in Normal Scan Mode

This graphic shows the relationship between master task and periodic task execution when the master task is configured in normal scan mode:



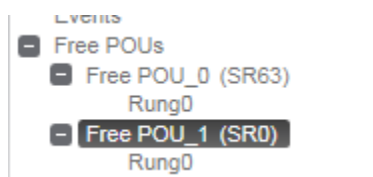
NOTE: The master task sleep time is at least 30% of the total cycle time with a minimum of 1 millisecond.

3.Event Task

ini biasa digunakan untuk memprogram High Speed Counter ,di sini bisa dibuat maksimum 4 task Event,event ini terhubung dengan digital input.

Free POU

FreePOU adalah subroutine yang bisa digunakan pada Periodical Task ataupun Event Task maksimumnya adalah 64 subroutine (SR0-SR63)



User Defined Function

Ini adalah sebuah block function yang dapat dipanggil di semua Subroutine/Function

User defined function ini dibuat menggunakan 3 buah jenis value

1.Return Value

digunakan untuk output function

2.Parameters

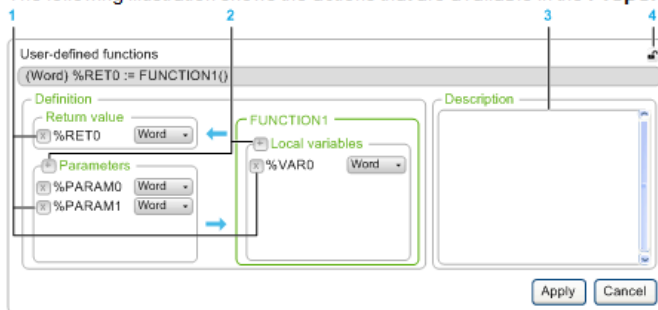
ini digunakan untuk memasukan inputan function

3.Local Variable

digunakan untuk menyimpan hasil dari parameters, bisa digunakan bisa tidak karena untuk program yang kecil mungkin tidak memerlukan sebuah local variable

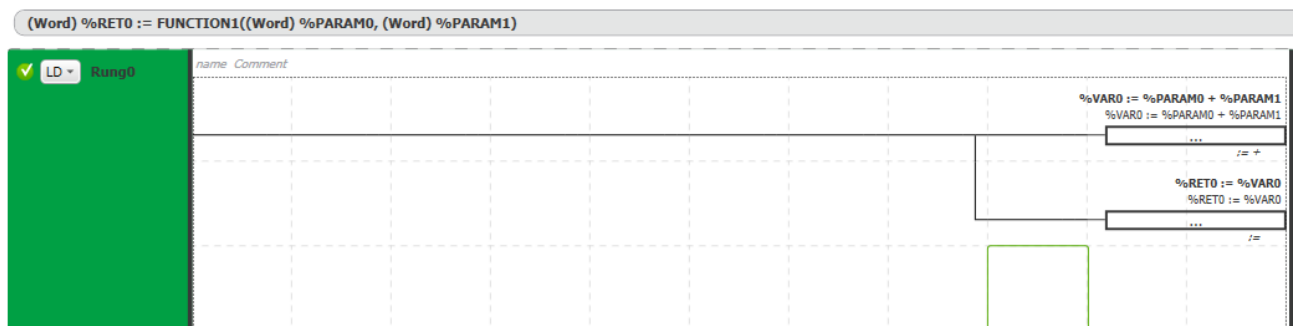
Presentation

The following illustration shows the actions that are available in the **Properties** view of the user-defined function:



- 1 Delete the **Return value**, an input **Parameter**, or a **Local variable**
- 2 Add a **Return value**, an input **Parameter**, or a **Local variable**
- 3 Optionally, write a description of the purpose of the user-defined function. This description appears in a tooltip when you use the user-defined function in an **Operation block**.
- 4 Detach the properties view

berikut ini adalah contoh dari user defined function



menambahkan input 1 dengan input 2 menyimpannya di var0 dan memindahkan ke Ret/Output



User defined function block

ini adalah sebuah function block yang hampir sama dengan user defined function perbedaannya pada user defined function tidak terdapat input output (boolean),serta cara memanggilnya juga berbeda

Pada User defined function block kita dapat membuat program dengan

1.Input

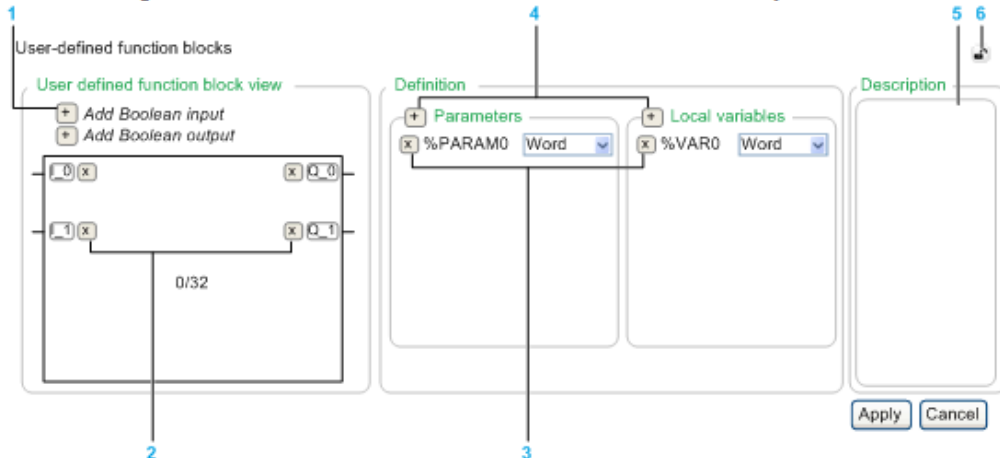
2.Output

3.Param

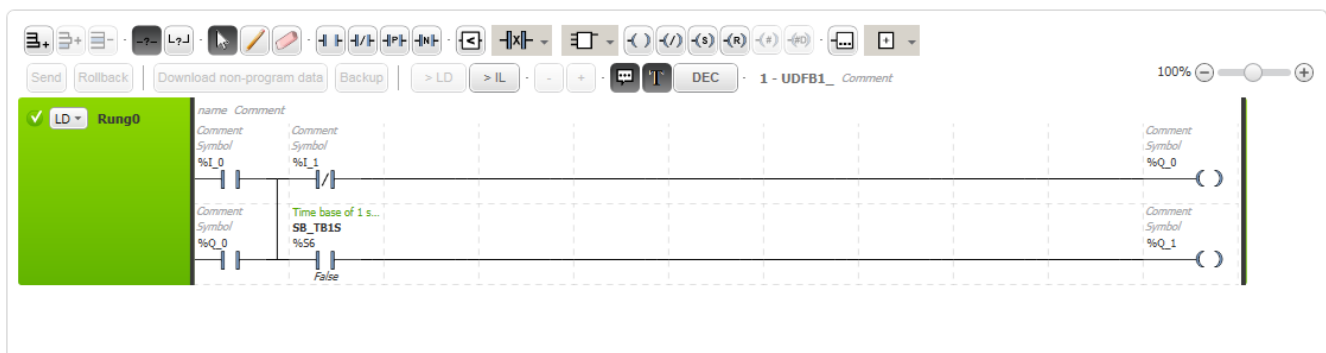
4.VaR(Local Variabel)

Presentation

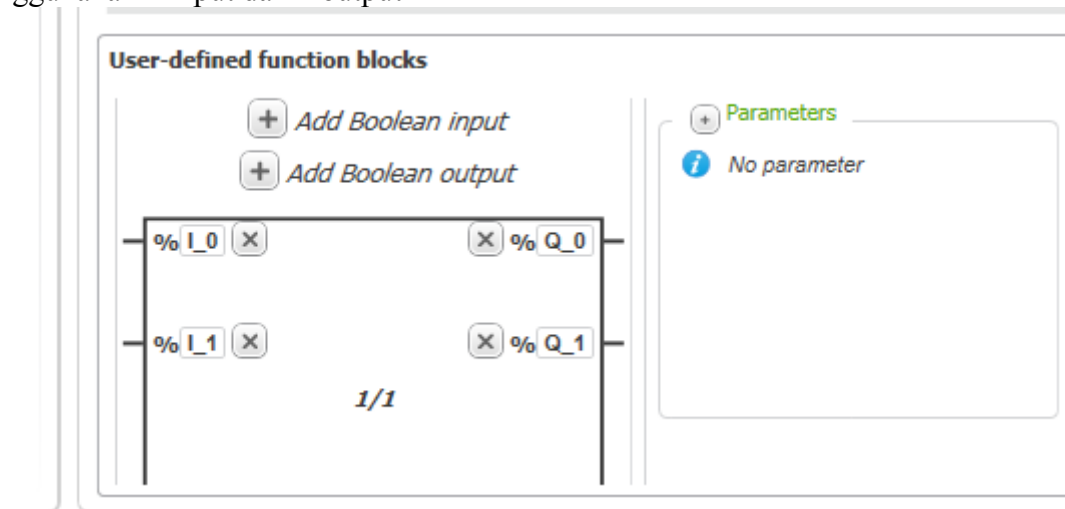
The following illustration shows the actions that are available in the **Properties** view of the user-defined function:



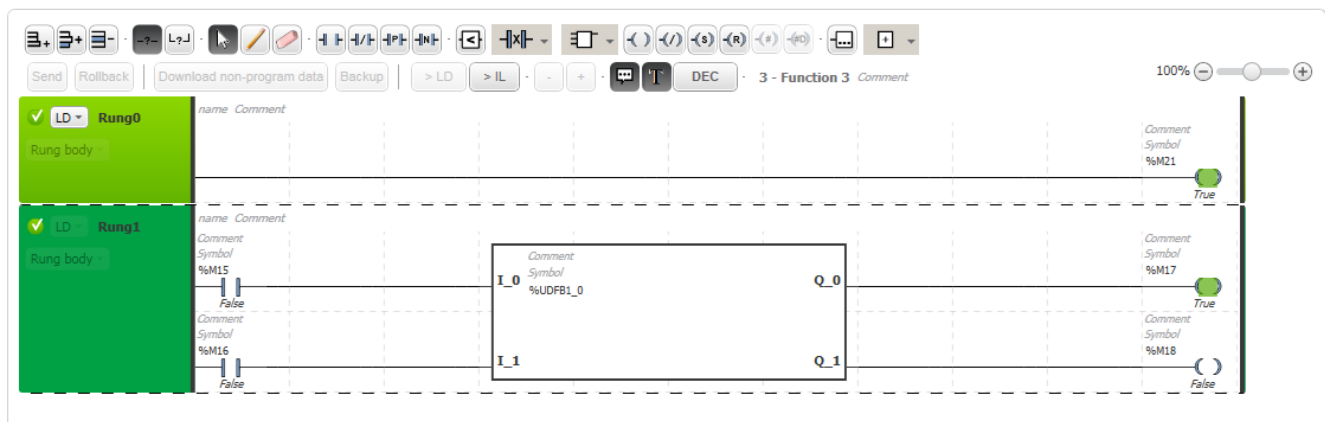
berikut ini adalah contoh program dari user defined function block



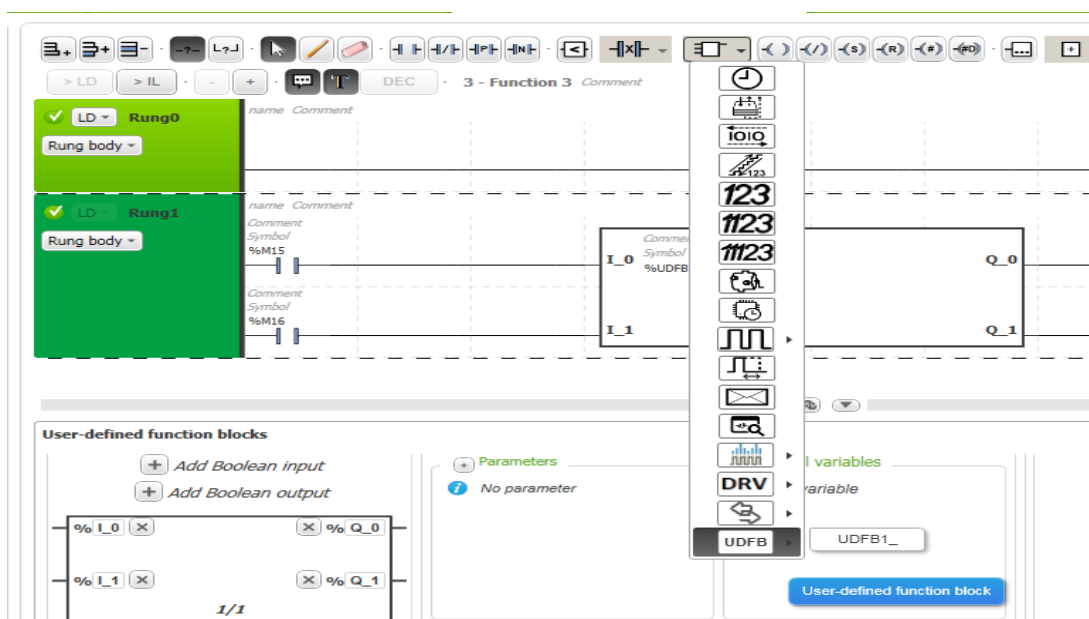
disini menggunakan 2 input dan 2 output



ini tampilan ketika dipanggil pada rung



Cara memanggil User defined Function Block



Kesimpulan :

1. Pada SoMachine Basic untuk memprogram dengan type data bit dapat menggunakan %M,%S,%X1(Grafcet)
2. Untuk memprogram menggunakan data type Word dapat menggunakan %MW,%KW,%SW,%IW,%QW
3. Untuk memprogram menggunakan data type Float atau Double Word dapat menggunakan %MD,%MF,%KD,%KF,dengan memperhatikan overlapnya.
4. Di SoMachine Basic terdapat tempat untuk memprogram High Speed Counter (HSC) dengan Event Task yang Subrutinanya mengambil dari Free POU.
5. Di SoMachine Basic bisa membuat Function atau Function Block dan juga menggunakan Parameters serta Local Variabel yang ada di dalamnya.
6. Free POU pada SoMachine Basic dapat digunakan untuk membuat Subroutine(SR) pada Periodical Task dan Event Task ,Maksimal Free POU 64 (SR0-SR63).
7. Untuk menyimpan data dan pada SoMachine dapat menggunakan data %KW,%KD,%KF
8. Untuk memasukan string/ascii dengan mudah bisa menggunakan %KW