

DR. BOB DAVIDOV

Генерация кода Программируемых Логических Контроллеров (ПЛК) в среде МатЛАБ, PLC Coder

Цель работы: освоение правил создания программного кода для ПЛК в среде МатЛАБ.

Задача работы: Трансляция модели Simulink в код ПЛК (PLC) и перевод кода PLC в m-файл для моделирования.

Приборы и принадлежности: Персональный компьютер, МатЛАБ (2010а или старше), Simulink PLC Coder.

ОБЩИЕ СВЕДЕНИЯ

Компания MathWorks разработала Simulink PLC Coder. Он позволяет автоматически генерировать код для Программируемых Логических Контроллеров (**ПЛК, PLC**) и программируемых контроллеров автоматизации согласно стандарту IEC 61131. Это позволяет использовать модельно-ориентированное проектирование для промышленного и силового оборудования, управляемого ПЛК.

С помощью Simulink PLC Coder, инженеры могут автоматически генерировать код для промышленных систем управления, включая замкнутые системы и системы контроля с обратной связью. Автоматическая генерация кода, неотъемлемая часть Модельно-ориентированного проектирования, помогает устранить ошибки, связанные с традиционным ручным написанием кода и уменьшает время разработки и валидации.

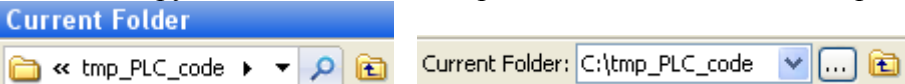
Simulink PLC Coder генерирует исходный код в структурированном текстовом формате из моделей Simulink, диаграмм Stateflow, и кода Embedded MATLAB, а затем использует интегрированную среду разработки (IDE) от поставщика средств промышленной автоматизации, для компиляции кода и запуска его на контроллерах ПЛК.

Для генерации ПЛК кода необходим пакет МатЛАБ начиная с версии 2010а. Ниже, на примерах, показан порядок разработки программы для ПЛК и перевод PLC кода в m-функцию для ее работы в составе Simulink модели.

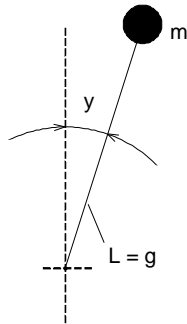
ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Задание 1. Трансляция модели Simulink в код PLC

1. Создайте рабочий каталог, загрузите МатЛАБ и настройте каталог МатЛАБ на рабочий

каталог, например, 

2. Запустите библиотеку **plclib**, или **Simulink** из командной строки Matlab.
3. Создайте файл модели в нем модель системы стабилизации положения маятника в верхнем положении [1] (см. Рис. 1).



Малые отклонения маятника от верхнего положения равновесия описываются дифференциальным уравнением второго порядка

$$(p^2-1)y(t) = u(t),$$

где $u(t)$ – управляющее воздействие.

Рис. 1. Маятник. Верхнее положение маятника является примером неустойчивого состояния.

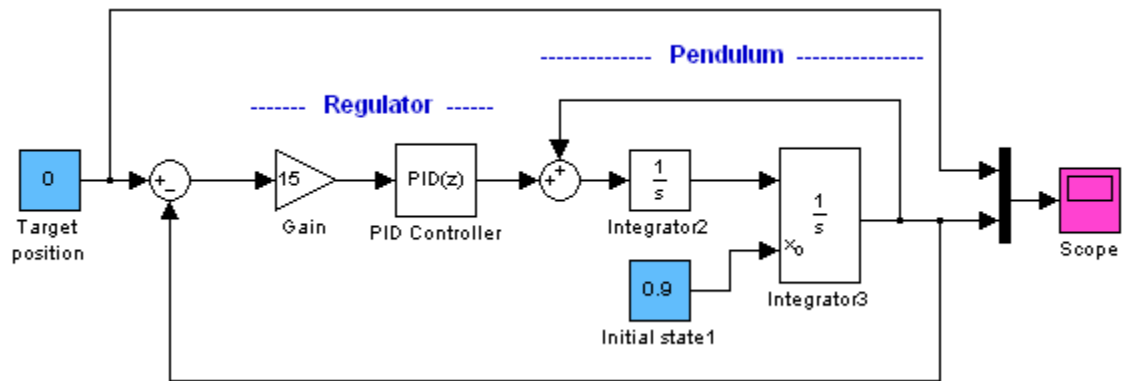
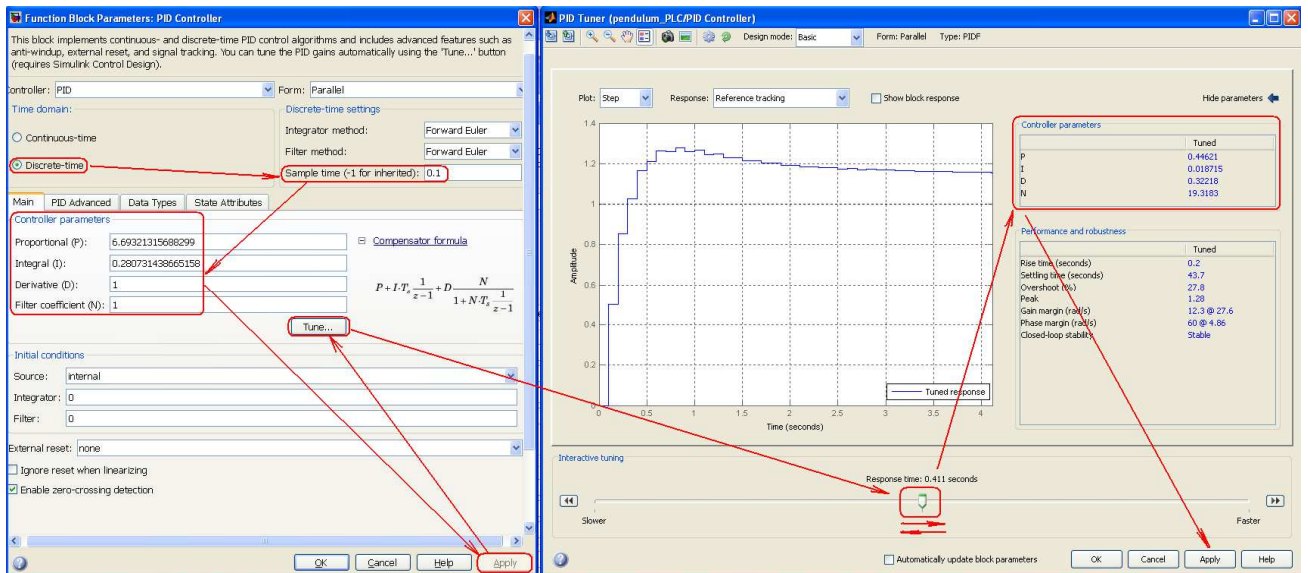


Рис. 2. Модель стабилизации маятника в верхнем положении. Возьмите блок **PID controller** в разделе **Simulink > Continuous >**

PID(s) переведите его из непрерывной **s** -

формы в дискретную форму **z** используя переключатель **Discrete-time**.

4. Задайте период дискретизации PID контроллера по времени 0.1 секунды **Sample time (-1 for inherited): 0.1**.
5. Выберите тип регулятора в поле **Controller**. Задайте (при необходимости) начальные параметры регулятора. Нажмите клавишу **Apply**, затем **Tune**. Перемещая движок под зоной графика и наблюдая за изменением параметров регулятора, настройте переходный процесс. Нажатием клавиши **Apply** перепишите параметры регулятора из зоны настройки **Controller Parameters** в поле **Main** окна **Function Block Parameters**.



6. Запустите модель, постройте переходный процесс.

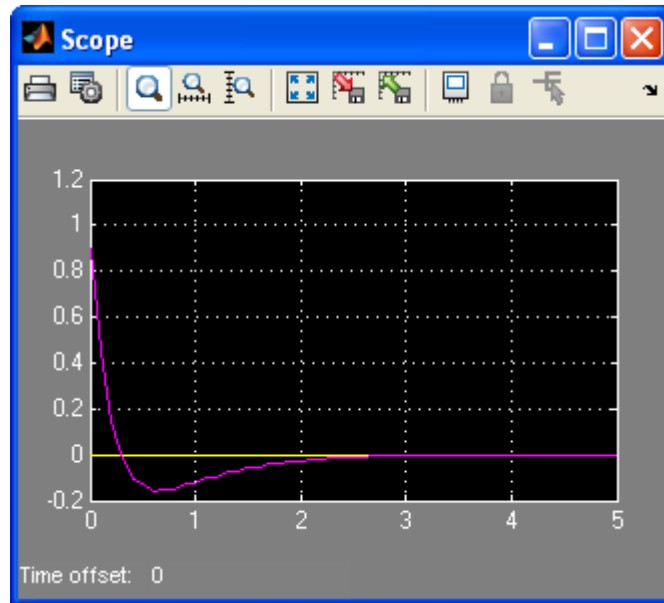
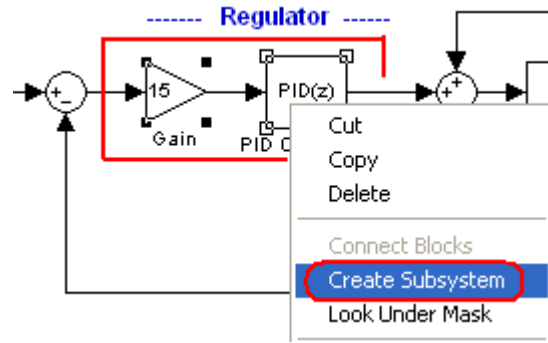
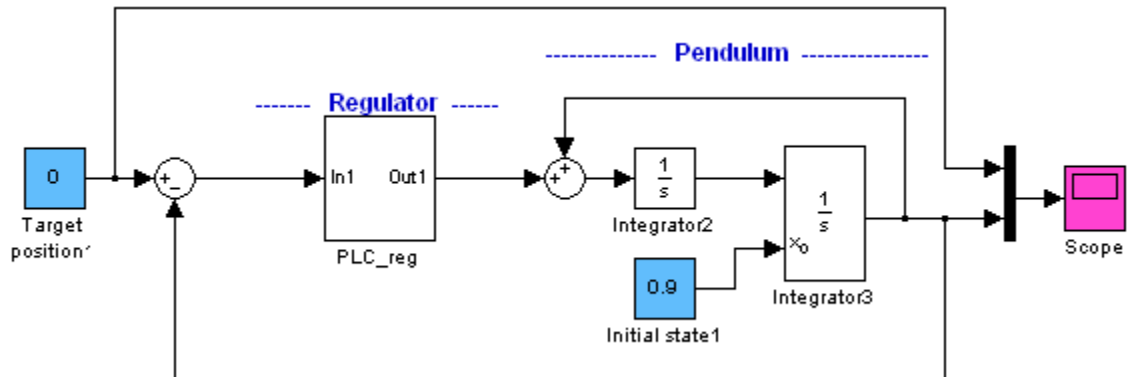


Рис. 3. Стабилизация маятника в верхнем (нулевом) положении.

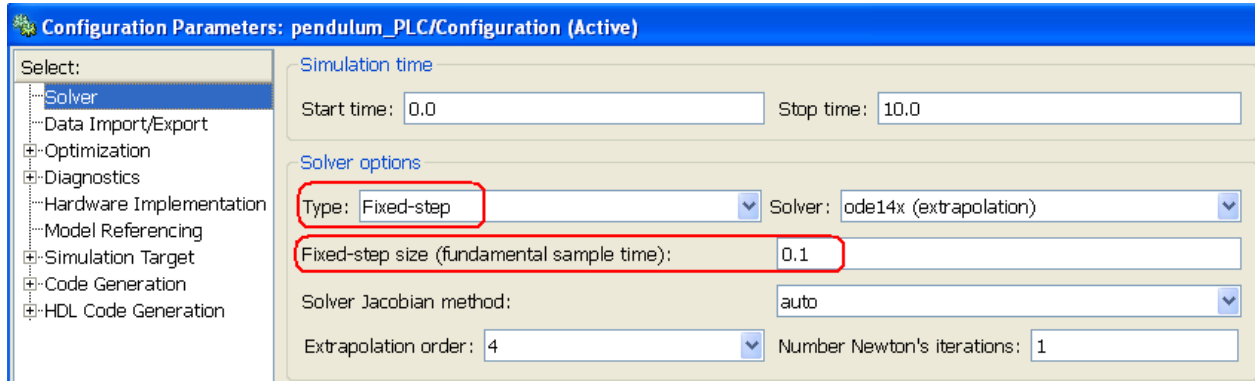
7. Выделите блоки регулятора функцию которых будет выполнять ПЛК (PLC) и постройте из них блок подсистемы (Subsystem).



8. Дайте название подсистеме, например, **PLC_reg**.



9. Настройте



10. Выберите блок **PLC_reg**, нажмите на него правой клавишей мыши (ПКМ) и выберите **Subsystem Params**. В появившемся окне выделите **Treat as atomic unit** и нажмите ОК.

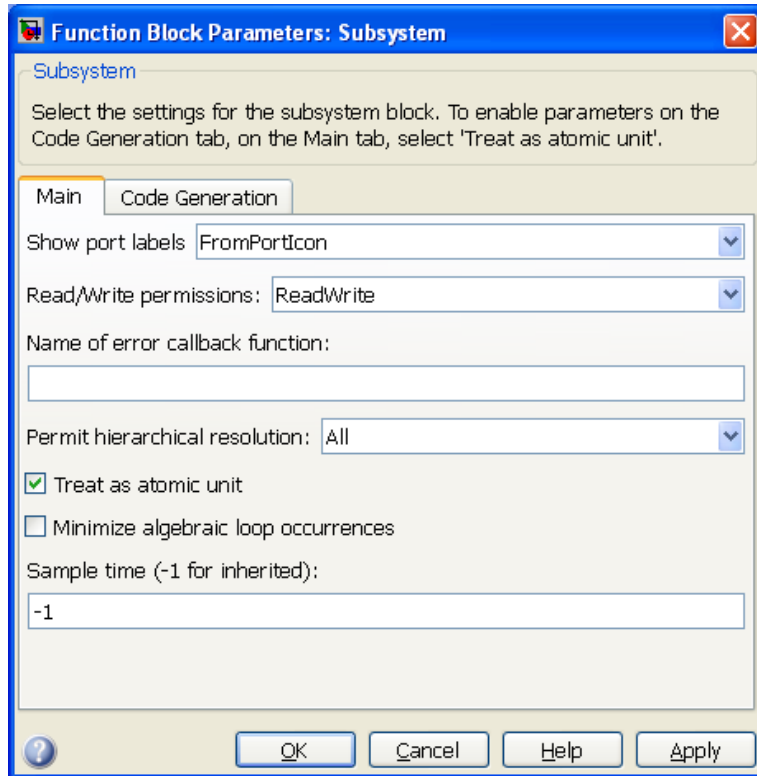
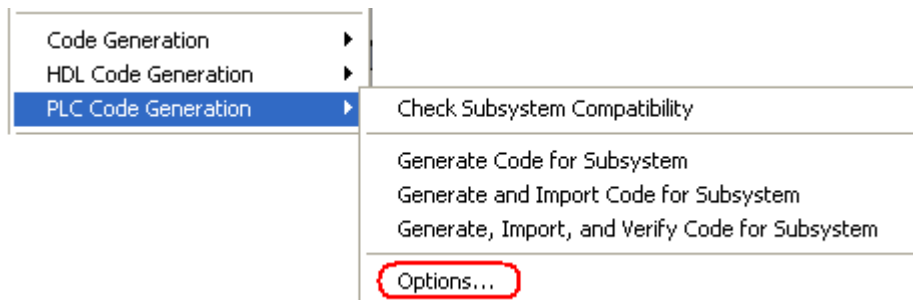
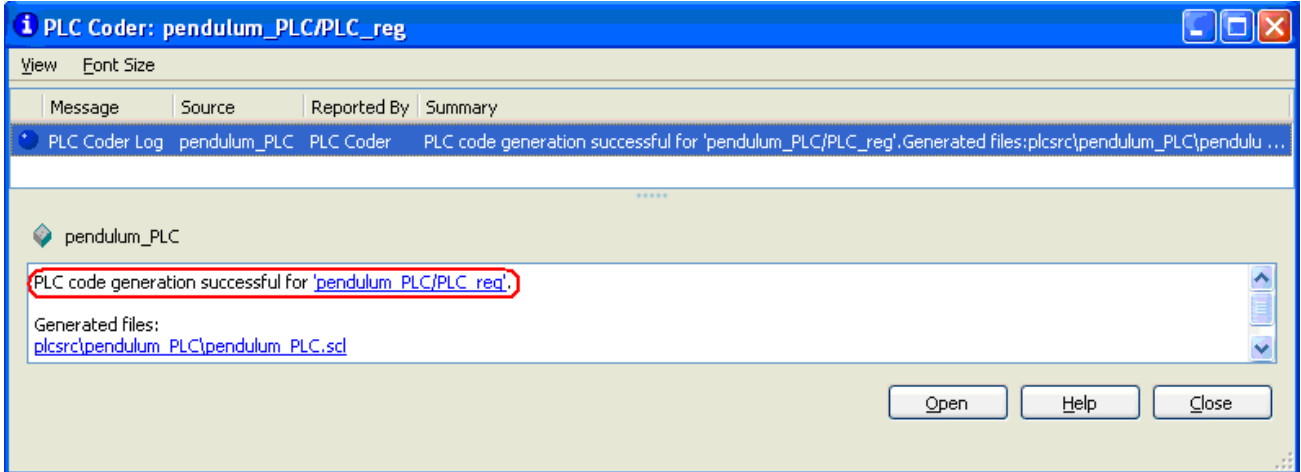
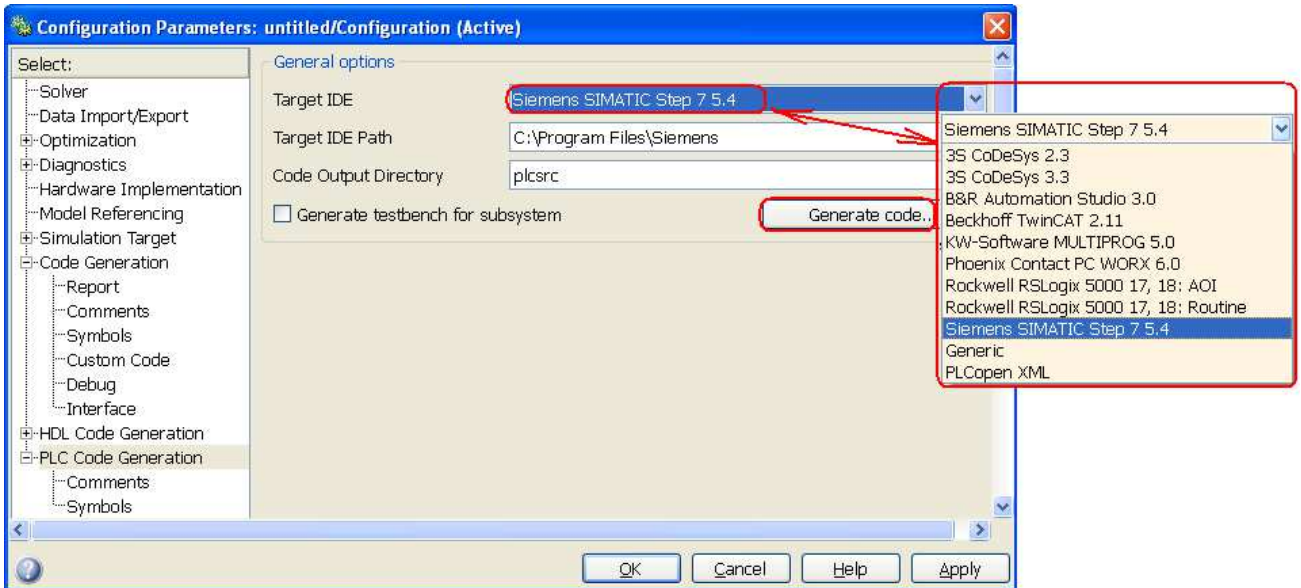


Рис. 4. Параметры блока pid_feedforward

11. Нажав ПКМ на блок **PID_reg** выберите **PLC Code Generation > Options**.



12. В разделе **PLC Code Generation > Target IDE** выберите тип PLC для которого необходимо создать код (**IDE**), например, Siemens SIMATIC Step 7 5.4 и нажмите клавишу **Generate code**.



13. После успешной генерации кода (PLC code generation successful) проверьте в рабочем каталоге наличие созданного кода для SIMATIC STEP 7

Name	Ext	Size
[..]		<DIR>
pendulum_PLC	scl	2,355

Таблица 1. Код регулятора Simulink модели системы стабилизации маятника.

```
(*
*
* File: pendulum_PLC.scl
*
* IEC 61131-3 Structured Text (ST) code generated for subsystem "pendulum_PLC/PLC_reg"
```

```
*  
* Model name           : pendulum_PLC  
* Model version        : 1.6  
* Model creator        : v_d  
* Model last modified by : Bob  
* Model last modified on : Tue Nov 26 16:22:42 2013  
* Model sample time    : 0.1s  
* Subsystem name       : pendulum_PLC/PLC_reg  
* Subsystem sample time : 0.1s  
* Simulink PLC Coder version : 1.3 (R2012a) 29-Dec-2011  
* ST code generated on  : Tue Nov 26 18:11:40 2013
```

```
*  
* Target IDE selection : Siemens SIMATIC Step 7 5.4  
* Test Bench included  : No
```

```
*
```

```
*)
```

```
FUNCTION_BLOCK PLC_reg
```

```
VAR_INPUT
```

```
    ssMethodType: INT;
```

```
    In1: REAL;
```

```
END_VAR
```

```
VAR_OUTPUT
```

```
    Out1: REAL;
```

```
END_VAR
```

```
VAR
```

```
    Integrator_DSTATE: REAL;
```

Filter_DSTATE: REAL;

rtb_Gain: REAL;

c_rtb_FilterCoeffic: REAL;

END_VAR

CASE ssMethodType OF

0:

(* InitializeConditions for DiscreteIntegrator: '<S2>/Integrator' *)

Integrator_DSTATE := 0.0;

(* InitializeConditions for DiscreteIntegrator: '<S2>/Filter' *)

Filter_DSTATE := 0.0;

1:

(* Gain: '<S1>/Gain' incorporates:

* Inport: '<Root>/In1' *)

rtb_Gain := 15.0 * In1;

(* Gain: '<S2>/Filter Coefficient' incorporates:

* DiscreteIntegrator: '<S2>/Filter'

* Gain: '<S2>/Derivative Gain'

* Sum: '<S2>/SumD' *)

c_rtb_FilterCoeffic := ((0.322182906257543 * rtb_Gain) - Filter_DSTATE) * 19.3182692452043;

(* Outport: '<Root>/Out1' incorporates:

* DiscreteIntegrator: '<S2>/Integrator'


```

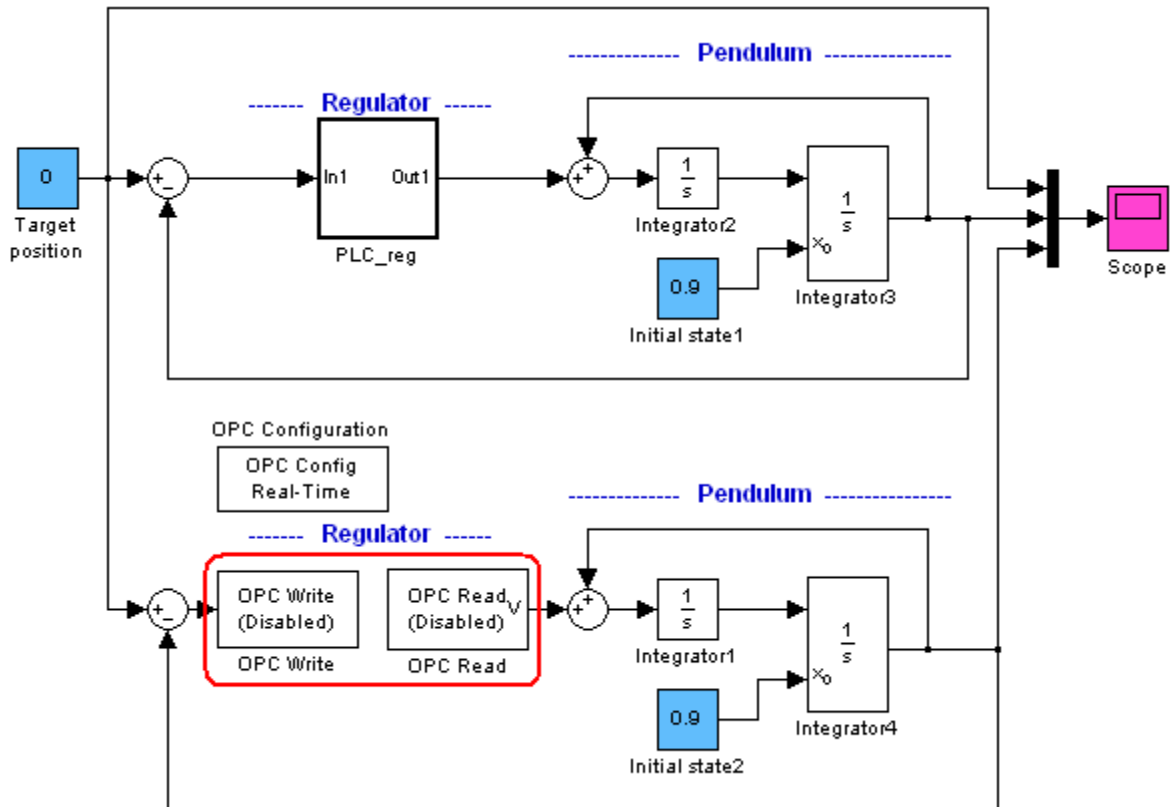
* Gain: '<S2>/Proportional Gain'
* Sum: '<S2>/Sum' *)
Out1 := ((0.446214210458868 * rtb_Gain) + Integrator_DSTATE) + c_rtb_FilterCoeff;

(* Update for DiscreteIntegrator: '<S2>/Integrator' incorporates:
* Gain: '<S2>/Integral Gain' *)
Integrator_DSTATE := ((0.0187154292443497 * rtb_Gain) * 0.1) + Integrator_DSTATE;

(* Update for DiscreteIntegrator: '<S2>/Filter' *)
Filter_DSTATE := (0.1 * c_rtb_FilterCoeff) + Filter_DSTATE;
END_CASE;
END_FUNCTION_BLOCK

```

14. Создайте проект в Siemens SIMATIC Step 7 и вставьте в редактор SCL созданный scl код. Результатом компиляции кода является создание функционального блока.
15. Для сравнения работы моделей контроллера Simulink и Siemens продублируйте модель стабилизации маятника в Simulink и дополните ее блоками OPC сервера.



16. Через MatLAB OPC сервер настройте связь Simulink модели с моделью или реальным контроллером Simens.

Примечание: При использовании реального ПЛК задействуйте Simatic Net OPC Server, при использовании симулятора ПЛК (PLC-Sim) необходимо применить SCADA-систему WinCC и WinCC OPC Server.

17. Постройте и сравните переходные процессы стабилизации виртуального маятника под управлением контроллера и его модели.

Задание 2. Трансляция PLC кода в m-файл модели Simulink.

1. Выделите в PLC коде (см. Таблица 1) строки с функциональными зависимостями.
2. Напишите m-функцию для контроллера с одним входом и одним выходом и с дополнительными входами и выходами для сохранения и использования предыдущих значений интегратора и фильтра.

```
function [Out1, Integrator_DSTATE_out, Filter_DSTATE_out] = fcn(In1, Integrator_DSTATE, Filter_DSTATE)
```

```
rtb_Gain = 15.0 * In1;
c_rtb_FilterCoeff = ((0.322182906257543 * rtb_Gain) - Filter_DSTATE) * 19.3182692452043;
Out1 = ((0.446214210458868 * rtb_Gain) + Integrator_DSTATE) + c_rtb_FilterCoeff;
Integrator_DSTATE_out = ((0.0187154292443497 * rtb_Gain) * 0.1) + Integrator_DSTATE;
Filter_DSTATE_out = (0.1 * c_rtb_FilterCoeff) + Filter_DSTATE;
```

3. Вставьте m-функцию в соответствующий блок Simulink модели (**MATLAB Function**).

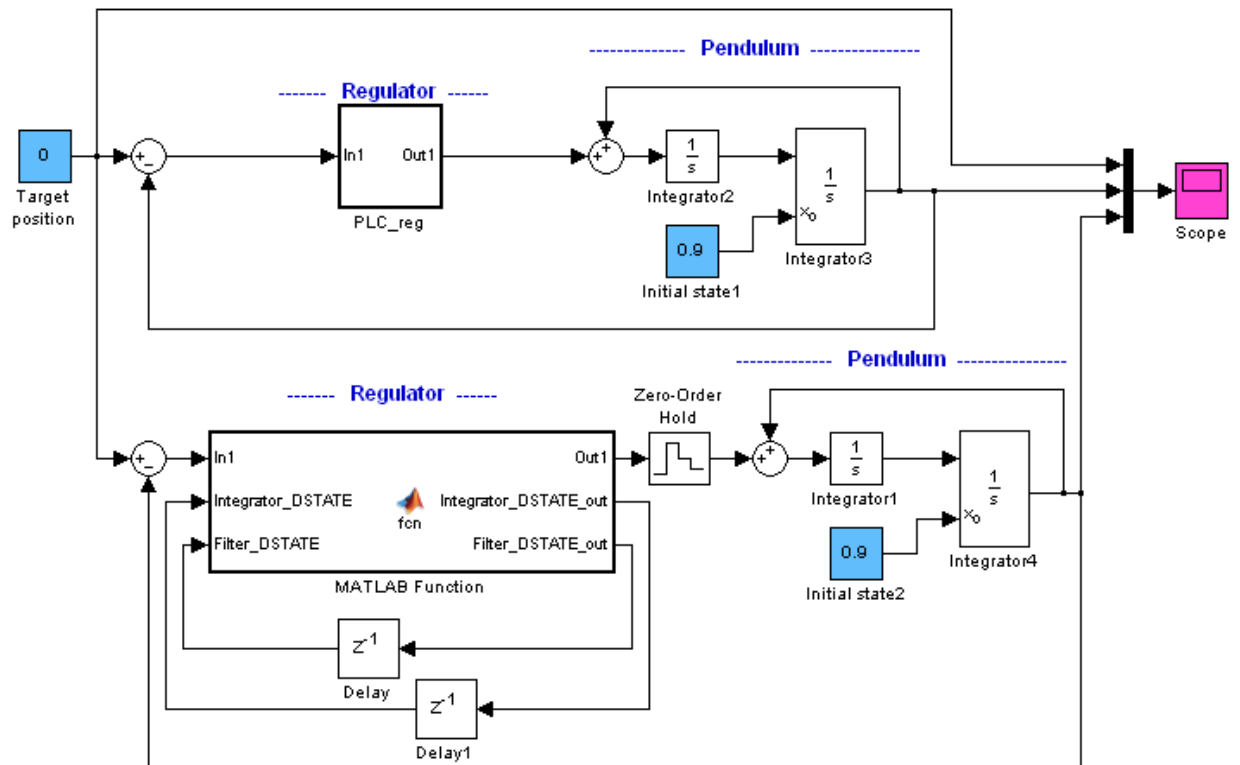


Рис. 5. Модель с МатЛАБ функцией, выделенной из PLC кода.

4. На выходе регулятора поставьте фиксатор нулевого порядка (**Zero-Order Hold**).
5. Сравните переходные процессы исходного виртуального регулятора и регулятора, функция которого выделена из PLC кода.

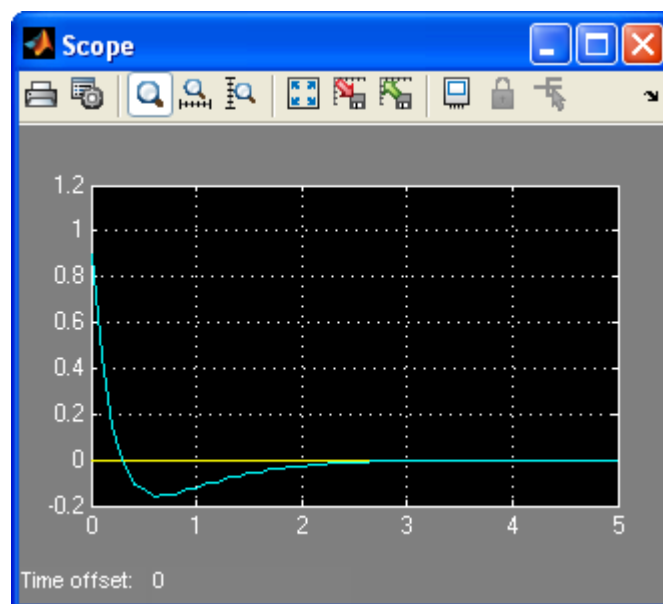


Рис. 6. Пример совпадения переходных процессов стабилизации маятника под управлением МатЛАБ регулятора и кода PLC контроллера.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие преимущества дает трансляция модели Simulink в PLC код?
2. Как можно обеспечить оптимальную настройку ПИД регулятора в МатЛАБ?
3. Для чего нужен фиксатор нулевого порядка в модели Рис. 5.?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. А.А. Алексеев, Д.Х. Имаев, Н.Н. Кузьмин, В.Б. Яковлев. Теория управления: Учеб./ СПб.: Изд-во СПбГЭТУ “ЛЭТИ”, 1999.
2. Dr. Bob Davidov. Компьютерные технологии управления в технических системах
<http://portalnp.ru/author/bobdavidov>.