**Лекция 5. Системы с цифровым управлением непрерывными объектами**

Задачи синтеза дискретных систем управления имеют те же постановки, что и соответствующие задачи синтеза систем по непрерывным моделям. Вместе с тем, необходимо учитывать особенности моделей систем цифрового управления непрерывными объектами.

Дискретизация времени и квантование уровней сигналов в большинстве современных систем управления обусловлена применением цифровых управляющих устройств. На рис. 1 изображена принципиальная схема системы цифрового управления непрерывным объектом. Выделены аналого-цифровой (АЦП) и цифроаналоговый (ЦАП) преобразователи, реализующие операции дискретизации и континуализации (операции обратные квантованию) сигналов.

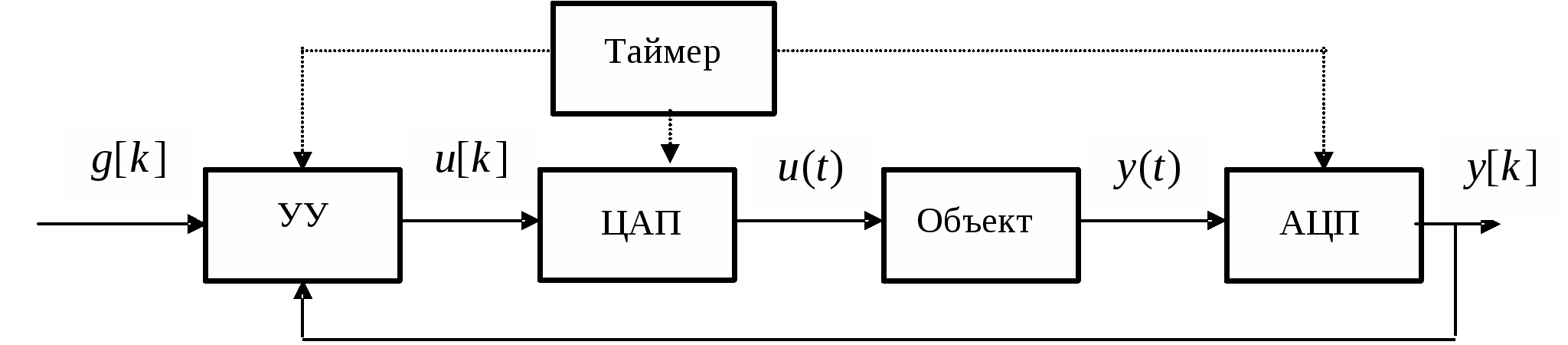


Рис. 1. Схема системы цифрового управления непрерывным объектом

Аналого-цифровые преобразователи содержат квантователи уровня. Число уровней определяется длиной машинного слова. Если число уровней велико, то квантованием уровня в большинстве случаев моделирования ***можно пренебречь***.

Рассмотрим задачу построения математических моделей отдельных элементов и системы в целом.

Динамика непрерывного объекта описывается дифференциальными уравнениями. Если ограничиться классом линейных стационарных моделей с сосредоточенными параметрами, объект будет описан обыкновенными линейными дифференциальными уравнениям с постоянными коэффициентами или в форме передаточной функции *W(s),* равной отношению изображений по Лапласу переменных выхода *Y(s)* и входа *U(s)* объекта при нулевых начальных условиях.

Пусть цифровое управляющее устройство УУ реализует алгоритм управления, описываемый в виде дискретной передаточной функции *R(z)*, равной отношению  *Z*-изображений *U(z)* и *E(z),* где *E(z)* − изображение ошибки системы.

Таким образом, модель системы (см. рис. 2) оказывается неоднородной (гибридной), так как она образована разнородными элементами, а переносимая между ними информация кодируется различными типами сигналов: аналоговыми − *u*t, *y*t и с дискретным временем − *u*k, *y*k, *e*k. Разнородные элементы взаимодействуют между собой с помощью АЦП и ЦАП (см. рис. 1).

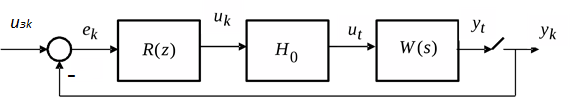


Рис. 2. Структурная схема системы цифрового управления

Пренебрегая эффектом квантования уровня, примем, что моделью АЦП является так называемый «ключ», который периодически замыкается на пренебрежимо малое (по сравнению со скоростью изменения переменной *y(t)*) время (рис. 3, *а*). Период замыкания обозначают *Ts* (*англ*. – sampling time). Ключ позволяет получать информацию о переменной ошибки *e(t)* в равноотстоящие моменты времени *ek = u*з*k - yk* = *e[k] = e(kTs); t = 0, Ts, 2Ts,…* .

*а) б)*

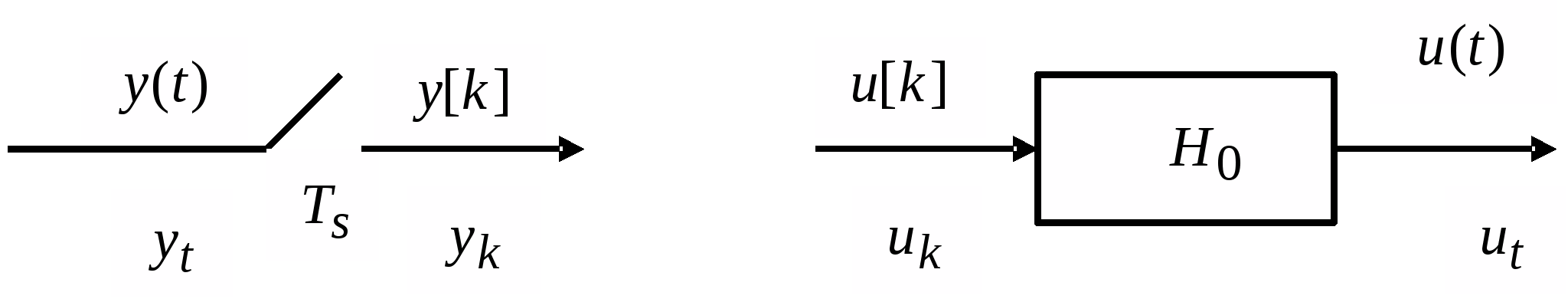


Рис. 3. Примеры графических изображений: дискретизатор времени непрерывного сигнала («ключ») (а); формирующее устройство (б)

ЦАП представляет собой *экстраполятор*или формирующее устройство *H*0(рис. 3, б). Его назначение − по определенному закону предсказывать значения функции *u(t)* до поступления на вход новой информации. Простейший и часто применяемый тип формирующего устройства − экстраполятор нулевого порядка или *фиксатор* сохраняет постоянное значение сигнала:

*u(t) = u[k]; kTs ≤ t < (k+1)Ts .*

Сигнал на выходе фиксатора *H*0 (см. рис. 3, *б*) имеет вид ступенчатой функции непрерывного времени *u(t)*.

Таким образом, дискретизацию времени осуществляет «ключ», а континуализацию — фиксирующее устройство (экстраполятор), условные изображения которых приведены на рис. 3. Эти элементы имеют дополнительные входы для сигналов синхронизации от таймера (см. рис. 1); на рис. 3, *а* рядом с ключом записан символ *ТS* – период замыкания ключа.

**Способы синтеза алгоритмов цифрового управления**

Алгоритмы цифрового управления непрерывными объектами можно

синтезировать двумя способами, как это иллюстрируется на рис. 4:

* дискретизация предварительно синтезированного аналогового регулятора;
* синтез дискретного регулятора по дискретной модели объекта.

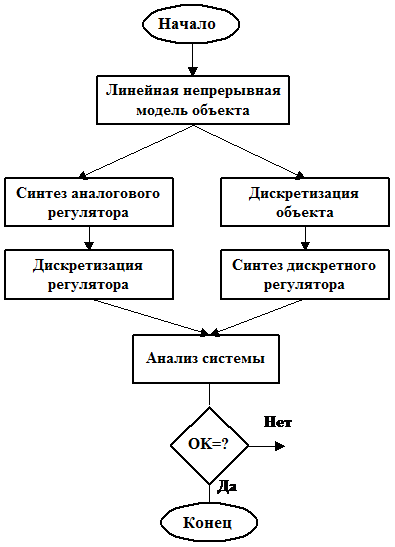


Рис. 4. Алгоритмы цифрового управления непрерывными объектами

Переход от непрерывной модели регулятора (левая ветвь алгоритма) или от непрерывной модели объекта управления (правая ветвь алгоритма) к эквивалентной дискретной модели любого подкласса, как было рассмотрено ранее, осуществляется с помощью функции MATLAB **c2d**. В то же время в среде Simulink имеются и другие возможности выполнения подобных преобразований, которые, впрочем, также базируются на использовании функции **c2d** в неявной форме.

Рассмотрим кратко основные методики синтеза цифровых алгоритмов управления на примере синтеза модальных и ПИД-регуляторов.

**Синтез цифрового модального регулятора**

Пусть имеется система, дифференциальные уравнения которой представлены в форме пространства состояний:

. (5.1)

Ее решение имеет вид:

. (5.2)

Так как на входе непрерывной системы стоит экстраполятор нулевого порядка, входная переменная остается постоянной от момента *kTs* до момента *(k+1)Ts*.  Примем за начало и конец отсчета моменты *kTs* и *(k+1)Ts*; тогда решение (5.2) запишется так:

.

После замены переменной интегрирования



получим разностное уравнение для дискретной системы в форме пространства состояний:

,

где:

****Как известно, функция от квадратной матрицы представляет собой матрицу того же размера. Ее собственные значения связаны с собственными значениями матрицы-аргумента той же функциональной зависимостью. Следовательно, собственные значения  матрицы состояний дискретной системы  связаны с собственными значениями матрицы непрерывной (5.3)  
 системы так:

На рис. 5 показаны плоскости собственных значений непрерывной и дискретной систем.

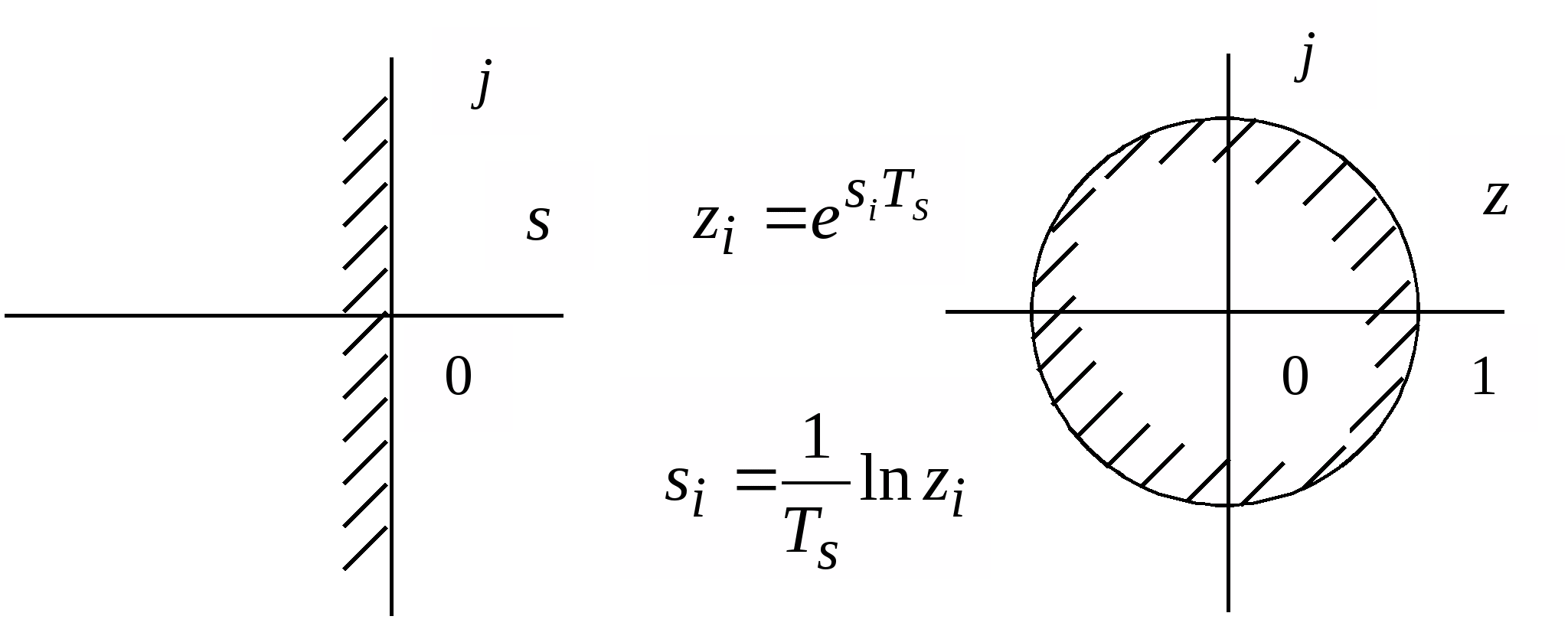


Рис. 5. Плоскости собственных значений непрерывной (*si*) и дискретной (*zi*) систем

По непрерывной стандартной форме можно сформировать соответствующую ей дискретную стандартную форму в соответствии с уравнением

*\*(z) = (z−.*

Можно также использовать непрерывную стандартную форму для получения вектора желаемых полюсов в *s*-плоскости, после чего получить вектор желаемых полюсов в плоскости *z*, используя выражение (5.3).

На основе этой методики достаточно просто можно синтезировать модальный регулятор для дискретной системы, используя рассмотренный ранее подход, на основе стандартных форм.

**Переоборудование регуляторов непрерывных систем в MATLAB**

На средствах вычислительной техники нельзя непосредственно реализовать управляющее устройство, которое описывается дифференциальными уравнениями в непрерывном времени. Задача переоборудования состоит в том, чтобы заменить спроектированный непрерывный регулятор его дискретным эквивалентом так, чтобы сохранить все существенные свойства непрерывной системы (устойчивость, качество, робастность).

Приведем простой пример. Пусть имеется непрерывная модель проектируемой системы **discretizer.slx** (см. рис. 6) и перед разработчиком стоит задача перевести какой-либо блок, например, блок регулятора (Regulator), в его дискретное представление.

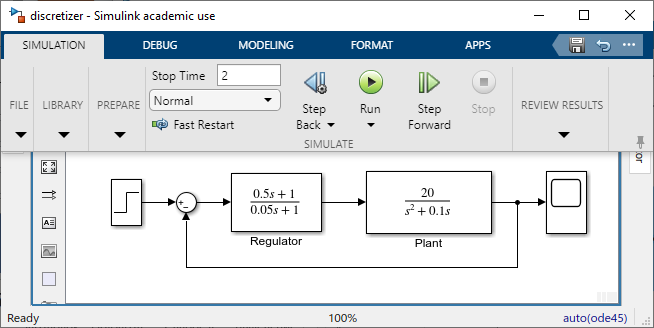


Рис. 6. Модель непрерывной системы

Для этого откроем вкладку APPS, развернем ее и выберем из библиотеки Control Systems блок Model Discretizer, как это показано на рис. 7. После этого, в открывшемся окне Simulink Model Discretizer из перечисленных блоков, входящих в модель системы управления, выберем блок Regulator, установим в строке Sample time требуемый период дискретности (T=0.02 с в нашем случае), а метод дискретизации оставим Zero-order hold (он установлен по умолчанию). После этого следует нажать клавишу S→Z. В результате на схеме модели в блоке регулятора появится дополнительная надпись zoh, что свидетельствует о выполненном преобразовании блока регулятора в дискретную форму. Описанные выше действия иллюстрируются рис. 8.

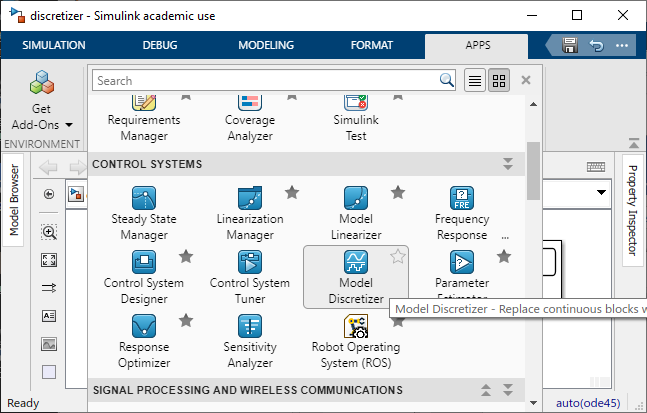
****

Рис. 7. Выбор блока Model Discretizer

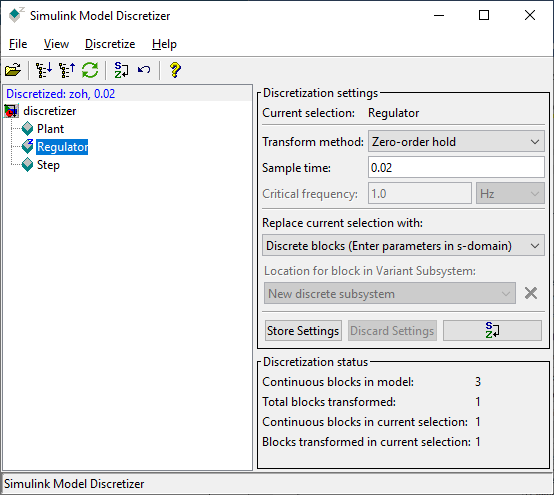
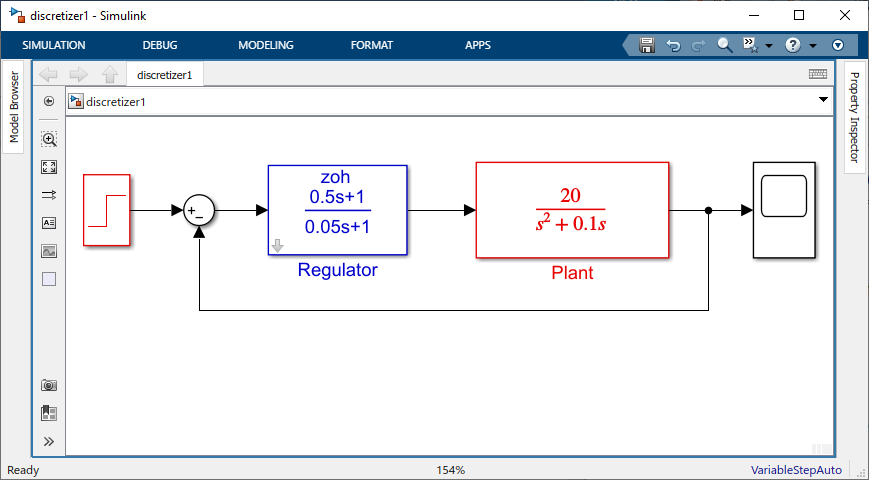
 

Рис. 8. Выбор блока Regulator непрерывной модели, и его дискретизация

В результате мы получили непрерывно-дискретную модель системы, содержащую дискретную передаточную функцию цифрового регулятора. Степень адекватности непрерывной (файл **discretizer.slx)** и непрерывно-дискретной (файл **discretizer1.slx**) модели проверяется моделированием. При этом, открыв параметры блока Regulator, можно изменять как период дискретизации, так и метод преобразования. Убедитесь в этом, установив значение T=0.01 c.

Описанные выше действия можно выполнить, введя в командное окно MATLAB команду

>> **slmdldiscui('discretizer')**

При необходимости, описанным выше способом можно преобразовать к дискретной форме и остальные блоки, входящие в Simulink-модель и перейти к полностью цифровой системе управления.

В качестве альтернативы можно заменить непрерывные блоки в модели Simulink эквивалентными блоками, дискретизированными в s-области, используя библиотеку Discretizing. Доступ к этой библиотеке осуществляется с помощью команды **discretizing**, вводимой **в командной строке** MATLAB.

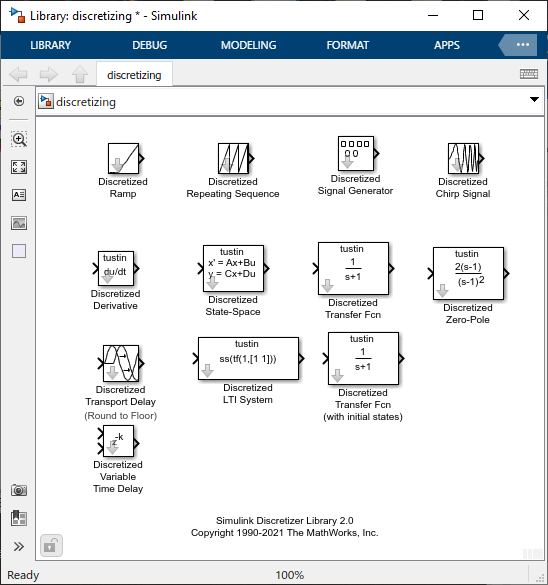


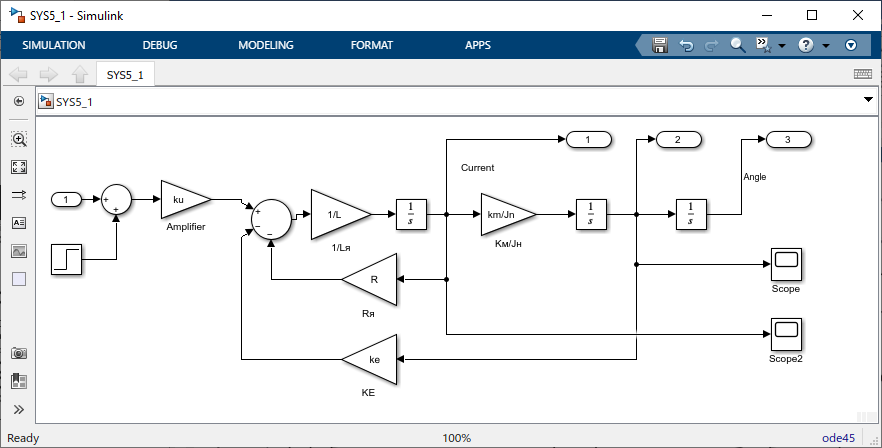
Рис. 9. Окно библиотеки Simulink Discretizing Library 2.0

По этой команде открывается окно библиотеки Discretizing, показанное на рис. 9. Видно, что в составе библиотеки содержатся блоки моделей различных подклассов. Таким образом в арсенале проектировщика имеется достаточно удобный инструментарий для переоборудования моделей непрерывных систем в цифроаналоговые и цифровые.

**Модальное управление в непрерывной и дискретной системе**

Далее рассмотрим методику синтеза дискретного регулятора по дискретной модели объекта (правая ветвь алгоритма на рис. 4) на основе модального управления.

Исходная Simulink-модель (файлы **pasport\_dpt\_SL261.m** и **SYS5\_1.slx**) непрерывной системы управления представлена на рис. 10:

 Рис. 10. Исходная непрерывная модель системы

Получим представление модели системы в форме пространства состояний (используйте скрипт **diskr\_mod.m**):

[A,B,C,D]=linmod('SYS5\_1')

Проверим управляемость системы:

Co = ctrb (A,B);

unctr = length (A) - rank (Co) ;% Число неуправляемых мод

if unctr == 0

disp ('Система полностью управляема')

else

T = 'Число неуправляемых мод равняется';

disp ([T unctr])

end

Протокол проверки:

>>

A =

1.0e+03 \*

-0.3643 -0.0017 0

3.9821 0 0

0 0.0010 0

B =

142.8571

0

0

C =

1 0 0

0 1 0

0 0 1

D =

0

0

0

Система полностью управляема

Произведем расчет желаемых полюсов системы в соответствии с распределением Баттерворта (загрузим на выполнение файл **batterwort.m**):

>> batterwort

Введите порядок системы n = 3

Введите желаемое время переходного процесса tgel = 0.02

tgel =

0.0200

Вектор желаемых полюсов

p =

1.0e+02 \*

-2.9800 + 0.0000i

-1.4900 + 2.5808i

-1.4900 - 2.5808i

Вычислим коэффициенты модального регулятора:

>> K = place(A,B,p)

K =

1.6220 0.3004 46.5197

Составим схему модели с модальным регулятором, показанную на рис. 11 (файл **SYS5\_2.slx**):

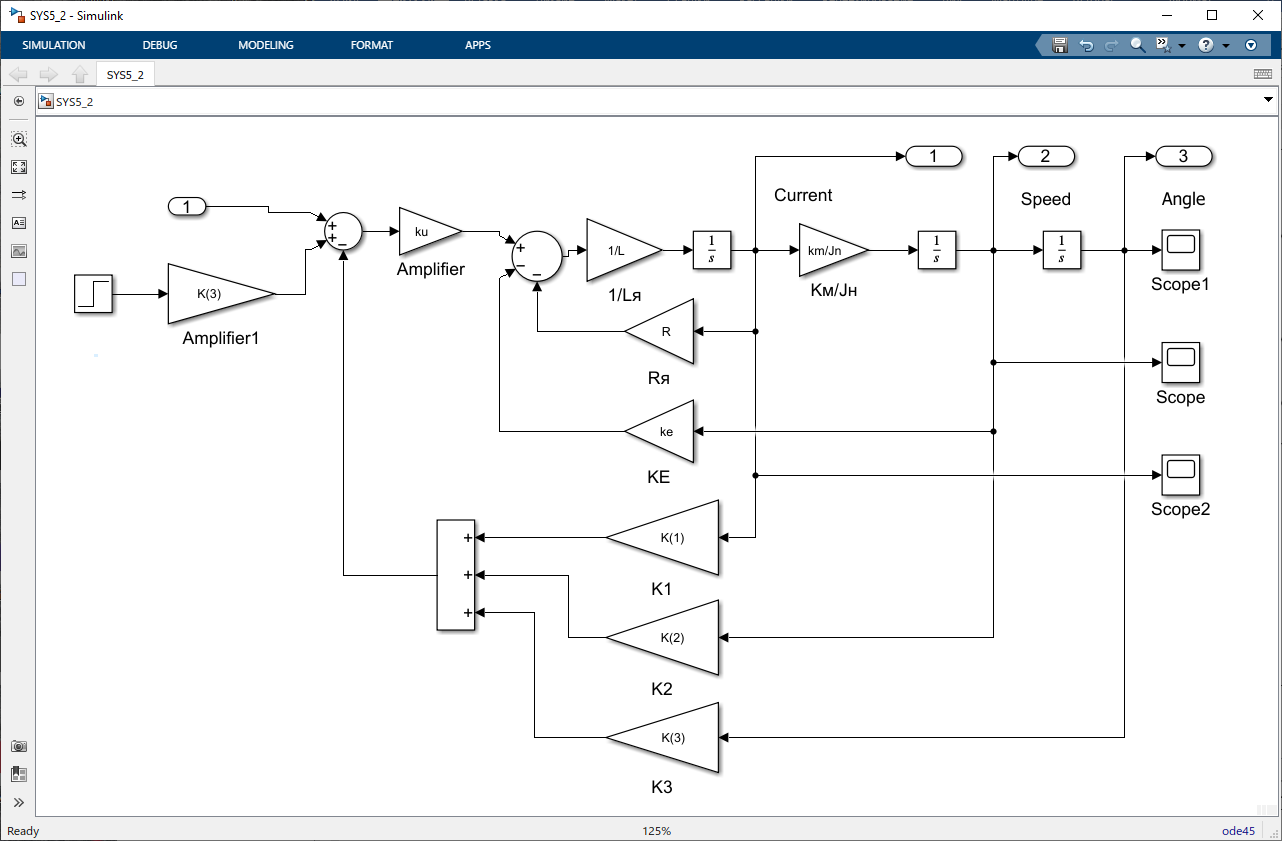


Рис. 11. Непрерывная система с модальным регулятором

Результат моделирования (переходный процесс по углу) показан на рис. 12.

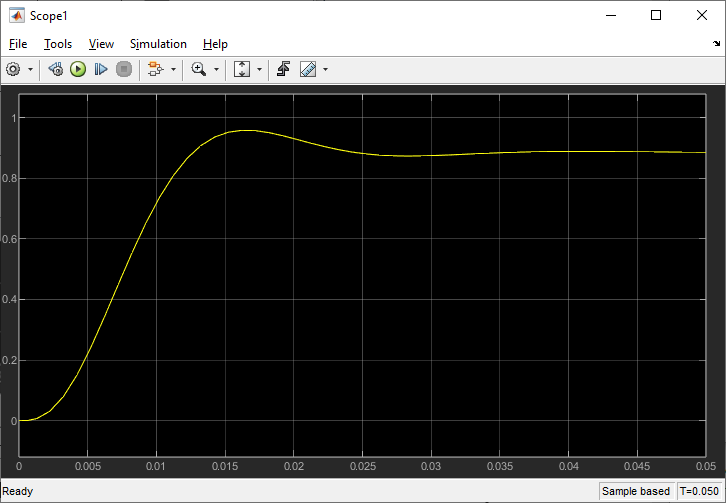


Рис. 12. Переходный процесс по углу

Рассмотрим переход от непрерывной к эквивалентной ей дискретной модели на примере системы управления, представленной выше.

Описанные ниже команды составляют содержимо файла **diskr\_mod.m** (запустите его на выполнение).

Найдем вектор желаемых полюсов дискретной системы в соответствии с выражением**:**

Ts=0.001; % Период дискретизации

pd=exp(p\*Ts) %Вектор желаемых полюсов дискретной системы

pd =

0.7423 + 0.0000i

0.8330 + 0.2199i

0.8330 - 0.2199i

В программе MATLAB/Control System Toolbox процедура дискретизации линейных моделей (класса LTI – *Linear Time-Invariant*) выполняется по команде **c2d**. Можно выбрать методы, предполагающие наличие фиксатора нулевого порядка на входе, метод Тастина и др.

Преобразуем непрерывную ss-модель системы в эквивалентную дискретную форму, воспользовавшись процедурой **c2d**:

sys=ss(A,B,C,D);

Sysd=c2d(sys,Ts,'zoh')

Sysd =

A =

x1 x2 x3

x1 0.692 -0.001413 0

x2 3.334 0.997 0

x3 0.001769 0.000999 1

B =

u1

x1 0.1196

x2 0.2527

x3 8.674e-05

C =

x1 x2 x3

y1 1 0 0

y2 0 1 0

y3 0 0 1

D =

u1

y1 0

y2 0

y3 0

Sample time: 0.001 seconds

Discrete-time state-space model.

Рассчитаем теперь коэффициенты дискретного модального регулятора для полученной дискретной формы модели системы и нормирующий коэффициент для единичного коэффициента передачи замкнутой системы:

Kd = place(Sysd.a,Sysd.b,pd)

Kd =

1.7122 0.2863 41.2254

Kdnorm1 =

41.2254

Изменим нашу схему моделирования, введя новые значения коэффициентов. Для сравнения дополним схему непрерывной моделью (файл **SYS5\_3.slx**, см. рис. 13). Настроим решатель на фиксированный шаг расчета, равный величине Ts = 0.001 с = 1/20·*t*пп (см. рис. 14). Этот же шаг следует указать в окне настройки блока Zero-Order Hold. Взятие меньшего шага расчета не оправдано с практической точки зрения.

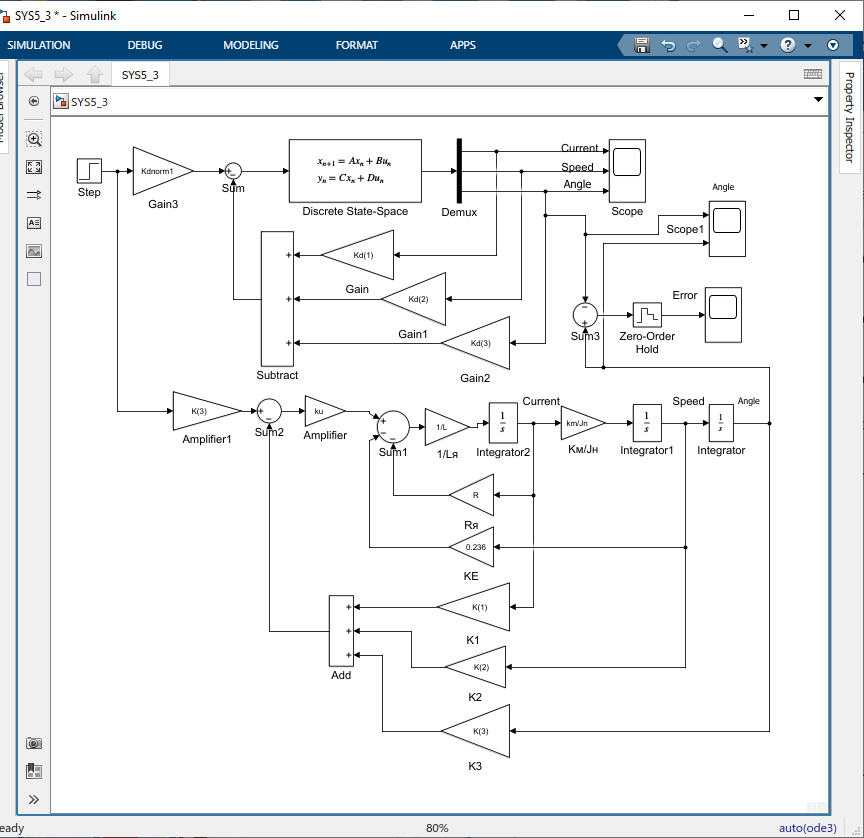


Рис. 13. Непрерывная (нижняя) и эквивалентная ей дискретная (верхняя) модель системы управления

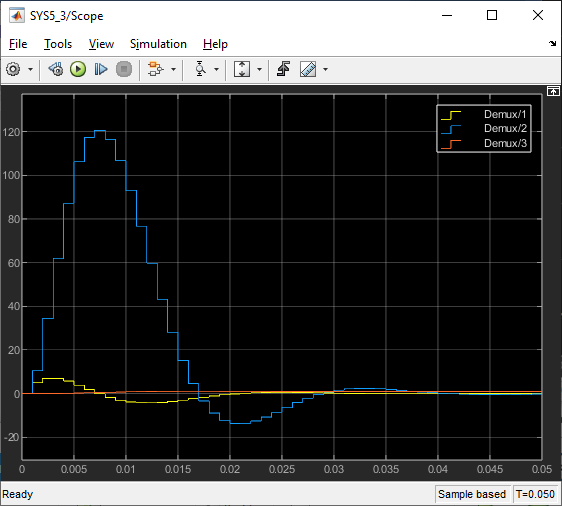
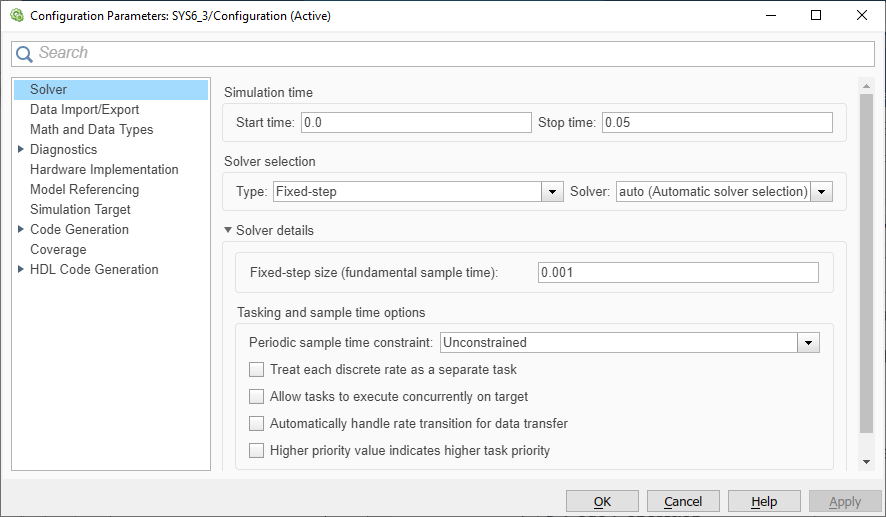


Рис. 14. Настройка решателя и результат симуляции

Выполним моделирование (см. рис. 14) и сравним результаты работы непрерывной и дискретной систем по углу поворота (см. рис. 15):

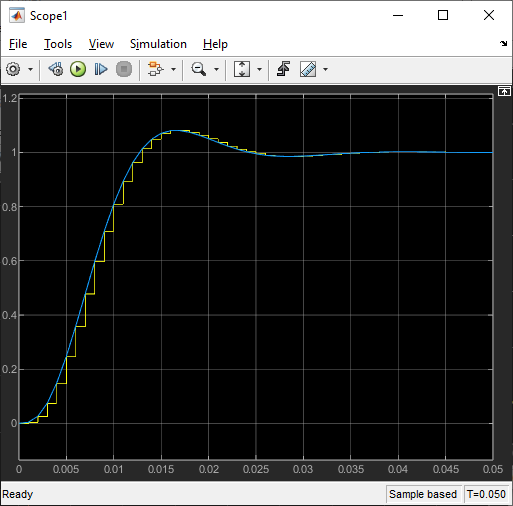
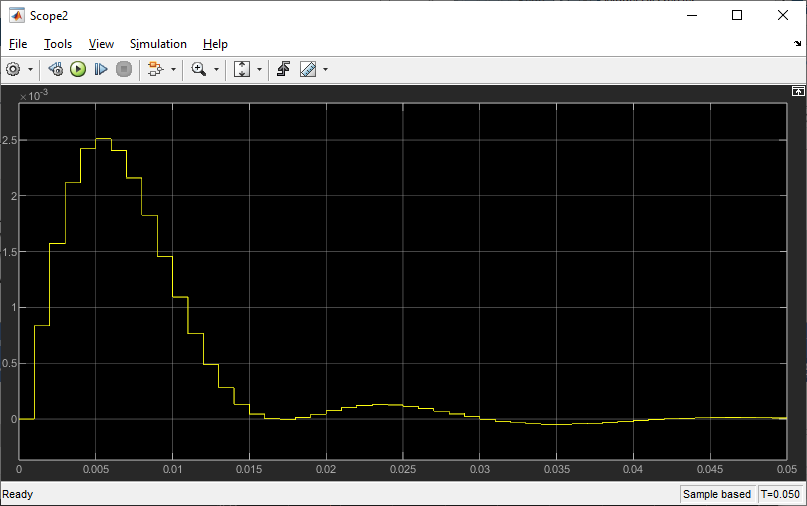


Рис. 15. Результаты моделирования непрерывной и дискретной модели по углу поворота

Выведем график ошибки между переходными процессами непрерывной и дискретной системами (рис. 16):



*Рис. 16. График ошибки между переходными процессами*

*непрерывной и дискретной системами*

Как видно, результаты хорошо совпадают. Максимальное значение ошибки e(kT) = 2.5⋅10-3 (0.25%).

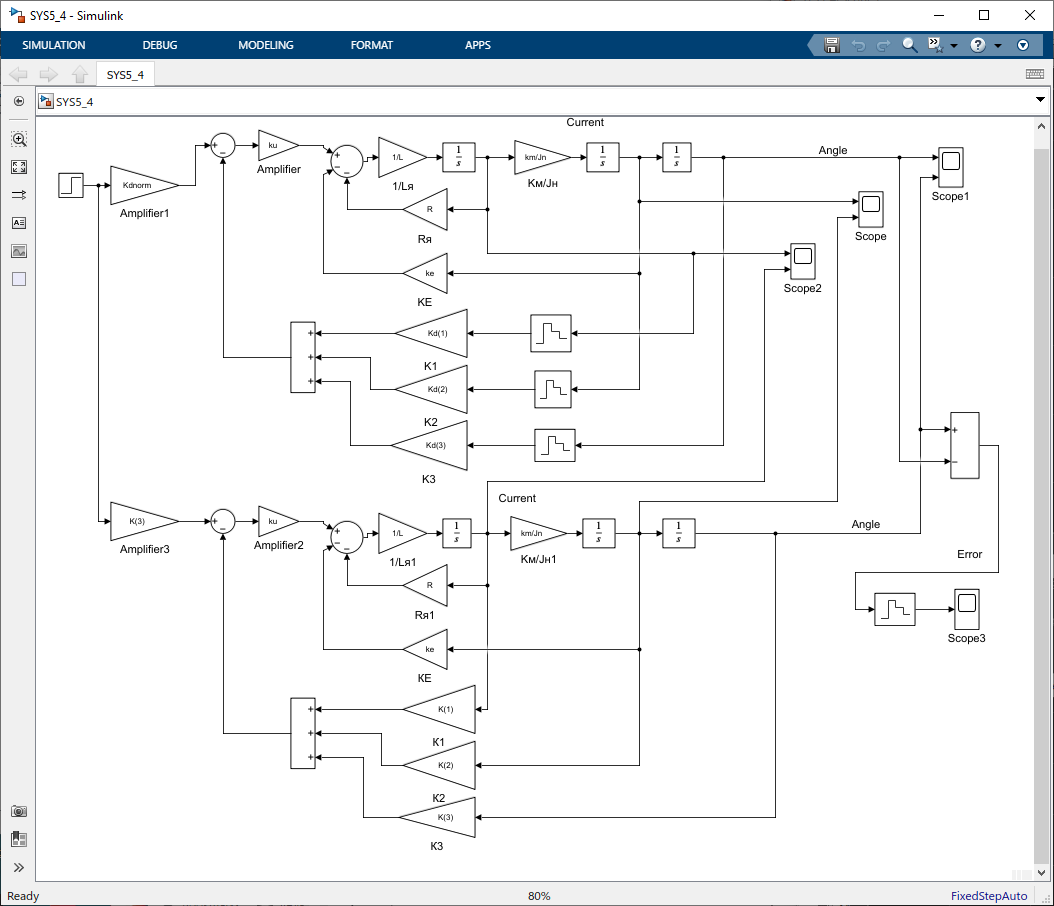


Рис. 17. Непрерывная (нижняя) и эквивалентная ей дискретно-непрерывная (верхняя) модель системы управления

Дополнительно на рис. 17 представлена система с дискретно-непрерывной моделью, которая будет использоваться на практике. Здесь подразумевается, что на практике наша модель сохраняет свою непрерывную форму. Разработанный нами цифровой модальный регулятор подключается к аналоговой модели объекта управления посредством АЦП (на схеме модели это блоки Zero-Order Hold). Результаты моделирования и в этом случае практически совпадают, в чем можно убедиться, загрузив и запустив на симуляцию файл Simulink-модели **SYS5\_4.slx** (убедитесь в этом самостоятельно).

**Практическое задание по материалу лекции 5**

**«Разработка и исследование систем цифрового модального управления непрерывными объектами»**

На основе методик, изложенных в лекции 5 и результатов предыдущих лабораторных и практических работ, выполнить разработку дискретного модального регулятора для своего варианта следящей системы управления с ДПТ независимого возбуждения. При этом должны быть выполнены все рассмотренные варианты построения регуляторов. Сравнить результаты разработанных аналоговых, цифровых и гибридных моделей системы управления в том числе с результатами, полученными в предыдущих работах.

Результаты работы оформить в виде отчета. Отчет должен содержать следующие разделы:

* Введение;
* Задание;
* Разработка моделей;
* Результаты моделирования и их сравнительная оценка;
* Выводы по работе.

**Приложение**

Текст файла-скрипта **diskr\_mod.m**

[A,B,C,D]=linmod('SYS5\_1')

Co = ctrb (A,B);

unctr = length (A) - rank (Co) ;% Число неуправляемых мод

if unctr == 0

disp ('Система полностью управляема')

else

T = 'Число неуправляемых мод равняется';

disp ([T unctr])

end

batterwort

K = place(A,B,p)

% Синтез модального регулятора для дискретной системы

Ts=0.001; % Период дискретизации

pd=exp(p\*Ts) %Вектор желаемых полюсов дискретной системы

sys=ss(A,B,C,D); % Описание непрерывной системы

Sysd=c2d(sys,Ts,'zoh') % Описание дискретной системы

Kd = place(Sysd.a,Sysd.b,pd) % Коэффициенты модального регулятора

% дискретной системы

Adk = Sysd.A -Sysd.B\*Kd; % Матрица A замкнутой дискретной системы

Fd =ss(Adk,Sysd.B,Sysd.C,Sysd.D,Ts); % Описание замкнутой дискретной

% системы

yduk=dcgain(Fd); % Установившееся значение на выходе

% замкнутой дискретной системы

Kdnorm1 = 1/yduk(3) % Нормирующий коэффициент для дискретной

% системы с коэффициентом передачи равным 1