Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Факультет: «Специальное машиностроение»

Кафедра: «Робототехнические системы и мехатроника»

Лабораторная работа № 4

по курсу «Теория автоматического управления»

Вариант 13

Выполнил: Садовец Роман
Группа: СМ7-62Б

Проверил(а):

Москва, 2024 г.

1. Моделирование Цифро-аналогового преобразователя

Цифро-аналоговый преобразователь (ЦАП) — устройство для преобразования цифрового (обычно двоичного) кода в аналоговый сигнал. Цифро-аналоговые преобразователи являются интерфейсом между дискретным цифровым миром и аналоговыми сигналами.

Назначим разрядность (англ. Digit Capacity) ЦАП и максимальное выходное напряжение

$$D\_C = 14 \text{ бит.}$$

$$U_{max} = 5 \text{ В.}$$

```
U_max = 5; % Max output voltage
D_C = 14; % Digit capacity of DAC
```

Соберём следующую принципальную схему ЦАП (рис. 1)

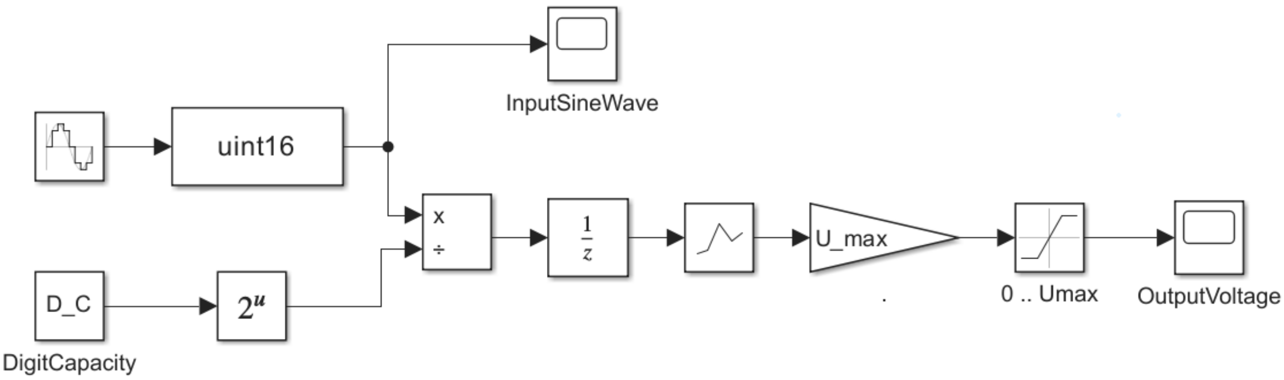

Рис. 1. Структурная схема ЦАП

На вход подаём дискретный синусоидальный сигнал. Его параметры представлены на рисунке 2, а на рисунке 3 показан формируемый синусоидальный сигнал при времени моделирования 1 секунда.

В реальности на вход ЦАП приходят двоичные (bin) числа, поэтому необходимо дополнительно проводить перевод числа из одной системы счисления в другую.

После моделирования, на выходе получим график, представленный на рисунке 4.

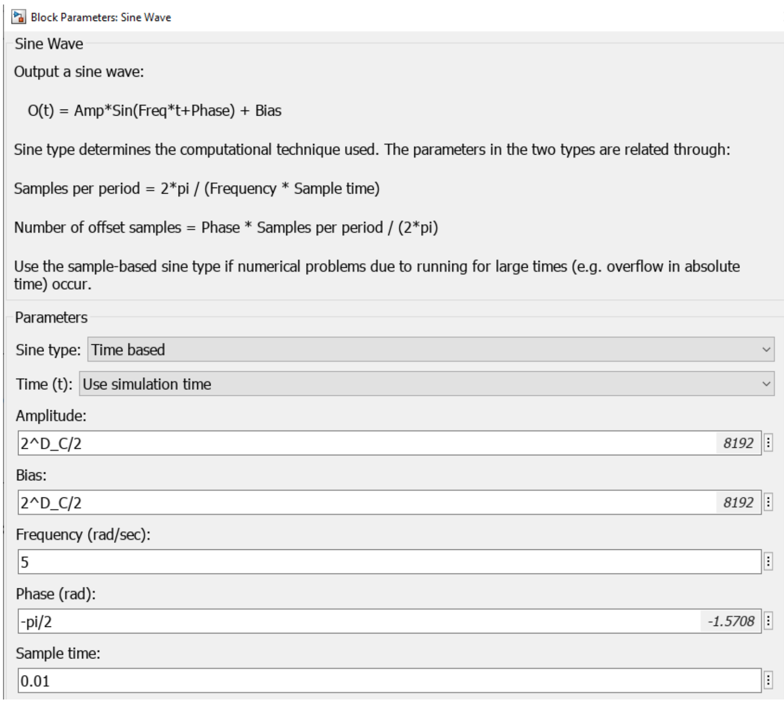

Рис. 2. Параметры дискретного синусоидального сигнала

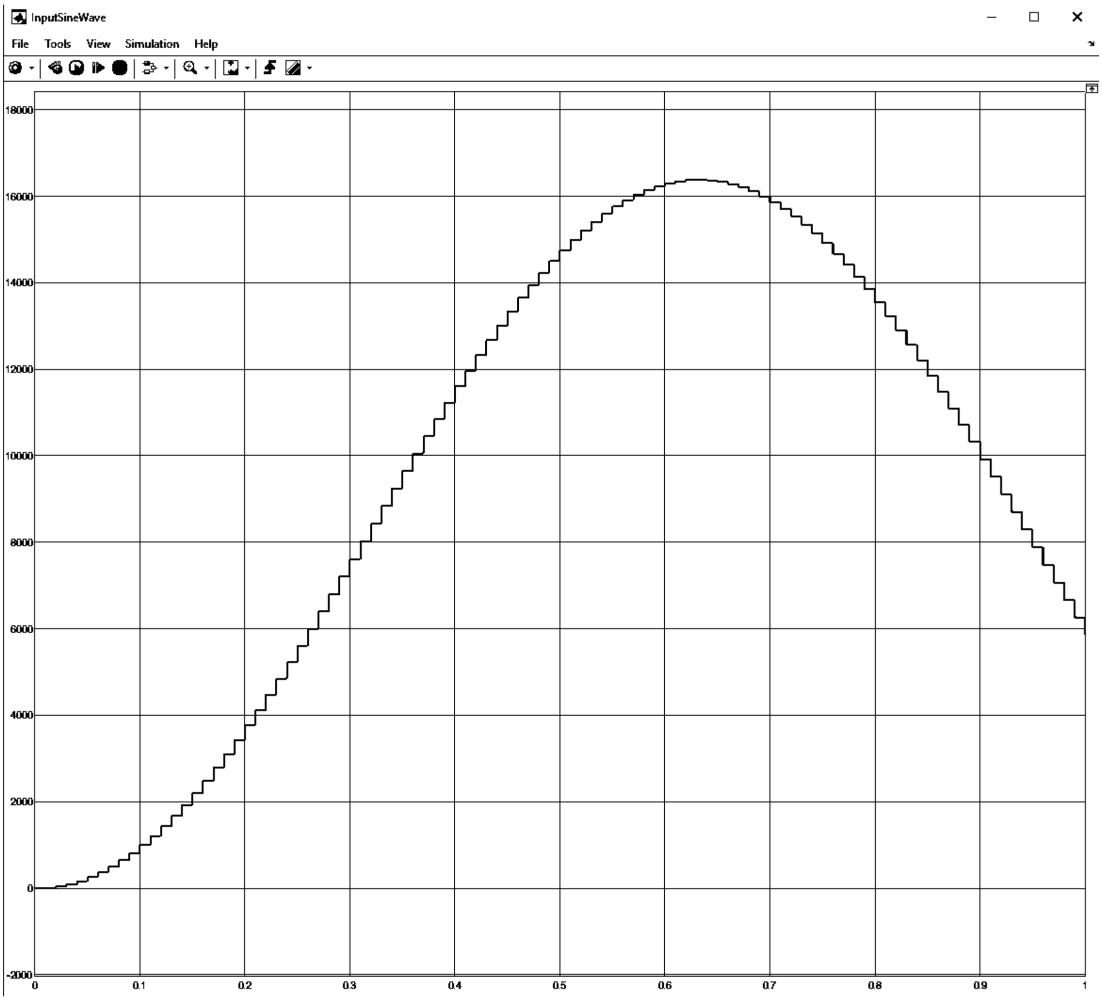

Рис. 3. Входной дискретный синусоидальный сигнал

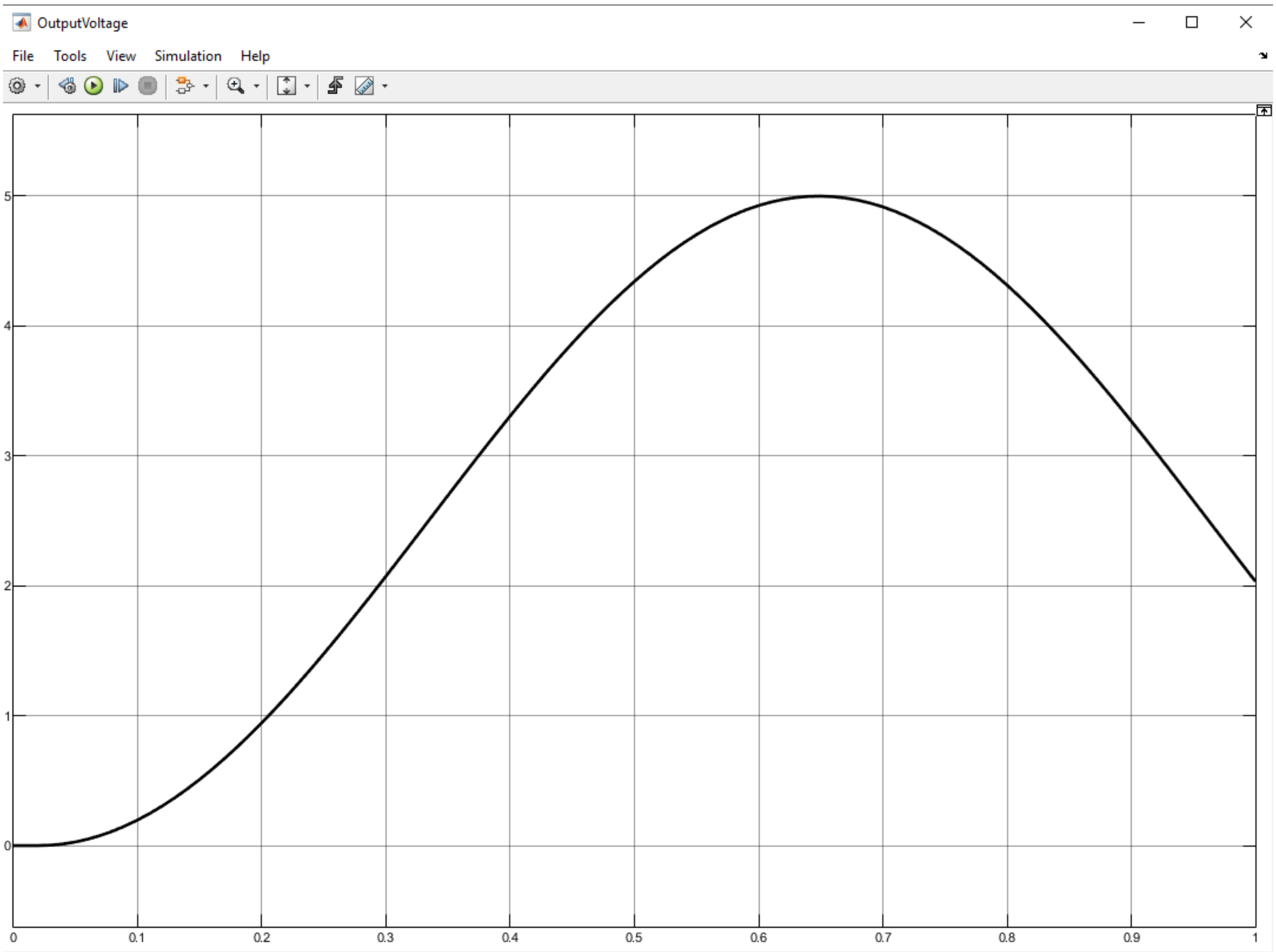

Рис. 4. Выходной аналоговый синусоидальный сигнал

Добиться устранения «ступенчатости» дискретного сигнала удалось за счёт блока задержки Unit Delay и блок аппроксимации дискретного сигнала First Order Hold.

2. ДЗ1 по курсу нелинейного ТАУ

```
x = linspace(36.1, 200, 1000);

figure('Name','q(a) plot');
plot(x,q(x),"DisplayName","q = q(a)",'LineWidth',2); %Plotting y=q(a)
yline(2/3, '-r', 'LineWidth', 2);                    %Plotting y=2/3
xlabel("a");
ylabel("q(a)");
grid on;

%%
function y = q(a)
    % Define func y = q(a)
    y = 2/pi .* asin( 36 ./ a ) +72/pi * sqrt(1 - 1296 ./ a.^2) ./ a;

end
```

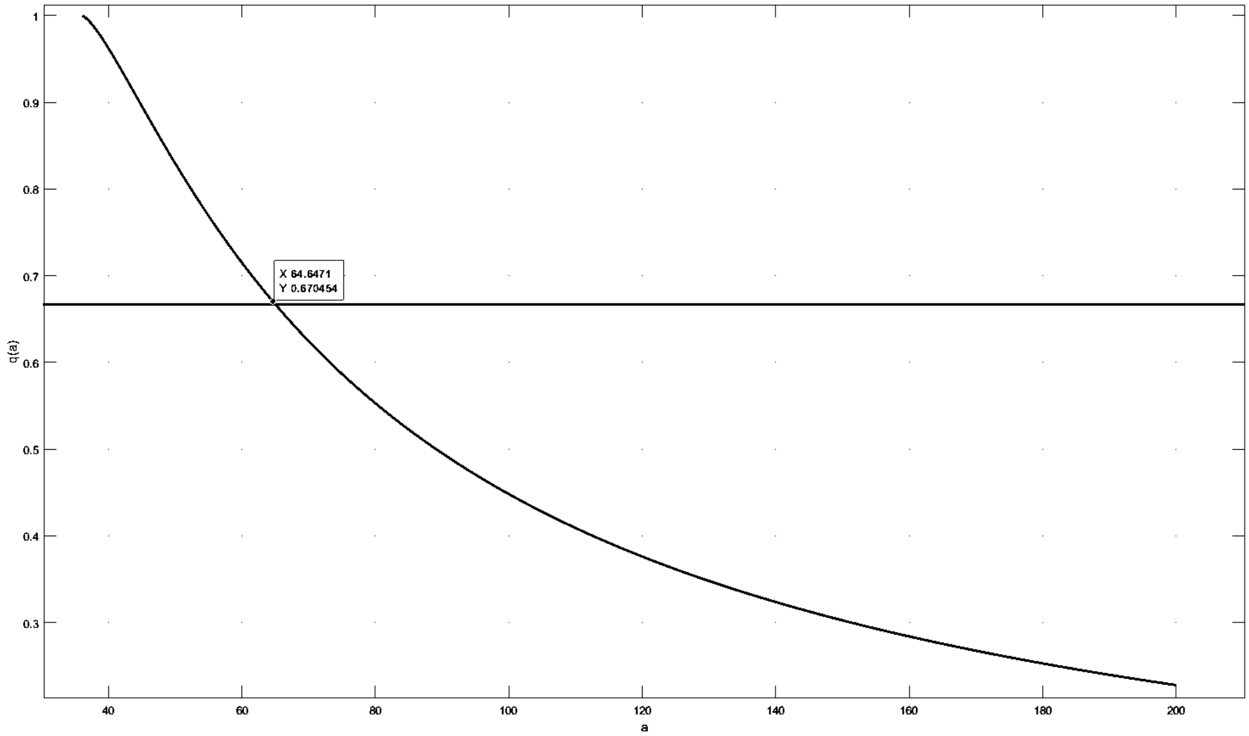

Рис. 5. График функции q(a) = 2/3

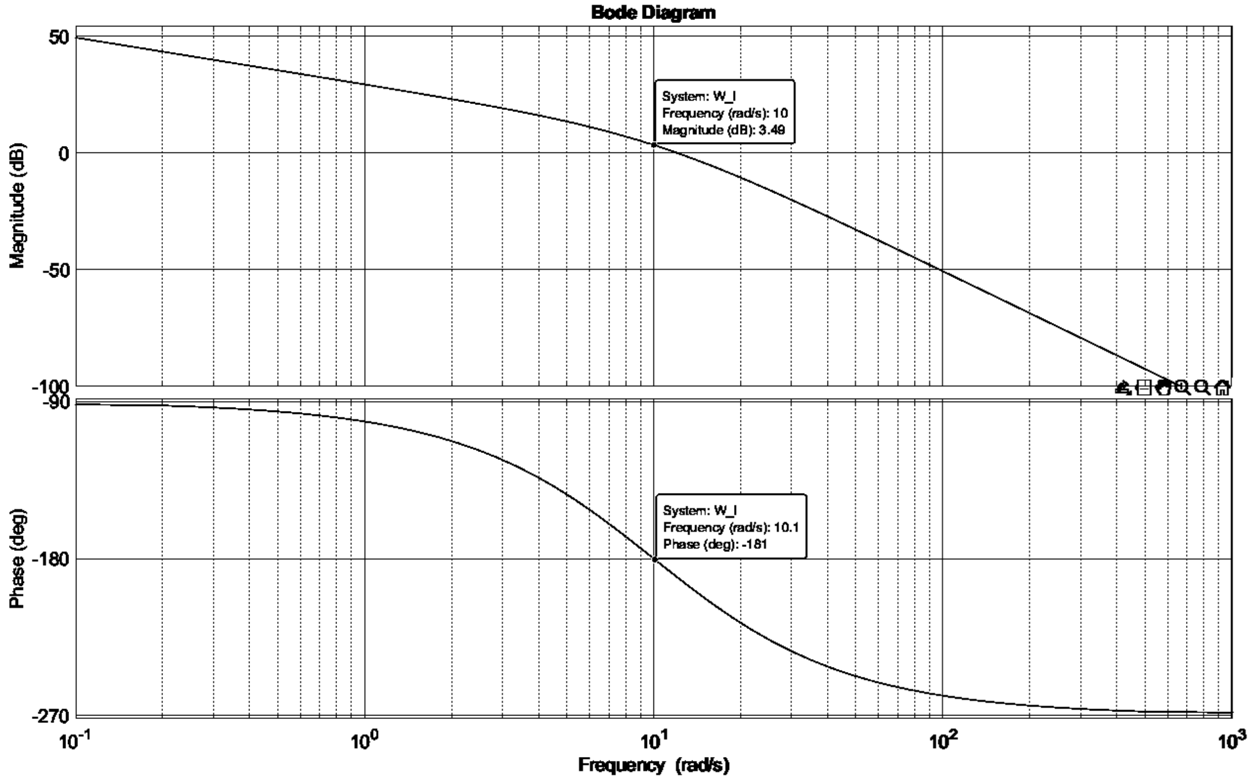

Рис. 6. ЛАЧХ и АФЧХ для линейной части системы

```
figure('Name','Nonlinear part plot')
semilogx(x, -20*log( q(x) ), "DisplayName","Nonlinear part plot",
'LineWidth', 2);
xlabel("a");
ylabel("-20lg( q(a) )");

grid on;
hold on;
```

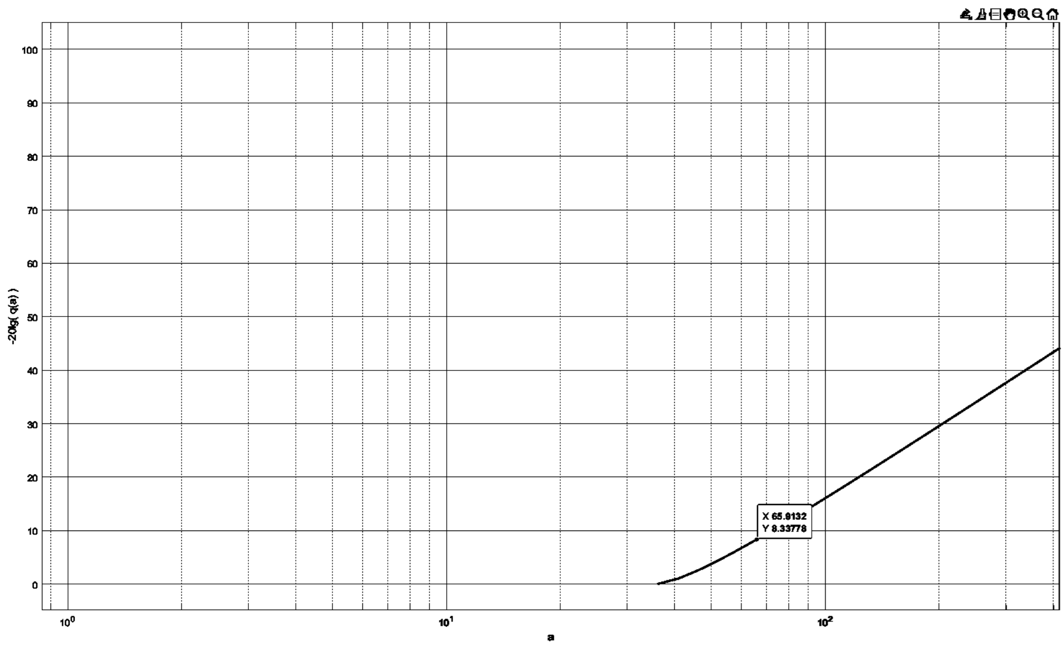

Рис. 7. ЛАЧХ для нелинейной части системы (y = -20lg(q(a)))

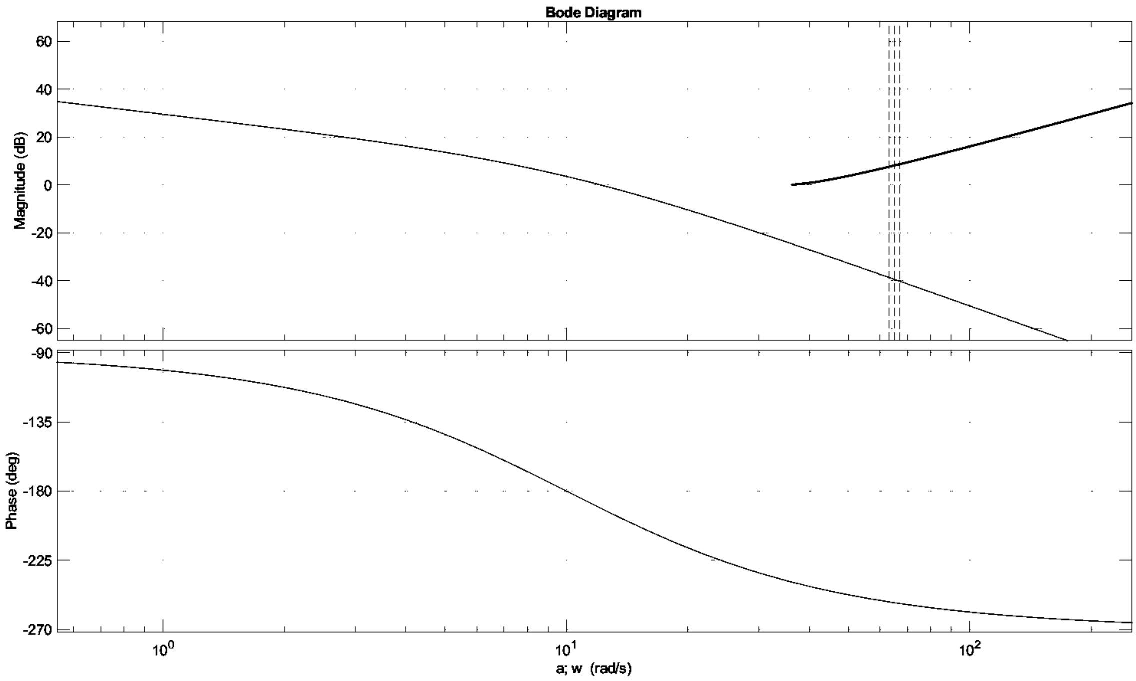

Рис. 8. Объединение ЛАЧХ и -20lg(q(a))

Компьютерный часть

```
%% Params for Linear part
k_l = 30;
T_1 = 0.1;
T_2 = 0.1;
```

```
function y = nonlin(eps)
% Define nonlinear part of system
    % Constant parameters
    c = 36;
    k = 1;

    y = k * x;
    for i = 1:length(y)
        if y >= c
            y = c;
        elseif y <= -c
            y = -c;
        end
    end
end
```

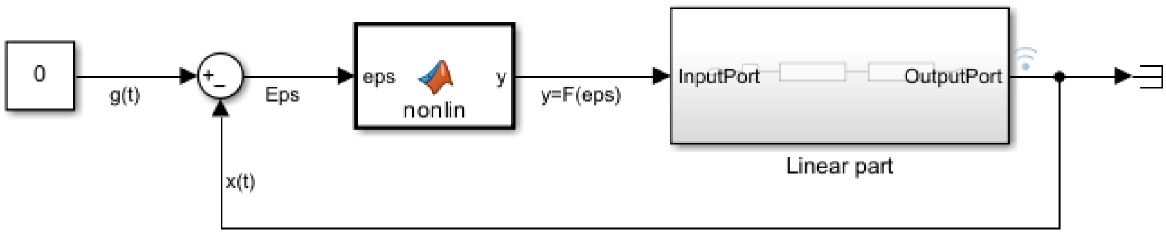

Рис. 9. Структурная схема в Simulink

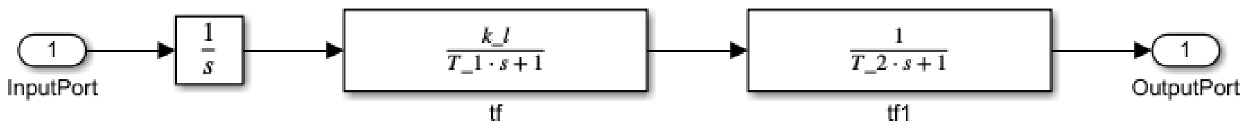

Рис. 10. Структурная схема линейной части в Simulink

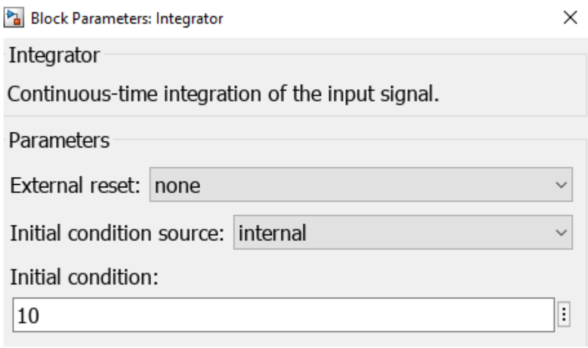

Рис. 11. Начальный условия (ω = 10 рад/с) в блоке интегратора

Наиболее подходящий солвер для решения задачи – ode23t. Назначим максимальный шаг, равный 0.5

Смоделируем систему на некотором произвольном участке времени, достаточным для осознания устойчивости автоколебаний. В ходе работы устанавливались различные промежутки времени моделирования системы, но после 2 секунды амплитуда автоколебаний не меняется. График для 5 секунд представлен ниже (рис. 12)

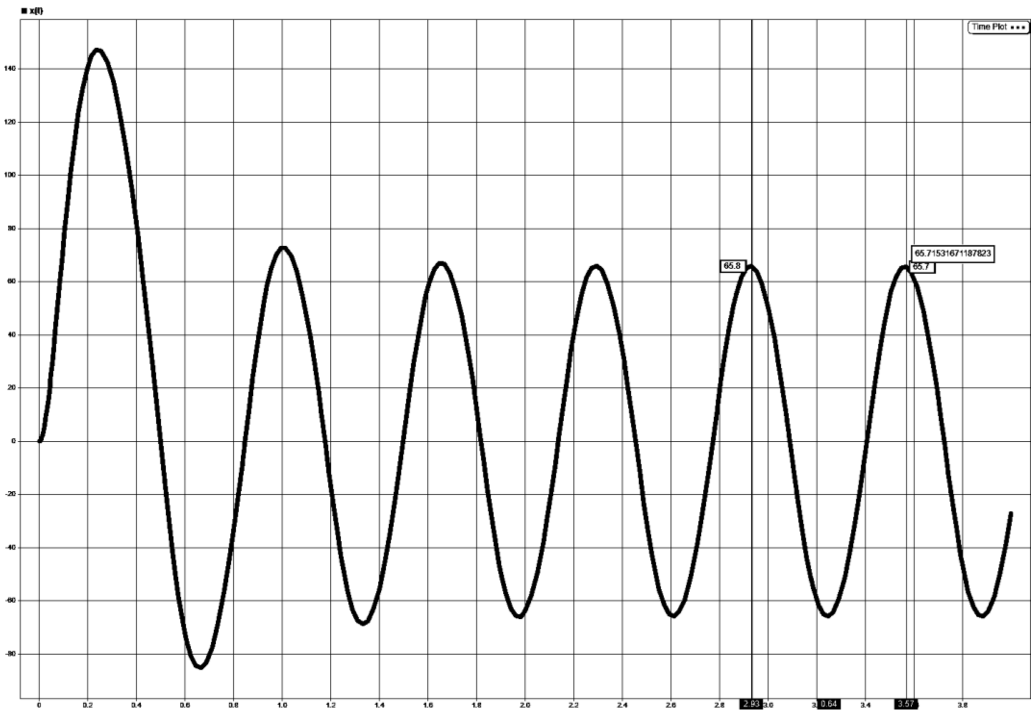

Рис. 12. График автоколебаний системы

По полученному графику можно получить приблизительный параметры системы:

$$a = 65.72$$

$$\Delta T = 0{,}64 \text{ сек.} \to f = \frac{1}{\Delta T} = \frac{1}{0.64} \sim 1.5625 \text{ Гц} \to \omega = f * 2\pi = 1.5625 * 2\pi$$

$$\to \omega = 9.82 \text{ рад/с}$$

Получаем автоколебания со следующими параметрами:

$$x(t) = 65.72 * \sin(9.82 * t)$$

Попробуем вывести систему из состояния устойчивости, задав значение для ω = 1000 рад/с (рис. 13)

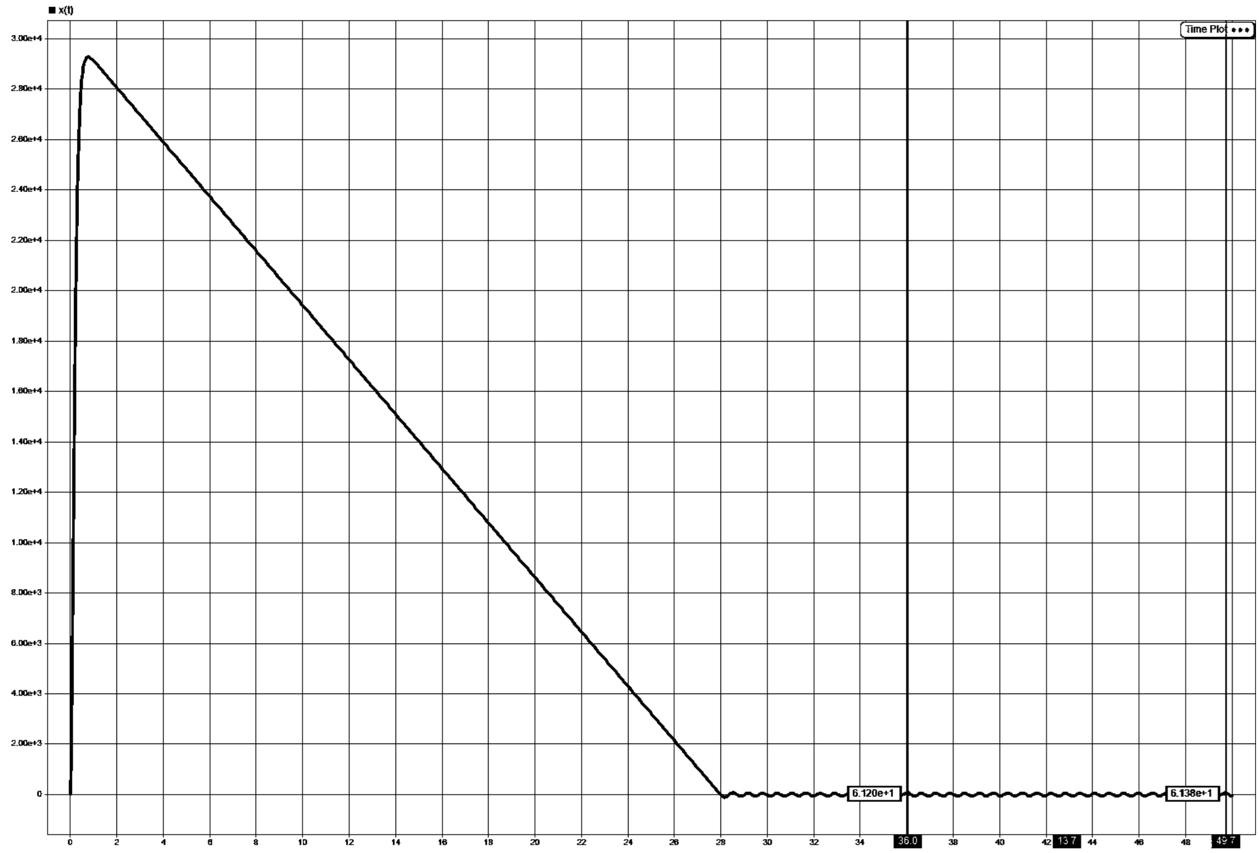

Рис. 13. График автоколебаний системы при ω = 1000 рад/с

Видно, что система не потеряла устойчивость автоколебаний.

При задании малых начальных условий (ω = 0.1 рад/с) система вначале начинает расходиться (рис. 10), но через какой-то промежуток времени возвращается на те же параметры автоколебаний.

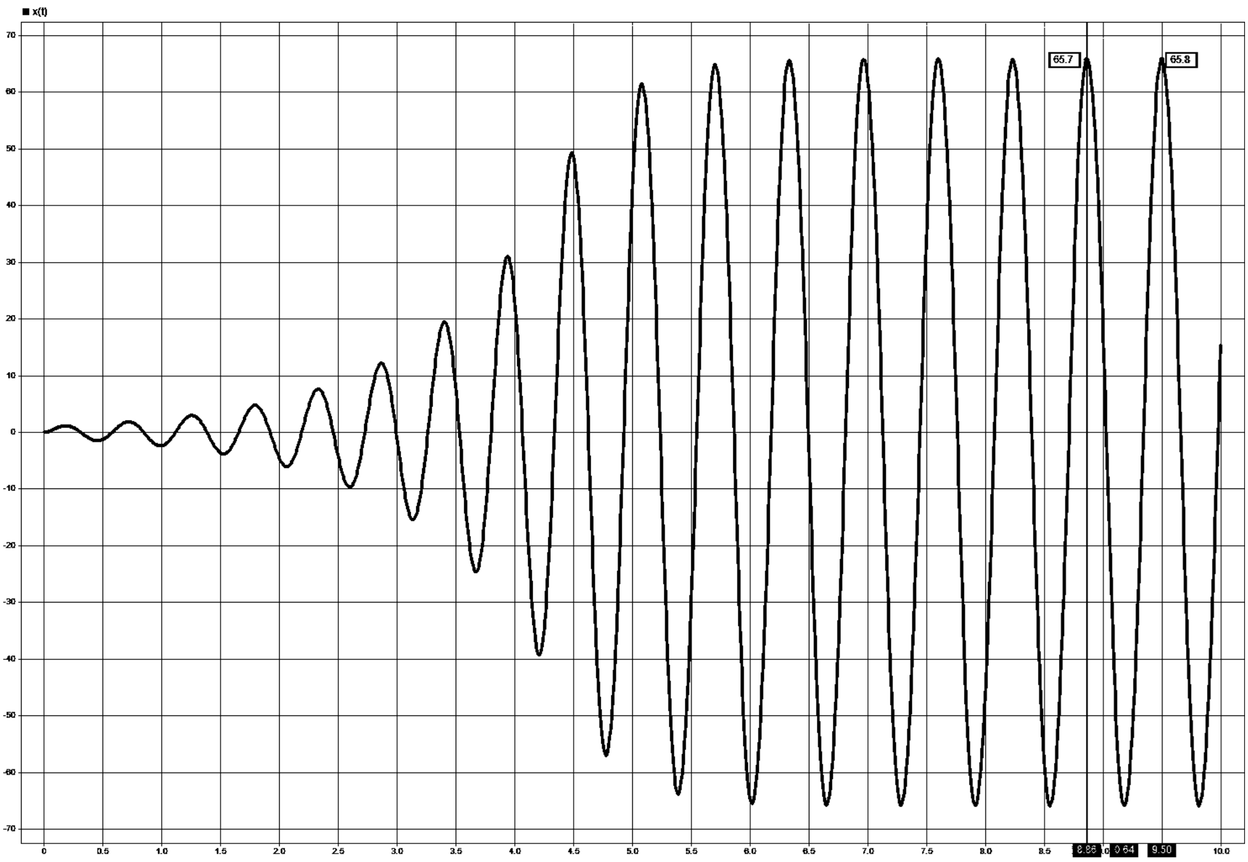

Рис. 14. График автоколебаний системы при ω = 0.1 рад/с

3. Синтез системы с нелинейным регулятором

В качестве неизменной части примем линейную часть ДЗ1 (см. рис. 10). Параметры неизменной части запишем в среде разработки MatLab

```
%% Linear part
k_l = 30;
T_1 = 0.1;
T_2 = 0.1;
```

Для этой системы необходимо провести синтез вначале непрерывного регулятора, затем дискретного.

Требования к системе:

- Перерегулирование $\leq 30\%$ при g(t) = 1
- Динамическая ошибка 1^o при g(t) = 50*sin(0.5t)

Составим модель системы с непрерывным ПИД-регулятором (рис. 15). Параметры синусоидального сигнала представлены на рисунке 16.

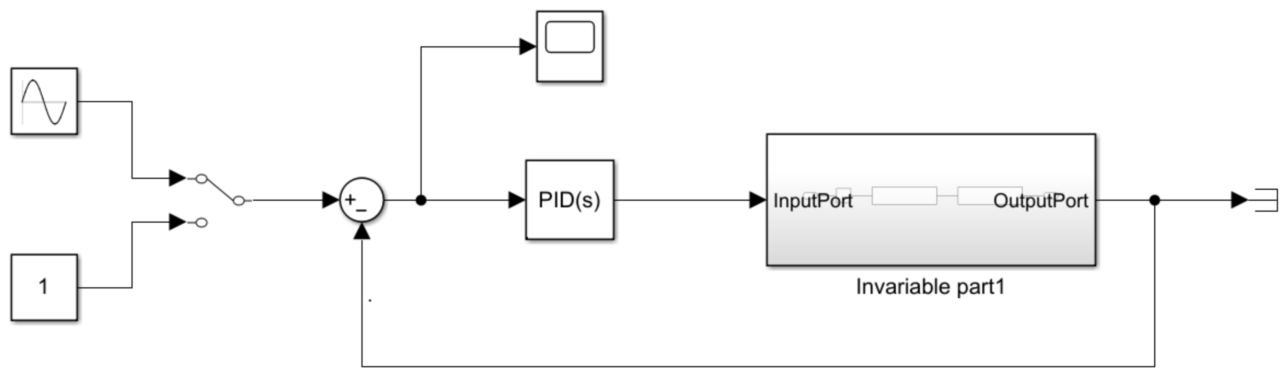

Рис. 15. Структурная схема модели с непрерывным регулятором.

Далее проведем синтез ПИД-регулятора для удовлетворения поставленным требованиям. Для этого воспользуемся внутренним тюнером ПИД-регулятора (“PID Controller” → Tune). Параметры робастности (Robust) и времени переходного процесса (Response Time) подбираем такими, чтобы удовлетворить поставленному техническому заданию. Путем перебора и проведения тестов, получаем параметры, представленные на рисунке 17. Перерегулирование при полученных параметрах составляет 17.8%. Если подать на вход сигнал g(t) = 50*sin(0.5t), то получим график, представленный на рисунке 18. Динамическая ошибка составляет менее 1 градуса (по модулю)

Рис. 16. Синусоидальный сигнал g(t) = 50*sin(0.5t)

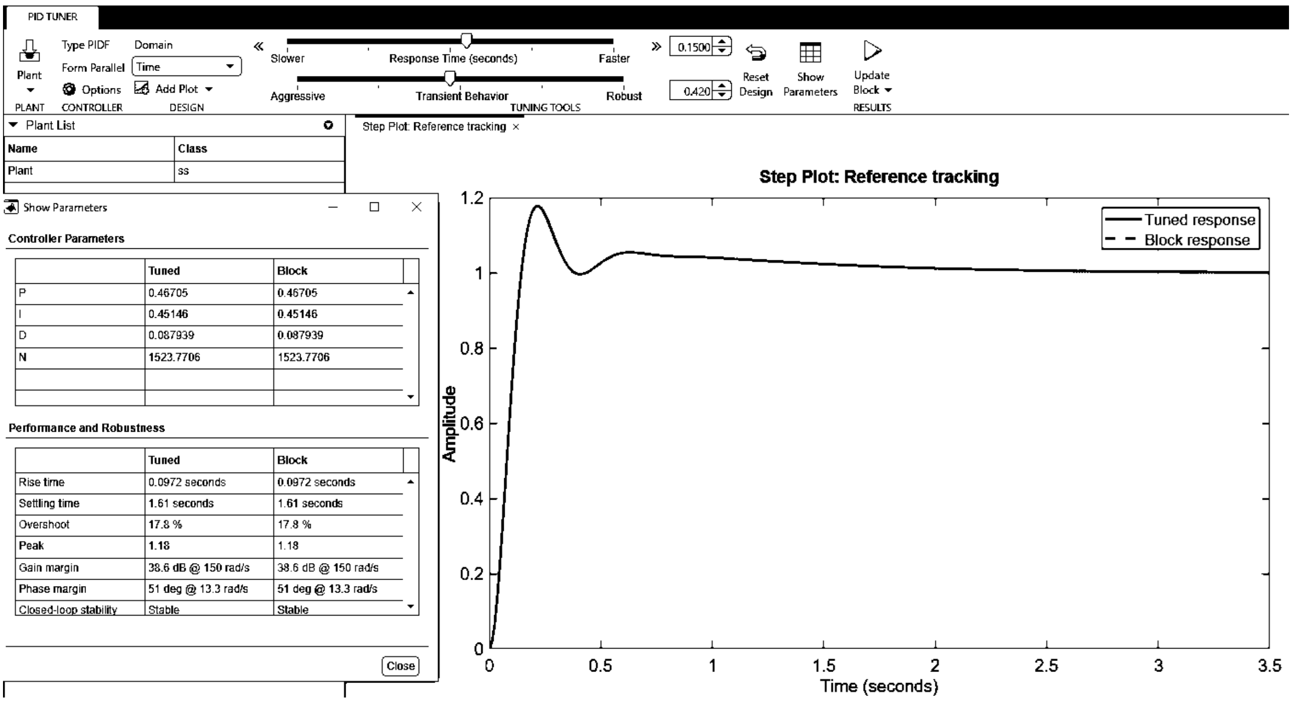

Рис. 17. Синтез ПИД-Регулятора. Перерегулирование

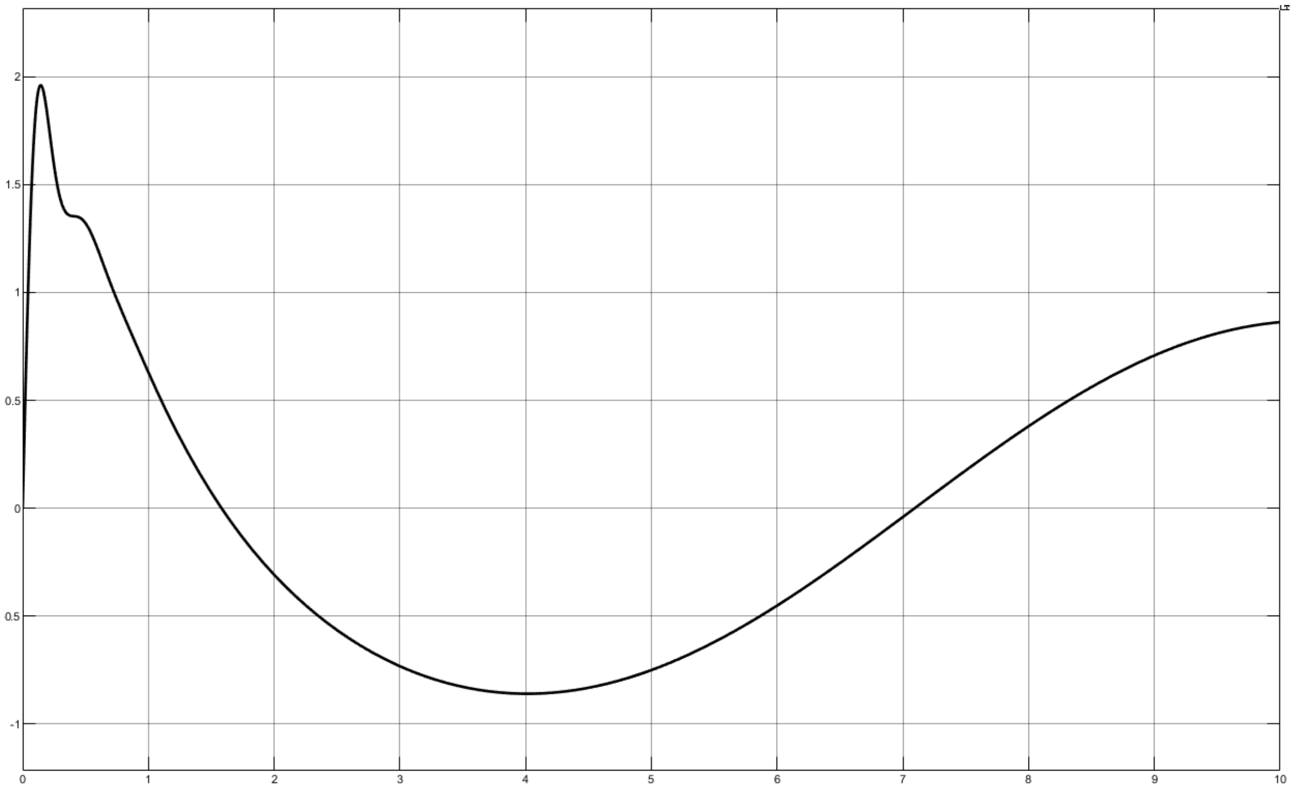

Рис. 18. Величина ошибки при g(t) = 50*sin(0.5t)

Получившиеся параметры экспортируем в Matlab.

```
%% PID parameters
k_p = 0.467049831227445;
k_i = 0.451462915126038;
k_d = 0.0879387159613378;
N = 1523.77063472214;
```

Следующий этап – создание с полученными параметрами дискретного ПИД-регулятора. В первую очередь, осуществим перевод передаточной функции ПИД-регулятора из непрерывного вида в дискретный. Передаточная функция непрерывного ПИД-регулятора:

$$W_{\text{ПИД}}(s) = k_{\text{П}} + k_{\text{И}} * \frac{1}{s} + k_{\text{Д}} * \frac{N}{\left(1 + N * \frac{1}{s}\right)},$$

где $k_{\text{П}}, k_{\text{И}}, k_{\text{Д}}$ – полученный коэффициенты ПИД-регулятора; N – коэффициент фильтра ПИД-регулятора.

Перевод из непрерывного вида в дискретный осуществим с помощью функции c2d() с использованием метода 'zoh'. Время дискретизации 0.001 с.

```
%% Compensator formula
s = tf('s');
Ts = 0.0001;

Compensator_formula = k_p + k_i * ( 1 / s ) +...
    k_d * N / ( 1 + N * ( 1 / s ) );

RegulatorDiscreteFunc = c2d(Compensator_formula, Ts, 'zoh');
```

После чего создадим модель с дискретным регулятором (рис. 19-21)

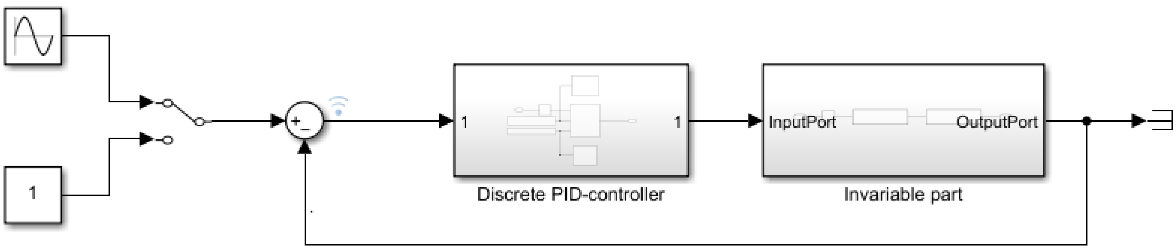

Рис. 19. Структурная схема модели с дискретным регулятором.

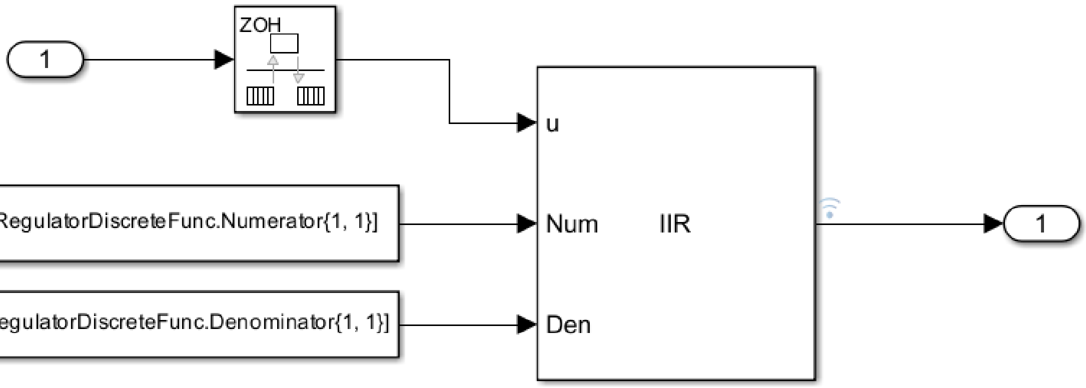

Рис. 20. Структурная схема дискретного регулятора

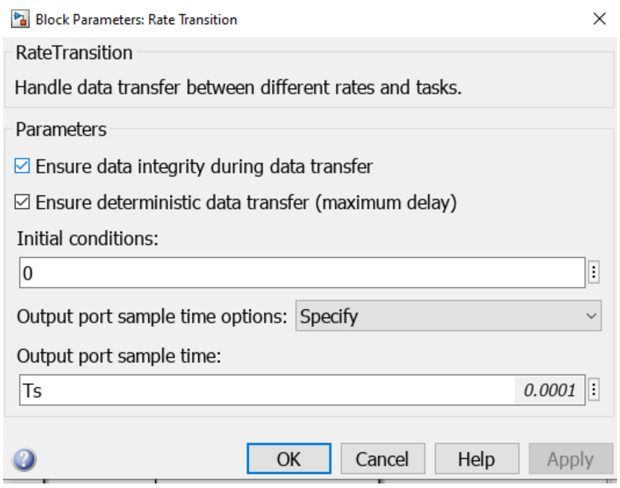

Рис. 21. Параметры блока Rate Transition

Подадим на вход g(t) = 1 и g(t) = 50*sin(0.5t) и выведем полученный графики системы (рис. 22 и 23-24 соотв.)

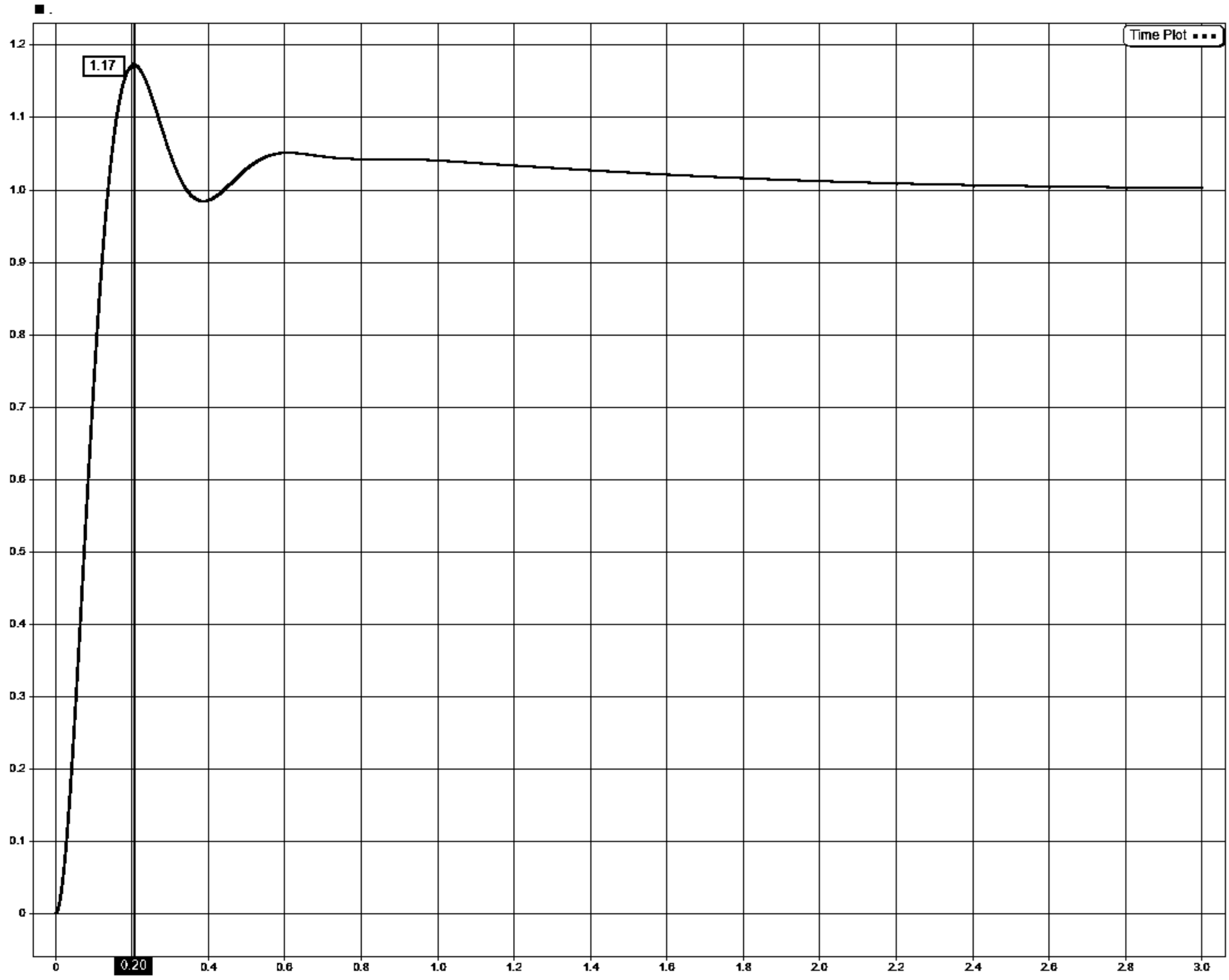

Рис. 22. Сигнал на выходе из линейной части при подаче g(t) = 1

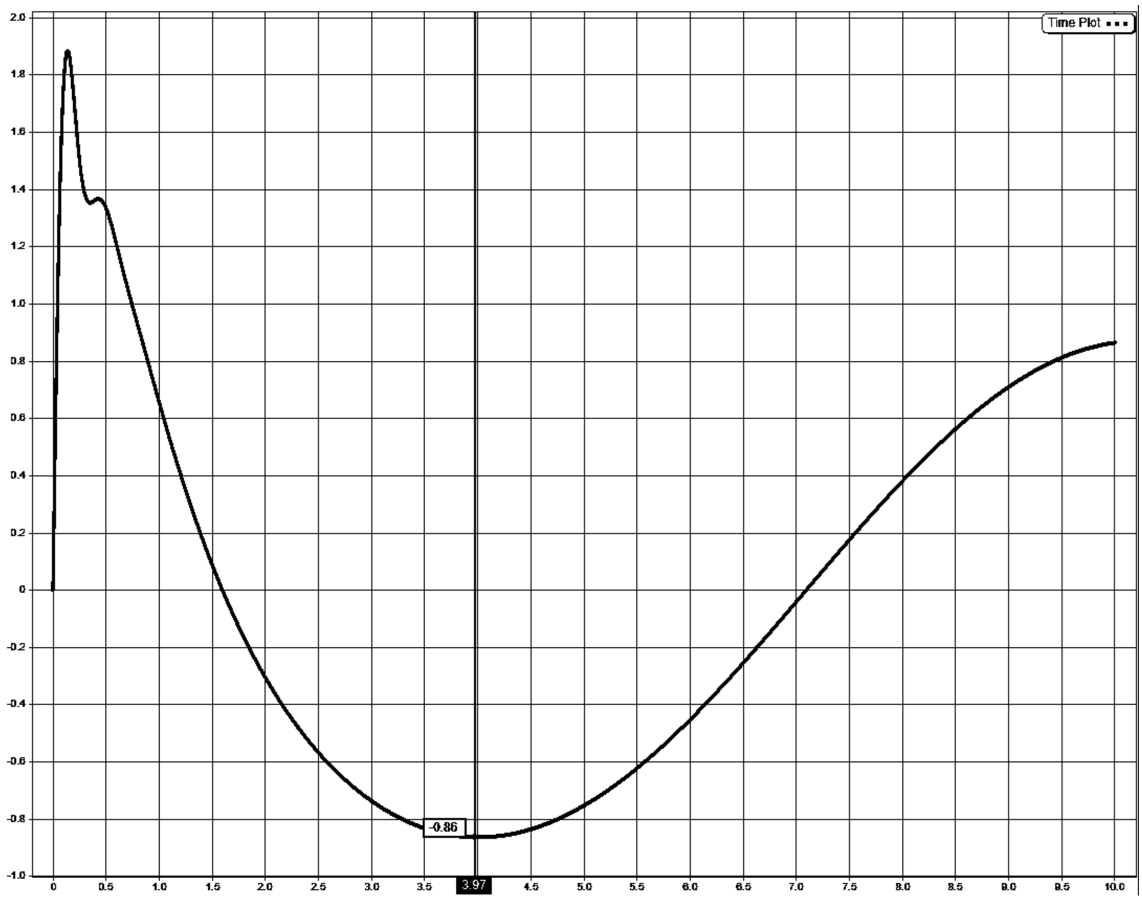

Рис. 23. Сигнал ошибки при подаче g(t) = 50*sin(0.5t)

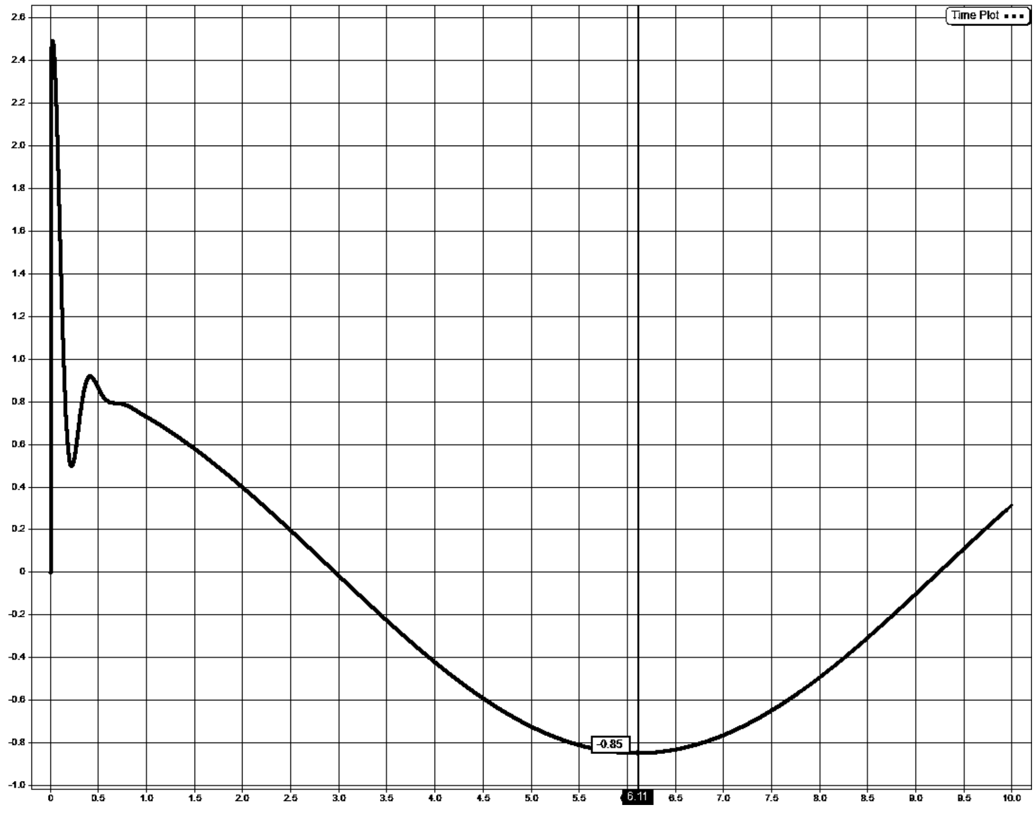

Рис. 24. Сигнал на выходе из регулятора при подаче g(t) = 50*sin(0.5t)

Приложение

1. Публичный репозиторий для лабораторных по ТАУ // GitHub URL: https://github.com/RiXenGC/Theory-of-Automatic-Control