



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
"МИРЭА – Российский технологический университет"

РТУ МИРЭА

Институт искусственного интеллекта
Кафедра проблем управления

Программное обеспечение мехатронных и робототехнических систем

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

Тема лабораторной работы: «Отладка программного обеспечения
робототехнических систем с использованием виртуального моделирования»

Выполнили: студенты группы
КРБО-01-20

Крохмальный Д. И.
Клиндухов Я. И.

Ведущий преподаватель
Морозов А.А.

Цель работы: получение навыков моделирования объекта управления в промышленных системах автоматического управления и создание функциональных блоков.

Задание: создать виртуальную систему управления (рис.1), включающую: модель объекта управления (рис. 2), ПИ-регулятор (рис. 3), сумматор и обратную связь. Передаточная функция объекта:

$$W = \frac{1}{k_e \cdot (T_M s + 1)}$$

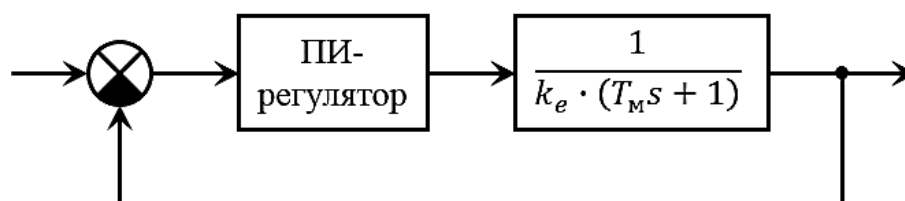


Рис. 1. Структура системы управления

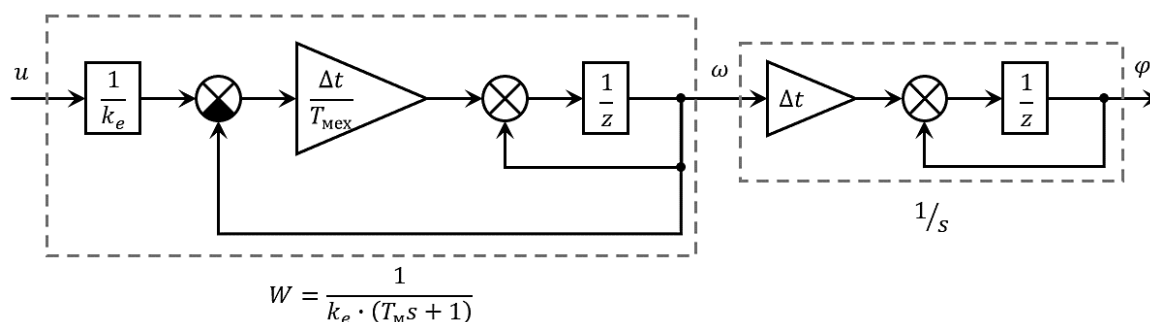


Рис. 2. Структура объекта управления

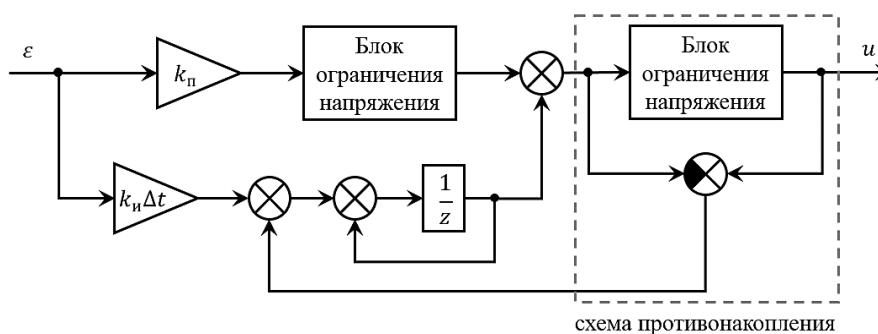


Рис. 3. Структура ПИ-регулятора

Ход работы:

Давайте начнем новый проект в среде Automation Studio без конфигурации оборудования. Нам не требуется настраивать соединение с установкой, так как будем работать с симуляцией.

Создадим в проекте следующие объекты:

1. ANSI C Program;
2. ANSI C Library «MotorControl».

В библиотеке создаем функциональные блоки:

1. «FB_Motor» — модель ДПТ;
2. «FB_Regulator» — модель ПИ-регулятора;
3. «FB_Integrator» — модель интегрирующего звена.

Детальное создание моделей начнем с интегратора, поскольку он необходим для функциональных блоков мотора и регулятора.

Таблица 1. Параметры функционального блока FB_Integrator

Конфигурация	Имя	Тип данных	Описание
ВХОД	in	REAL	вход интегрирующего звена
ВЫХОД	out	REAL	выход интегрирующего звена
внутреннее состояние	dt	REAL	шаг расчета [с]

Заносим параметры функционального блока FB_Integrator в Automation Studio согласно Таблице 1.

Name ▲	Type	& Reference	Scope
FB_Integrator		<input type="checkbox"/>	
in	REAL	<input type="checkbox"/>	VAR_INPUT
out	REAL	<input checked="" type="checkbox"/>	VAR_OUTPUT
dt	REAL	<input type="checkbox"/>	VAR

Рис. 4. Параметры функционального блока FB_Integrator

На структурных схемах блок интегратора представляет собой следующую структуру:

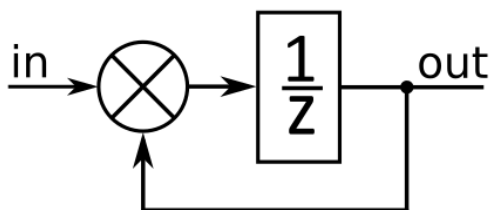


Рисунок 5. Структурная схема интегратора

Логика работы интегратора заключается в накоплении сумм разностей входного и выходного значений в соответствии с шагом расчета. Это реализуем в программном коде данного функционального блока (см. приложение А).

Далее наполним функциональный блок FB_Motor.

Таблица 2. Параметры функционального блока FB_Motor

Конфигурация	Имя	Тип данных	Описание
вход	U	REAL	входное напряжение [В]
выход	W	REAL	частота вращения [об/мин]
выход	Phi	REAL	положение [рад]
внутреннее состояние	integrator	FB_Integrator	интегратор
внутреннее состояние	Tm	REAL	электрохимическая постоянная времени [с]
внутреннее состояние	Ke	REAL	постоянная ЭДС двигателя [В•мин/об]
внутреннее состояние	Dt	REAL	шаг расчета [с]

Заносим параметры функционального блока FB_Motor в Automation Studio согласно Таблице 2.

FB_Motor			
u	REAL	<input type="checkbox"/>	VAR_INPUT
w	REAL	<input checked="" type="checkbox"/>	VAR_OUTPUT
phi	REAL	<input checked="" type="checkbox"/>	VAR_OUTPUT
Tm	REAL	<input type="checkbox"/>	VAR
ke	REAL	<input type="checkbox"/>	VAR
dt	REAL	<input type="checkbox"/>	VAR
integrator	FB_Integrator	<input type="checkbox"/>	VAR

Рис. 6. Параметры функционального блока FB_Motor

Расчет значения на выходе блока происходит в соответствии со схемой ДПТ (см. рис. 2) с помощью программного кода (см. приложение Б).

Крайний функциональный блок FB_Regulator.

Таблица 3. Параметры функционального блока FB_Regulator

Конфигурация	Имя	Тип данных	Описание
вход	e	REAL	рассогласование между задающим воздействием и реальной скоростью вращения вала ДПТ [об/мин]
выход	u	REAL	напряжение, подаваемое на вход ДПТ [В]
внутреннее состояние	k_p	REAL	пропорциональный коэффициент регулятора
внутреннее состояние	k_i	REAL	интегральный коэффициент регулятора
внутреннее состояние	integrator	FB_Integrator	интегратор
внутреннее состояние	iyOld	REAL	хранение предыдущего значения схемы противонакопления
внутреннее состояние	max_abs_value	REAL	граница блока ограничения [В]
внутреннее состояние	dt	REAL	шаг расчета [с]

Заносим параметры функционального блока FB_Regulator в Automation Studio согласно Таблице 3.

FB_Regulator		<input type="checkbox"/>	
e	REAL	<input type="checkbox"/>	VAR_INPUT
u	REAL	<input type="checkbox"/>	VAR_OUTPUT
k_p	REAL	<input type="checkbox"/>	VAR
k_i	REAL	<input type="checkbox"/>	VAR
iyOld	REAL	<input type="checkbox"/>	VAR
max_abs_value	REAL	<input type="checkbox"/>	VAR
& dt	REAL	<input checked="" type="checkbox"/>	VAR
integrator	FB_Integrator	<input type="checkbox"/>	VAR

Рис. 7. Параметры функционального блока FB_Regulator

Расчет значения на выходе блока происходит в соответствии со схемой ПИ-регулятора (см. рис. 3) с помощью программного кода (см. приложение В).

Объединим объект и регулятор в систему управления в основной программе с применением разработанных функциональных блоков.

В основной программе Main создадим следующие переменные:

Таблица 4. Переменные основной программы

Имя	Тип данных	Описание
fb_controller	FB_Controller	рассогласование между задающим воздействием и реальной скоростью вращения вала ДПТ [об/мин]
fb_motor	FB_Motor	напряжение, подаваемое на вход ДПТ [В]
Speed	REAL	уставка по скорости
Enable	BOOL	интегральный коэффициент регулятора
dt	REAL	шаг расчета [с]

В основной программе, в части инициализации «Init», заполняем все постоянные (коэффициенты регуляторов, постоянные времени, граничные значения и шаги расчета) созданных объектов fb_controller и fb_motor.

Добавляем второй мотор, указав в полях инициализации данные, аналогичные уже созданному ранее мотору. Добавить исполнение

функционального блока второго мотора в основной цикл программы, подавая на его вход уставку speed.

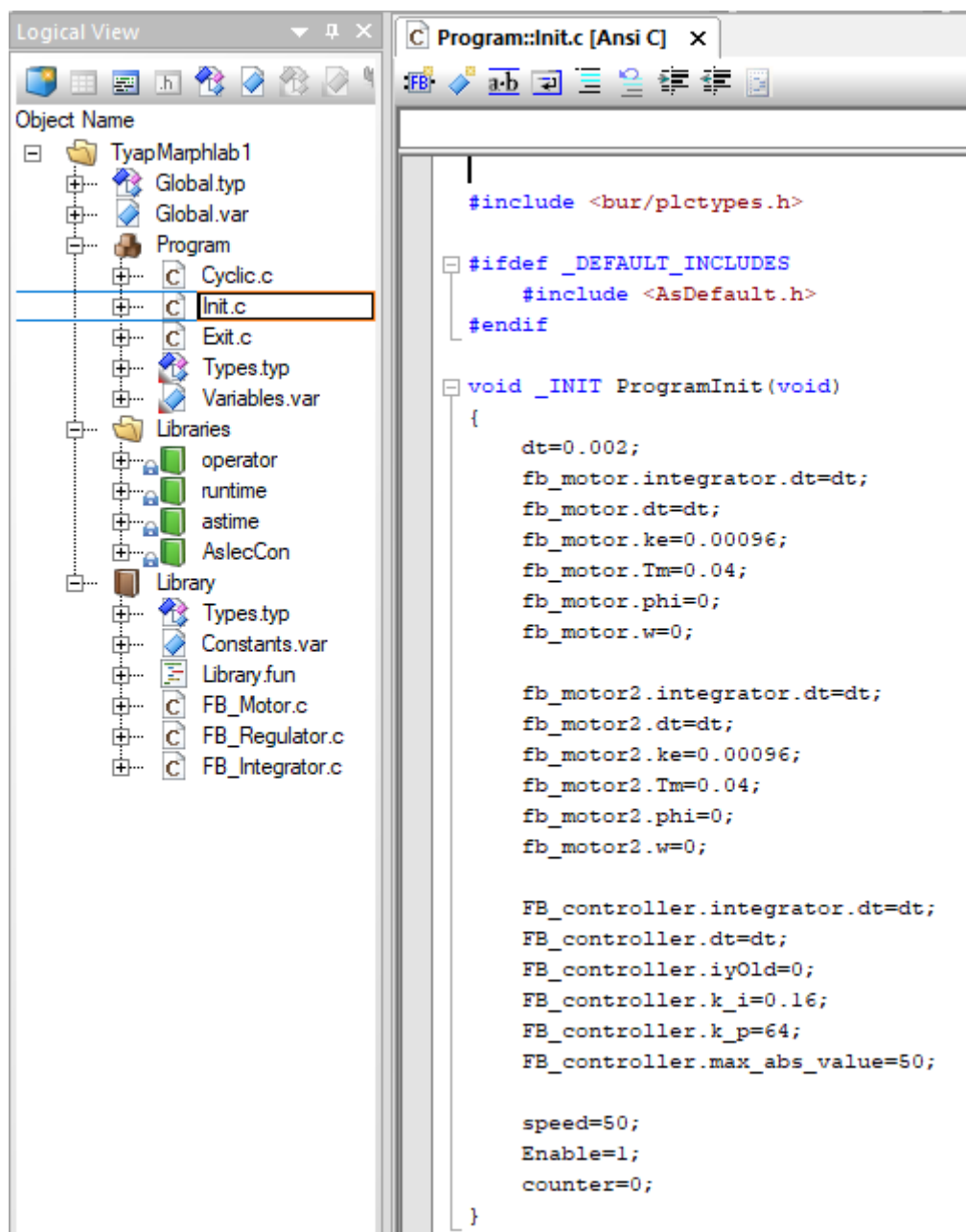


Рис. 8. Параметры fb_controller, fb_motor и fb_motor2

Снимаем графики с помощью средства Trace.

::Program.tc [Trace] X		
Name	State	Type
TARGET_CONFIGURATION		Module size=20.0 KByte Buffer size=122
Program:speed		REAL
Program:fb_motor.w		REAL
Program:counter		UINT
Program:fb_motor2.w		REAL

Рис. 9. Конфигурация Trace

Далее подбираем параметры регулятора для мотора. Начинаем с изменения интегрального коэффициента k_i при неизменных значениях k_p , max_abs_value (рис. 10-14).

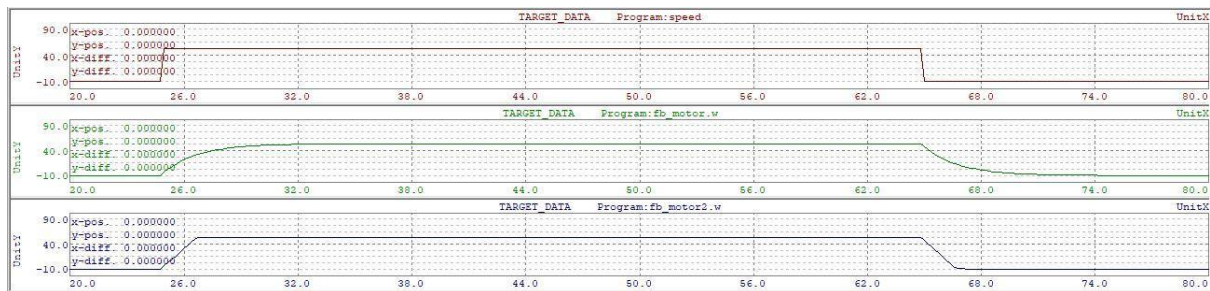


Рис. 10. График уставки, $k_i = 0.16$

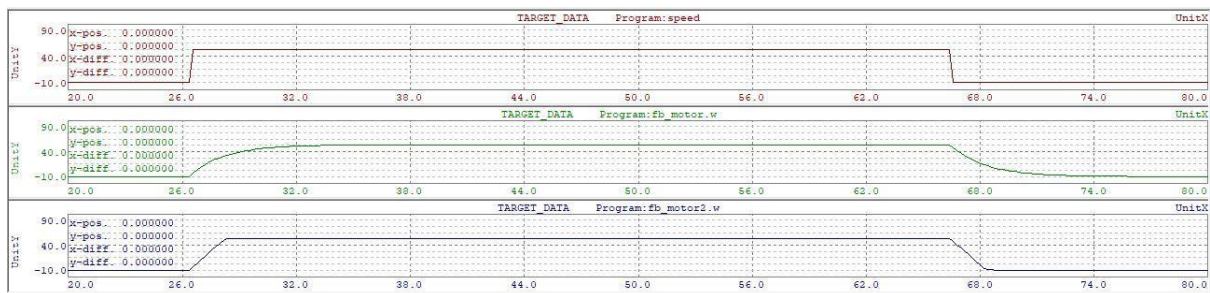


Рис. 11. График уставки, $k_i = 1.6$

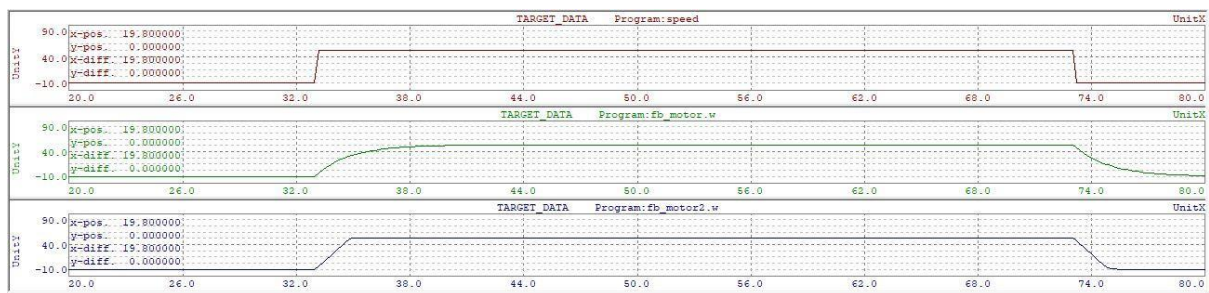


Рис. 12. График уставки, $k_i = 16$

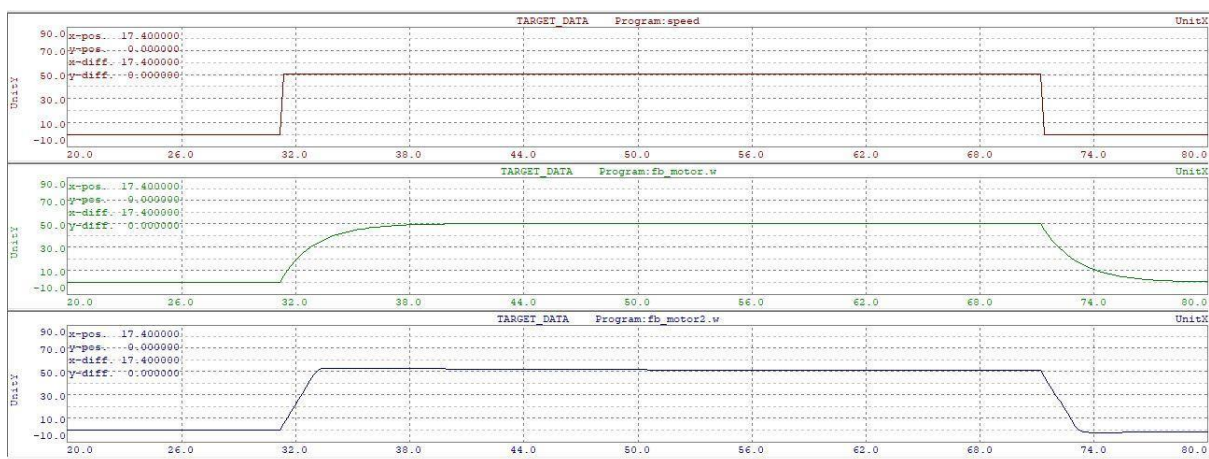


Рис. 13. График уставки, $k_i = 16000$

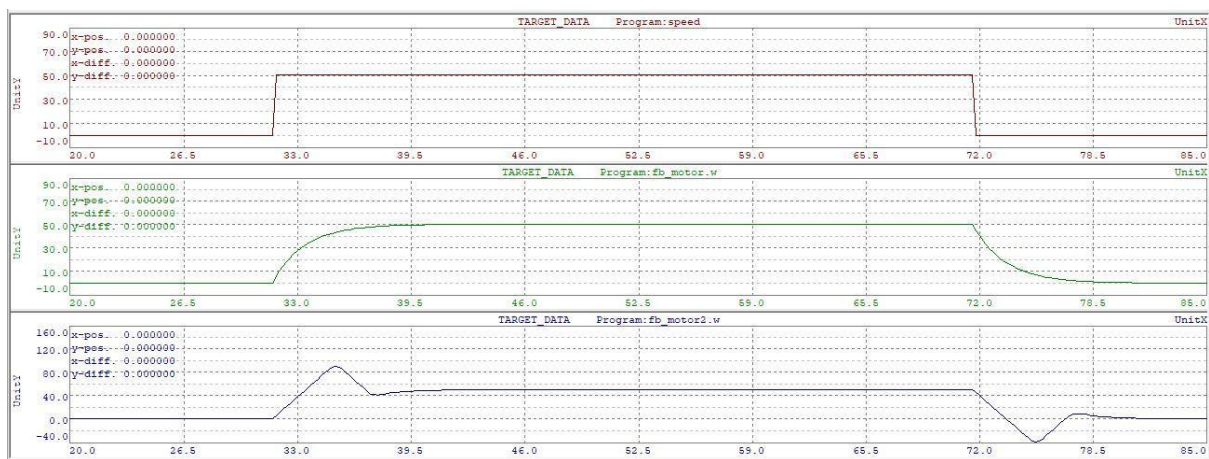


Рис. 14. График уставки, $k_i = 160000$

Оптимальное значение $k_i=0.16$. Теперь изменим значения \max_abs_value при неизменных значениях k_p и k_i (рис. 15-19).

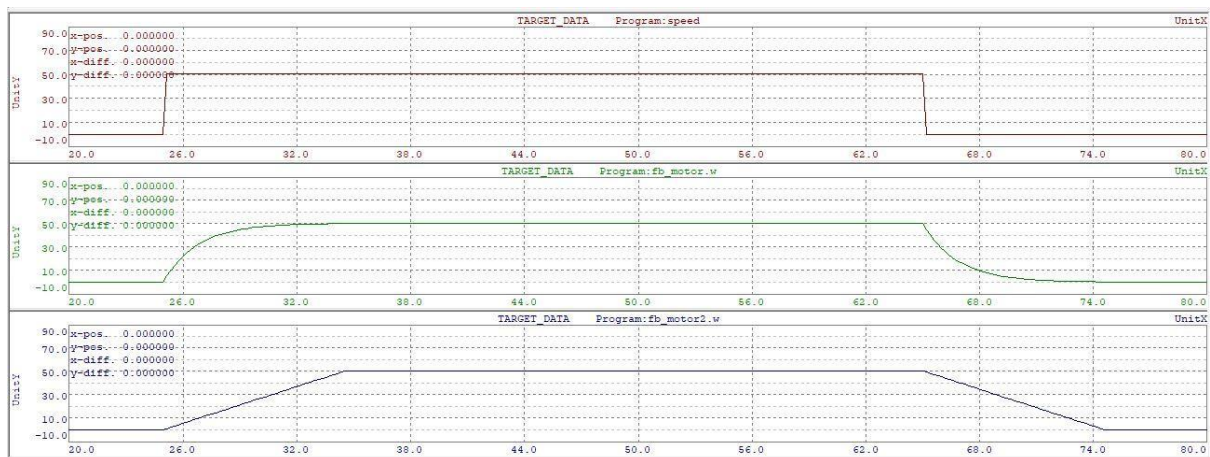


Рис. 15. График уставки, max_abs_value = 10



Рис. 16. График уставки, max_abs_value = 30

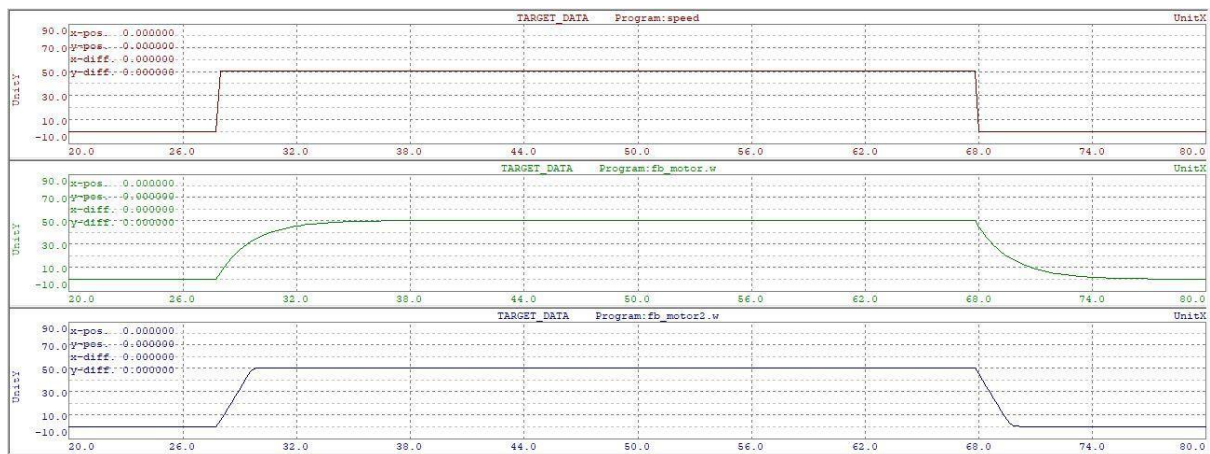


Рис. 17. График уставки, max_abs_value = 50

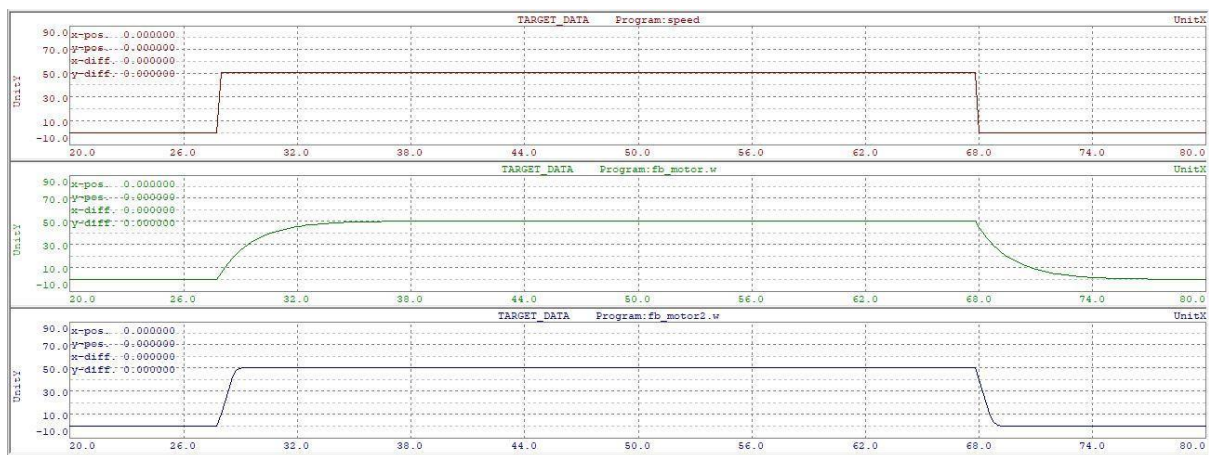


Рис. 18. График уставки, $\max_abs_value = 100$

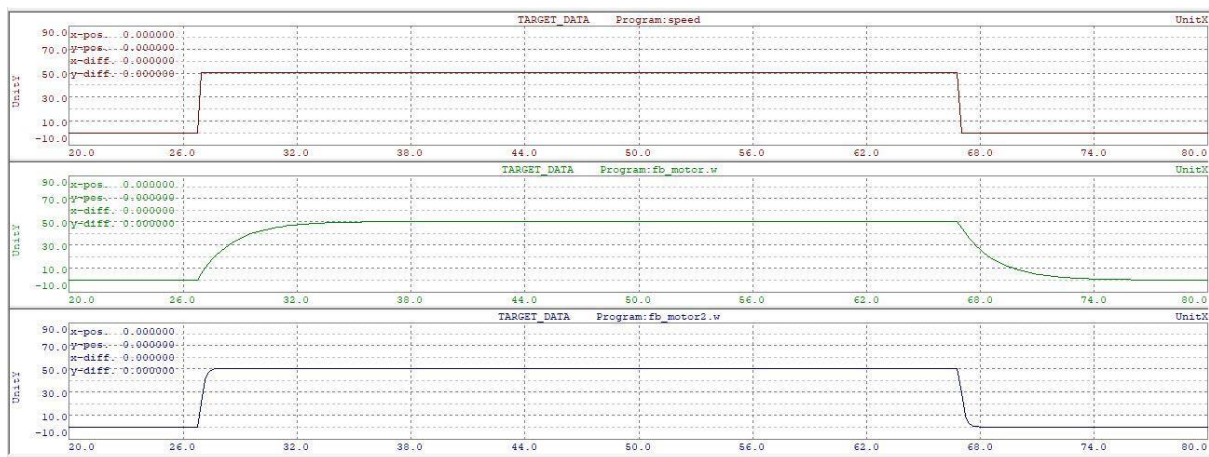


Рис. 19. График уставки, $\max_abs_value = 200$

Оптимальное значение $\max_abs_value=200$. Теперь изменим значения k_p при неизменных значениях \max_abs_value и k_i (рис. 20-22).

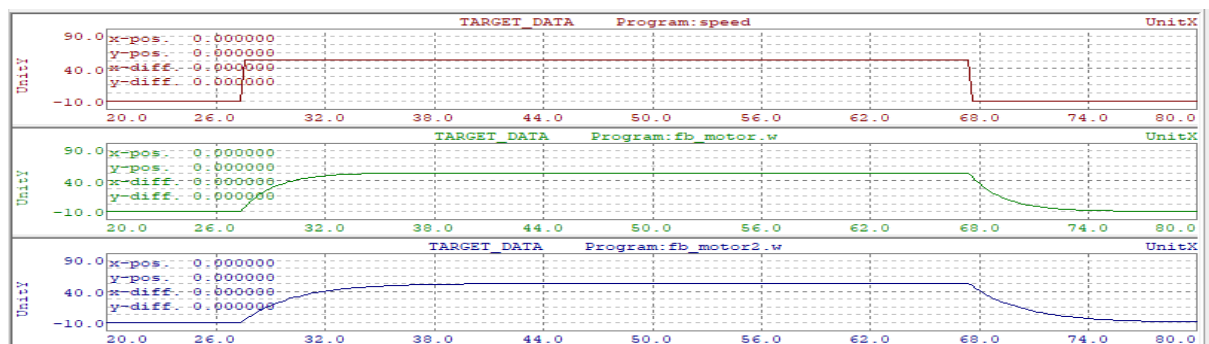


Рис. 20. График уставки, $k_p=0.64$



Рис. 21. График уставки, $k_p=6.4$

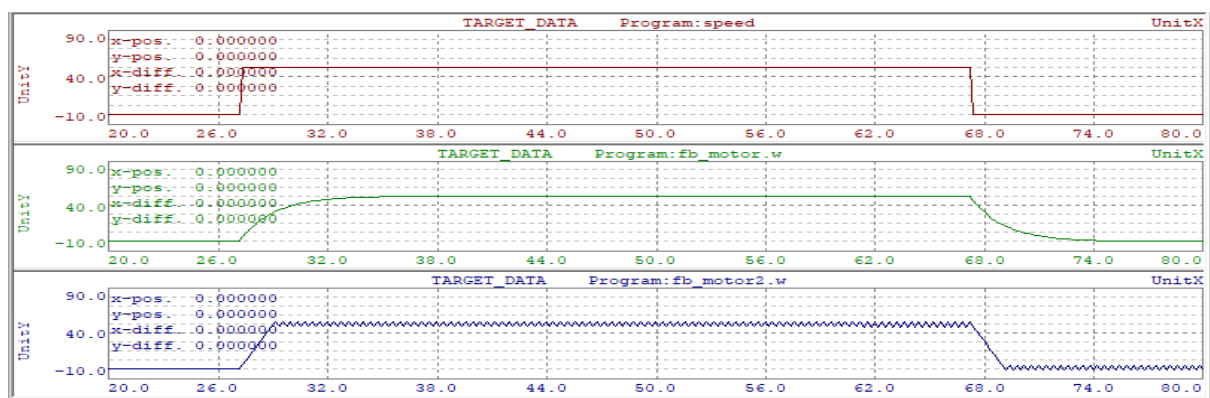


Рис. 22. График уставки, $k_p=64$

Оптимальное значение $k_p=6.4$.

Вывод:

В ходе выполнения лабораторной работы было освоено моделирование двигателя и его регулятора с использованием функциональных блоков в виртуальной среде Automation Studio. Мы применяли ступенчатое воздействие на уставку скорости ($\text{speed}=50$) и нашли оптимальные параметры для ПИ-регулятора при шаге расчета $\text{dt}=0.002$: $k_p=6.4$, $k_i=0.16$, и максимальное абсолютное значение выходного сигнала ограничено 200.