

К приказу № **2383-3** Приложение №1
от **15 СЕН 2026**

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования
«Новосибирский национальный исследовательский государственный университет»

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА НГУ

УТВЕРЖДАЮ

Проректор по учебной работе


С. Г. Саблина
(подпись)

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

технической направленности

**«ИИ в разработке электроники- программирование STM32 на CMSIS с помощью
искусственного интеллекта»**

Категория обучающихся: 12-17 лет

Срок реализации программы: 3 месяца.

Объем программы: 64 ак. часа.

Авторы-составители:

Быков Виталий Валерьевич,
Ведущий специалист ОДО ПИШ НГУ

Ёлкина Елизавета Викторовна
Специалист по УМР ОДО ПИШ НГУ

г. Новосибирск, 2026

Разработал(и):

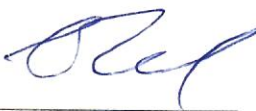
Ведущий специалист ОДО ПИШ НГУ Быков В.В.

Специалист по УМР ОДО ПИШ НГУ Ёлкина Е.В.

Согласовано:

Директор ПИШ НГУ,

д. ф.-м. н. С.В. Головин



Оглавление

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ.....	4
1.1. Краткое описание программы.....	4
1.2. Планируемые результаты обучения.....	6
2. УЧЕБНЫЙ ПЛАН.....	10
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК	11
4. РАБОЧИЕ ПРОГРАММЫ РАЗДЕЛОВ	15
4.1. Тематическое планирование программы.....	15
4.2. Содержание программы	18
5. ПРАВИЛА АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПРОГРАММЫ.....	23
6. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ.....	24
6.1. Формы реализации программы	24
6.2. Материально-техническое обеспечение	24
6.3. Кадровое обеспечение	24
6.4. Список информационных ресурсов	24
7. ВНЕШНЯЯ РЕЦЕНЗИЯ.....	27
8. ВНУТРЕННЯЯ РЕЦЕНЗИЯ.....	28

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Краткое описание программы

Программа технической направленности ориентирована на обучение школьников практическому программированию микроконтроллеров STM32 на уровне регистров через CMSIS (Cortex Microcontroller Software Interface Standard). Курс обеспечивает плавный переход с уровня Arduino на профессиональное программирование встраиваемых систем и формирует культуру ответственного применения ИИ-инструментов (DeepSeek, Claude, ChatGPT, Qwen, Gemini и др.) как помощника разработчика, а не средства автоматического написания кода.

Актуальность:

ИИ стремительно становится ежедневным инструментом инженера. Для школьников, начинающих путь в электронику, ИИ часто остаётся «чёрным ящиком» или источником копируемых без понимания решений. Программа решает эту проблему, встраивая ИИ в процесс обучения как помощника. Переход от Arduino к STM32 открывает сложность регистров и периферии, и именно на этом уровне ИИ особенно полезен: он предлагает заготовку кода, но ученик обязан проверить её по технической документации и понять каждую строку. Практико-ориентированный характер повышает мотивацию и формирует устойчивый интерес к техническим дисциплинам, а также даёт конкурентное преимущество при поступлении в технические вузы.

Целевая аудитория:

Для обучения принимаются все желающие обучающиеся (мальчики и девочки) без предварительной подготовки в возрасте 12–17 лет, проявляющие выраженный интерес к микроэлектронике, техническому творчеству, робототехнике, информационным технологиям и инженерным практикам.

Цель программы:

Развитие инженерных компетенций в области программирования 32-разрядных микроконтроллеров STM32 на уровне регистров (CMSIS) и формирование культуры ответственного применения ИИ-инструментов в разработке встраиваемого программного обеспечения через систему практико-ориентированных групповых занятий и индивидуальную проектную деятельность.

Задачи программы:

Предметные:

- формировать представление об архитектуре STM32 и принципе программирования «на регистрах» через CMSIS (тактирование, порты ввода-вывода, SysTick);
- формировать знания и умения по настройке базовой периферии: таймеры и ШИМ, АЦП, USART, подключение дисплея по I2C;
- сформировать навык базовой аппаратной отладки: ST-Link и основные команды GDB (точка останова, пошаговое выполнение, просмотр переменных и регистров);
- сформировать навык систематической работы с ИИ-сервисами: точный запрос, сравнение ответов, проверка кода по Reference Manual, разбор ошибок компиляции и логики;
- развивать инженерный навык диагностики «железо или код?»: сбор фактов приборами и отладчиком, их интерпретация;

- развивать интерес к встраиваемым системам, программированию и современным инструментам разработки.

Метапредметные:

- познавательные УУД: анализировать техническую документацию (Reference Manual, Data Sheet) и сверять с ней решения, в том числе от ИИ; декомпозировать задачу создания прибора на программные модули и этапы;
- регулятивные УУД: планировать этапы проекта (от ТЗ до защиты); оценивать функциональность устройства, вести журнал ошибок и решений;
- коммуникативные УУД: формулировать точный технический запрос к ИИ с указанием контекста (модель чипа, регистры, требования); презентовать решения, аргументировать выбор подхода и объяснять роль ИИ.

Личностные:

- сформировать ответственное отношение к применению ИИ: критическая проверка ответов, недопустимость использования непонятного кода;
- развивать инженерное мышление: диагностика по фактам, критическая оценка решений, поиск причин неисправностей;
- повышать мотивацию к профессиональному самоопределению в высокотехнологичных отраслях.

Особенности организации образовательного процесса. Реализация программы предусматривает сочетание очного и дистанционного форматов обучения в группах 8–12 человек. Очные занятия проводятся в соответствии с учебным расписанием и включают практическую работу под руководством преподавателя. Дистанционная часть включает теоретические блоки и инструктивные материалы. Дистанционный модуль обучающиеся осваивают самостоятельно параллельно с очными занятиями.

Направленность программы: техническая.

Категория слушателей: обучающиеся в возрасте 12–17 лет.

Срок реализации программы: 3 месяца.

Объем программы: 64 академических часа.

Режим занятий по программе: 4 очных академических часа в неделю (Форматы занятий на весь период обучения (по выбору преподавателя): 2 раза в неделю по 2 ак. ч. или 1 раз в неделю 4 ак. ч.). Отдельные темы программы сопровождаются дистанционными занятиями (7 ак.ч. за весь период обучения).

Продолжительность одного академического часа — 45 мин. Перерыв между занятиями — 15 минут.

Форма обучения: очно-заочная, с применением электронного обучения и дистанционных образовательных технологий.

Язык программы: русский язык.

Уровень программы: стартовый.

1.2. Планируемые результаты обучения

Направления работы со слушателями	Компетенции	Знания	Умения	Практический опыт
Переход с Arduino на STM32 через CMSIS и регистры	К-1. Способность перейти от Arduino к работе с STM32 «на регистрах» через CMSIS: настраивать порты ввода-вывода и таймбазу, пользоваться технической документацией и базовой отладкой, а также применять ИИ как помощника при изучении нового материала и поиске ошибок.	Знает, чем STM32 отличается от Arduino и что значит «программирование на регистрах». Назначение CMSIS и регистров тактирования (RCC) и портов ввода-вывода (вход/выход). Принцип работы системного таймера SysTick и неблокирующих задержек. Назначение отладчика и несколько базовых команд GDB. Возможности ИИ как помощника при освоении новой платформы и базовые правила запроса.	Создавать, собирать и прошивать проект для STM32 на CMSIS (PlatformIO + ST-Link). Включать тактирование периферии и настраивать выводы порта. Делать таймбазу на SysTick (аналог millis()) и работать без блокирующих delay. Переносить знакомую архитектуру (структура состояния, функции updateX()) на STM32. Обращаться к ИИ за подсказкой, объяснением кода и помощью в поиске ошибок, проверяя полученный ответ.	Мигание светодиодом и опрос кнопки на чистом CMSIS. Прошивка и запуск программы через ST-Link. Базовая отладка в GDB: точка останова, просмотр регистра и переменной. Использование ИИ как помощника при первых программах на CMSIS
Настройка и использование базовой периферии	К-2. Способность настраивать и использовать базовую периферию STM32 на CMSIS: таймеры и ШИМ, АЦП, последовательный порт	Что такое прерывание и таймер; как задать период; кратко ШИМ для плавной яркости. Как АЦП измеряет напряжение: разрядность, опорное напряжение, перевод в величину.	Получать события от таймера и обрабатывать кнопку через прерывание. Измерять датчики через АЦП и переводить показания в физическую величину. Выводить данные в терминал по	Генерация событий и ШИМ на таймере. Измерение аналоговых датчиков с помощью АЦП. Обмен данными с ПК по последовательному

	(Serial/USART) и простой дисплей, в том числе с привлечением ИИ для получения заготовок кода и их последующей проверки по документации.	Что такое UART и скорость обмена; зачем нужен вывод в терминал. Виды простых дисплеев (I2C LCD/OLED) и как подключить дисплей.	Serial и принимать простые команды. Выводить информацию на дисплей.	порту; отладочный вывод. Вывод данных на экран. Получение заготовок настройки периферии у ИИ и их проверка по Reference Manual.
Использование ИИ как помощника разработчика	К-3. Способность использовать бесплатные ИИ-сервисы как помощника: ставить точные запросы для кода на CMSIS, искать с ИИ ошибки компиляции и логики, объяснять поведение программы — и обязательно проверять ответы ИИ по документации.	Обзор бесплатных ИИ (DeepSeek, Claude, ChatGPT, Qwen, Gemini и др.), их сильные и слабые стороны и ограничения. Четыре правила хорошего запроса применительно к коду на регистрах. Типичные ошибки ИИ на конкретном чипе и как их ловить. Как искать ошибки с ИИ: текст ошибки компилятора, факты измерений, данные отладчика.	Сравнивать ответы разных ИИ и выбирать корректный. Давать ИИ контекст: модель чипа, нужные регистры. Проверять код ИИ по документации и находить ошибки. Использовать ИИ для разбора ошибок компиляции и поведения программы. Понимать, когда ИИ доверять нельзя.	Постановка точного запроса к ИИ для настройки периферии на CMSIS. Проверка кода ИИ по документации. Поиск ошибок компиляции и логики с помощью ИИ. Диагностика «железо или код?» с ИИ, приборами и GDB.
Проектная деятельность — создание простого законченного прибора	К-4. Способность самостоятельно сделать простой законченный прибор-автоматику с экраном на STM32: от ТЗ и архитектуры до сборки, отладки с помощью ИИ и защиты.	Этапы небольшого проекта: ТЗ, архитектура, реализация по частям, сборка, отладка, тест, защита. Как устроена архитектура прибора (датчик → логика → действие → экран). Как отлаживать устройство с ИИ, отладчиком и приборами.	Формулировать простое ТЗ и архитектуру прибора. Реализовывать части устройства на CMSIS с помощью ИИ и проверять их. Собирать устройство целиком и отлаживать его. Тестировать по чек-листу и защищать проект.	Сборка простого прибора-автоматики на STM32 с экраном. Отладка с помощью ИИ, GDB и приборов. Короткое описание проекта и его публичная защита.

		Как подготовить короткую защиту проекта.		
--	--	--	--	--

Предметные результаты

- Обучающийся способен объяснить архитектуру микроконтроллера STM32 и принцип программирования «на регистрах» через CMSIS: назначение и взаимодействие регистров тактирования (RCC), портов ввода-вывода (GPIO), системного таймера SysTick; демонстрирует понимание отличия STM32 от Arduino и сути подхода «программирование на регистрах».
- Обучающийся умеет настраивать и использовать базовую периферию STM32 на CMSIS: таймеры и ШИМ, АЦП (с переводом показаний в физическую величину), последовательный порт USART, подключение дисплея по I2C; может получать события от таймера и обрабатывать кнопку через прерывание.
- Обучающийся умеет работать с программатором-отладчиком ST-Link и базовыми командами GDB (точка останова, пошаговое выполнение, просмотр переменных и регистров); демонстрирует навык диагностики состояния «железо или код?» с помощью измерительных приборов и отладчика.
- Обучающийся способен применять ИИ-сервисы как помощника разработчика: умеет ставить точный технический запрос с указанием модели чипа и нужных регистров; может сравнивать ответы разных ИИ и проверять предложенный код по Reference Manual; демонстрирует использование ИИ для разбора ошибок компиляции и логики, понимая границы доверия к нему.

Метапредметные результаты

- Обучающийся умеет анализировать техническую документацию (Reference Manual, Data Sheet) и сверять с ней предлагаемые решения, в том числе полученные от ИИ; может декомпозировать задачу создания прибора на программные модули и этапы.
- Обучающийся умеет планировать этапы создания проекта (от технического задания до защиты); может оценивать функциональность устройства и соответствие заявленным требованиям; демонстрирует ведение журнала ошибок и решений.
- Обучающийся способен формулировать точный технический запрос к ИИ-инструменту с указанием контекста (модель чипа, регистры, требования); умеет презентовать решения, аргументировать выбор подхода и объяснять роль ИИ в выполненной работе.
- Обучающийся умеет организовывать совместную деятельность с педагогом и сверстниками; может работать индивидуально и в группе, распределять роли и нести ответственность за общий результат проекта; демонстрирует навыки учебного сотрудничества.

Личностные результаты

- Обучающийся демонстрирует ответственное отношение к применению ИИ-инструментов: критически проверяет ответы ИИ, не допускает использования непонятного кода; способен понимать, что решение за инженером, а ИИ — лишь помощник.
- Обучающийся демонстрирует инженерное мышление: умеет диагностировать по фактам, критически оценивать решения и находить причины неисправностей; может проявлять настойчивость, точность и аккуратность в процессе отладки и доведения проекта до работающего прототипа.
- Обучающийся демонстрирует устойчивый интерес к встраиваемым системам, программированию и современным инструментам разработки; способен к осознанному выбору образовательной и профессиональной траектории в высокотехнологичных отраслях.
- Обучающийся способен к рефлексии, самоанализу и непрерывному обучению; демонстрирует понимание социальной значимости технических профессий и роли инженера в развитии общества; может выстраивать собственную траекторию саморазвития.

2. УЧЕБНЫЙ ПЛАН

Компоненты образовательной программы	Общая трудоемкость, ч	Учебные занятия, ч			Аттестация ч.	Форма аттестации	Код компетенции
		всего	теория	практика			
Раздел 1. От Arduino к STM32. Основы CMSIS	15	15	5	10			К-1
Раздел 2. Периферия STM32: таймеры, АЦП, Serial и дисплей	15	15	5	10			К-2
Раздел 3. Искусственный интеллект как помощник разработчика	15	15	5	10			К-3
Раздел 4. Проект — простая автоматика с экраном (частично дистанционно)	10	10	8,5	1,5			К-4
Входной контроль	1				1	Дистанционное тестирование	
Экскурсия	4	4	0	4			
Итоговая аттестация	4				4	Защита проекта	К-1, К-2, К-3, К-4
ИТОГО	64	59	23,5	35,5	5		