

АРХИТЕКТУРА ПАРАЛЛЕЛЬНЫХ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ УСТРОЙСТВ ОБРАБОТКИ СИГНАЛОВ

Андреев В. С., Бутусов Д. Н., Красильников А. В.

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)», Санкт-Петербург, Россия (197376, Санкт-Петербург, ул. Проф. Попова, д. 5), e-mail: valery.s.andreev@gmail.com

В статье предлагается модификация параллельного способа декомпозиции динамических систем, описываемых передаточными функциями. Данный способ пригоден для создания цифровых устройств управления и обработки сигналов с более высокими точностными и скоростными характеристиками по сравнению с устройствами того же класса за счет распараллеливания вычислительного процесса. Приводится алгоритм программы, позволяющий осуществить автоматический переход от описания динамической системы в виде передаточной функции либо набора нулей полюсов и коэффициента передачи к сумме передаточных функций не более второго порядка. Анализ достоверности предлагаемого способа описания производится на примере синтеза и моделирования эллиптического фильтра 9 порядка в среде LabVIEW. Производится анализ эффективности и достоверности предлагаемого способа декомпозиции во временной и частотной области. Показано, что степень эквивалентности цифрового фильтра, представленного предлагаемым способом относительно классической реализации, достаточна для решения современных задач цифровой обработки сигналов.

Ключевые слова: компьютерное моделирование, передаточная функция, цифровая обработка сигналов, параллельные вычисления.

ARCHITECTURE OF THE HIGH PERFORMANCE PARALLEL SIGNAL PROCESSING DEVICES

Andreev V. S., Butusov D. N., Krasilnikov A. V.

Saint-Petersburg State Electrotechnical University, Saint-Petersburg, Russia. (197376, 5, Professora Popova st., Saint-Petersburg, Russia), e-mail: valery.s.andreev@gmail.com

The article describes the modification of design method, based on parallel decomposition of dynamic systems described by transfer function. This method is suitable for the development of digital control and signal processing systems with higher accuracy and speed characteristics by parallelizing the computation process. The authors give the program algorithms that allow the automatic conversion for description of the dynamical system in the form of a transfer function or zero-pole-gain to the sum of first and second order transfer functions. A validation of the described method is made by creating and simulating elliptic filter of order 9 in LabVIEW development system. Analysis of the efficiency and accuracy of the method of decomposition in the time and frequency domain is performed. It is shown that the proposed implementation of a digital filter is equivalent to the classical implementation, and is suitable for solving the current problems of digital signal processing.

Keywords: computer modeling, transfer function, digital signal processing, parallel computing.

Введение

Цифровые фильтры применяются практически во всех областях техники, требующих обработки сигналов. Сюда относятся спектральный анализ, обработка изображений, видео, звук и др. Классический синтез цифровых фильтров заключается в получении описывающей его передаточной функции минимального порядка, соответствующей заданным характеристикам [3]. Это обусловлено тем, что при данном подходе фильтры реализуются аппаратно на базе R-L-C-элементов интегральных схем.

С появлением таких аппаратных платформ, как программируемые логические интегральные схемы (ПЛИС, FPGA), цифровые сигнальные процессоры (DSP) и системы на кристалле (SoC), стал осуществим новый подход, заключающийся в программной реализации цифровых фильтров. Данный тип синтеза позволяет улучшить точностные характеристики отфильтрованного сигнала за счет повышения порядка фильтра, однако имеет и свои недостатки: низкая скорость обработки сигнала, зависящая от быстродействия исполняющего процессора, сложность реализации цифровых фильтров высокого порядка, большие времена запаздывания сигнала, возрастающие с ростом порядка фильтра.

Современные программные пакеты моделирования, такие как MATLAB\Simulink и NI LabVIEW, позволяют синтезировать различные цифровые фильтры с желаемыми характеристиками и автоматически генерировать программный код под множество аппаратных платформ, список которых постоянно расширяется [4]. Таким образом, трудоемкость процесса проектирования устройств фильтрации значительно сокращается. Однако для увеличения точности обработки сигнала необходимо увеличивать порядок передаточной функции искомого цифрового фильтра, что приводит к росту времени обработки сигнала [5].

Представляется целесообразным распараллелить передаточную функцию фильтра, синтезируемого в инструментальном пакете моделирования, представив ее в виде суммы передаточных функций меньшего порядка. Такой подход позволит увеличить быстродействие устройств фильтрации без увеличения аппаратных затрат.

Описание решения

Поскольку любая передаточная функция является правильной рациональной дробью, то в соответствии со свойствами рациональных функций ее можно представить в виде суммы элементарных дробей (звеньев), что широко используется при аналитическом интегрировании.

$$H(s) = \frac{B(s)}{A(s)} = \sum_{i=1}^k \frac{r_i}{(s - p_i)} \quad (1)$$

, где В и А – многочлены с вещественными коэффициентами, причем степень В меньше степени А. Коэффициенты p_i – вещественные или комплексные. Комплексные коэффициенты образуют сопряженные пары. Коэффициент r_i – вещественная постоянная.

Поскольку реализация вычислительного алгоритма с использованием комплексных чисел затруднительна, необходимо привести звенья к виду, когда они не содержат комплексных коэффициентов. Перемножим звенья с комплексно-сопряженными коэффициентами p_i и получим передаточные функции вида:

$$H_i(s) = \frac{l_i s + r_i}{s^2 + m_i * s + p_i} \quad (2)$$

, где l_i, r_i, m_i, p_i – вещественные числа.

Таким образом, любую передаточную функцию можно представить в виде суммы отдельных передаточных функций, порядок которых не превышает второго (3). При этом все коэффициенты суммы являются вещественными числами, что важно при реализации передаточных функций на специализированных вычислителях.

$$H(s) = \sum_{i=1}^n H_i(s) \quad (3)$$

, где $H_i(s)$ – передаточная функция с вещественными коэффициентами либо первого порядка (1), либо второго порядка (2).

Для увеличения точности обработки сигнала часто прибегают к увеличению порядка описывающей фильтр передаточной функции, что приводит к росту времени обработки сигнала. При реализации фильтра в форме (3) рост порядка передаточной функции будет оказывать незначительное влияние на общее время вычислений при реализации на платформах, поддерживающих параллельные вычисления, таких как ПЛИС или многоядерные системы, поскольку каждое интегрирующее звено из общей суммы может вычисляться независимо от остальных. Таким образом, можно достичь выигрыша в точности вычислений за счет увеличения аппаратных затрат, при этом сохраняя общее время обработки сигнала практически неизменным.

Рассмотрим пример параллельной реализации эллиптического фильтра в среде LabVIEW с использованием функций MathScript. Наиболее распространенными способами математического описания фильтров в средах MATLAB и LabVIEW являются передаточная функция (Transfer Function, tf) и набор нулей полюсов и коэффициента передачи (Zero-Pole-Gain, zpk). Синтезируем искомый фильтр в обеих формах tf и zpk (4) и проанализируем результаты. На рисунке 1 показан фрагмент блок-диаграммы, реализующий синтез эллиптического фильтра в узле MathScript Node в виде передаточной функции и набора нулей полюсов и коэффициента передачи. Параметры синтезируемого эллиптического фильтра следующие:

- порядок передаточной функции – 9;
- величина пульсаций АЧХ в полосе пропускания – 0,001 дБ;
- подавление сигнала в полосе задерживания – 50 дБ;
- частота среза – 7,8 Гц;
- вид фильтра – фильтр нижних частот;
- тип передаточной функции – непрерывная (s-область);

```

1 [b,a]=ellip(9,0.001,50,7.8,'low','s');
2 H=tf(b,a);
3 [z,p,k]=ellip(9,0.001,50,7.8,'low','s');
4

```

Рисунок 1. Фрагмент блок-диаграммы виртуального прибора, реализующий синтез эллиптического фильтра в узле MathScript Node в виде tf и zpk

Передаточная функция синтезированного фильтра (4).

$$\frac{0,192939S^8 + 145,675S^6 + 33787,8S^4 + 3,11113 * 10^6S^2 + S^9 + 24,8317S^8 + 466,97S^7 + 5963,23S^6 + 60373,1S^5 + 474624S^4 + 2,93918 * 10^6S^3 + 9,97145 * 10^7}{+1,39547 * 10^7S^2 + 4,65209 * 10^7S + 9,97145 * 10^7} \quad (4)$$

На рисунке 2 показана амплитудно-частотная характеристика синтезированного эллиптического фильтра.

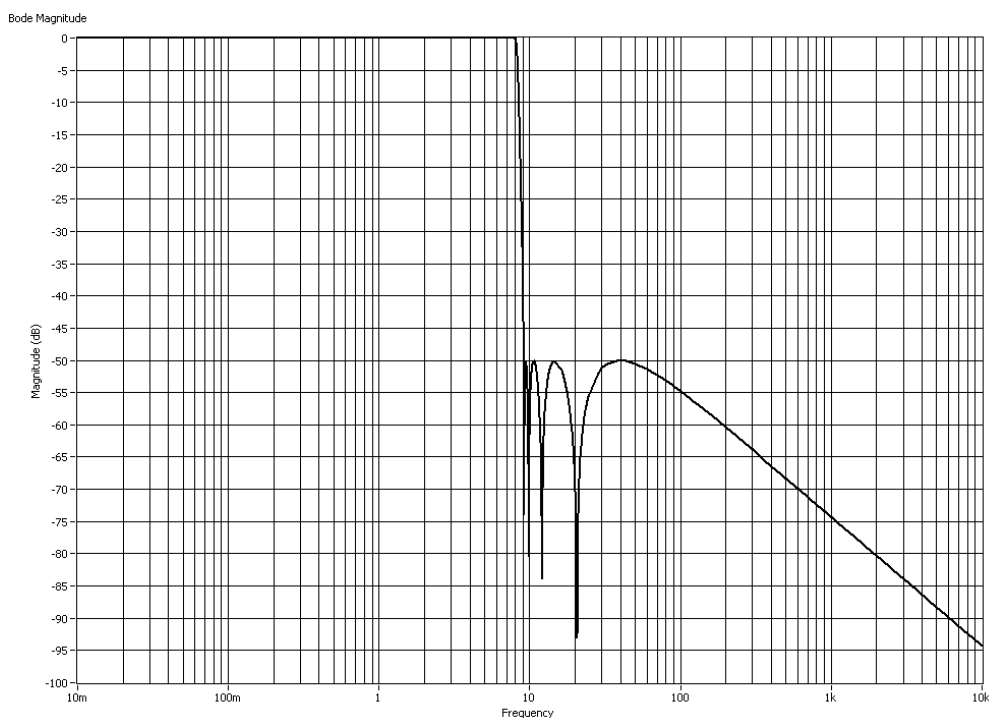


Рисунок 2. Амплитудно-частотная характеристика синтезированного эллиптического фильтра 9 порядка

Целесообразно автоматизировать процесс перехода от стандартного математического описания фильтра к описанию в форме (3). Для этого необходимо программно реализовать два алгоритма преобразования для каждой формы представления фильтра (tf и zpk). На рисунке 3(a) показан узел MathScript Node, содержащий текст программы для перехода от

формы описания в виде *tf* к массиву передаточных функций первого и второго порядка, сумма которых эквивалентна исходной передаточной функции фильтра. На рисунке 3(б) показаны узлы MathScript Node с текстом скрипта перехода от *zpk* к аналогичному массиву передаточных функций.

a) *tf to sum(tf)*

```

1 trns_in=tf(A,B)
2 [R, P, K] = residue(A,B);
3 n = length(R);
4 i = 1;
5 trns = [];
6 while(i <= n)
7     if imag(P(i)) ~= 0 %если комплексный
8         num = R(i)*[1 -P(i+1)] + R(i+1)*[1 -P(i)];
9         den = conv([1 -P(i)],[1 -P(i+1)]);
10        i = i+1;
11    else %если вещественный
12        num = R(i);
13        den = [1 -P(i)];
14    end
15    i = i+ 1;
16    trns = [trns tf(num,den)];
17 end
18 if length(K) > 0
19     trns = [trns tf(K,1)];
20 end
21

```

б) *zpk to sum(tf)*

```

1 n = length(p);
2 pPar = zeros(n,1);%парные
3 ind = zeros(n,1);%индексатор добытых из массива p
4 k = 1;
5 for i=1:n
6     if ind(i) == 0 %если корень неюзаный
7         pPar(k) = p(i);
8         ind(i) = 1;
9         k = k + 1;
10    end
11    if imag(p(i)) ~= 0 %ага, комплексный! Найдем пару
12        invImag = -imag(p(i));
13        j = k;
14        while(j < n)
15            j = j+1;
16            if imag(p(j)) == invImag %если нашли
17                pPar(k) = p(j);
18                k = k + 1;
19                ind(j) = 1;
20                break;
21            end
22        end
23    end
24    %а если не комплексный, то ничего не делаем
25 end
26 p = pPar;

```

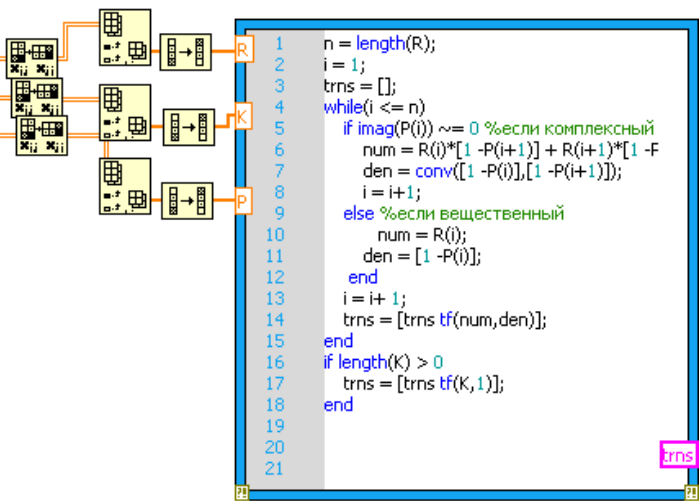


Рисунок 3. Алгоритмы перехода от одной передаточной функции к набору звеньев в узлах Mathscript Node

Преобразование исходной передаточной функции синтезированного фильтра (4) к виду (3) из форм представления в виде *tf* и *zpk*, полученных на этапе синтеза эллиптического фильтра (рисунок 1), дает практически идентичные результаты. Коэффициенты полученных передаточных функций различаются лишь в 14 знаке после запятой. Общий вид полученной суммы представлен ниже в выражении (5).

$$H(s) = \frac{14,2618}{s + 6,78067} + \frac{-15,3998s - 6,97511}{s^2 + 10,162s + 52,2705} + \frac{-1,39453s - 72,4776}{s^2 + 5,11543s + 61,4499} + \frac{3,08042s + 8,89451}{s^2 + 2,15735s + 66,6542} + \frac{-0,35494s + 5,23091}{s^2 + 0,577364s + 68,6879} \quad (5)$$

Для анализа параллельной декомпозиции цифрового фильтра (5) во временной и частотной области соберем модель в Simulation Module для вариантов синтеза в tf и в zpk, как показано на рисунке 4.

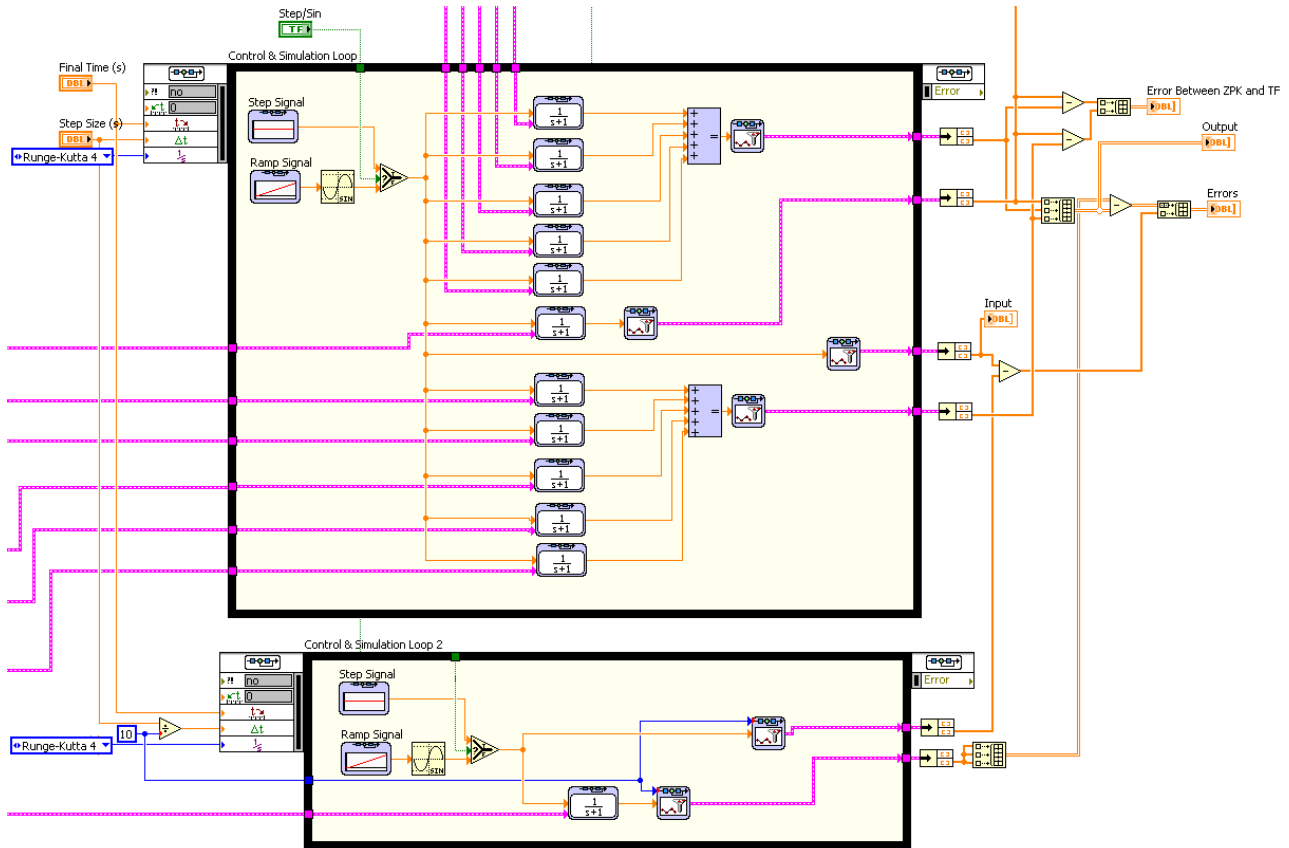


Рисунок 4. Фрагмент блок-диаграммы, реализующий параллельный фильтр в узле Simulation Loop для вариантов синтеза в tf и в zpk

На рисунке 5 приводятся погрешности параллельной модели эллиптического фильтра (5) во временной области для вариантов перехода к параллельным звеньям от единой передаточной функции высокого порядка (tf) и от представления в виде нулей и полюсов синтезированного фильтра (zpk). Погрешность вычисляется относительно исходной передаточной функции (4), моделируемой средствами инструментальной среды LabVIEW.

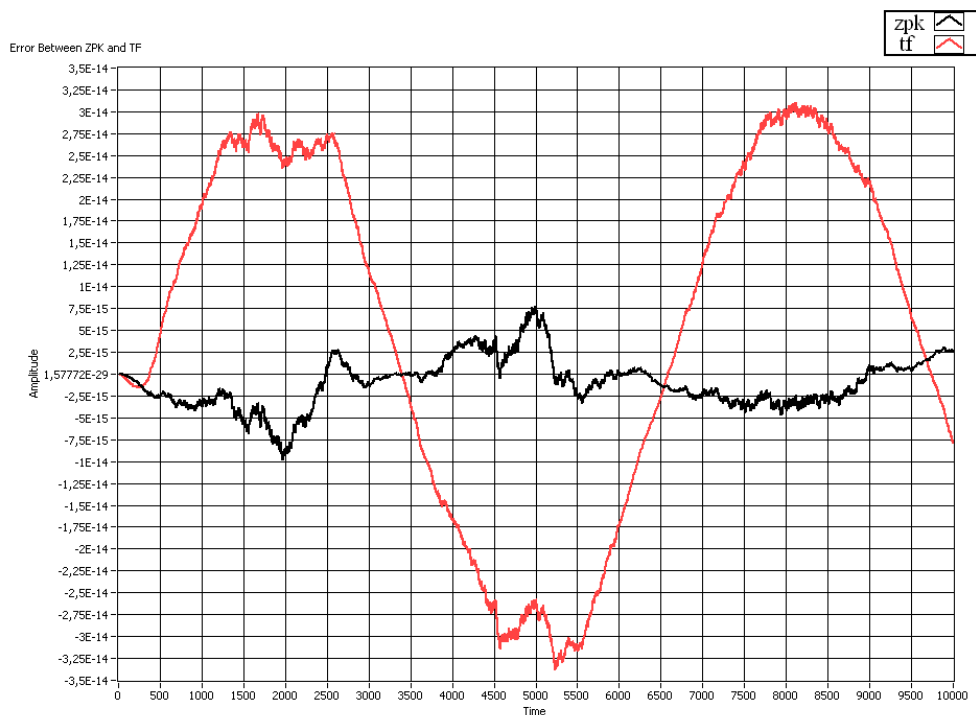


Рисунок 5. Сравнение погрешностей параллельной реализации фильтра относительно исходной передаточной функции для вариантов синтеза в tf и в zpk

На рисунке 5 можно увидеть, что погрешность при изначальном синтезе фильтра в форме zpk практически на порядок меньше погрешности передаточной функции фильтра, синтезированного в виде tf. Это объясняется неточностью нахождения корней полиномов числителя и знаменателя при разбиении на сумму параллельных звеньев, в то время как форма zpk изначально представляет собой набор этих корней. Для цифровой реализации необходимо заменить непрерывные звенья первого и второго порядка на цифровые интегрирующие процессоры, основанные на численных методах интегрирования. Представляется целесообразным использовать численные методы, ориентированные на аппаратную реализацию, в т. ч. пригодные для распараллеливания вычислительного процесса. Такими свойствами, например, обладает метод, описанный в работе [1].

Заключение

Несмотря на выявленные недостатки, можно заключить, что предлагаемое разбиение цифровых фильтров, синтезированных как в форме zpk, так и в форме tf, на сумму интегрирующих звеньев первого и второго порядка имеет высокую степень эквивалентности относительно исходной передаточной функции, достаточную для современных задач цифровой обработки сигналов. В качестве аппаратной платформы для реализации полученных компьютерных моделей цифровых устройств рекомендуется использовать оборудование с вычислителями на базе ПЛИС, поскольку данный класс устройств обладает

естественным параллелизмом, что позволяет увеличивать порядок фильтра для достижения необходимой точности, не уменьшая при этом скорость обработки сигнала.

Работа выполнена при финансовой поддержке Министерства образования и науки Российской Федерации (Договор № 14.В37.21.0589).

Список литературы

1. Бутусов Д. Н., Жуков К. Г. Реализация цифровых фильтров методом совместного интегрирования // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. – С.: СПбГПУ, 2009. – № 6. – С. 26-34.
2. Жуков К. Г., Бутусов Д. Н. Коррекция погрешности решения уравнения гармонического осциллятора методом последовательного интегрирования // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. – 2010. – № 6 (113). – С. 135-144.
3. Каплун Д., Меркучева Т. Исследование реализации цифровых фильтров с симметричными амплитудными функциями // Компоненты и технологии [электронное издание]. – 2009. – № 10. – Режим доступа к публикации: http://kit-e.ru/articles/circuit/2009_10_108.php
4. Кехтарнаваз Н., Ким М. Н. Цифровая обработка сигналов на системном уровне с использованием LabVIEW: Пер. с английского под ред. д-ра техн. наук, проф. В. К. Макухи. – М.: Издат. дом «Додэка - XXI», 2007.
5. Оппенгейм А., Шафер Р. Цифровая обработка сигналов. – Изд. 2-е, испр. – М.: Техносфера, 2009.

Рецензенты:

Анисимов В. И., д.т.н., профессор кафедры систем автоматизированного проектирования, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)», г.Санкт-Петербург.

Сольницев Р. И., д.т.н., профессор кафедры систем автоматизированного проектирования, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)», г.Санкт-Петербург.