**СПЕЦИАЛЬНЫЕ ВОПРОСЫ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ**

1. **Основные положения**

Одной из целей данной выпускной квалификационной работы является вычисление среднего времени анализа в частотной области на одной временной позиции. Данный этап является очень важным для оценки эффективности предложенного метода поиска сигнала, так как именно благодаря этому можно оценить эффективность алгоритма. В связи с этим, очень важно точно и правильно описать модель поиска сигналов, чтобы полученная оценка максимально описывала действительность.

Разработанный алгоритм реализован программным способом и, таким образом, был разработан соответствующий метод определения среднего времени поиска сигнала в частотной области. Написание программы осуществлялось на персональном компьютере с использованием среды технический разработки Matlab. Для обеспечения комфорта, производительной работы и безопасности необходимо, чтобы интерфейс Matlab соответствовали ГОСТ Р ИСО 9241.

Эргономика – одна из научных дисциплин, которая изучает взаимодействие человека с другими элементами системы [ссылка на ГОСТ Р ИСО 9241]. Жизнь без ПК на данный момент сложно представить, ведь каждый человек использует его не только на работе или учебе, но и дома. Поскольку взаимодействие с компьютером осуществляется через интерфейсы программ, важно поддержание их эргономичности, в целях обеспечения комфортной и безопасной среды для человека. В случае длительной и частой работы при несоблюдении требований ГОСТ Р ИСО 9241 у пользователя может возникнуть снижение продуктивности, а также могут появиться проблемы со здоровьем.

Поэтому, чтобы снизить риски для безопасности человека, необходимо соответствие используемого и разрабатываемого алгоритма требованиям ГОСТ Р ИСО 9241.

1. **Эргономика интерфейса Matlab**

В данном разделе анализируется пригодность используемой среды разработки Matlab с точки зрения эргономики.

Человек взаимодействует с ПО с помощью пользовательского интерфейса. Пользовательский интерфейс (ПИ) – компоненты интерактивной системы, которые предоставляют пользователю информацию и возможность управления для выполнения задач [ссылка на ГОСТ Р ИСО 9241]. Таким образом, с помощью пользовательского интерфейса человек ведет диалог с ПК: управляет его работой, посылает различные задачи, получает на них ответы на запросы и обменивается другой информацией компьютером.

ГОСТ Р ИСО 9241 определяет следующие принципы осуществления общения между человеком и пользовательским интерфейсом разрабатываемого программного обеспечения:

* пригодность для выполнения задачи;
* информативность;
* соответствие ожиданиям пользователя;
* пригодность для обучения;
* управляемость;
* устойчивость к ошибкам;
* пригодность для индивидуализации.

Для оценки соответствия Matlab требуемым принципам разработки диалога, далее будет подробно рассмотрено каждое из них, применительно к используемой среде разработки.

На рисунке 1 представлено окно, которое появляется на старте работы среды Matlab, возникающее при запуске программы.

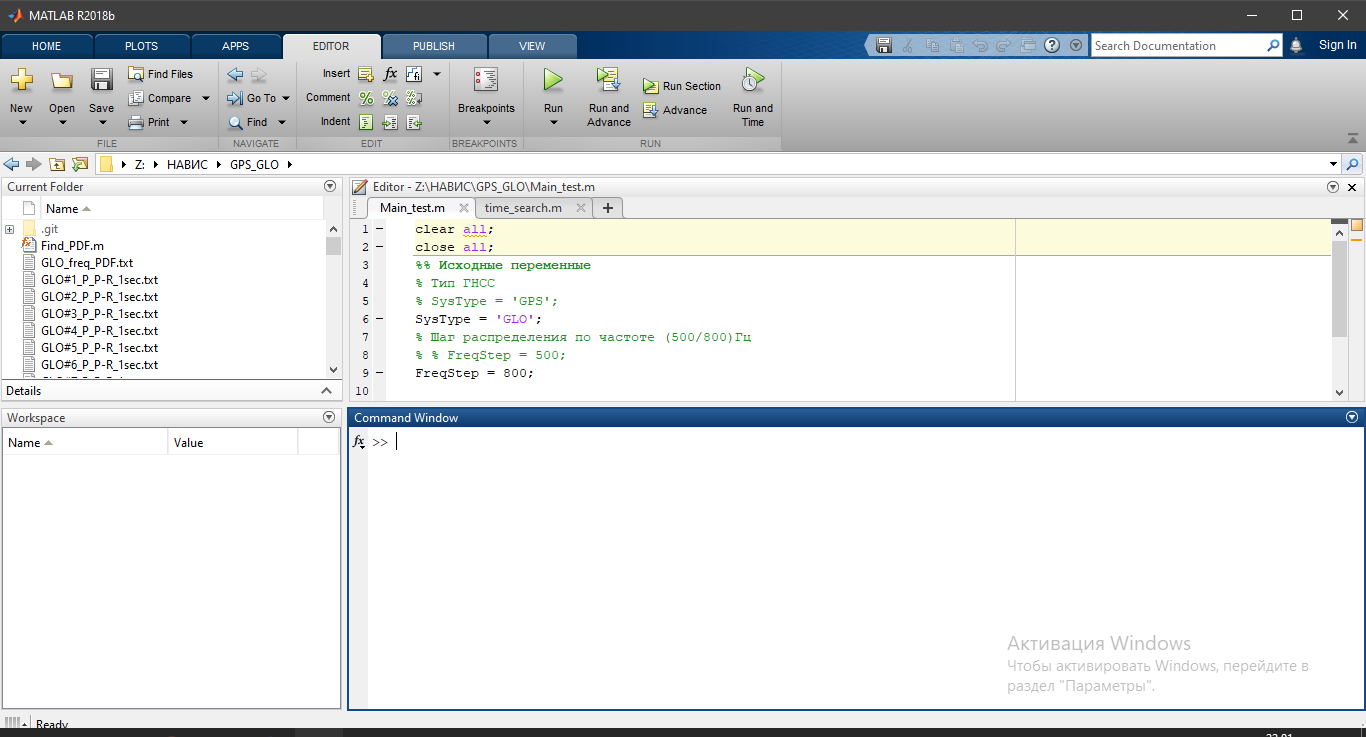


Рисунок 1 – Стартовое окно Matlab

Среда разработки предоставляет высокий уровень информативности. В основном части окна пользовательского интерфейса, в левой верхней части находятся вкладки и кнопки, позволяющие быстро перейти к осуществлению требуемой операции: открыть проект, создать новый, открыть пример, выбрать нужной для анализа инструмент и т.д. При создании нового проекта, всплывает дополнительное окно, в котором можно указать путь расположение проекта (рисунок 2).

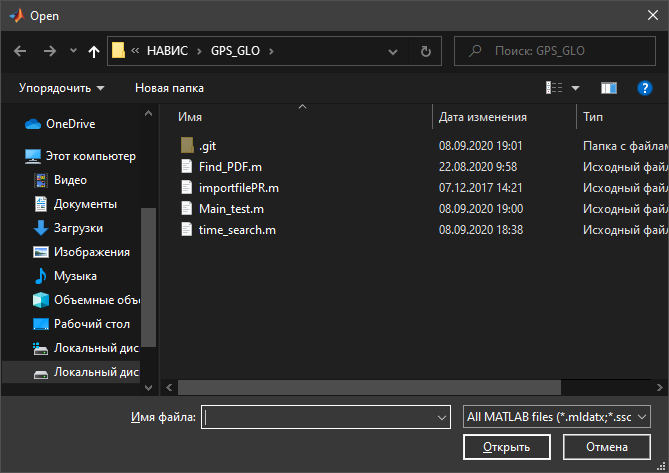


Рисунок 2 – Окно создания проекта

В левой части окна представлены директории операционной системы Windows 10, с помощью которых можно указать путь расположения проекта. В центральной части представлены сами кусочки проекта. В нижней части нужно указать имя открываемого файла или же выбрать его из представленного списка с помощью средств клавиатуры или же мышки.

Matlab имеет возможности для получения доступа к документации сразу из командной строки среды разработки. Рассмотрим стартовое окно программы, продемонстрированное ранее (рисунок 1). В основную область окна «Command Window», предназначенного для отправки инструкций, можно написать код в виде шаблона «doc “название команды”», которое выдаст документацию на введенную команду (рисунок 3). Помимо документации, открывшаяся вкладка также содержит в себе некоторое количество примеров для использования данной инструкции, которые позволят неопытному пользователю в быстро начать работу в данной среде разработки. Недостаток данного способа получения документации заключается в том, что вся документация, встроенная в Matlab, предоставляется на английском языке, что может доставить некоторые трудности для человека не знакомого с этим языком.

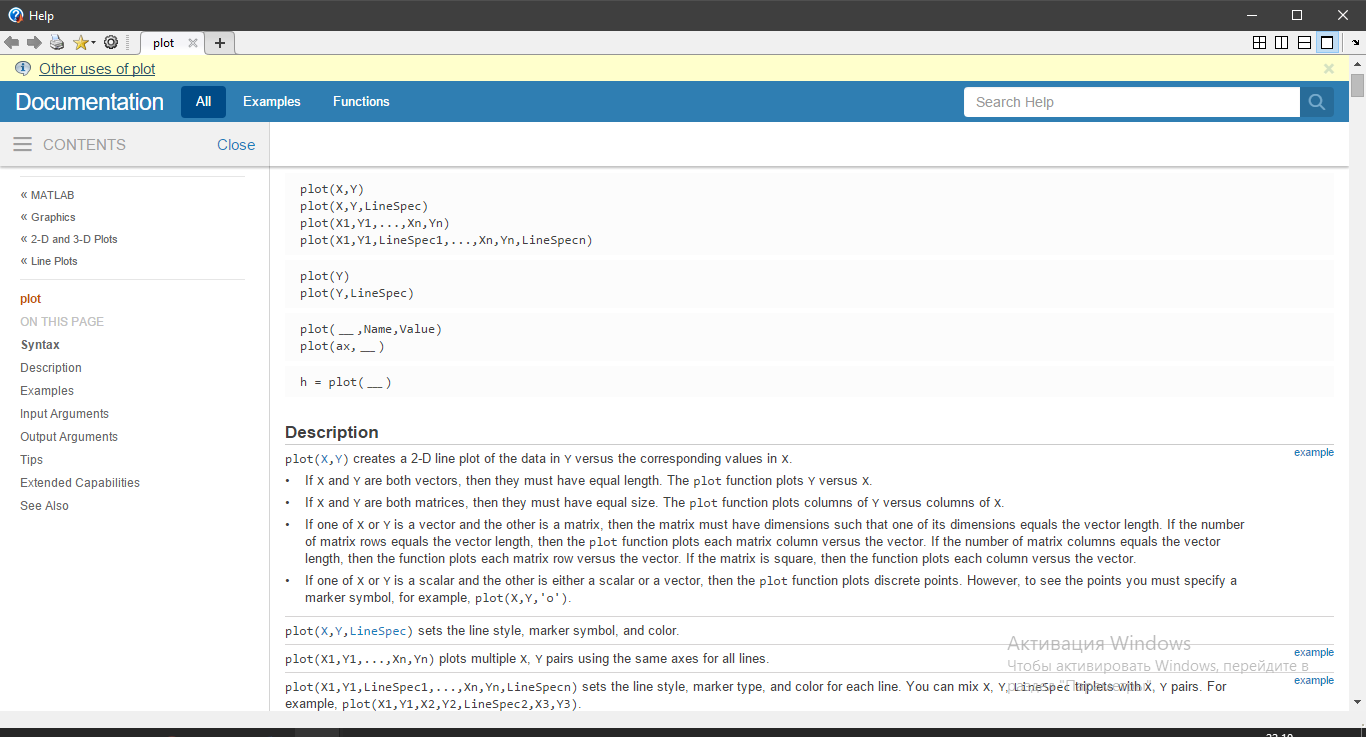


Рисунок 3 – Содержимое документации на инструкцию «plot»

Среда разработки Matlab предоставляет возможность выбрать каждому человеку пользовательский интерфейс под свои нужды, в целях обеспечения удобства каждого. С помощью стрелочек, появляющихся при наведении на край отдельного блок интерфейса, можно менять их масштаб (рисунок 4). Здесь в верхней части окна присутствуют несколько заранее отмасштабированных шаблонов расположения встроенных окон относительно друг друга. Окно редактора скриптов можно разделить дополнительно на 2 и более области, как это показано на рисунке 5. Размеры всех этих областей можно менять исходя из потребностей, а также добавлять новые и закрывать имеющиеся.

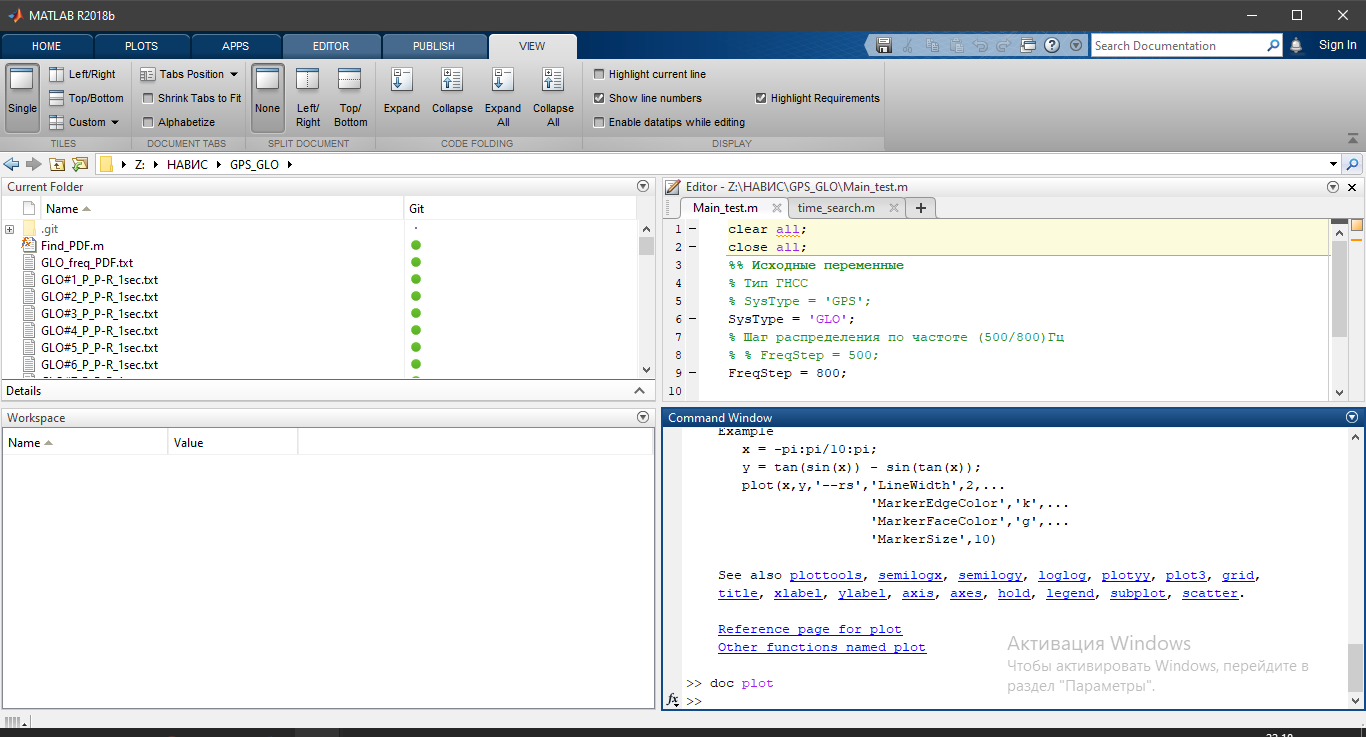


Рисунок 4 – Масштабирование окон Matlab

В случае, если в программном коде возникают ошибки во время написания, большая часть из них обнаруживается средой разработки во время выполнения скрипта (по нажатию зеленого треугольника в верхней части экрана с надписью «Run»), а строки с ошибками выводятся в окно «Command Window» с подписью на каком номере строки кода появилась была допущена ошибка, как это представлено на рисунке 6.

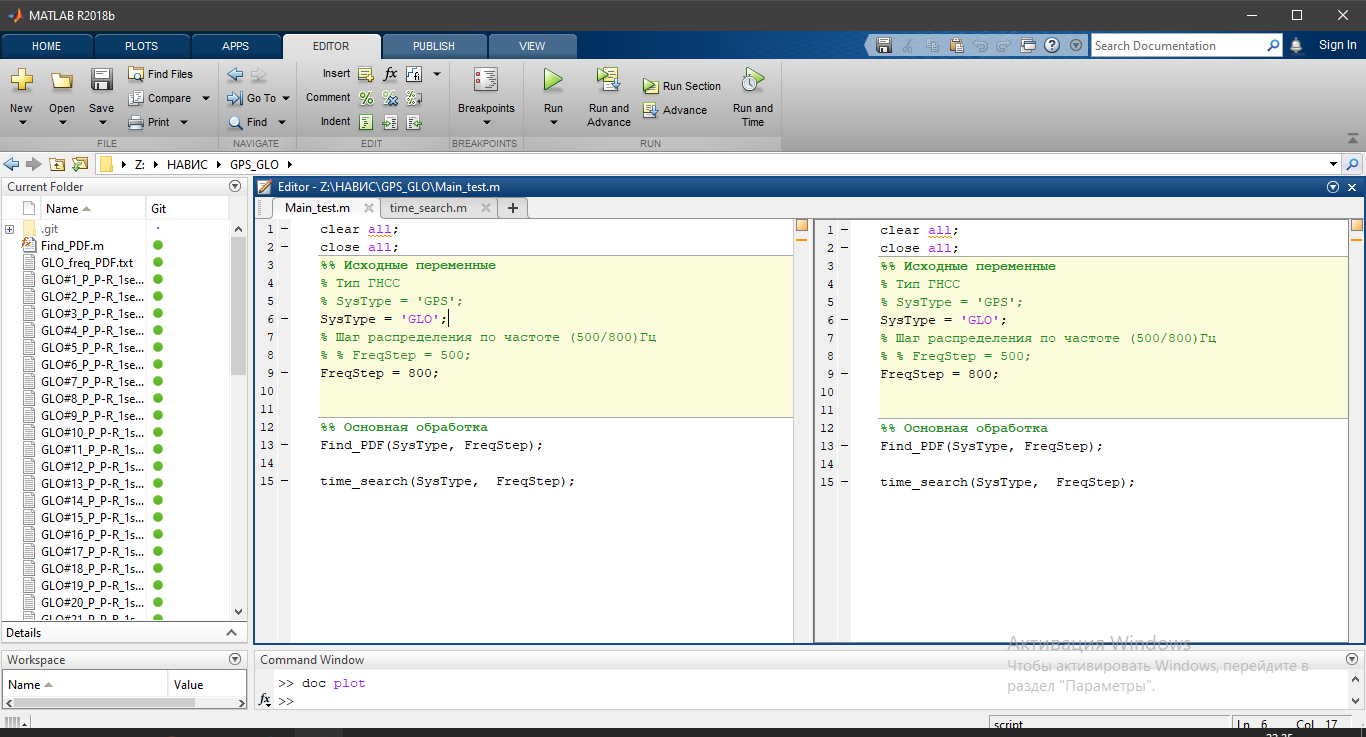


Рисунок 5 – Окно редактора кода

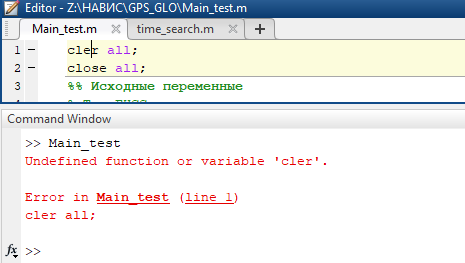


Рисунок 6 – Обнаружение ошибок в коде

Запуск программы осуществляется по нажатию соответствующей кнопки в верхней части интерфейса (рисунок 7) или же по нажатию кнопки «F5» на клавиатуре. Здесь также присутствуют кнопки для выставления точки остановки в режиме отладка и запуска только определенных, выделенных в скрипте, строк.

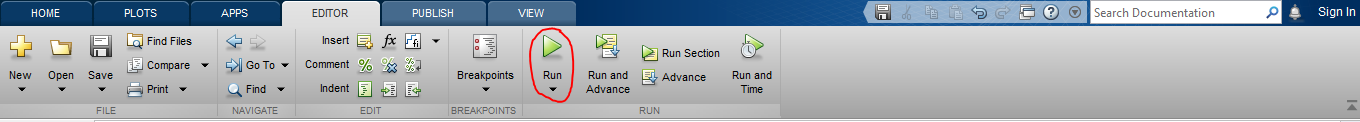


Рисунок 7 – Окно запуска программы

Среда разработки Matlab представляет собой не только программу, с помощью которой можно написать тот или иной скрипт, но и набор различных заранее встроенных инструментов. Из чего можно сделать вывод о том, что Matlab востребован для работы во различных отраслях, в том числе для решения задач, поставленных в данной работе.

Исходя из всего вышесказанного, можно заключить, что интерфейс данной среды разработки эргономичен и соответствует требованиям и принципам ГОСТ Р ИСО 9241.

1. **Эргономика интерфейса разработанного алгоритма**

Помимо принципов разработки интерфейса программного обеспечения, упомянутых в предыдущем подразделе, оно также должно обладать результативностью, эффективностью, удовлетворенностью и доступностью [ссылка на ГОСТ Р ИСО 9241]. На рисунке 8 представлено окно с задаваемыми начальными значениями.

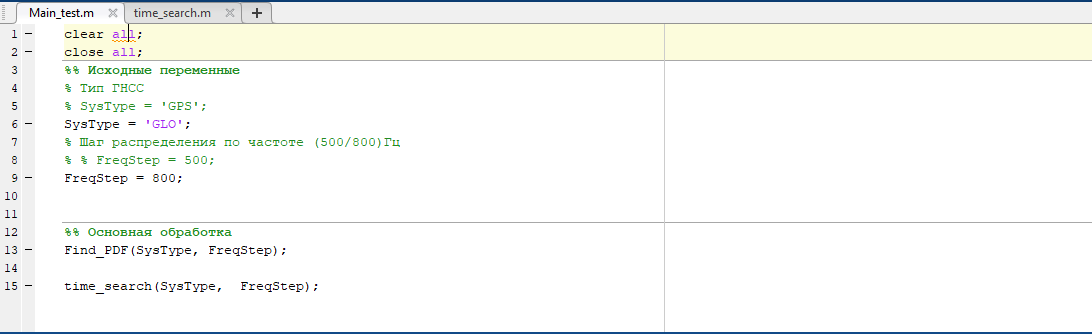


Рисунок 8 – Начальное окно программы.

В данном окне выбирается спутниковая радионавигационная система (СРНС), в данном случае ГЛОНАСС или GPS, а также пользователь может выбрать для какого шага по частоте стоит выполнять расчет (500 Гц или 800 Гц). Для вывода информации используется окно «Command Window», как это представлено на рисунке 9. Навигация по окну, в котором были получены интересующие результаты, осуществляется с помощью мыши полосой прокрутки в правой части консольного окна программы.

Вывод рассчитанных значений не перегружен излишней информацией и выводит только то, что необходимо пользователю.

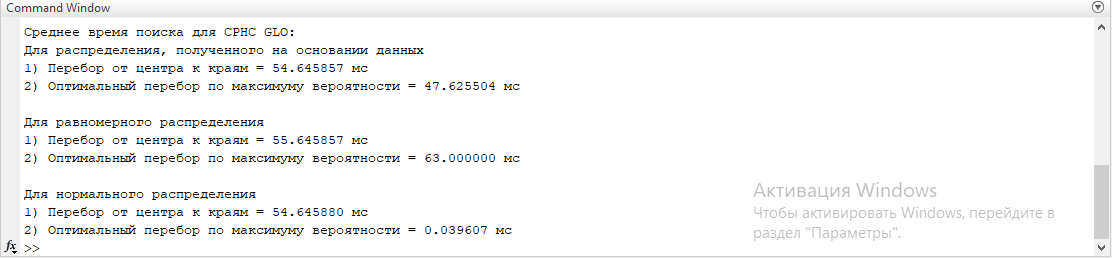


Рисунок 9 ­– Вывод информации после исполнения программы

При выборе другой системы для расчета параметров, пользовательский вывод также показывает, что за система была (рисунок 10).



Рисунок 10 – Демонстрация способности вывода значений для выбранной системы

В процессе выполнения отдельно написанных для программы функция, в стороннем окне открывается вкладка с частотным распределением (рисунок 11) той или иной системы. В итоге, все компоненты как текстового, так и графического вывода имеют заголовки, подписи и короткие описания для удобства работы пользователя.

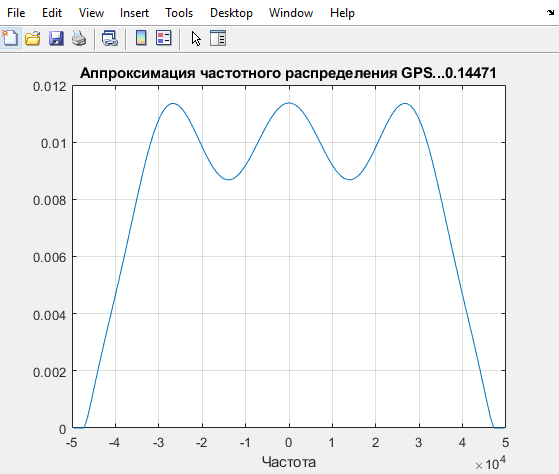


Рисунок 11 – Вывод графика с частотным распределением

Исходя из вышесказанного, можно заключить, что пользовательский интерфейс разработанного алгоритма соответствует требованиям ГОСТ Р ИСО 9241 и пригоден для выполнения расчета среднего времени поиска сигнала на одной временной позиции.

1. **Аспекты функциональной безопасности**

Персональный компьютер является сложным устройством, которое способно решать широкий спектр потребительских задач и выполнять огромное количество функций. Однако правильность работы зависит не только от правильности функционирования его аппаратной части, но и программных средств. Из-за возможности свободно редактировать код, надежность представленной программы обладает своими особенностями: зачастую, внося изменения в программу, программист не знает, как это изменение может повлиять на работоспособность алгоритма. Тем самым, любая модернизация программных средств может привести к новым ошибкам, которые сложно (а зачастую невозможно) определить на этапе тестирования, поскольку они могу проявляться только при определенных условиях.

Чтобы дать оценку рискам возникновения неполадок, необходимо составить перечень возможных отказов программы, ранжировать по приоритету действия по ликвидации отказов и по их предупреждению. Это можно сделать с помощью FMECA анализа с вычислением значения приоритетности риска (англ. Risk Priority Number – RPN).

Для любого программного обеспечения можно выделить следующие причины нарушения работоспособности:

* ошибки, скрытые в программе;
* искажение входной информации;
* неверные действия пользователя;
* неисправность аппаратуры установки, на которой реализуется вычислительный процесс.

Каждый хороший программист перед выпуском своей программы должен провести этап ее отладки. Однако не всегда получается обнаружить все ошибки и исправить их с помощью предоставленного инструмента. В итоговом счете, в коде отдельных скриптов могут оставаться скрытые, заранее необнаруженные, ошибки, искажающие правильность работы всей программы. К ним можно отнести ошибки ввода-вывода. Они могут проявляться в неверном формировании инструкций в разрабатываемом алгоритме, в связи с чем, он будет выполнять неправильные действия, что, несомненно, приведет к ошибочным результатам. Данный риск можно оценить по рангу тяжести на 8 (очень высокая) – потеря основного функционала, по рангу появления – 3 (поскольку появление такой ошибки является случайностью), ранг обнаружения – 1 (такая ошибка будет обнаружена при первом же запуске программы).

Следующая причина проявляется при выходе входных данных из области допустимых значений. Это может произойти в случае, если пользователь, например, выберет слишком большой шаг по. Также сюда можно отнести ошибки в документации, которая используется при подготовке данных. Такой вид ошибок пагубно влияет на работоспособность программы (вплоть до потери функциональности). Таким образом ранг тяжести – 7, ранг появления – 1, вероятность обнаружения – 2.

Неверные действия пользователя в первую очередь связаны с неправильной интерпретацией сообщений от программы и с неправильными действиями в работе с программой. В случае, если программное обеспечение эргономично и имеет качественную программную документацию, то риск возникновения ошибок минимален. Ранг тяжести – 5, ранг появления – 1, вероятность обнаружения – 1.

В результате неисправности аппаратуры может произойти искажение данных и текста самой программ в основной и внешне памяти. Подобная ошибка трудно диагностируема, поскольку не зависит от разработчика ПО и приводит в полной потере работоспособности программы. Ранг тяжести – 10, ранг появления – 1, вероятность обнаружения – 5.

Определив основные риски возникновения неполадок, необходимо рассчитать RPN и составить ранжированный список.

RPN рассчитывается по формуле (1) [ссылка на ГОСТ Р МЭК 61505 ]

 (1)

где S – ранг тяжести, О – ранг появления, D – ранг обнаружения.

В результате расчета RPN можно составить ранжированный список приоритета предотвращения рисков, 1 – элемент, которому необходимо уделить особое внимание, а о последнем можно беспокоится в последнюю очередь:

1. ошибки, скрытые в программе;
2. искажения входной информации;
3. неверные действия пользователя.

Следующим шагом является определение доли безопасных отказов (англ. Safe Failure Fraction, SSF), которая рассчитывается по формуле (2):

 (2)

где  – интенсивность безопасных отказов,  – интенсивность опасных отказов,  – суммарная интенсивность отказов. Из выше представленных причин неполадок опасными являются: ошибки, скрытые в программе и искажение входной информации. Безопасным можно считать только неверные действия пользователя.

Таким образом, доля безопасных отказов равна 1. Это можно объяснить тем, что при расчете SSF принимаются во внимание только диагностируемые неисправности. В разработанном алгоритме маловероятно появление ошибок, которые не диагностируются. При детальном рассмотрении доля безопасных отказов может снизится, но вряд ли опуститься ниже 90%. Таким образом, руководствуясь ГОСТ Р МЭК 61508, разработанному программному обеспечению можно присвоить третий уровень полноты безопасности (УПБ 3).

1. **Выводы по разделу**

Программа для расчета среднего времени поиска сигналов по частоте на одной временной позиции, разработанного в ходе выполнения выпускной квалификационной работы, обладает понятным, доступным и простым пользовательским интерфейсом, а также соответствует требованиям ГОСТ.

Так же в разделе были представлены возможные причины возникновения отказов разработанного программного обеспечения. В результате анализа диагностируемых факторов, которые могут повлиять на функциональные возможности разработанной программы, ему был присвоен третий уровень полноты безопасноти.