

Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет  
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”  
Завідувач кафедри кібербезпеки  
та програмного забезпечення  
д.т.н., професор  
\_\_\_\_\_ Олексій СМІРНОВ  
« \_\_\_\_ » \_\_\_\_\_ 2023 р.

**ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА**  
**за другим (магістерським) рівнем вищої освіти**  
на тему  
**“Дослідження та програмна реалізація системи GPS навігатора**  
**на основі чипу серії SiRF Atlas V”**

Виконав здобувач вищої освіти  
II курсу, групи КН-22М-1  
ОПП «Комп’ютерні науки»  
спеціальності 122 «Комп’ютерні науки»  
\_\_\_\_\_ Яценко Д.Р.  
« \_\_\_\_ » \_\_\_\_\_ 2023 р.

Керівник проекту  
кандидат технічних наук  
\_\_\_\_\_ Улічев О.С.  
« \_\_\_\_ » \_\_\_\_\_ 2023 р.  
Рецензент \_\_\_\_\_  
\_\_\_\_\_

Центральноукраїнський національний технічний університет  
Факультет *Механіко-технологічний*  
Кафедра *Кібербезпеки та програмного забезпечення*  
Рівень вищої освіти *магістр*  
Галузь знань *12* "Інформаційні технології"  
Спеціальність *122* "Комп'ютерні науки"  
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерні науки"

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 6 » вересня 2023 року

## ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

*Яценку Денису Романовичу*

(прізвище, ім'я, по батькові)

1. Тема роботи *Дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V*

2. Керівник роботи *Улічев Олександр Сергійович, канд. техн. наук*

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 32-13 від 04.08.2023 року

3. Строк подання студентом роботи до захисту *10.12.2023 р.*

4. Мета та завдання випускної кваліфікаційної роботи: *Метою розробки є дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V*

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

- |  |   |
|--|---|
| <i>1. Призначення та область використання.</i>           | <i>6. Наукова новизна.</i>                              |
| <i>2. Перегляд аналогічних існуючих систем.</i>          | <i>7. Економічна ефективність розробленої програми.</i> |
| <i>3. Опис і обґрунтування проектних рішень.</i>         | <i>8. Заходи з охорони праці та техніки безпеки.</i>    |
| <i>4. Етапи програмування системи.</i>                   | <i>9. Висновки.</i>                                     |
| <i>5. Впровадження системи в промислову експлуатацію</i> |   |

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

- |  |                 |
|--|-----------------|
| <i>Наукова новизна</i>                     | <i>1 аркуш</i>  |
| <i>Структурна схема системи</i>            | <i>1 аркуш</i>  |
| <i>Функціональна схема системи</i>         | <i>1 аркуш</i>  |
| <i>Діаграма процесів</i>                   | <i>1 аркуш</i>  |
| <i>Блок-схема алгоритму роботи додатку</i> | <i>2 аркуша</i> |
| <i>Показники економічної ефективності</i>  | <i>1 аркуш</i>  |

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Савеленко Г.В.	05.10.2023	14.11.2023
Охорона праці	Оришака О.В.	06.10.2023	16.11.2023

7. Дата видачі завдання « 6 » вересня 2023 р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2023 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2023 р.	
3.	Розробка моделі компонента	20.10.2023 р.	
4.	Розробка структур даних	25.10.2023 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2023 р.	
6.	Програмування алгоритмів	10.11.2023 р.	
7.	Розрахунок економічної ефективності	13.11.2023 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2023 р.	
9.	Оформлення ПЗ	17.11.2023 р.	
10.	Попередній захист роботи	10.12.2023 р.	

Дата видачі завдання  
« 6 » вересня 2023 р.

Підпис керівника

\_\_\_\_\_  
(прізвище та ініціали)Завдання прийнято до виконання  
« 6 » вересня 2023 р.

Підпис здобувача

\_\_\_\_\_  
(прізвище та ініціали)

## АНОТАЦІЯ

**Ященко Д.Р. Дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2023.**

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи GPS навігатора на основі чипу серії SiRF Atlas V.

Метою розробки є дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.

Об'єктом дослідження є процес GPS навігатора на основі чипу серії SiRF Atlas V.

Предметом дослідження є методи GPS навігатора на основі чипу серії SiRF Atlas V.

Методи дослідження базуються на методах теорії геопозиціонування, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows 10/11.

Програму розроблено в середовищі Embarcadero Delphi.

**Ключові слова:** комп'ютерні науки, GPS, SiRF Atlas V

## ABSTRACT

**Yashchenko D.R. Research and software implementation of the GPS navigator system based on the SiRF Atlas V chip series. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2023.**

In this graduation thesis for the second (master's) level of higher education, software is developed, which is intended for the GPS navigator system based on the SiRF Atlas V series chip.

The goal of the development is the research and software implementation of the GPS navigator system based on the SiRF Atlas V series chip.

The object of the study is the GPS navigator process based on the SiRF Atlas V series chip.

The subject of research is GPS navigator methods based on the SiRF Atlas V series chip.

The research methods are based on geo-positioning theory methods, mathematical statistics methods, and software development methods.

The result of the work is a software implementation of the GPS navigator system based on the SiRF Atlas V series chip.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on PCs of IBM PC architecture with Windows 10/11 OS.

The program was developed in the Embarcadero Delphi environment.

**Keywords:** computer science, GPS, SiRF Atlas V

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ .....	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ .....	7
1.1 Призначення системи.....	7
1.2 Область застосування.....	8
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ .....	13
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	13
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	22
2.3 Розгорнута постановка завдання .....	28
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ .....	30
3.1 Опис функціонування системи .....	30
3.2 Розробка структурної схеми.....	36
3.3 Розробка функціональної схеми .....	46
3.4 Розробка діаграми процесів.....	49
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	51
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	51
4.2 Захист розробленого програмного забезпечення.....	58
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ .....	59
6 НАУКОВА НОВИЗНА .....	61

						ВКРМ-122.23.0026.00.00.ПЗ		
Вим	Арк.	№ докум.	Підп.	Дата				
Розроб.	Яценко Д.Р.				Дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V	Літ.	Аркуш	Аркушів
Перев.	Улічев О.С.					М	1	101
Н.контр.	Коваленко А.С.				ЦНТУ КН-22М-1			
Затв.	Смірнов О.А.							

7 ЕКОНОМІЧНА ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ.....	62
7.1 Техніко економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	62
7.2 Розрахунок трудомісткості розробки програмної продукції.....	64
7.3 Визначення чисельності виконавців і планового фонду зарплати.....	66
7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника.....	71
7.5 Визначення собівартості розробки та ціни програмної продукції.....	75
7.6 Визначення об'єму капітальних вкладень та експлуатаційних витрат у споживача програмної продукції.....	78
7.7 Визначення експлуатаційних витрат.....	79
7.8 Визначення економічної ефективності програмної продукції.....	80
7.9 Висновок.....	82
8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ .....	83
8.1 Вступ.....	83
8.2 Аналіз умов праці на робочому місці ІТ-фахівця.....	84
8.3 Пропозиції щодо підвищення працездатності ІТ-фахівця.....	86
8.4 Пожежна безпека.....	88
8.5 Розрахункова частина .....	89
9 ОСНОВНІ ВИСНОВКИ.....	93
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	95

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

- КПК – кишеньковий персональний комп'ютер
- РМД – робоче місце диспетчера
- СЕР – Circular Error of Probability – окружність можливої помилки
- GPS – глобальна система місцевизначення.
- GSM – система цифрового стільникового телефонного зв'язку
- IOD – issue of data – часовий ідентифікатор
- LAAS – навігаційний комплекс
- LAT – широта
- LON – довгота
- MFC – Microsoft Foundation Class library – бібліотека класів
- OLE – технологія зв'язування й вбудовування об'єктів
- POI – point of interest – вбудована база «цікавих» точок
- SA – режим обмеженого доступу
- WAAS – система супутників і наземних станцій, що робить коректування сигналів GPS

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

## ВСТУП

**Актуальність теми.** Глобальна система позиціонування (GPS) – це мережа супутників і приймальних пристроїв, які використовуються для визначення місцезнаходження чогось на Землі. Деякі GPS-приймачі настільки точні, що можуть визначити своє місцезнаходження з точністю до 1 сантиметра. Приймачі GPS забезпечують визначення місцезнаходження за широтою, довготою та висотою. Вони також дають точний час. GPS включає 24 супутники, які обертаються навколо Землі на точних орбітах. Кожен супутник робить повний оберт навколо Землі кожні 12 годин. Ці супутники постійно посилають радіосигнали. Приймачі GPS запрограмовані на отримання інформації про те, де кожен супутник знаходиться в будь-який момент. GPS-приймач визначає своє власне місцезнаходження, вимірюючи час, потрібний для надходження сигналу до його місцезнаходження принаймні з чотирьох супутників. Оскільки радіохвилі поширюються з постійною швидкістю, приймач може використовувати вимірювання часу для розрахунку відстані від кожного супутника. Використання кількох супутників робить дані GPS більш точними. Якщо приймач GPS обчислює свою відстань лише від одного супутника, це може бути точна відстань від супутника в будь-якому напрямку. Думайте про супутник як про ліхтарик. Коли ви посвітите ним на землю, ви отримаєте коло світла. З одним супутником GPS-приймач може бути де завгодно в цьому колі світла. З двома супутниками більше, є ще два кола. Ці три кола перетинаються, або перетинаються, лише в одному місці. Це місцезнаходження GPS-приймача. Такий спосіб визначення місця розташування називається трилатерацією. Літаки, кораблі, підводні човни, потяги та космічні човники використовують GPS для навігації. Багато людей користуються магнітолами під час водіння автомобіля. GPS-приймач наносить на електронну карту місце розташування автомобіля, яке постійно змінюється. Карта показує напрямки до місця призначення людини. І місцезнаходження, і транспортний засіб наносяться за допомогою супутникових даних. Деякі туристи використовують GPS, щоб допомогти їм знайти дорогу, особливо коли вони не на

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

маркованих стежках. Іноді виникають перешкоди для отримання чіткого сигналу GPS. Гравітація може трохи вивести супутники GPS з орбіти. Частина земної атмосфери іноді спотворюють супутникові радіосигнали. Древа, будівлі та інші споруди також можуть блокувати радіохвилі. GPS контроль і станції моніторингу по всьому світу відстежують супутники та постійно контролюють їхні сигнали. Потім вони обчислюють поправки, які транслуються на приймачі GPS. Ці поправки роблять GPS набагато точнішим.

**Мета й завдання дослідження.** Метою роботи є дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем GPS навігатора на основі чипу серії SiRF Atlas V.
- Дослідження системи GPS навігатора на основі чипу серії SiRF Atlas V.
- Програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.

*Об'єктом дослідження* є процес GPS навігатора на основі чипу серії SiRF Atlas V.

*Предметом дослідження* є методи GPS навігатора на основі чипу серії SiRF Atlas V.

*Методи дослідження* базуються на методах теорії геопозиціювання, методах математичної статистики, методах розробки програмного забезпечення.

**Наукова новизна отриманих результатів.** У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод GPS навігатора на основі чипу серії SiRF Atlas V.
- Розроблено вітчизняний продукт GPS навігатора на основі чипу серії SiRF Atlas V, який має більш широкі можливості, на відміну від існуючих аналогів.

**Практична цінність отриманих результатів** полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі GPS навігатора на основі чипу серії SiRF Atlas V.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

**Достовірність наукових результатів** підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти «Наука – виробництву», 2023, основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №14.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБГІЗ-2023

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		<b>6</b>

# 1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

## 1.1 Призначення системи

Ваша уява є єдиним обмеженням для можливостей технологій глобальної системи позиціонування (GPS). Системи GPS надзвичайно універсальні, і їх можна знайти майже в будь-якому секторі промисловості. Сьогодні GPS використовується для картографування лісів, допомагає фермерам збирати врожай на полях і пілотам літаків орієнтуватися на землі чи в повітрі. Системи позиціонування є невід'ємною частиною військових застосувань і для екстрених екіпажів для визначення місцезнаходження людей, які потребують допомоги. Технології GPS часто працюють у багатьох сферах, які ми зазвичай не розглядаємо.

### Приклади використання GPS

Програми глобальної системи позиціонування зазвичай поділяються на 5 основних категорій:

1. Локація – визначення позиції.
2. Навігація – переміщення з одного місця в інше.
3. Трекінг – моніторинг руху об'єкта або особи.
4. Картографування – створення карт світу.
5. Визначення часу – надання світу точного часу.

У будь-якому випадку використання GPS базується на вимірюванні відстані людини від кількох супутників. GPS – це лише одна з кількох груп супутників, які використовуються для визначення положення. Galileo, ГЛОНАСС, BeiDou і GPS – це чотири основні супутникові угруповання, які використовуються в усьому світі. Технології використовують сигнали цих супутників для вимірювання відстані людини до цих супутників. Завдяки цим вимірюванням технології GPS можуть визначити, де людина перебуває у світі та

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

як перейти до іншого місця. GPS використовується для розрахунку дуже конкретного та точного часу завдяки рубідієвим годинникам на кожному супутнику. Ці годинники дозволяють супутникам знати своє точне місцезнаходження в дуже точний час, і ця інформація про час використовується в різних програмах, включаючи морську гідрографію, автономні транспортні засоби та точне землеробство.

## 1.2 Область застосування

Від гірничої промисловості до авіації, від сільського господарства до морської галузі, від відпочинку до оборони, існує багато застосувань для технології GPS. Сьогодні всі: лікарі, науковці, фермери, солдати, пілоти, туристи, водії доставки, моряки, рибалки, диспетчери, спортсмени та люди з багатьох інших верств суспільства використовують системи GPS таким чином, щоб зробити свою роботу продуктивнішою та безпечнішою, і простішою.

Три основні галузі використання GPS включають точне сільське господарство, автомобільну промисловість і оборону. У кожному прикладі GPS вирішує проблему позиціонування.

Визначення положення за допомогою глобальних навігаційних супутникових систем (GNSS) залежить від кількох факторів, зокрема геометрії супутника, доступності, якості сигналу, атмосферної затримки та середовища. Приймач GNSS повинен правильно оцінювати помилки GNSS і враховувати ці фактори, щоб забезпечити точне рішення позиціонування.

Ми поєднуємо досвід GNSS і злиття датчиків, глобальну інфраструктуру служб коригування, автентифікацію та цілісність, щоб пом'якшити обмеження GNSS і забезпечити точне й надійне позиціонування. Ми об'єднуємо додаткові технології, рішення та послуги, щоб забезпечити найточніші позиції в будь-якому середовищі.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8



одночастотні приймачі повинні або покладатися на трансляцію ефемерид, або на альтернативне джерело коригувань, як-от космічні системи доповнення (SBAS), щоб моделювати іоносферні помилки. Точність цих методів обмежена, а у випадку SBAS вона залежить від регіону та може потребувати додаткової інфраструктури.

Багаточастотні приймачі також забезпечують більшу стійкість до перешкод. Якщо є перешкоди в одній частотній смузі, багаточастотний приймач все одно відстежуватиме інші сигнали, щоб забезпечити постійне позиціонування.

### **Перевага кількох сузір'їв**

Коли ми говоримо про багатосузір'я, це означає, що приймач може отримати доступ до сигналів із кількох сузір'їв: GPS, ГЛОНАСС, BeiDou та Galileo, наприклад. Використання інших сузір'їв на додаток до GPS призводить до збільшення кількості супутників у полі зору, що має такі переваги:

- Скорочений час отримання сигналу.
- Покращено позицію та точність часу.
- Зменшення проблем, викликаних такими перешкодами, як будівлі та листя.
- Покращено просторовий розподіл видимих супутників, що призводить до покращеного зменшення точності.

Коли приймач використовує сигнали з різних сузір'їв, у рішення вбудовано резервування. Якщо сигнал заблоковано через робоче середовище, існує дуже висока ймовірність того, що приймач може просто вловити сигнал з іншого угруповання, забезпечуючи безперервність рішення.

### **Методи позиціонування**

#### **Точне позиціонування точки (PPP)**

PPP – це метод позиціонування за допомогою GNSS, який забезпечує розташування на субметровому або кращому рівні шляхом поєднання поправок глобального супутника GNSS і сигналу з моделюванням помилок приймача

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

GNSS і алгоритмами оцінки положення. Дані корекції, необхідні для рішення PPP, включають супутниковий годинник GNSS, орбіту та поправки зміщення сигналу, створені мережею глобальних опорних станцій. Після обчислення поправок вони доставляються кінцевому користувачеві через супутник або Інтернет. Типове рішення PPP вимагає певного періоду часу для зближення. Цей період збіжності необхідний для оцінки локальних зміщень вимірювання. Фактично досягнута точність і необхідний час збіжності залежать від якості поправок, умов локального спостереження та складності алгоритмів приймача.

### **Супутникова система посилення (SBAS)**

Системи SBAS – це геосинхронні супутникові системи, які надають послуги для підвищення точності, цілісності та доступності основних сигналів GNSS. Точність позиціонування підвищується завдяки передачі широкомасштабних поправок на помилки діапазону GNSS. Цілісність підвищується за рахунок швидкого виявлення помилок супутникового сигналу та надсилання сповіщень одержувачам про те, що вони не повинні відстежувати несправний супутник. Опорні станції, які територіально розподілені по всій зоні обслуговування SBAS, отримують сигнали GNSS і передають їх на головну станцію. Оскільки розташування опорних станцій точно відоме, головна станція може точно розрахувати широкомасштабні поправки. Поправки передаються на супутник SBAS, а потім транслюються на приймачі GNSS у всій зоні покриття SBAS. Обладнання користувача отримує виправлення та застосовує їх до обчислень діапазону.

SiRFatlas – чипсет для GPS-приймачів. SiRFatlas сполучає на одному чипі ARM11 процесор (500/664 MHz):

- автономне DSP-ядро для обробки сигналів GPS і Galileo з технологією SiRFAlwaysFix;
- контролери пам'яті DDR, DDR2, SD/MMC/MMC+ і NAND;
- аудіо ЦАП;
- контролер сенсорної панелі;

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

- акселератор пост-обробки відео;
- USB 2.0 і інші інтерфейси обміну даними;
- повне керування електроживленням.

Такий високий рівень інтеграції зменшує загальну кількість деталей, знижує витрати сировини, знижує споживану потужність і вимоги до електроживлення, зменшує габарити монтажної плати, спрощує виробництво, що прискорює його реалізацію.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБГПЗ-2023

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		<b>12</b>

## 2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

### 2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

На початку 2023 року вимоги до окремих навігаторів трохи зросли. Екрани 5" стали практично стандартними, повністю витиснувши моделі 3.5" і 4". Зрідка попадаються гарні GPS навігатори з діагоналлю 4.3". У більшості випадків використовується платформа Sirf Atlas V і обов'язкова наявність хоча б одного бездротового інтерфейсу. Незмінним в усіх залишається тільки одне – операційна система. Але недалеко та година, коли її теж замінять на що-небудь більше сучасне й швидке.

#### **Lexand SU-533**

Недорогий і практичний навігатор з ультратонким корпусом буквально за перший місяць продажів завоював визнання з боку користувачів. Його відрізняє стабільність програмного забезпечення, що не завжди можна сказати про пристрої за його ціну.

Як уже було відзначено, апарат укладений у тонкий корпус, товщиною 11 мм. Пристрій пластиковий з матовим покриттям Soft-touch – таким, яке й повинне бути в навігатора. У корпусі немає ніяких блискучих частин, а виходить, яскраве сонце не буде відволікати від навігації.

Екран самий звичайний TFT з діагоналлю 5" і розрішенням 480x272. Яскрава й чітка картинка забезпечена й не буде відволікати дрібними деталями.

Серцем пристрою є навігаційний процесор Sirf Atlas V із частотою 600 МГц. Гарна продуктивність і швидкий пошук супутників забезпечені. Обсяг оперативної пам'яті 128 Мб. Операційна система працює швидко й стабільно.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

Убудованої пам'яті 4 Гб цілком достатньо для потреб навігації. Роз'єм карт пам'яті до 32 Гб на місці.



Рисунок 2.1 – Інтерфейс користувача Lexand SU-533

З бездротових модулів є тільки Bluetooth 2.0, що дозволяє виводити звук на зовнішній пристрій, а також використовувати зовнішній телефон для доступу в Інтернет або одержання інформації про пробки.

Тонкий корпус зіграв злий жарт із акумулятором. Незважаючи на те, що виробник заявляє про чотири годинки автономності, пристрій ледь устигає проробити друга година від одного заряду. Втім, у машині можна хоч безупинно заряджати пристрій, але деякі незручності це доставляє.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

Апарат відразу поставляється із програмою «Навітел» п'ятої версії. Можуть бути поставки із програмами «Сітігід» або «Прогород». Операційна система дозволяє використовувати навігатор як мультимедійний пристрій за умови, що будуть установлені сторонні програми. Стандартні в ні на що не годяться.

Звичайно, хотівся б і GSM-модуль, і акумулятор більше ємний, але це тільки збільшило б ціну навігатора. Апарат відмінно справляється зі своїми основними функціями, але, як і будь-який пристрій, не ідеальний.

Переваги:

- Тонкий і практичний корпус.
- Гарний 5" екран.
- Відмінна продуктивність.
- Bluetooth з DUN і HSP.
- Невисока ціна.

Недоліки:

- Маленька ємність акумулятора.

### **Prestigio GeoVision 5600GPRSHD**

Традиційно для Prestigio навігатор GeoVision 5600GPRSHD сполучить у собі стильний дизайн і відмінну функціональність. По дизайну навігатор чимсь нагадує iPhone 4, тільки всі панелі пластикові, як і торці. Хіба що рамку навколо екрана могли б зробити матовою.

Дисплей з діагоналлю 5" має розрішення 800x480. Забагато для навігатора з такою діагоналлю, де потрібна точність сенсора, особливо з огляду на ємнісний сенсорний екран. Але, як показує практика, особливих незручностей це не доставляє. Навпаки, краща чутливість екрана скрашує невеликі незручності.

Серцем навігатора є процесор Sirf Atlas V із частотою 533 Мгц. Холодний старт пристрою займає мінуту, гарячий – кілька секунд. Це гарні показники. Оперативної пам'яті 128 Мб, убудованої – 2 Гб. Апарат здатний бути повноцінним комунікатором, оскільки має убудований GSM-модуль. Ніяких

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

мучень із налаштуванням окремого телефону. А от інших бездротових з'єднань в апарату немає, що, втім, у цьому випадку розбудовує не сильно.

Акумулятор ємністю 1500 мАч дозволяє навігаторові проробити до 2,5 годин. Але якщо включити всі комунікації, то час скорочується до години.

Як програма навігації використовується «Navitel Навігатор 5». Гарна функціональність, докладні деталізовані карти й більша база точок POI. Для більшості можливостей програми досить. Установити іншу програму, якщо буде потрібно, ніхто не заважає.



Рисунок 2.2 – Інтерфейс користувача Prestigio GeoVision 5600GPRSHD

Операційна система дозволяє, завдяки достатку софту, використовувати всі мультимедійні можливості, які здатний надати навігатор. Стандартні програми по зрозумілих причинах ні на що не годяться.

Критичних недоліків навігатор GeoVision 5600GPRSHD не має. Як завжди, Prestigio створила відмінний пристрій, що користується популярністю.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

Ще б корпус не блищав і бездротових з'єднань побільше, і було б взагалі все гарно.

Переваги:

- Стильний дизайн.
- Якісний екран 6".
- Процесор Sirf Atlas V.
- GSM-модуль.

Недоліки:

- Блискучий корпус.
- Крім GSM немає бездротових з'єднань.

### **Garmin Nuvi 3490 LT**

Nuvi 3490LT – новий флагман зручних і функціональних навігаторів практично без додаткових функцій і по жахливо дорогій ціні. Так, пристрою від Garmin завжди такі. Цей апарат відрізняється дуже тонким для навігатора корпусом і високою якістю використовуваних матеріалів. Задня кришка металева, торці із хромованого металу, екран скляний. Пластик використовується тільки в панелі динаміка на задній кришці.



Рисунок 2.3 – Інтерфейс користувача Garmin Nuvi 3490 LT

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17



– Фірмові сервіси від Garmin.

Недоліки:

– Практично повна відсутність додаткових функцій.

– Висока ціна.

### **Pocket Nature RD-500**

Головна особливість Pocket Nature RD-500 у тому, що він має убудований антирадар. Причому дані, отримані з його, інтегруються в програму навігації, відображаючи стан у реальному часі. Пристрій повністю пластиковий, не має блискучих частин. Це добре, тому що нічого не буде відволікати під час водіння.



Рисунок 2.4 – Інтерфейс користувача Pocket Nature RD-500

Екран самий звичайний TFT з діагоналлю 5" і розрешенням 800x480. Програмне забезпечення спеціально адаптоване під такий розрешення екрана. Всі елементи великі і якісно відмальовані.

					VKPM-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19



## TeXet TN-610 Voice HD

TeXet TN-610 Voice HD – один із самих технологічних навігаторів на ринку у своєму класі й при цьому за невисокою ціною. Апарат пластиковий, причому матовий. Ніяких відблисків на сонце. Товщина пристрою невелика, але й не сама маленька.



Рисунок 2.5 – Інтерфейс користувача TeXet TN-610 Voice HD

Розрішення екрана з діагоналлю 5" становить 800x480. Чітке зображення гарантоване.

Серцем навігатора виступає платформа Sirf Atlas V із частотою 500 МГц. Холодний старт займає до 40 секунд, гарячий – 1-2 секунди. Відмінні показники, які забезпечує такий процесор. Оперативної пам'яті 128 Мб, убудованої – 4 Гб, що досить для завантаження повного набору карт. Навігатор здатний працювати

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

як повноцінний комунікатор, маючи GSM-модуль на борту. Пробки можна одержувати по мобільному Інтернету. Є також модуль Bluetooth 2.1.

Акумулятор ємністю 1500 мАч дозволяє пристрою проработи до чотирьох годин автономно. При використанні GSM цей показник знижується до двох годин і нижче.

Навігатор може комплектуватися програмами «Navitel навігатор» або City Guide. Свої переваги є в кожній, але можна також установити й інші.

Операційна система – звична для всіх навігаторів. Багатий вибір стороннього софта дозволяє по-різному використовувати пристрій. На стандартний софт краще не покладатися.

Як таких недоліків апарат не має. Так, хотівся б більше ємний акумулятор, щоб з GSM довше працював. Але для навігатора наявного акумулятора цілком достатньо. Комуś не буде вистачати FM-трансмiтера, однак для голосного зв'язку можна й Bluetooth використовувати.

Переваги:

- Практичний дизайн і матеріали корпусу.
- Процесор Sirf Atlas V.
- GSM-модуль.
- Одна із двох програм навігації на вибір.

Недоліки:

- Акумулятор хотiлося б більш ємний.
- Немає FM-трансмiтера.

## **2.2 Обґрунтування вибору засобів для побудови системи та мови програмування**

Embarcadero Delphi, раніше Borland Delphi і Codegear Delphi, – інтегроване середовище розробки ПЗ для Microsoft Windows, Mac OS, iOS і Android мовою Delphi (що раніше носила назву Object Pascal), створена спочатку фірмою Borland

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

і на даний момент приналежна й розроблювальна Embarcadero Technologies. Embarcadero Delphi є частиною пакета Embarcadero RAD Studio і поставляється в чотирьох редакціях: Community (поширюється безкоштовно й має обмежену ліцензію на використання в комерційних цілях), Professional, Enterprise і Architect.

### **Delphi 10.4 Sydney**

Випущено 26 травня 2020 року. RAD Studio Delphi 10.4 забезпечує значно поліпшену високопродуктивну нативну підтримку Windows, кращу продуктивність розробки, миттєві підказки code completion, прискорення виконання коду із синтаксисом керованих записів, поліпшення виконання паралельних завдань на сучасних багатоядерних CPU, а також містить більш 1000 виправлень багів, поліпшення продуктивності середовища й бібліотек і багато чого крім того.

#### **Основні можливості Delphi 10.4.1:**

– Істотні розширення для Windows: поліпшення для застосунків на моніторах 4K High DPI, інтеграція з новим WebView2 на базі Chromium, використання розширених title bars, таких же, як в Office, Explorer, Google Chrome.

– Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

– Істотне поліпшення Delphi Code Insight (без можливого блокування IDE – в окремому процесі), що допоможе при роботі з великими проектами.

– Тип даних Delphi «record» тепер підтримуть довільні ініціалізацію, фіналізацію й операції копіювання.

– Розширена підтримка бібліотек C++: ZeroMQ, SDL2, SOCI, libSIMDpp і Nematode.

– Відладник Win 64 (на LLDB) і збирач для C++.

– Поліпшення для C++: Включена велика кількість поліпшень STL з Dinkumware.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	<i>Арк.</i>
<i>Вим.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		<b>23</b>

- Підтримка Metal Driver GPU для macOS і iOS.
- Вбудований Fmxlinux.
- Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.

Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

Реалізований заново стилізуємий FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку ІМЕ.

- Численні поліпшення швидкості й стабільності роботи нашої бібліотеки The Parallel Programming Library (PPL).

- Додані оновлені драйвери для FireBird, PostgreSQL і SQLite.

- Клієнтські бібліотеки HTTP і REST Client розширені застосунковими можливостями роботи з HTTPS. Також були розширені можливості підтримки Amazon AWS services

- У технологію Visual LiveBindings внесена безліч поліпшень, у тому числі швидкодії, що стосуються, застосунків на VCL і FireMonkey

RAD Studio 10.4 Короткий огляд:

- Істотні розширення для Windows. Створення застосунків, що чудово виглядають, із чіткими елементами інтерфейсу на 4к моніторах High DPI за допомогою нової гнучкої підтримки стилів елементів керування на екрані. Інтеграція із сучасними, безпечними web-технологіями від Microsoft – новим WebView2 на базі Chromium. Використання сучасних розширених title bars, таких же, як в Office, Explorer, Google Chrome, у своїх проектах. Істотні поліпшення надійності налагодження в новому відладнику для С++ Windows 64-bit.

- Зросла продуктивність розробки. Ріст продуктивності за рахунок миттєвої реакції підказок code completion у середовищі IDE. Краща сумісність із уже наявною кодовою базою, і спрощення програмування за рахунок уніфікованої архітектури керування пам'яттю. Швидке зв'язування даних і візуальних елементів за допомогою розширеної технології Visual LiveBindings з підвищеною швидкодією. Просте використання розповсюджених бібліотек С++,

						<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			<b>24</b>

наприклад, ZeroMQ, SDL2, SOCI, libSIMDpp і Nematode. Оновлена підтримка Amazon AWS cloud.

– Поліпшення швидкодії і якості. Більш 1000 поліпшень швидкодії і якості. Краща ефективність коду за допомогою нового синтаксису `custom managed records`. Більш швидке виконання паралельних завдань на сучасних багатоядерних CPU. Переконаєтеся в прискоренні відображення на екрані з підтримкою Metal API на macOS і iOS. Краща сумісність із уже наявною кодовою базою й спрощення програмування за рахунок уніфікованої архітектури керування пам'яттю.

### **Істотне поліпшення Delphi Code Insight**

Як найбільше й головне поліпшення інструментів програмування Delphi за багато років, в 10.4 Delphi Code Insight реалізований через Language Server Protocol (LSP). LSP – це технологія генерації результатів для code completion, навігації й інших сервісів в окремому процесі. Це значить, що code completion і Code Insight одержать більш точні результати без блокування IDE. 10.4 забезпечує набагато більш високу продуктивність розроблювачів, які працюють із більшими проектами, що містять мільйони рядків коду.

### **Delphi Custom Managed Records**

Ключове розширення мови Delphi: тип даних Delphi «record» тепер підтримуть довільні ініціалізацію, фіналізацію й операції копіювання. Управляйте тем, як ці структури створюються, копіюються й звільняються з допомоги вашого коду, який буде виконуватися у відповідний момент.

Це розширює потужність конструкцій records в Delphi, які використовуються щоб одержати більшу ефективність у порівнянні із класами.

### **Єдине керування пам'яттю**

Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

У порівнянні з Automatic Reference Counting (ARC), це дає кращу сумісність із існуючим кодом і спрощує написання компонентів, бібліотек і застосунків.

ARC модель керування пам'яттю model залишилася для керування рядками й посиланнями на тип інтерфейсу на всіх платформах. Для C++ це означає, що при створенні й звільненні Delphi-style класів в C++ використовується звичайне керування пам'яттю, як у будь-якого heap-allocated класу C++, що значно знижує складність коду.

### **Розширена підтримка бібліотек C++**

В 10.4 ми портували багато популярних бібліотек C++ у C++Builder.

Забезпечивши оптимізовану підтримку бібліотек ZeroMQ, SDL2, SOCl, libSIMDpp і Nematode, поряд із уже підтримуваними Boost і Eigen, які можуть бути додані за допомогою менеджера пакетів Getit.

### **Win 64-відладник і збирач для C++**

В 10.4 з'явився новий відладник C++ для Windows 64-bit. Відладник заснований на LLDB і показує значне збільшення стабільності при налагодженні 64-bit застосунків поряд з новими відладочними можливостями, такими як перегляд і інспекція типів начебто рядків C++ і Delphi, а також колекцій STL, включаючи std::vector, std::map і інших. Крім того, згенерована для застосунку відладочна інформація має інший внутрішній формат, сприяючи більш стабільному й багатому на можливості процесу налагодження, більш докладним перегляду й інспекції в debug-time.

### **Підвищення якості й швидкодії інструментів**

- Велика кількість поліпшень STL від Dinkumware.
- Поліпшені деякі найважливіші методи й області RTL, на базі поліпшень сумісності з популярними бібліотеками C++.
- Поліпшена підтримка Stafe.
- Велика кількість виправлень для підвищення стабільності і якості.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

– Відновлення Windows API – Обновлено й додали безліч декларацій API щоб добитися ще більшої інтеграції із платформою Windows.

– Загальні вдосконалення в бібліотеці доступу до БД FireDAC, включаючи оновлені драйвера для FireBird, PostgreSQL і SQLite. Вибір статичного або динамічного підключення SQLite до застосунку.

### **Змінені стилі VCL для High DPI**

В 10.4, архітектура стилізації VCL була суттєво розширена для підтримки High DPI і 4K моніторів. Тепер усі елементи UI на формі VCL автоматично масштабуються під відповідне до монітора дозвіл для показу форми. Був оновлений API стилізації для підтримки стилів high DPI.

Кожний графічний елемент UI може бути обраний з наборів різних масштабів і масштабований до потрібного DPI, що дає чітке зображення елементів UI на всіх моніторах.

### **Нові High DPI стилі й стилізація окремих VCL компонент**

Обновлено велике число вбудованих і преміальних VCL стилів для підтримки нового режиму стилізації High-dpi. Це дозволяє вам створювати застосунку з відмінним дизайном для всіх моніторів.

Розроблювачі VCL застосунків тепер можуть використовувати трохи VCL стилів на різних формах в одному застосунку або в різних компонентах на одній формі. Це також включає стилізацію компонентів загальною темою для платформи. Крім застосункової гнучкості використання стилів, це дозволяє використовувати нестилізуємі компоненти із зовнішніх бібліотек в VCL застосунках, що використовують стиль.

### **Поліпшена кроссплатформеність**

- Додана підтримка Metal Driver GPU для macOS і iOS.
- Крім підтримки останнього iOS SDK, в RAD Studio 10.4 розроблювачі можуть задовольнити нові вимоги Apple до набору стартових екранів.
- Реалізований заново стилізуємі FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку IME.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

- Користувачам редакцій Enterprise або Architect доступна повна інтеграція Fmxlinux з IDE для створення клієнтських застосунків Linux з GUI.
- Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.
- Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

### **Оновлений менеджер пакетів Getit**

Менеджер пакетів Getit в IDE був значно вдосконалений.

Дати випуску релізів пакетів тепер видні, і можливе сортування списку по цих датах; відбір тільки встановлених пакетів, контенту, доступного тільки при наявності підписки, багато чого іншого.

### **Універсальний інсталятор для установки Online і Offline**

В 10.4 включений новий універсальний інсталятор, який використовує технологію на базі Getit. Цей інсталятор підтримує як online, так і offline (з ISO) варіанти установки.

Тепер обоє варіанта установки дозволяють вам указати початковий набір можливостей RAD Studio для установки, наприклад, свою комбінацію мов програмування й цільових платформ, мов інтерфейсу, і додавати до нього або видаляти непотрібне в будь-який момент.

## **2.3 Розгорнута постановка завдання**

Згідно з технічним завданням на випуск кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи GPS навігатора на основі чипу серії SiRF Atlas V.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

- а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	<i>Арк.</i>
<i>Вим.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		<b>28</b>

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

## 3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

### 3.1 Опис функціонування системи

Перш, ніж ми почнемо розглядати різні режими роботи навігаторів, необхідно визначитися з термінологією. У перекладах описів до навігаторів зустрічається різна термінологія, але ми будемо дотримуватися загальновідомої.

Шляхова точка (Waypoint) – точка зі своїм номером або назвою, занесена вами до пам'яті навігатора й зберігається там до того, поки ви її примусово не зітрете. Звичайно в пам'яті сучасного навігатора можна зберігати 250 – 500 шляхових точок.

Шляхову точку можна запам'ятати, безпосередньо перебуваючи на конкретному місці. Наприклад, можна відзначити місце залишеної машини, намет, знайдені у водоймі за допомогою ехолота яму, банку, звалення й т.д.

Шляхову точку можна екстраполювати, тобто нанести на карту й відзначити, як таку, що представляє для нас який-небудь інтерес. Цей випадок добре використовувати для визначення відстаней і напрямків до тих точок, у яких ми ще не були, але куди прагнемо потрапити. Даний режим повною мірою можна використовувати тільки в тих навігаторах, у яких є можливість завантаження карт місцевості (тут не розглядається випадок, коли у вас є звичайна «паперова» карта з топографічною сіткою). Приклад роботи в цьому режимі розглянемо в наступній статті. Там же, для користувачів найпростіших навігаторів, покажемо, як «прив'язати» точку, задану навігатором, до звичайної карти.

Ну й, нарешті, шляхову точку можна поставити, просто вводячи її географічні координати, тобто широту й довготу. Цей режим найбільше часто використовується у всіляких змаганнях типу спортивного орієнтування, які проводяться з використанням супутникового навігатора.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

Рух до обраної шляхової точки – це основний режим використання навігатора. Будь-який, навіть найпростіший і дешевий навігатор, здатний надати вам можливість працювати в цьому режимі.

Шлях (Track) – траєкторія вашого руху, що теж складається із точок, але нумерованих і запам'ятовуваних навігатором автоматично в той час, коли він включений, і коли сила сигналів від супутників достатня для визначення місця розташування. У більше простих моделях ці точки відносяться навігатором через постійний проміжок часу (звичайно через 1 секунду). Краще зроблені моделі дозволяють програмувати режим установки точок шляхи через обраний вами проміжок часу або яку-небудь пройдену відстань (допустимо, через кожні 10 метрів).

Взагалі, режим установки точок шляхи через обрану відстань більше зручний з точки зору економії пам'яті приладу. Справа в тому, що під точку шляху приділяється цілком певне місце в пам'яті приладу. У простих моделях – це 1500-2000 точок, у більше сучасних – 10000 і більше. У міру заповнення пам'яті, відведеної підточування шляху, координати нових точок записуються на місце старих, і інформація про більше ранні точки безповоротно губиться.

При включенні режиму Інверсія шляху (Track Back) можна дуже точно пройти свій шлях у зворотному напрямку. Цей режим іноді буває дуже корисний, наприклад, коли потрібно пройти в темряві по фарватеру.

Маршрут (Route) – серія зв'язаних між собою шляхових точок. Цей режим руху використовується для переміщення від точки до точки. Основних способів використання цього режиму руху – два. Перший, це пересування по дорогах від перехрестя до перехрестя. Другий, перетворення режиму Шлях у режим Маршрут. Справа в тому, що зберігати траєкторію вашого пересування в режимі Шлях не вигідно з погляду економії пам'яті навігатора. Тому дуже часто буває розумно перетворити Шлях у Маршрут, розставивши в характерних місцях шляхові точки вже після закінчення руху, і внести його до пам'яті під своїм ім'ям.



## **Сторінка «НАВІГАЦІЯ» («NAVIGATION»)**

Основна сторінка при використанні приладу. Навігатори бувають дорогі і дешеві, прості і складні, але такий екран є в кожному. Як уже говорилося, головний і найбільше часто використовуваний режим навігації – це рух до наміченої точки. Даний режим наочніше всього реалізується на цій сторінці.

## **Сторінка «ВИСОТОМІР» («ALTIMETER»)**

Висоту, щоправда, дуже приблизно, визначає будь-який навігатор, що «бачить» не менш 4-х супутників. Висота обчислюється відповідно до тієї моделі Землі, що обрана в початкових установах навігатора. Для точного виміру висоти служить Альтиметр (Барометричний висотомір).

Ця функція є не у всіх навігаторів, тільки в дорогих моделях. Принцип роботи – у визначенні атмосферного тиску й перерахування його у висоту над рівнем моря або над рівнем поверхні землі. Зрозуміло, що Висотомір має потребу в калібруванні перед кожним його використанням. Якщо атмосферний тиск міняється по яких-небудь погодних причинах, відповідно до зміни тиску будуть змінюватися й показання Висотоміра. Так як Висотомір по своїй суті є барометром, то з його допомогою корисно стежити за зміною атмосферного тиску для прогнозування зміни погоди.

## **Сторінка «ШЛЯХОВИЙ КОМП'ЮТЕР» («TRIP COMPUTER»)**

Дуже цікава й корисна навігаційна функція. У найпростіших приладах як окрема сторінка відсутня, але основні елементи є на сторінці Навігації.

Шляховий комп'ютер визначає безліч параметрів вашого руху й видає їх на екран у зручному для вас виді. Цих параметрів може бути дуже багато, і кожний може для себе вибрати ті, які йому потрібні в цей момент. Перелічу деякі з них:

- Пройдена відстань.
- Час у русі.
- Час стоянок.
- Передбачуваний час прибуття в намічену точку.

- Поточна швидкість.
- Середня швидкість.
- Середня швидкість переміщення по всьому маршруті з урахуванням стоянок.
- Максимальна швидкість.
- Відстань до поточної точки.
- Відстань до кінця маршруту.
- Курс.
- Відхилення від курсу.
- Інші.

### **Сторінка «ГОЛОВНЕ МЕНЮ» («MAIN MENU»)**

Як і сторінка Навігація, є в кожному GPS-приймачі. Із цієї сторінки можна викликати будь-яку функцію навігатора, там же зосереджені основні налаштування.

Підміну розділів Головного меню позначені піктограмами, які у всіх навігаторах виглядають приблизно однаково.

Відзначити (Mark) – дозволяє відзначити ваше місце розташування у вигляді шляхової точки. Кожній шляховій точці автоматично привласнюється тризначний номер. Ви можете відредагувати назву, замінивши номер ім'ям і призначивши точці який-небудь значок у вигляді піктограм (будинку, табору, готелю й т.д.).

Пошук (Find). Як уже було сказано, у пам'яті навігатора можна зберігати кілька сотень шляхових точок і цей режим допомагає швидко орієнтуватися у всьому їхньому великому списку.

Маршрути (Routes). У цьому режимі ви можете створювати й редагувати потрібні вам маршрути з наявних у пам'яті вашого приймача шляхових точок.

Шляхи (Tracks). Тут ви зберігаєте шляхи ваших переміщень, активізуєте режим руху уздовж обраного шляху, здійснюєте необхідні налаштування.



убудованим акумулятором вам може дуже підійти. Для тих, хто відривається від цивілізації на кілька днів, необхідний навігатор з батареями. У подорож звичайно беруть ще ліхтарики, радіостанції, фотоапарати й іншу техніку, і якщо всі вони працюють на батареях одного типу (це в ідеалі), то в критичний момент знайдеться, чим харчувати навігатор. До багатьох навігаторам продаються адаптери для живлення від автомобільного прикурювача.

Практично всі сучасні навігатори мають інтерфейсний роз'єм для зв'язку з комп'ютером. У комп'ютері ви можете зберігати свої шляхові точки й маршрути, завантажувати їх у навігатор у міру потреби, можете завантажувати з Інтернету електронні карти місцевості (якщо така можливість є у вашого навігатора).

Чутливість прийомної частини навігаторів можна підвищити, підключивши зовнішню антену. Зовнішня антена корисна при використанні навігатора в автомобілі або на катері. Жаль тільки, що роз'єм для зовнішньої антени є тільки в самих «навернених» і найдорожчих моделях.

На закінчення додам, що навігатори мають 7-й клас вологозахисності, а це означає, що їх можна короткочасно роняти в неглибокі калюжі. Деякі з них при цьому ще й мають позитивну плавучість, тобто не тонуть.

### 3.2 Розробка структурної схеми

На рисунку 3.1 представлена структурна схема роботи системи. При розробці системного програмного забезпечення визначення геопараметричних даних об'єкта з відображенням на мобільні пристрої використовувалася технологія GPS (Global Positioning System) – "Система глобального позиціонування" система визначення місця розташування об'єктів, заснована на використанні штучних супутників Землі (більш докладно розглянуто вище).

#### Конвертування карт для завантаження в GPS

Іноді буває так, що карти, які хотілося б завантажити в GPS є у форматі від GPS далекому. Опишемо процес рішення тих самих проблем – як до них

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

підходити взагалі, де чекати підводних каменів, і так далі. Є векторна карта у форматі "ГІС ІНГЕО-4". Завдання, яке треба буде розв'язати, складається із двох великих частин. По-перше, необхідно вихідну базу даних перетворити в один з форматів так званого торованого ланцюжка перетворень. Інакше кажучи, необхідно невідому базу даних привести до того формату, з якого ми вже вміємо одержувати карту для GPS. По-друге, необхідно, щоб дані, наведені до потрібного формату, були коректні для GPS, – важливіше всього домогтися, щоб координати об'єктів, вивантажених з невідомої бази, були коректні й у відомій системі координат.

В "ГІС ІНГЕО-4" дані перетворюються в DXF-формат (autocad-овський) і в MIF-формат (MapInfo-шний формат обміну). Дані зберігаються в локальній декартовій системі координат, що якось потрібно перетворювати в систему координат, зрозумілу подальшим програмам технологічного ланцюжка.

### **Перетворення координат і експорт**

Перетворимо карту у формат Пулково-1942. Для початку необхідно прийняти кілька допущень. По-перше, припустити, що координати карти в "ГІС ІНГЕО-4" є прямокутні декартові. По-друге, припустити, що вихідна карта ІНГЕО є зміщеною й розтягнутою, але не поверненою щодо координат Пулково-1942.

Виходячи із цих припущень і знання основ лінійної алгебри можна записати, як нові координати (Пулково-1942) обчислюються з бази ІНГЕО:

$$n_x = x_x * s_x + d_x \quad n_y = y_y * s_y + d_y, \quad (3.1)$$

де  $x_x, y_y$  – координати точки в базі ІНГЕО,  $s_x, d_x$  – відповідно, розтягання й зсув по осі  $x$ , а  $s_y, d_y$  – розтягання й зсув по осі  $y$ . Як видно, я припускаю, що спочатку до точок ІНГЕО потрібно застосувати розтягання, а потім зсув, щоб одержати Пулково-1942.

Для обчислення параметрів розтягання й зсувів, необхідно мати хоча б дві точки на карті, для яких ми знаємо й старі й нові координати. Загалом кажучи, точок потрібно як мінімум дві, можна більше, але не обов'язково чим більше тим

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

краще. Справа в тому, що задаючи дві точки, і обчислюючи по них параметри перетворення, ми домагаємося того, що ці дві точки будуть однозначно збігатися на місцевості й карті, – тобто в них прив'язка буде абсолютно точною. Погрішність прив'язки точок, що відстоять недалеко від двох точок прив'язки, буде невелика, але вона буде зростати при видаленні від них. Якщо прив'язувати карту по багатьом точкам, то неможливо буде знайти параметри перетворення, що абсолютно прив'язують всі ці точки, зате карта в цілому, можливо, буде прив'язана краще. Формули для обчислення параметрів прив'язки у випадку більше двох точок прив'язки можна виписати, наприклад, скориставшись будь-яким методом апроксимації лінійної функції – хоч методом найменших квадратів.

Отже, точка1 має координати на місцевості ( $nx1=439395$ ,  $ny1=6072078$ ), а в карті ( $xx1=3208.34$ ,  $yy1=5030.94$ ); а точка2 – ( $nx2=433902$ ,  $ny2=6068556$ ) на місцевості, і ( $xx2=-397.74$ ,  $yy2=-427.93$ ) у карті.

Помітимо із самого початку, що в Пулково перша координата задає зсув по горизонталі, а друга по вертикалі, а в ІНТЕГРО, як і в WGS84, навпаки. Тому щоб обчислити параметри перетворень коректно, необхідно поміняти місцями координати в карті. Тоді перша точка на карті буде мати координати ( $xx1=5030.94$ ,  $yy1=3208.34$ ), а друга ( $xx2=-427.93$ ,  $yy2=-397.74$ );

Підставивши координати двох точок в (3.1) одержуємо систему рівнянь, з яких перебувають параметри перетворень у такий спосіб:

$$sx = (nx1 - nx2) / (xx1 - xx2)$$

$$dx = (nx2 * xx1 - nx1 * xx2) / (xx1 - xx2)$$

$$sy = (ny1 - ny2) / (yy1 - yy2)$$

$$dy = (ny2 * yy1 - ny1 * yy2) / (yy1 - yy2),$$

підставляючи куди наші координати одержуємо:  $sx = 1.0062522097064$ ,  $dx = 434332.6055081$ ,  $sy = 0.976683822876919$ ,  $dy = 6068944.46622371$ .

Тепер можна переходити до вивантаження об'єктів з ІНГЕО. Вибіримо там сервіс-експорт-MID/MIF. Вибираємо шари, які необхідно експортувати (наприклад, тільки вулиці), указуємо, куди експортувати, і на наступному вікні

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

задаємо параметри трансформації координат. Задаємо перше перетворення: "збільшення-стиск", 1.0062522097064 по X і 0.976683822876919 по Y. Потім задаємо два послідовних перетворення, що приводять до того, що X і Y поміняються місцями (відбиття відносно прямої  $y=x$ ): друге перетворення: "поворот" на 90 градусів, третє перетворення: "збільшення-стиск" на -1 по X і 1 по Y. І останнє перетворення: "зсув", 10434332.6055081 по X і 6068944.46622371 по Y. Цифри 10 я приписав до споконвічного зсуву по X, рівному 434332.6055081, для того, щоб формально привести використовувані координати до виду Пулково-1942, де, як відомо, до зсуву по X приписується ліворуч номер зони, що для України є 10 (центральний меридіан – 57). Натискаю ОК і в обраній директорії зберігаються файли MID і MIF для кожного із шарів. Дані вивантажені.

### **Імпорт даних у відомий формат**

Всі вивантажені шари лежать у файлах MID і MIF. Для початку перейменую їх у коротких, мовців назви, якщо це ще не так. Далі конвертуємо MIF-MID-файли у формат SHP. Найкраще створити bat-файл, що запускає перекачування в SHP і внести в нього рядки типу:

```
mifshape.exe point streets streets_pt
```

```
mifshape.exe text streets streets_t
```

```
mifshape.exe line streets streets_l
```

```
mifshape.exe poly streets streets_p
```

(за умови, що експортовані файли streets.mif і streets.mid лежать у тій же каталозі, що й конвертер mifshape). Програма створювала по трьох файлу для кожного типу даних, які будуть скачані з MIF-формату: для точок, для тексту, для ліній, і для замкнутих полігонів.

Після завершення цього процесу, ми одержали набір SHP-файлів (із супутніми їм) для всіх даних, які ми хочемо бачити на карті. Якщо всі перетворення були виконані вірно, то координати в них більш-менш коректні. На цьому етапі, можна сказати, завершується робота по підготовці карти, специфічна

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

для завдання; далі йдуть дії, виконувані по тому самому алгоритму при підготовці будь-якої карти, будь вона оцифрована або отримана яким-небудь іншим шляхом.

### **Підготовка карти для GPS**

Коротко відзначимо ті проблеми, що відносяться саме до вивантаженої карти з невідомої бази даних.

Для кожного файлу програма спочатку запросить як відображати об'єкти даного файлу в GPS (чи це дороги, чи струмки, чи ще що), потім запропонує вибрати джерело для підписів до об'єктів, потім запросить використовувану систему координат. У наступному вікні можна вибрати, у які шари потрібно завантажувати карту (про шари як-небудь потім), і нажавши там на Finish, ми нарешті-те побачили результати своєї праці.

Або ж побачили зовсім не те що очікували. На жаль, пророчити, якого роду проблеми можуть виникнути, дуже складно.

Після того, як всі шари завантажені й карта вийшла в більш-менш пристойному вигляді, необхідно перевірити, наскільки добре вона прив'язалася. Як мінімум для цього потрібно перевірити які координати відображає програма для точок, по яких обчислювалося перетворення – вони повинні збігатися з реальною місцевістю. Як правило, є наявності деяка погрішність прив'язки, і на цьому етапі її можна виправити. Для цього зручно завантажити якнайбільше точок і треків, і спробувати побачити, є чи деяка загальна тенденція в їхній помилці. Наприклад, якщо вони всі виявилися зрушені в одну сторону, то за допомогою інструмента Transform можна спробувати внести корекцію в карту. У цей інструмент може виконувати просто зрушення, що задається однією точкою (шляхом вказівки її старої й нової позицій); афінне перетворення, що задається трьома точками й квадратичне перетворення, що задається шістьма точками. Найкраще спробувати виправити ситуацію за допомогою простого зрушення, а якщо це не вийшло, тоді вже намагатися застосовувати афінне перетворення,

попередньо зберігши карту в безпечне місце, через деяку непередбачуваність поведження об'єктів карти при такому складному перетворенні.

### Завантаження карти в GPS

Після того як карта виходить прив'язаною із задовільною точністю, все інше виконується як з будь-якою іншою картою. А саме, настраюються властивості карти (її ID і NAME, обов'язково CODEPAGE – 1251, якщо ви хочете, щоб український текст відображався коректно), потім карта зберігається в форматі MP, і з тої ж програми викликається програма, що готує файл IMG, заливається безпосередньо в GPS за допомогою розробленого програмного забезпечення або його візуального розширення img2gps.

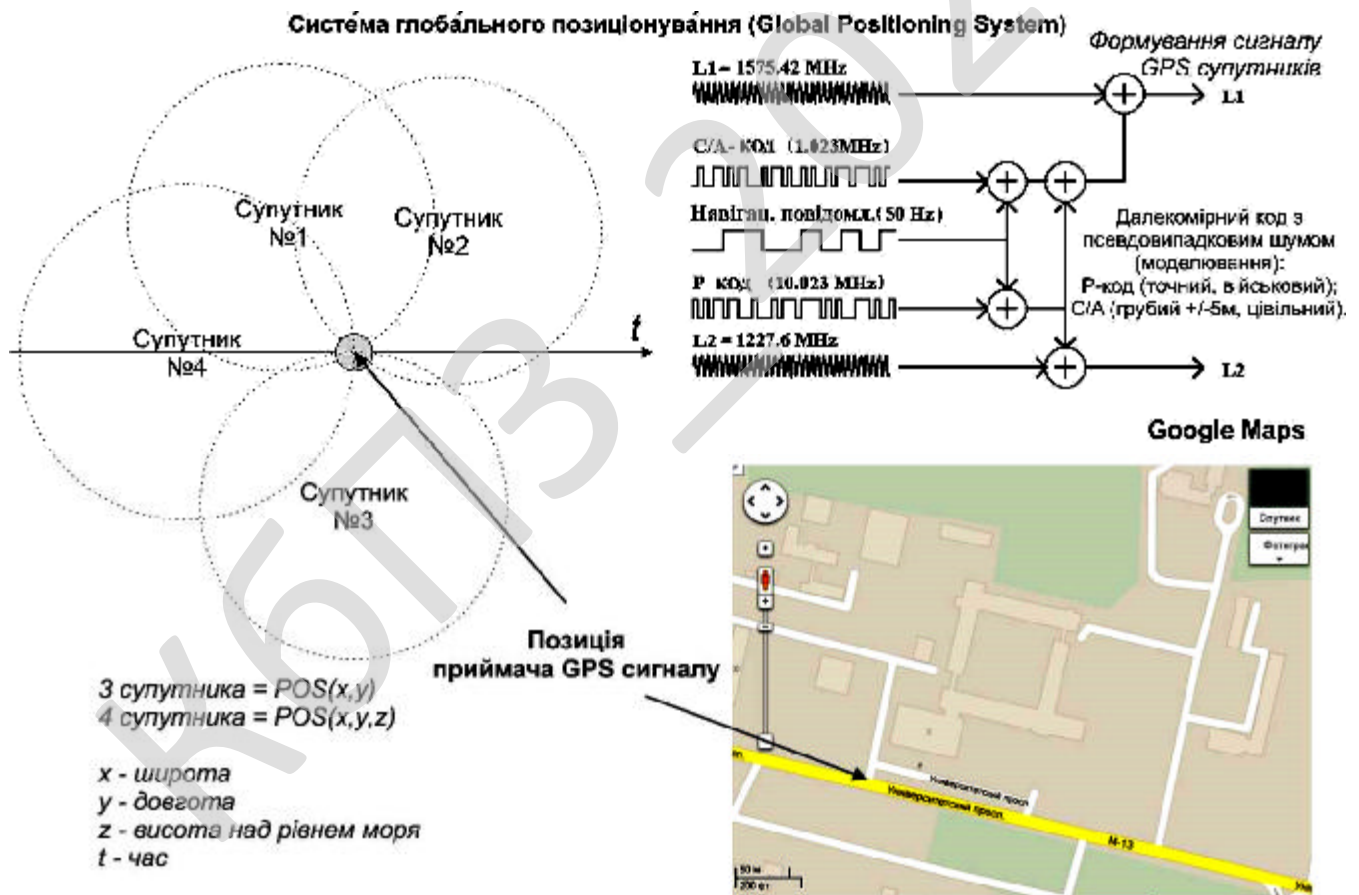


Рисунок 3.1 – Структурна схема системи

## Час «старту» GPS навігатора

Час «старту» необхідний навігаційному приймачу на визначення позиції після включення, залежить від наявної в пам'яті початкової інформації. Виділяються наступні режими:

- «Холодний» старт («автопошук») – час, позиція, альманах і ефемериди невідомі.
- «Теплий» старт – позиція й ефемериди невідомі, час і альманах відомі.
- «Гарячий» старт («перезахват») – альманах, ефемериди відомі, час і позиція відомі з деякою помилкою.

Навігаційні повідомлення передані із супутників містять два типи даних – ефемериди й альманах супутників. В альманасі передаються параметри орбіти, за допомогою яких можна обчислити зразкове місце розташування супутників з достатньо великим ступенем погрішності. Альманах, що зберігається в пам'яті приймача, постійно оновлюється, тому що кожний супутник передає дані альманасу для всіх супутників угруповання. Час «життя» альманасу становить 2-3 місяці. Далі, величина накопиченої помилки в розрахунках буде неприпустимою.

Дані ефемерид містять параметри, що дозволяють більш точно обчислити поточне місце розташування супутників. На відміну від альманасу, кожний із супутників передає, тільки свої власні ефемериди. Час «життя» ефемерид не перевищує 4-6 годин.

Інформація даних ефемерид і альманасу, передана із супутників, постійно коректується. Це відбувається один раз у добу. Мережа наземних станцій, одержує інформацію із супутників, за аналогією зі звичайними користувачами, аналізує виміри, порівнює їх з опорними, розраховує коригувальні виправлення й передає їх на головну станцію, з якої здійснюється передача даних на супутники.

«Холодний» старт приймача може бути зв'язаний не тільки з його тривалою бездіяльністю, але переміщенням на велику відстань у виключеному стані.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

Якщо перший випадок пов'язаний із застарілим альманахом і помилкою у визначенні поточного точного часу, то в другому випадку приймач, не знаючи про своє переміщення, буде намагатися знайти супутники, яких повинні бути видимі на «старому» місці. Користувач може «допомогти» приймачу й зменшити час «холодного» старту, указавши на базовій карті, зразкове «нове» місце розташування. Під час «холодного» старту приймач сканує весь діапазон можливих значень частот і часових затримок навігаційних сигналів. При цьому, у багатоканальних приймачах, кілька каналів можуть використовуватися для пошуку одного супутника, щоб прискорити час його захвата. Після того, як сигнал хоча б від одного супутника буде отриманий і розібраний, приймач буде мати повну інформацію про альманах всього угруповання й, по суті, перейде до «теплого» старту

При «теплому» старті, приймач, включений після 6-ї годин бездіяльності, почне «пошук» сигналів супутників, використовуючи значення поточного часу й дані, що зберігаються в пам'яті, альманаху. Буде здійснюватися пошук тільки тих супутників, які, по теоретичних розрахунках перебувають у видимій півкулі й повинні бути доступні приймачу. Відповідно, відомий досить вузький діапазон частот і часових затримок, що потрібно просканувати у процесі пошуку сигналів. Ця інформація істотно прискорює час захвата супутників, у порівнянні з «холодним» стартом, коли пошук ведеться на широкому діапазоні всіх можливих значень затримок і частот

Варто відзначити, що в момент включення, багатоканального приймача починає пошук сигналів з декількох супутників одночасно. Інформація передана із супутників прив'язана до єдиної шкали часу, містить однакову структуру й досягає антени приймача, приблизно в один й той же час. Тому дані ефемерид, одночасно захоплених супутників, надійдуть у приймач майже що одночасно. Якщо кількість таких супутників більше або рівняється трьом, то це дозволяє приймачу відразу ж розрахувати позицію. У випадку, коли сигнали блокуються перешкодами, то може знадобитися досить тривалий час на визначення позиції.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		<b>43</b>

Наявність повністю отриманих ефемерид, не гарантує використання цього супутника у підрахунку позиції. Інформація передана в ефемеридах може бути неправильною, помилковою, або пов'язаною з несправністю в роботі супутника. Це може бути зв'язано не тільки з несправністю супутника, але й діагностичними роботами проведеними на його борту, процесом уведення його в експлуатацію або тестуванням нових режимів.

«Гарячий» старт пов'язаний з короткочасним вимиканням приймача (до 6-ї годин) не вимагає тривалого часу на визначення позиції. Це пояснюється тим, що отримані раніше ефемериди містять «свіжі» дані, використовувані для визначення точних координат супутників і можуть використовувати в обчисленні позиції. У випадку включення приладу після граничного часу, ефемериди розглядаються застарілими й починає діяти принцип «теплого» старту. Якщо на момент включення приймача видимими залишилися менш 3-х супутників з «свіжими» ефемеридами, то для визначення позиції буде потрібно якийсь час на збір даних ефемерид нового супутника.

Дані ефемерид передаються в складі трьох пакетів. Кожний з пакетів містить однаковий часовий ідентифікатор (IOD – issue of data) по якому можна об'єднати загальну інформацію. Інформація ефемерид передана із супутників кожні 30 секунд, змінюється раз в 2 години, і містить однаковий на цей час IOD. Якщо один з пакетів був пропущений, або отриманий з помилками, то можна виділити аналогічний пакет з наступного повідомлення, перевірити його ідентифікатор і не чекаючи наступних пакетів, використовувати його з раніше отриманими. Це дозволяє приймачу прискорити час «старту».

Існує мінімальний можливий час, необхідний приймачу на «старт», і це визначається структурою переданого сигналу із супутників. Виробники навігаційної апаратури, використовуючи стандартні методи навігації, можуть наблизитися до цього часу, але зменшити його не зможуть. Одним з методів, призначених для рішення цієї проблеми, є Assisted-GPS (A-GPS). Його принцип полягає в обчислення точного місця розташування супутників без інформації

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

ефемерид, на одержанні яких потрібен час. Обчислення здійснюється на використанні точних моделей орбіт супутників, доступних через спеціальні Інтернет – сервіси.

З іншого боку, максимальний час «старту» може значно перевищувати заявлене в технічній специфікації на навігатор час. Це пояснюється навколишніми умовами, у яких відбувається «захват» супутників і «старт» приймача. Якщо приймач перебуває в умовах сильних фізичних перешкод, то навігаційний сигнал піддається зовнішньому впливу, містить помилки й неправильно декодується. Більше, того геометричний фактор цих супутників, що є одним із критеріїв точно визначення позиції, сильно погіршується. Всі ці умови можуть значно збільшити час «старту» приймача.

### **Обчислення радіуса окружності помилки для оцінки точності GPS-вимірів**

Показник CEP – окружність можливої помилки (Circular Error of Probability) один з можливих шляхів оцінити точність вироблених GPS вимірів у даній точці тепер. Завдяки великій кількості факторів зовнішнього середовища впливаючих на виміри – в одній точці показання приладу будуть різними в різні моменти часу. До таких факторів відносяться вплив іоносфери, вплив нижніх шарів атмосфери, багатопроменевість, наявність перешкод на шляху сигналу. Показник CEP використовує опорну точку, або задаваему користувачем, або що обчислюється як середнє геометричне між всіма вимірами, для того, щоб побудувати серію окружностей що показують відповідно 50, 90, 95, 99% можливої помилки.

Для того, щоб визначити CEP повинна бути взята серія вимірів зроблена в одній точці. Наприклад, включений і нерухливий GPS з інтервалом в 2-5 сік реєструє точки треку, які потім завантажуються, конвертуються в share-файл і аналізуються.

Очевидна регулярність розташування точок пов'язана з розрішенням цифрових значень видаваних GPS. Наприклад точність із якої GPS Garmin 12

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

видає координати – 0.000005 десяткових градусів по довготі, і 0.000005 по широті.

Для обчислення CEP дані повинні бути спроектовані. Для обчислення вимірюються відстані між середньою точкою й кожним виміром, а потім вираховується на якій відстані перебуває потрібний відсоток точок.

Результати обчислення CEP 4 різних окружностей: Average = 6.999e+006 5.82936e+006, SD = 7.00012e+006 5.83025e+006, Circular Error Probabilities (CEP), 50% = 3.42281, 90% = 7.36774, 95% = 9.52791, 98% = 14.2946.

Приклад показує, що 50% точок перебувають на відстані 3.4 метра від середнього значення, 98% точок на відстані 14.2 метра від середнього. З діаграми також видний розкид помилки.

Розроблена програма є універсальною системою, що може використовуватися у всіх мобільних пристроях, підтримуючих систему навігації GPS і маючих віконний інтерфейс на основі операційної системи Windows mobile. Коли користувач налаштовує й запускає розроблену програму, відбувається пошук штучних супутників Землі для визначення координат. Сучасні приймачі GPS вбудовані в мобільні пристрої одночасно можуть приймати дані з 12 супутників, що забезпечує високу точність позиціонування (50 – 100 метрів). На схемі представлена геопараметрична сітка з поточним місцем розташування об'єкта й система супутників, що забезпечує визначення місця розташування. При поганому зв'язку із супутниками (знаходження в приміщеннях) необхідно, принаймні 3 супутники, що знаходяться не на одній лінії для визначення поточного положення.

### 3.3 Розробка функціональної схеми

На рисунку 3.2 показана функціональна схема роботи системи. У ній можна чітко простежити як працює приймач GPS і розроблена програма.

					VKPM-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		46

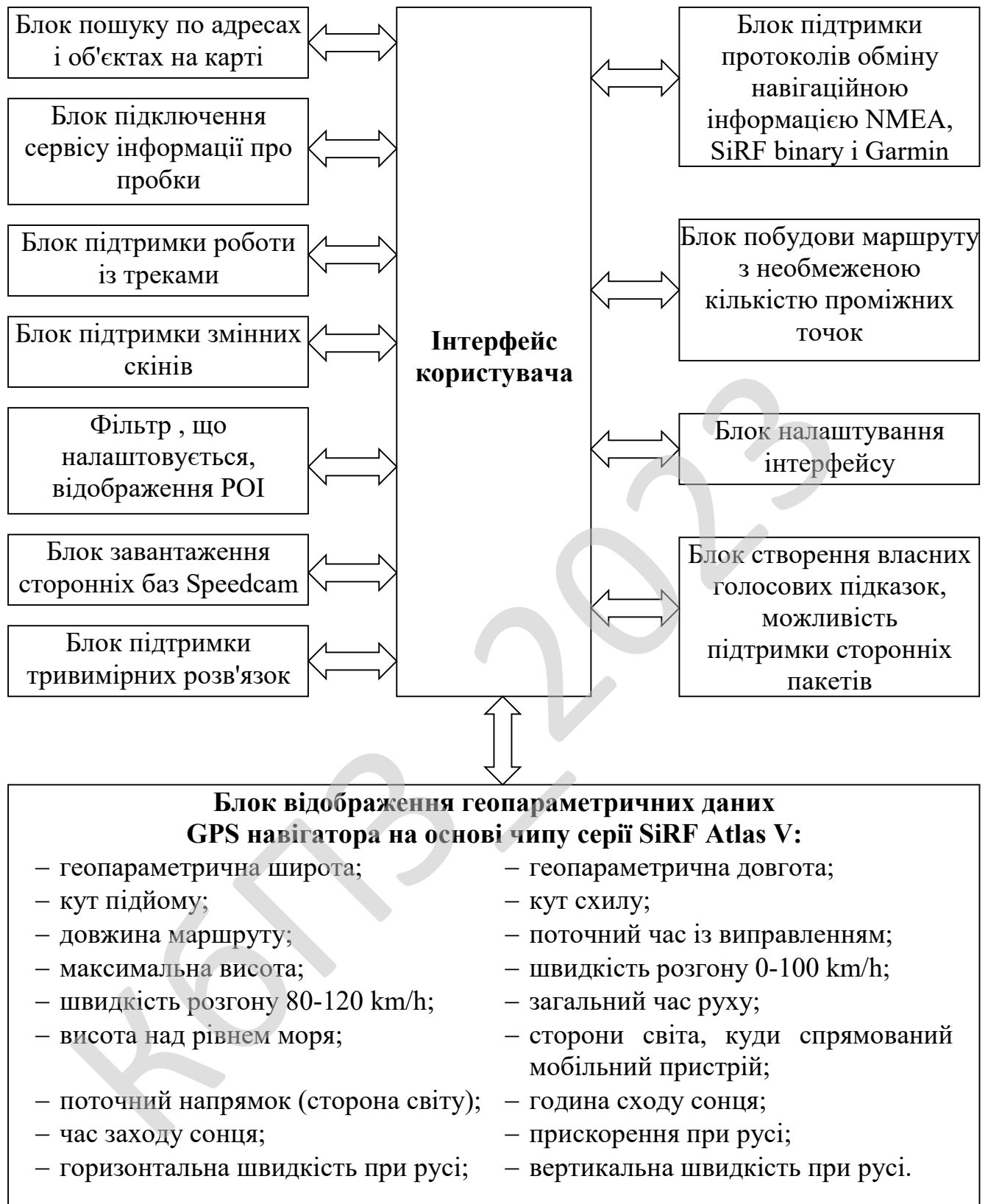


Рисунок 3.2 – Функціональна схема системи

Розглянемо схему зверху долілиць. Штучні супутники Землі подають сигнал на геопараметричну поверхню (землю) сигнал, мобільний пристрій з GPS приймачем через апаратну частину й операційну систему приймають цей сигнал у розроблену програму по запиту користувача.

У розробленої програми існують наступні можливості:

– Підтримка протоколів обміну навігаційною інформацією NMEA, SiRF binary і Garmin.

– Підтримка роботи із треками.

– Підтримка змінних скінів (візуальних оболонок програми).

– Фільтр, що налаштовується, відображення POI (точок інтересу – об'єктів різних типів, наприклад, заправлень, кафе, пересувними й стаціонарними камерами спостереження й т.д.).

– Можливість завантажувати сторонні бази Speedcam (радары, пересувні й стаціонарні камери спостереження за швидкістю, обмеження швидкості, попередження про погане дорожнє покриття, інше).

– Пошук по адресах і об'єктах на карті.

– Можливість побудови маршруту з необмеженою кількістю проміжних точок.

– Інтерфейс, що налаштовується, (можна відключати або включати кнопки керування в програмі).

– Підтримка сторонніх пакетів голосових підказок, можливість створювати власні.

– Підтримка тривимірних розв'язок

– Можливість підключення сервісу інформації про пробки.

Через розроблений інтерфейс програми, при правильному налаштуванні й запуску можна побачити наступні геопараметричні дані:

– геопараметрична широта;

– геопараметрична довгота;

– кут підйому;

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

- кут схилу;
- довжина маршруту;
- поточний час із виправленням;
- максимальна висота;
- швидкість розгону 0-100 km/h;
- швидкість розгону 80-120 km/h;
- загальний час руху;
- висота над рівнем моря;
- сторони світа, куди спрямований мобільний пристрій;
- поточний напрямок (сторона світу);
- година сходу сонця;
- час заходу сонця;
- прискорення при русі;
- горизонтальна швидкість при русі;
- вертикальна швидкість при русі.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

### 3.4 Розробка діаграми процесів

Діаграма взаємодії процесів системи, розробленої у результаті виконання магістерського проектування, наведена на рисунку 3.3. Після початку роботи ПЗ виконується наступна послідовність дій: Основний блок ПЗ; Модуль захисту ПЗ; Інтерфейс ОС; Бібліотеки налаштування та взаємодії з бездротовою лінією зв'язку (Wi-Fi); Інтерфейс розробленого ПЗ; Взаємодія з Google Maps; Авторська форма ПЗ; Тестування лінії зв'язку; Налаштування ПЗ; Отримання первинних даних; Обробка та виведення розрахункових даних.

Обробляються шляхові точки – точки зі своїм номером або назвою, занесення до пам'яті навігатора й зберігається там до того, поки її примусово не

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		<b>49</b>

зітруть. Звичайно в пам'яті сучасного навігатора можна зберігати 250 – 500 шляхових точок. Шляхову точку можна запам'ятати, безпосередньо перебуваючи на конкретному місці. Наприклад, можна відзначити місце залишеної машини, намет, знайдені у водоймі за допомогою ехолота яму, банку, звалення й т.д.

Шляхову точку можна екстраполювати, тобто нанести на карту й відзначити, як таку, що представляє для нас який-небудь інтерес. Цей випадок добре використовувати для визначення відстаней і напрямків до тих точок, у яких ми ще не були, але куди прагнемо потрапити.

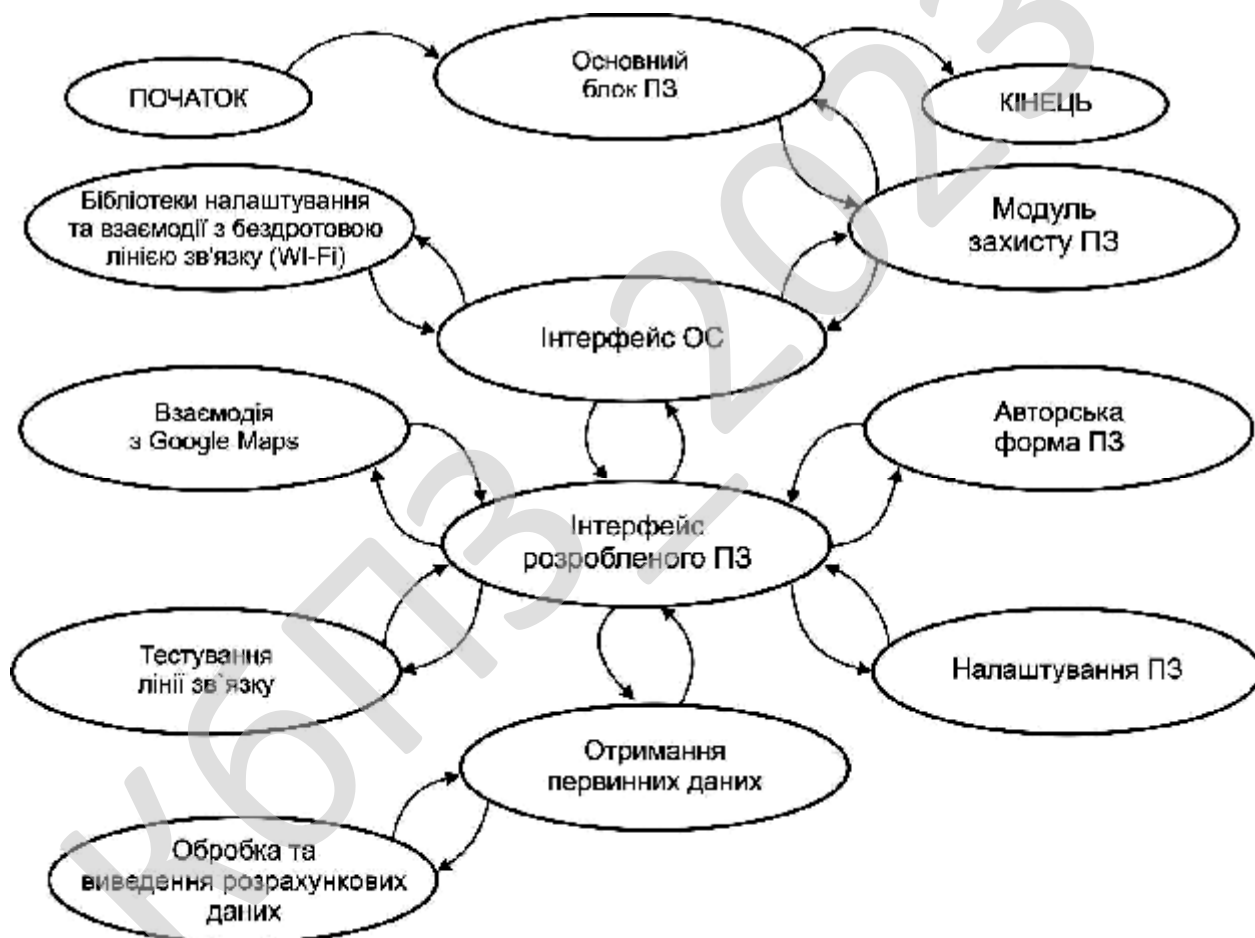


Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

# 4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

## 4.1 Блок-схеми та опис алгоритмів функціонування системи

На рисунку 4.1 наведено блок-схему основної програми. Робота програми розподілена на декілька зон, розглянемо ці зони та дії які проводяться в цих зонах.

Ініціалізація ПЗ:

- Виділення пам'яті ПЗ та ініціалізація початкових змінних.
- Підключення до GPS модулю.
- Підключення файлу налаштування ПЗ.

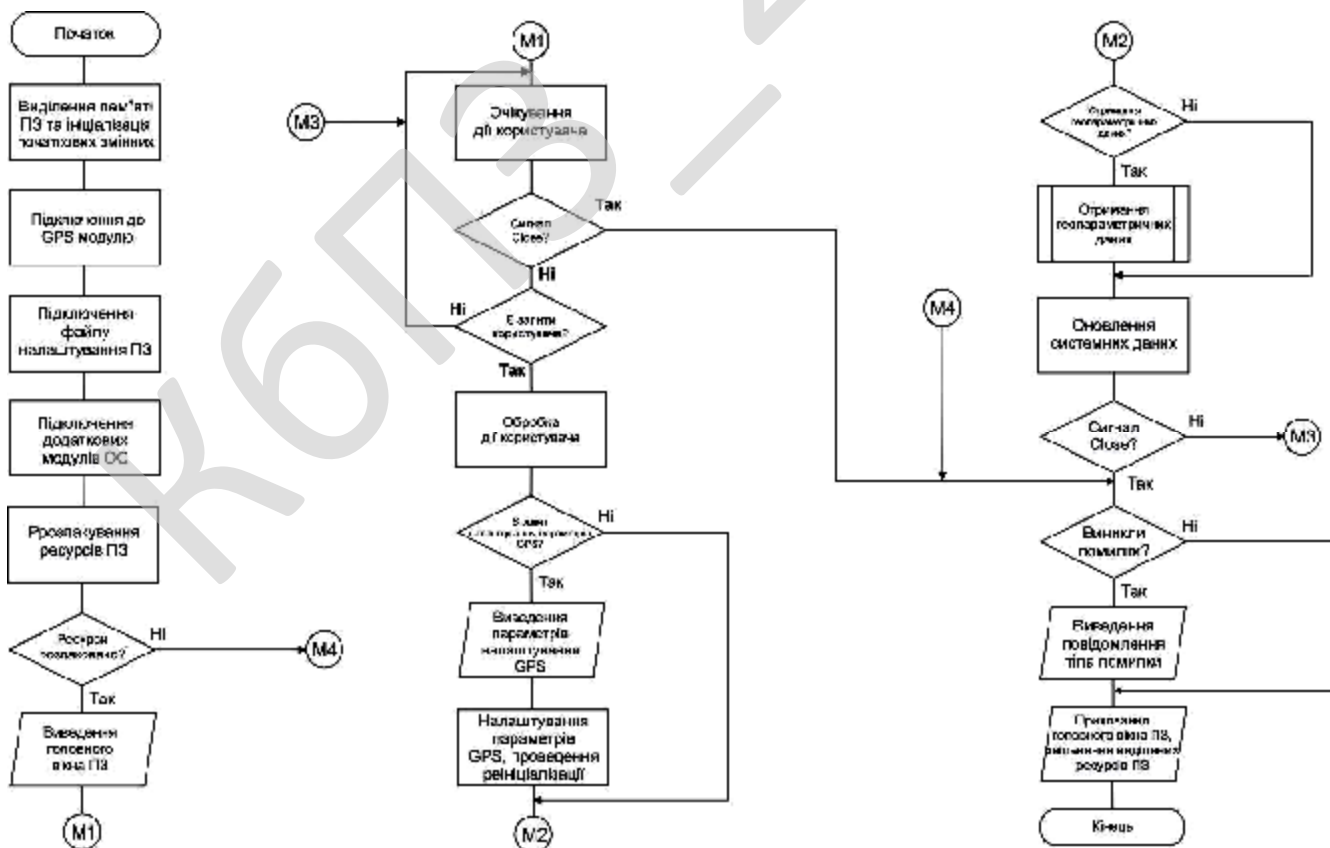


Рисунок 4.1 – Блок-схема основної програми

- Підключення додаткових модулів ОС.
- Розпакування ресурсів ПЗ.
- Ресурси розпаковано?

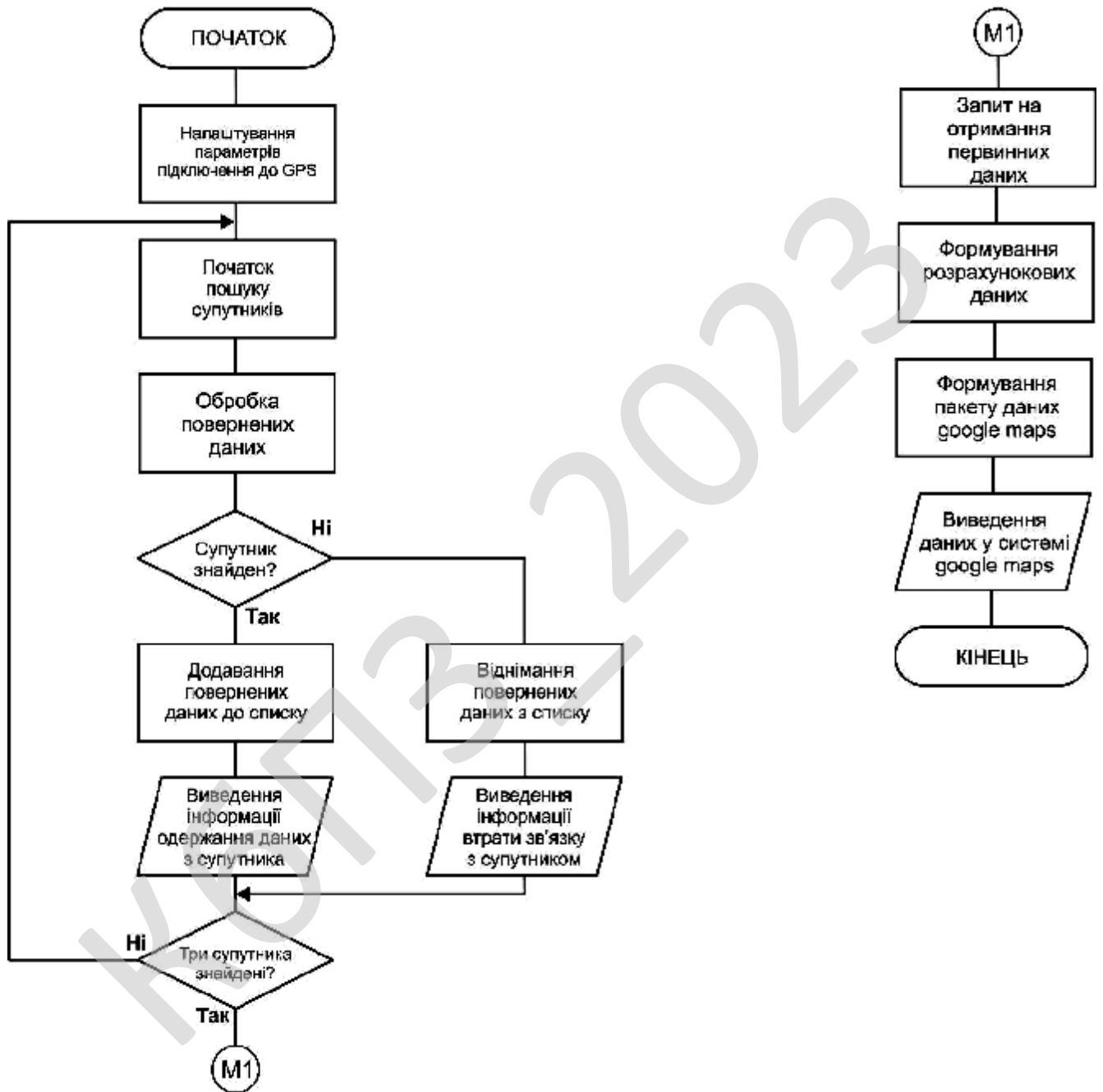


Рисунок 4.2 – Блок-схема роботи підпрограми отримання геопереметричних даних

Робота ПЗ:

- Виведення головного вікна ПЗ.
- Очікування дії користувача.
- Сигнал Close?
- Є запити користувача?
- Обробка дії користувача.
- Є запит налаштування параметрів GPS?
- Виведення параметрів налаштування GPS.
- Налаштування параметрів GPS, проведення реініціалізації.
- Отримання геопараметричних даних?

Виклик підпрограми отримання геопараметричних даних:

- Налаштування параметрів підключення до GPS.
- Початок пошуку.
- Обробка повернених даних.
- Супутник знайдено?
- Додавання повернених даних до списку.
- Виведення інформації одержання даних з супутника.
- Три супутника знайдені?
- Запит на отримання первинних даних.
- Формування розрахункових даних.
- Формування пакету даних Google maps.
- Виведення даних у системі Google maps.

Завершення роботи ПЗ:

- Оновлення системних даних.
- Сигнал Close.
- Виникли помилки.
- Виведення повідомлення тіла помилки.
- Приховання головного вікна ПЗ, звільнення виділених ресурсів ПЗ.

При розробці програми під GPS навігатор автор стовкнувся із проблемою – зберігання даних. Необхідно дуже швидко передавати та обробляти дані та їх зберігати, але існуючі апаратні ресурси GPS навігатора не дозволяють це робити. Для усунення цих недоліків було реалізовано бібліотеку обробки та стиску даних. Розглянемо реалізацію бібліотеки.

Для того, щоб архівувати папку, необхідно спочатку склеювати всі файли, що перебувають у деякій папці. Для цього був розроблений наступний формат:

- 1-3 байт сигнатура, що позначає, що це файл необхідного формату;
- 4-7 байт кількість файлів у склеєному файлі;
- 8-N байт масив структур, описуючий всі файли в склеєному файлі, з

наступною структурою: 2 байти довжина ім'я файлу, 4 байти – розмір файлу. У даній структурі максимальний розмір одного файлу може бути максимум  $2^{32}$  байт.

Розглянемо функцію одержання списку всіх файлів у папці, що передає шлях до папки у форматі TStringList.

```
function Get_All_Files (Filter, Folder: string):TFilesList;
var
  sr: TSearchRec;
  sDirList, _FilesList, _LST: TStringList;
  i, j: Integer;
begin
  j:=0;
  _FilesList:= TStringList.create;
  _FilesList.Clear;
```

На початку створюється об'єкт результат, потім відбувається пошук файлів, які перебувають у папці, шлях до якої був переданий як другий параметр.

Перші два знайдених файли ігноруються тому, що це папки й при переході на які відбувається перехід на один рівень нагору.

```
if FindFirst (Folder + Filter, faAnyFile , sr) = 0 then
  repeat
    j:=j+1;
    if j 0;
    FindClose(sr);
```



```

begin
{перевірка формату}
DATA:=CreateFile(pchar(ArchivePath),GENERIC_READ+GEN
                ERIC_WRITE,FILE_SHARE_READ,0,OPEN_EXISTING,0,0);
    ReadFile(DATA,value,4,_readed,0);
{кількість файлів}
    ReadFile(DATA,_Count,4,_readed,0);
    for i:=1 to _Count do
        begin
            path_in_archive_file := Read_LS(DATA);
            ReadFile(DATA,CurrFileSize,4,_readed,0);
            for j:=1 to CurrFileSize div MainBufferSize do
                begin
                    ReadFile(DATA,pBuff^,MainBufferSize,_readed,0);
                    WriteFile(CurrFile,pBuff^,_readed,_writed,0);
                end;
            ReadFile(DATA,pBuff^,CurrFileSize mod MainBufferSize,_readed,0);
            WriteFile(CurrFile,pBuff^,_readed,_writed,0);
            if FolderPath[Length(FolderPath)]'\\' then
                Delete(FolderPath,Length(FolderPath),1);
        if path_in_archive_file[1]'\\' then
            path_in_archive_file:= '\'+path_in_archive_file;
            path_in_archive_file:=FolderPath+path_in_archive_file;
            mkdir(ExtractFileDir(path_in_archive_file));

```

**Розглянемо розроблену функцію, що буде передавати параметри функції склеювання файлів (функція зворотного виклику).**

```

type
TActionFuntion=function(SourceFileName,DestFileName:string):boolean;

function DoFolderAction(FolderPath, ArchivePath:string;
    ActionFunction :TActionFuntion):boolean;
var
    DATA,CurrFile :THandle;
    Value,i,CurrFileSize,_readed,_writed:DWORD;
    _Files:TStringList;
    path_in_archive_file :string;
    pBuff:Pointer;
begin
    Result:=false;
    if FolderPath[Length(FolderPath)]'\\' then FolderPath:=FolderPath+'\\';
    _Files:=GetAllFiles('*.*',FolderPath);
    DATA:=CreateFile(pchar(ArchivePath), GENERIC_READ+GENERIC_WRITE,FILE_SHARE_READ,0,
        CREATE_ALWAYS,0,0);

```

						<b>БКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			<b>56</b>



```

VirtualFree (pBuff, MEM_RELEASE, 0);
CloseHandle (CurrFile);
DeleteFile (bkpFile);
end;
CloseHandle (DATA);
Result:=True;
end;

```

#### Використовуваний основний запуск склейки файлів:

```

function DODO (f11, f12:string):boolean;
begin
    Result:=CopyFile (pchar (f11), pchar (f12), false);
end;

```

## 4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Khufu. Khufu – це 64-бітовий блоковий шифр. 64-бітовий відкритий текст спочатку розщеплюється на дві 32-бітові половини,  $L$  і  $R$ . Над обома половинами й певними частинами ключа виконується операція XOR. Потім, аналогічно DES, результати проходять деяку послідовність раундів. У кожному раунді молодший значущий байт  $L$  використовується як вхід S-блоку. У кожного S-блоку 8 вхідних біт і 32 вихідних біта. Далі обраний в S-блоці 32-бітовий елемент піддається операції XOR з  $R$ . Потім  $L$  циклічно зрушується на число, кратним восьми біткам,  $L$  і  $R$  міняються місцями, і раунд завершується. Сам S-блок не статичний, він міняється кожні вісім раундів. Нарешті, по закінченні останнього раунду, над  $L$  і  $R$  виконується операція XOR з іншими частинами ключа, і половини поєднуються, утворюючи блок шифртексту.

Хоча частини ключа використовуються для операції XOR із блоком шифрування на початку й кінці виконання алгоритму, головне призначення ключа – генерація S-блоків. Ці S-блоки секретні, по суті, це частина ключа. Повний розмір ключа алгоритму Khufu дорівнює 512 біт (64 байт), алгоритм надає спосіб генерації S-блоків по ключу.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

## 5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо ПЗ, на рисунку 5.1 зображено вікно розробленого ПЗ. ПЗ працює під операційною системою Android, як можна побачити розроблене програм має інтуїтивно зрозумілий інтерфейс. Представлено знайдені угруповання супутників систем США Navstar GPS та Galileo Європа. Чим більше супутників буде знайдено тим точніше буде поточна геопозиція.

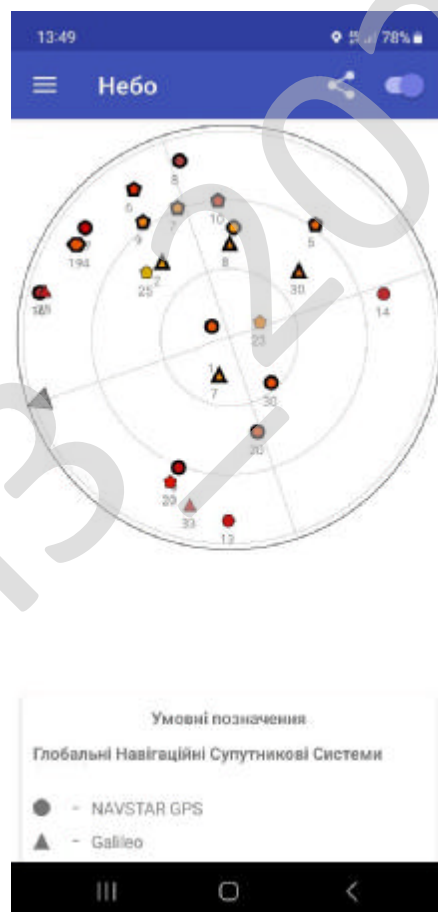


Рисунок 5.1 – Головне вікно ПЗ

Відображує кількість знайдених супутників у текстовому та графічному режимі. Показує якість сигналу зі супутників та геопараметричні

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

характеристики – широта, довгота, висота над рівнем моря. Вказується поточний час та розраховується поточний рух. Виведення даних та їх розрахунок починається зі знайдення мінімальної кількості супутників – 3.

На рисунку 5.2 зображена форма авторського права з виказанням даних розробника, дати розробки та поточної широти та довготи. Обраний тип ліцензії – безкоштовний.



Рисунок 5.2 – Авторське право

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

## 6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи GPS навігатора на основі чипу серії SiRF Atlas V.

*Метою розробки є дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.*

*Об'єктом дослідження є процес GPS навігатора на основі чипу серії SiRF Atlas V.*

*Предметом дослідження є методи GPS навігатора на основі чипу серії SiRF Atlas V.*

*Методи дослідження базуються на методах теорії геопозиціонування, методах математичної статистики, методах розробки програмного забезпечення.*

**Наукова новизна отриманих результатів.** У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод GPS навігатора на основі чипу серії SiRF Atlas V.
- Розроблено вітчизняний продукт GPS навігатора на основі чипу серії SiRF Atlas V, який має більш широкі можливості, на відміну від існуючих аналогів.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		<b>61</b>



Продовження таблиці 7.1

1	2	3
7. Кількість макетів вхідної інформації	–	8
8. Кількість форм вихідної інформації.	–	6
9. Мова програмування (1-6)	–	2
10. Попередній досвід (1-6)	–	3
11. Гнучкість проекту ПП (1-6)	–	3
12. Детальність проекту ПП (1-6)	–	1
13. Рівень спрацьованості колективу (1-6)	–	2
14. Ступінь вимірності процесів (1-6)	–	3
15. Необхідна надійність програмного забезпечення (1-6)	–	3
16. Розмір бази даних (порівняно з розміром програми) (1-6)	–	4
17. Складність кінцевого програмного продукту (1-6)	–	5
18. Необхідний рівень забезпечення повторного використання (1-6)	–	2
19. Документованість відповідно до планованого життєвого циклу (1-6)	–	3
20. Вимоги до швидкодії ПП (1-6)	–	3
21. Обмеження на розміри основного сховища даних (1-6)	–	2
22. Різноманітність використовуваних обчислювальних платформ (1-6)	–	4
23. Професійний рівень аналітиків (1-6)	–	3
24. Професійний рівень програмістів (1-6)	–	4
25. Постійність складу команди розробників (1-6)	–	2
26. Досвід розробки додатків (1-6)	–	1
27. Досвід роботи з обчислювальною платформою (1-6)	–	2

Продовження таблиці 7.1

1	2	3
28. Досвід роботи з мовою і інструментами середовища розробки (1-6)	–	2
29. Досвід роботи з програмними інструментами розробки (1-6)	–	3
30. Розробка ПЗ для декількох серверів одночасно (1-6)	–	3
31. Вимоги до дотримання встановленого графіка робіт (1-6)	–	2
32. Вартість ПЗ у розробника (НМА), грн.	–	260000
33. Норматив додаткової зарплати, % :	Н <sub>д</sub>	10
34. Норматив відрахувань у соціальні фонди, %	Н <sub>с</sub>	22
35. Норматив загальногосподарських витрат, %	Н <sub>г</sub>	15
36. Норматив витрат на освоєння нових мов програмування, %	Н <sub>п</sub>	15
37. Рівень рентабельності програмної продукції, %	Р <sub>е</sub>	40
38. Ставка податку на додану вартість, %	Н <sub>дв</sub>	20

## 7.2 Розрахунок трудомісткості розробки програмної продукції

Значення трудомісткості розробки програмного забезпечення для стадій ТЗ, ЕК, ТП та ВП визначаємо по типовим нормам часу приведеним в додатках МВ. Стадія РП є найбільш тривалою і трудомісткою, що робить значний вплив на інші стадії проекту.

Визначимо трудомісткість розробки ПЗ для стадії РП.

Обчислюємо номінальні трудовитрати, люд-міс.:

$$T_{ном} = A \text{ Size}^B, \quad (7.1)$$

де:  $A$  – коефіцієнт Боема,  $A = 2,45$ ;

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64



Таблиця 7.2 – Визначення трудомісткості розробки програмного забезпечення

Стадії розробки	Трудомісткість за типовими нормами та розрахунками	
	Величина, люд/дні	Підстава
Технічне завдання	9	Д5
Ескізний проект	10	Д6
Технічний проект	15	Д7
Робочий проект	122	Ф 7.1-7.4
Впровадження	15	Д13
Всього	171	–

### 7.3 Визначення чисельності виконавців і планового фонду зарплати

Чисельність ставок інженерів-програмістів для розробки програмного забезпечення визначається за формулою:

$$Ч = \frac{T_{нз} N}{F_{pq} - H_{ев}}, \quad (7.5)$$

де:  $F_{pq}$  – плановий фонд робочого часу одного спеціаліста, днів;

$T_{нз}$  – трудомісткість розробки програмного забезпечення люд-дні.

$$Ч = \frac{171 \cdot 1}{48-3} = 3,8 \text{ ставки.}$$

Чисельність інженерів-електронщиків для проведення технічного обслуговування та ремонту комп'ютерних мереж визначається в залежності від наявності технічних засобів і норм витрат часу на виконання профілактичних робіт на протязі року.

Визначаємо затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за період розробки. Результати розрахунку зводимо до таблиці 7.3.

Таблиця 7.3 – Затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за розрахунковий період

Найменування обладнання	Профілактичне обслуговування			
	Кількість хв. на один. обл.	Кількість обладнання	Затрати часу в хв.	Затрати часу в год.
Системний блок ПК	90	7	630	10,5
Монітор	60	7	420	7
Клавіатура	30	7	210	3,5
Маніпулятор «мишка»	30	7	210	3,5
Принтер матричний	60	0	0	0,0
Принтер лазерний	120	1	120	2
Принтер струминний	60	1	60	1
Сканер	20	1	20	0,33
Концентратор-маршрутизатор	30	2	60	1
Кабельні господарства ЛОМ на 1 м.п.	2,5	200	500	8,33
Копіювальний апарат	140	1	140	2,33
Усього за рік:			3 <sub>ч</sub>	39,49

Час на профілактику обладнання в загальному балансі робочого часу інженерів-електронщиків не повинен складати більше 10%.

Виходячи з цього фонд робочого часу інженерів-електронщиків складає:

$$\Phi_{op}^c = \frac{3_{ч} \cdot n_{mic}}{1,2}, \quad (7.6)$$

$$\Phi_{op}^c = \frac{40 \cdot 2}{1,2} = 67 \text{ год.}$$

Визначаємо необхідну кількість ставок штатного персоналу сектора ТО:

$$Ч_{ел} = \frac{\Phi_{op}^c}{F_{op} \cdot T_{зм}}, \quad (7.7)$$



Продовження таблиці 7.4

Посада	Вид роботи	Час	К-ть штатних одиниць
Продакт-менеджер	Презентації нової продукції, пошук каналів збуту	2	0,5
	Підтримка постійних клієнтів	1	
	Оформлення договорів, ведення тендерів	0,5	
	Контроль взаєморозрахунків з постачальниками	0,5	
Всього		4	
Дизайнер WEB	Розробка концепції оформлення та інтерфейсу сайту, оптимізація дизайну існуючих, проектує їх структуру та навігацію	2	0,5
	Створення графічних і стилістичних елементів сайту	1	
	Оформлення банерів і промо-сторінок	0,5	
	Розміщення графіки і контенту на Інтернет сторінках	0,5	
Всього		4	
Інженер верстальник	Розробка та верстка макетів рекламної продукції та технічної документації	1	0,25
	Верстка друкованих видань	0,5	
	Додрукова підготовка макетів	0,25	
	Розміщення графіки і контенту на Інтернет сторінках	0,25	
Всього		2	

Складемо штатний розклад виконавців.

Таблиця 7.5 – Штатний розклад виконавців

Посада	Кількість ставок	Середньомісячний оклад, грн.	Всього за період розробки, грн.
Керівник (ІТ-менеджер)	1	17706	35412
Продакт-менеджер	0,5	15000	15000
Інженер-програміст	3,8	17500	133000
Інженер-електронщик	0,2	15000	6000
Інженер-системотехнік	0,25	15000	7500
Адміністратор мережі	0,5	15000	15000
Системний програміст	0,25	15000	7500
Дизайнер WEB	0,5	16000	16000
Інженер-верстальник	0,25	15000	7500
Бухгалтер-економіст	0,5	16000	16000
Всього за період розробки	$R_{cn} = 7,75$	-	$\Phi_{роб} = 258912$

Розрахуємо середньоденну зарплату одного виконавця:

$$z_{cd} = \frac{\Phi_{роб}}{R_{cn} F_{pq}}, \quad (7.8)$$

де:  $\Phi_{роб}$  – загальна сума зарплати за плановий період, грн.

$$z_{cd} = \frac{258912}{7,75 \cdot 48} = 696 \text{ грн.}$$

#### 7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника

Балансова вартість будівель визначається з урахуванням кількості робочих місць виконавців, питомої площі на одне робоче місце, та вартості одного квадратного метра виробничої площі:

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

$$B_{y\partial} = R_{cn}^1 S_y \Pi_{nl}, \quad (7.9)$$

де:  $R_{cn}^1$  – кількість робочих місць виконавців, шт. Приймаємо 12 робочих місць;

$S_y$  – питома площа на одне робоче місце,  $m^2$ ;

$\Pi_{nl}$  – вартість одного квадратного метра площі, грн.

Згідно даних інтернет ресурсу DOM.RIA (<https://dom.ria.com>) ціна одного квадратного метра площі, вік якої не перевищує 30 років, по місту складає 500...1600 у.о./ $m^2$ . Враховуючи, що курс складає 1 у.о. = 38 грн. приймаємо для розрахунку вартість одного метра квадратного рівною 20000 грн./ $m^2$ . На кожне робоче місце у середньому потрібно 8  $m^2$ . З урахуванням цього:

$$B_{y\partial} = 12 \cdot 8 \cdot 20000 = 1920000 \text{ грн.}$$

Вартість передавальних пристроїв складає 10% від вартості будівель, і у даному випадку вона складе: 192000 грн.

Балансова вартість інвентарю розраховується за нормою 3500 грн. на одне робоче місце. Тобто:

$$I_{nb} = R_{cn}^1 \cdot \Pi_m, \quad (7.10)$$

де:  $\Pi_m$  – ціна меблів для одного робочого місця, грн.

$$I_{nb} = 12 \cdot 3500 = 42000 \text{ грн.}$$

Балансова вартість обчислювальної техніки визначається по оптовим цінам постачальника з врахуванням витрат на транспортування.

Специфікація на обчислювальну техніку наведена в таблиці 7.7.

Дані по оптовій ціні на обладнання та комплектуючі вибирались по прайсу фірми Компбест за 25.10.23 – джерело <https://compbest.com.ua/>

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71



Продовження таблиці 7.6

Найменування комплектуючої або обладнання	Тип	Оптова ціна
інше	Клавіатура, мишка	Подарунок
Монітор	22" TFT, ASUS VW223D ( 5ms, 300/3000: 170/160, D-SUB, Wide)	3200
Принтер лазерний	Canon i-SENSYS LBP6030W	2700
Принтер струминний	Epson Stylus Photo P50 (C11CA45341) + USB cable	5500
Сканер	Epson Perfection V37 Photo	2970
Копіювальний апарат	Canon i-SENSYS MF217W with Wi-Fi	5965
Пристрій безперебійного живлення	UPS APC BACK-UPS ES 525VA 230V RUSSIA (BE525-RS)	1496

Витрати на транспорт, монтаж та випробування можуть бути прийняті в межах до 10% від оптової ціни.

Для визначення необхідної кількості капітальних вкладень складемо таблицю 7.8.

Таблиця 7.7 – Балансова вартість обчислювальної техніки

Найменування обчислювальної техніки	Кількість, шт.	Ціна за одиницю, грн.	Витрати на транспортування, монтаж та випробування.	Загальна вартість, грн.
Персональні комп'ютери	12	11186	13423,2	147655,2
Принтер лаз.	2	2700	540	5940
Принтер струм.	1	5500	550	6050

Продовження таблиці 7.7

Найменування обчислювальної техніки	Кількість, шт.	Ціна за одиницю, грн.	Витрати на транспортування, монтаж та випробовування.	Загальна вартість, грн.
Сканери	1	2970	297	3267
Копіюв. апарат	1	5965	596,5	6561,5
Всього	–	–	–	169473,7

Таблиця 7.8 – Вартість основних фондів та амортизаційні відрахування розробника

Групи та види основних фондів	Балансова вартість, грн.	Амортизація	
		Норма, %	Відрахування, грн.
1	2	3	4
Група 3			
1. Будівлі	1920000	-	-
2. Передавальні пристрої	192000	-	-
Всього по групі	2112000	5	105600
Група 4			
3. Обчислювальна техніка	169474	-	-
Всього по групі	169474	50	84737
Нематеріальні активи			
4. Нематеріальні активи	260000	10	26000

Продовження таблиці 7.8

1	2	3	4
Група 5, 6			
5. Вимірювальні пристрої	5190	25	1297,5
6. Транспортні засоби	72500	20	14500
7. Господарський інвентар	42000	25	10500
Всього по групі	119690	-	26297,5
Разом	$K_p = 2661164$		$A_p = 242634,5$

Примітка: вартість автомобіля Sens 2010 взята по даним з автосалону авто-РІА, джерело [https://auto.ria.com/uk/auto\\_zaz\\_sens\\_33451541.html](https://auto.ria.com/uk/auto_zaz_sens_33451541.html), складає 72500 грн.

### 7.5 Визначення собівартості розробки та ціни програмної продукції

Визначимо основну зарплату виконавців:

$$Z_o = \frac{Z_{cd} \cdot T_{nz}}{N_e}, \quad (7.11)$$

де:  $N_e$  – кількість екземплярів програм, шт.

$$Z_o = 696 \cdot 171 / 260 = 458 \text{ грн.}$$

Визначимо додаткову зарплату (оплата відпусток, виконання державних та суспільних обов'язків) на рівні 10%:

$$Z_d = Z_o \cdot H_q \cdot 0,01, \quad (7.12)$$

де:  $H_q$  – норматив додаткової зарплати, %.

$$Z_d = 458 \cdot 10 \cdot 0,01 = 46 \text{ грн.}$$

Відрахування на соціальні потреби за нормативом  $H_c = 22\%$  від суми основної та додаткової зарплати:

$$C_{ou} = 0,01 \cdot H_c (Z_o + Z_d), \quad (7.13)$$

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

де:  $H_c$  – відрахування на соціальні потреби, %.

$$C_{oc} = 0,01 \cdot 22(458+46) = 111 \text{ грн.}$$

Визначимо загальногосподарські витрати (електроенергію, ремонт і утримання приміщень і т.д) за нормативом  $H_z = 15\%$  від основної зарплати:

$$G_{ocn} = 3_o \cdot H_z \cdot 0,01, \quad (7.14)$$

де:  $H_z$  – загальногосподарські витрати, %.

$$G_{ocn} = 458 \cdot 15 \cdot 0,01 = 69 \text{ грн.}$$

Визначимо витрати на матеріали для розробки програмної продукції за нормами споживання та діючими цінами за одиницю виміру:

$$Z_M = (Z_{M1} + Z_{M2} + Z_{M3})/N_e, \quad (7.15)$$

де:  $Z_{M1}$  – вартість паперу, грн.;

$Z_{M2}$  – вартість запам'ятовуючих пристроїв, грн.;

$Z_{M3}$  – вартість фарби, картриджей, тонеру, грн.;

$N_e$  – кількість екземплярів програм, шт.

Згідно прийнятих норм на підприємстві  $n_{вум}$  приймаємо 1 пачку паперу на період розробки. Тоді, враховуючи, що вартість пачки паперу складає  $C_n=240$  грн., визначаємо вартість паперу за період розробки:

$$Z_{M1} = C_n \cdot N_m. \quad (7.16)$$

$$Z_{M1} = 240 \cdot 1 = 240 \text{ грн.}$$

Згідно прийнятих норм по комплектації до вартості запам'ятовуючих пристроїв входить вартість CD/DVD дисків. Їх кількість дорівнює кількості коробочних версій запропонованого продукту (приймаємо 100):

$$Z_{M2} = \sum C_d, \quad (7.17)$$

де:  $C_d$  – вартість дисків CD/DVD: CDR box – 23,6 грн./шт., DVD-R box – 33,93 грн./шт.

$$Z_{M2} = 100 \cdot 33,93 = 3393 \text{ грн.}$$

Згідно норм одноразовій заправці підлягають усі друкуючі пристрої і становить:

$$Z_{M3} = \sum C_z, \quad (7.18)$$

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

де:  $C_3$  – вартість розхідних матеріалів друкуючих пристроїв: відновлення та заправка картриджу для Canon i-SENSYS LBP6030W – 574 грн.; картридж для Epson Stylus Photo P50 – 558 грн.; відновлення картриджу для MF217W – 570 грн.

$$Z_{M3} = 574 + 558 + 570 = 1702 \text{ грн.}$$

$$Z_M = (240 + 3393 + 1702) / 260 = 21 \text{ грн.}$$

Визначимо витрати на освоєння нових мов програмування або операційних систем за нормативом ( $H_n = 15\%$ ) від основної зарплати виконавців:

$$O_n = Z_o \cdot H_n \cdot 0,01, \quad (7.19)$$

де:  $H_n$  – норматив витрат на освоєння нових мов програмування, %.

$$O_n = 458 \cdot 15 \cdot 0,01 = 69 \text{ грн.}$$

Визначимо витрати на амортизацію основних фондів з урахуванням загальної річної суми амортизаційних відрахувань та кількості екземплярів програм ( $N_e = 260$  прим.):

$$A_m = \frac{A_p \cdot N_{\text{міс}}}{N_e \cdot 12}, \quad (7.20)$$

де:  $A_p$  – загальна річна сума амортизаційних відрахувань, грн.

$$A_m = 242634 \cdot 2 / (260 \cdot 12) = 156 \text{ грн.}$$

Повна собівартість ПЗ визначається як сума витрат за попередніми статтями калькуляції:

$$C_n = Z_o + Z_d + C_{oc} + \Gamma_{ocn} + Z_M + O_n + A_m. \quad (7.21)$$

$$C_n = 458 + 46 + 111 + 69 + 21 + 69 + 156 = 930 \text{ грн.}$$

Визначимо плановий прибуток за рівнем рентабельності ( $P_n$ ) програмної продукції, яка залежить від складності програми та ступеня новизни задачі.

Для даного програмного забезпечення рівень рентабельності складає 40%.

$$P_p = 0,01 \cdot P_n \cdot C_n, \quad (7.22)$$

де:  $P_n$  – рівень рентабельності, %.

$$P_p = 0,01 \cdot 40 \cdot 930 = 379 \text{ грн.}$$

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

Величини ціна підприємства, податок на додану вартість, відпускна ціна програмної продукції визначаються за формулами, приведеними в таблиці 7.9

Таблиця 7.9 – Нормативна калькуляція собівартості розробки програмного забезпечення задачі

Найменування статей витрат	Позначення	Величина, грн
1	2	3
1. Основна зарплата виконавців	$Z_o$	458
2. Додаткова зарплата виконавців	$Z_d$	46
3. Відрахування на соціальні потреби	$C_{oc}$	111
4. Загальногосподарські витрати	$\Gamma_{ocn}$	69
5. Витрати на матеріали	$Z_M$	21
6. Освоєння нових операційних систем, мов програмування	$O_n$	69
7. Амортизація основних фондів	$A_m$	156
8. Повна собівартість програмного забезпечення	$C_n$	930
9. Плановий прибуток	$P_p$	379
10. Ціна підприємства $C_n = C_n + P_p$	$C_n$	1309
11. Податок на додану вартість $ПДВ = 0.01 \cdot H_{dv} \cdot C_n$	$ПДВ$	261,8
12. Відпускна ціна програмної продукції $C = C_n + ПДВ$	$C$	1570,8

### 7.6 Визначення об'єму капітальних вкладень у споживача програмної продукції

Об'єм капітальних вкладень у споживача програмної продукції визначаємо на основі балансової вартості основних фондів, яка враховує ціну, транспортно-заготівельні витрати, вартість будівель, монтажних та





$$E_e = (C_n - C_n) \cdot N_e - \sum_{i=1}^m E_{p_m} \cdot K_{p_m}, \quad (7.25)$$

де:  $K_p$  – балансова вартість основних фондів розробника, грн.;  $E_p$  – розрахунковий коефіцієнт капіталовкладень.

$$E_e = (1309 - 930) \cdot 260 - (0,05 \cdot 2112000 + 0,5 \cdot 169474 + 0,25 \cdot 47190 + 0,2 \cdot 72500 + 0,1 \cdot 260000) \cdot 2/12 = 58101 \text{ грн.}$$

Визначимо період окупності додаткових капітальних вкладень у виробника програмної продукції:

$$T_e = \frac{K_p^*}{(C_n - C_n) \cdot N_e}, \quad (7.26)$$

де:  $K_p^*$  – балансова вартість основних фондів розробника без врахування вартості ОФ третьої групи, так як їх строк служби на порядок більший ніж період розробки ПЗ.

$$T_e = \frac{549164}{(1309 - 930) \cdot 260 \cdot 12 / 2} = 0,9 \text{ року.}$$

Визначимо величину економічного ефекту у користувача програмної продукції за формулою:

$$E_{cn} = (I_{\bar{o}} - I_n) - E_n(K_n - K_{\bar{o}}), \quad (7.27)$$

де:  $I_{\bar{o}}$ ,  $I_n$  – величина експлуатаційних витрат за базовим и новим варіантом відповідно;

$K_{\bar{o}}$ ,  $K_n$  – об'єм капітальних вкладень за варіантами, що порівнюються.

$$E_{cn} = (24404 - 14452) - 0,5 \cdot 1571 = 9167 \text{ грн.}$$

Визначимо період окупності додаткових капітальних вкладень у споживача програмної продукції за рахунок зниження експлуатаційних витрат:

$$T_{cn} = \frac{K_n - K_{\bar{o}}}{I_{\bar{o}} - I_n}, \quad (7.28)$$

$$T_{cn} = \frac{1571}{24404 - 14452} = 0,2 \text{ року.}$$

Показники економічної ефективності програмної продукції зводимо до таблиці 7.13.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		81

Таблиця 7.13 – Показники економічної ефективності програмної продукції

Найменування показників	Одиниця виміру	Величина
1. Кількість екземплярів програми	Прим.	260
2. Повна собівартість розробленої програми	Грн.	930
3. Ціна розробленої програми	Грн.	1309
4. Плановий прибуток від реалізації розробленої програми	Грн.	379
5. Рентабельність програмної продукції	%	40
6. Об'єм додаткових капітальних вкладень у виробника програмної продукції	Грн.	2661164
7. Загальний прибуток від реалізації програмної продукції	Грн.	98540
8. Величина економічного ефекту при виготовлені програмної продукції	Грн.	58101
9. Період окупності додаткових капітальних вкладень у виробника програмної продукції	Років	0,9
10. Об'єм додаткових капітальних вкладень у споживача програмної продукції	Грн.	1571
11. Величина економічного ефекту у користувача програмної продукції	Грн.	9167
12. Період окупності додаткових капітальних вкладень у користувача програмної продукції	Років	0,2

### 7.9 Висновки

Розроблена програма економічно вигідна. За рахунок впровадження програмного забезпечення досягається скорочення часу обробки інформації, підвищується культура праці, підвищення якості приймаючих управлінських рішень.

## 8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

### 8.1 Вступ

Охорона праці – система збереження життя і здоров'я працівників у процесі трудової діяльності, що включає правові, соціально-економічні, організаційні, технічні, санітарно-гігієнічні, лікувально-профілактичні, реабілітаційні та інші заходи. Загальні положення державної політики, щодо галузі охорони праці зазначені у Законі України “Про охорону праці”. Цей Закон визначає основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я у процесі трудової діяльності, на належні, безпечні і здорові умови праці, регулює за участю відповідних органів державної влади відносини між роботодавцем і працівником з питань безпеки, гігієни праці та виробничого середовища і встановлює єдиний порядок організації охорони праці в Україні [20]. Законодавство про працю містить норми і вимоги з техніки безпеки і виробничої санітарії, норми, що регулюють робочий час і час відпочинку, звільнення та переведення на іншу роботу, норми праці щодо жінок, молоді, гігієнічні норми і правила тощо. Загальний нагляд за додержанням норм охорони праці покладено на прокуратуру, спеціальний – на професійні спілки. Контроль за безпекою праці здійснюють також, державні й відомчі спеціалізовані інспекції. Науково-технічний прогрес вніс серйозні зміни в умови виробничої діяльності робітників розумової діяльності. Їх праця стала більш інтенсивною, напруженою і вимагає значних витрат розумової, емоційної і фізичної енергії. Це призвело до необхідності у знаходженні комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці та відпочинку. Охорона здоров'я робітників, забезпечення безпеки умов праці, ліквідація та профілактика професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

## 8.2 Аналіз умов праці на робочому місці ІТ-фахівця

На робочому місці ІТ-фахівця (або програміста) виникають небезпечні та шкідливі для безпечної життєдіяльності фактори:

- підвищений рівень шуму;
- несприятливі мікрокліматичні умови;
- недостатній рівень освітленості;
- шкідливі речовини;
- підвищений рівень електромагнітних випромінювань радіочастот;
- висока напруга електричної мережі;
- статична електрика та інші.

Робота програміста супроводжується також підвищеним ступенем напруженості трудового процесу. При систематичному впливі виробничих факторів, які не відповідають нормативним показникам, зростає рівень професійно зумовленої захворюваності працюючих та можуть виникнути професійні захворювання органів зору, руху, нервової системи. Таким чином, вивчення умов праці на робочому місці програміста є необхідною умовою запобігання негативних наслідків впливу небезпечних та шкідливих факторів. Робоче місце, добре пристосоване до трудової діяльності інженера, правильно і доцільно організоване, щодо простору, форми, розміру забезпечує йому зручне положення при роботі і високу продуктивність праці при найменшому фізичному і психічному напруженні.

Нормування параметрів проводиться в залежності від періоду року та категорії важкості виконуваних робіт. Для постійних робочих місць, якими є робочі місця ІТ-фахівців, встановлені оптимальні параметри мікроклімату, а за неможливості їх дотримання використовують допустимі параметри. Робота ІТ-фахівця за важкістю відноситься до Іа (роботи, що виконуються сидячи і не потребують фізичного напруження) та Іб (роботи, що виконуються сидячи, стоячи або пов'язані з ходінням та супроводжуються деяким фізичним напруженням)

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84



Створення сприятливих умов праці і правильне естетичне оформлення робочих місць на виробництві має велике значення як для полегшення праці, так і для підвищення його привабливості, позитивно впливає на продуктивність праці. Забарвлення приміщень і меблів повинні сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою. У службових приміщеннях, у яких виконується одноманітна розумова робота, що вимагає значної нервової напруги і великого зосередження, забарвлення повинно бути спокійних тонів – малонасичені відтінки холодного зеленого або блакитного кольорів.

При розробці оптимальних умов праці програміста необхідно враховувати освітленість. Раціональне освітлення робочого місця є одним з найважливіших факторів, що впливають на ефективність трудової діяльності людини, що попереджають травматизм і професійні захворювання. Правильно організоване освітлення створює сприятливі умови праці, підвищує працездатність і продуктивність праці. Освітлення на робочому місці програміста повинно бути таким, щоб працівник міг без напруги зору виконувати свою роботу. Стомлюваність органів зору залежить від ряду причин: недостатність освітленості; надмірна освітленість; неправильний напрям світла. Недостатність освітлення приводить до напруги зору, ослабляє увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає засліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань. [21]

### **8.3 Пропозиції щодо підвищення працездатності ІТ-фахівця**

Практичне значення заходів щодо підвищення працездатності впливає із закономірностей її динаміки і зводиться ось до чого:

- збільшення фази стійкого стану у фонді робочого часу;
- прискорення процесу працювання;

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		86

- віддалення фази розвитку втоми;
- забезпечення високої продуктивності праці за нормальних фізіологічних затрат.

Комплекс заходів щодо підвищення і збереження працездатності працівників на оптимальному рівні реалізується на техніко-організаційному, соціально-економічному, санітарно-гігієнічному, медико-біологічному, психологічному напрямках.

Вагомим фактором високої працездатності і продуктивності праці є оптимізація трудових навантажень на основі механізації і автоматизації виробничих процесів, удосконалення технології, скорочення і ліквідації важкої ручної праці. Доведено, що при правильній організації праці на легких роботах спостерігається найбільша тривалість фази стійкого стану, а на важких роботах вона нетривала.

Високий рівень працездатності безпосередньо залежить від умов праці, оскільки поліпшення їх супроводжується зменшенням енергетичних затрат організму на подолання несприятливого впливу факторів виробничого середовища.

Важливим напрямком підвищення працездатності працюючих є ритмізація трудових процесів, оптимізація темпу роботи, а також раціоналізація трудових рухів на фізіологічній основі, що сприяє формуванню і закріпленню робочих динамічних стереотипів, а отже зменшенню м'язових і вольових зусиль. Ритмічна робота підвищує функціональні можливості організму, сприяє його тренуваності і забезпечує економізацію енергетичних затрат. [1]

Багатьом програмістам постійно доводиться працювати з великою кількістю програм одночасно. Часте перемикання туди-сюди між IDE та довідкою суттєво зменшує продуктивність фахівця. Однак вирішення цієї проблеми досить просте та очевидне: встановлення більшої кількості моніторів.

Оптимальним варіантом є два монітори. Все ж таки це найпростіший з апаратної точки зору варіант. Крім того, якби їх було більше, то ними було б

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		87

важче керувати, та й столі просто не вистачить місця на ще один монітор. Але тут ще залежить розміру моніторів. Є системи із 4 або 6 відносно невеликими екранами, які кріпляться на кронштейні. Але оптимальним є два 27-дюймові монітори, на яких все добре видно, особливо коли працювати доводиться в основному з текстом [3].

#### 8.4 Пожежна безпека

Вимоги до пожежної безпеки на підприємстві неухильно повинен дотримуватися кожен співробітник, а організаційна складова при цьому покладається на посадових осіб за відповідним рішенням керівництва і прописується в посадових інструкціях і положеннях по структурним підрозділам.

Зокрема, вказуються конкретні території, ділянки, зони, об'єкти, цілі будівлі і їх частини, поверхи, на яких відповідального співробітника повинне проводити такі організаційні роботи.

Відповідальні особи зобов'язуються розробити, впровадити та підтримувати в певному інструкцією і положенням на ввірених їм об'єктах протипожежний режим і інструкції відповідно до вимог, викладених в нормативних актах.

Передбачено також створення підрозділу добровільної пожежної охорони та пожежно-рятувальної команди в його складі.

Встановлений режим включає порядки з описом місць спеціального призначення та правила їх користування та утримання, наприклад:

- евакуаційних шляхів;
- так званих «курилок»;
- місць складування продукції та сировини;
- стоянки транспорту.

Також встановлюється порядок роботи та технічного обслуговування:

- вентиляційного устаткування;

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		88

- засобів пожежогасіння і захисту від загорянь;
- нагрівальних приладів;
- електрообладнання.

Розробляються і впроваджуються правила роботи з відкритим вогнем і горючими матеріалами. Створюються графіки проходження інструктажів з пожежної безпеки співробітників, а також порядок і терміни перевірок знань пожежно-технічного мінімуму, в тому числі, тих працівників, які відповідальні за цю ділянку роботи на підприємстві. При цьому можуть передбачатися внутрішні лекції, семінари, тренінги та практичні заняття на підприємстві, а також зовнішні – на базі спеціалізованих навчальних центрів з професійними викладачами.

Важливою складовою протипожежного режиму на будь-якому об'єкті є розробка і впровадження порядку дій при виникненні пожежі. Неодмінно має бути план евакуації, описано, як повинні відключатися електроустановки, що і в якій послідовності необхідно робити співробітникам.

Відповідно, для кожного об'єкта, кожного приміщення (крім коридорів, санвузлів, басейнів і подібних приміщень), окремих видів робіт складаються інструкції, за якими повинен працювати персонал, залучений на певних ділянках і в виконанні окремих видів робіт. За інструкціями проводиться навчання (інструктаж) персоналу з подальшим контролем знань.

Детально про те, як розробити протипожежний режим, прописати порядки та інструкції, пояснюють на тематичних курсах і семінарах. [24]

### 8.5 Розрахункова частина

Для захисного штучного заземлення застосовуються вертикальні електроди: металевий куток 63-63-6 мм., (згідно з ДСТУ 2251-93 «Кутики сталеві гарячекатані рівнополічні. Сортамент») довжиною  $L=2$  м., та горизонтальний електрод – металева полоса з перетином 60-5 мм. Напряга –

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		89





## Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз приміщення, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи.

Тільки повна усвідомленість працівника про можливі небезпеки, що можуть підстерігати його на робочому місці та дотримання вимог нормативних актів з питань охорони праці та відповідних рекомендацій фахівців, дозволять значною мірою знизити негативний вплив шкідливих та небезпечних факторів при роботі з комп'ютером на організм людини.

Виконано розрахунок захисного штучного заземлення, як одного з ключових факторів безпеки програміста.

					VKPM-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		92

## 9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи GPS навігатора на основі чипу серії SiRF Atlas V.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів GPS навігатора на основі чипу серії SiRF Atlas V.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем GPS навігатора на основі чипу серії SiRF Atlas V.

– Досліджена система GPS навігатора на основі чипу серії SiRF Atlas V.

– На основі отриманих результатів досліджень створена програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання GPS навігатора на основі чипу серії SiRF Atlas V.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		93

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Embarcadero Delphi. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Khufu.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Розроблена програма має реальний економічний ефект від її впровадження у виробництво у сумі 9167 грн. З урахуванням вартості розробки програми та обладнання, строк окуплення становить 0,2 роки.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		94

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Яценко Д.Р. Дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V // Збірник праць молодих науковців ЦНТУ. – Вип. 14. – Кропивницький: ЦНТУ, 2023.
2. Adam Freeman. Pro Go The Complete Guide to Programming Reliable and Efficient Software Using Golang. Apress Media. 2022. 1078 p.
3. Fernando Doglio. Skills of a Successful Software Engineer. Manning. 2022. 182 с.
4. M. Holmes He. Creating Apps with React Native. Apress Media. 2022. 445 p.
5. Maurício Aniche. Effective Software Testing. Manning Publications. 2021. 372 p
6. Priscila Heller. Automating Workflows with GitHub Actions. Packt Publishing. 2021. 216 p.
7. JJ Geewax. API Design Patterns. Manning Publications Co. 2021. 481 p.
8. Prateek Prasad. App Design Apprentice. Razeware LLC. 2020. 272 p.
9. Dawn Griffiths, David Griffiths. Head First Android Development. O'Reilly Media, Inc. 2021. 1414 p.
10. Nathan Metzler. Kotlin Programming for Beginners. Independently published. 2021. 158 p.
11. Aaron Torres. Go Programming Cookbook Second Edition. Packt Publishing Ltd. 2019. 427 p.
12. Мелешко Є.В., Якименко М.С., Поліщук Л.І. Алгоритми та структури даних: Навчальний посібник для студентів технічних спеціальностей денної та заочної форми навчання. – Кропивницький: Видавець – Лисенко В.Ф., 2019. – 156 с.

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		95

13. Knuth D. The Art of Computer Programming, Vol. 1: Fundamental Algorithms, 3rd Edition 3rd Edition. – Addison-Wesley Professional, 2019. – 672 p.
14. Knuth D. The Art of Computer Programming: Vol. 3: Sorting and Searching 2nd Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 800 p.
15. Knuth D. Art of Computer Programming, Vol. 2: Seminumerical Algorithms 3rd Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 672 p.
16. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. Introduction to Algorithms, 3rd Edition (The MIT Press) 3rd Edition – The MIT Press, 2019. – 1292 p.
17. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». Lecture Notes on Data Engineering and Communications Technologies, 2023, 178, pp. 208–223.
18. Smirnov, O., Karapetyan, A., Fedorov, E., «Creating Neural Network and Single Solution Human-Based Metaheuristic Methods of Solving the Traveling Salesman Problem». CEUR Workshop Proceedings, Volume 3312, 2022, pp. 47-58.
19. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». SN Computer Science, Vol 2, 337, 2021. <https://doi.org/10.1007/s42979-021-00739-w>.
20. Smirnov O., Kovalenko O., Kovalenko A., Kavun S. «Quantitative Risk Assessment Method Development in the Context of the SDLC-model». 2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 2021, pp. 203-208, doi: 10.1109/PICST54195.2021.9772143
21. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». Communications in Computer and Information Science, 2021, vol 1486. Springer, Cham. pp 169-184.
22. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-

feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

23. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

24. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

25. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V. «Detecting Hidden Information in FAT». International Journal of Computer Network and Information Security (IJCNIS). Vol. 12, No. 3, 2020. PP.33-43.

26. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

27. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

28. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

29. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of

Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

30. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

31. Kuznetsova, T., «Code-Based Schemes for Post-Quantum Digital Signatures», 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019; Metz; France; 18-21 September 2019. P. 707-712.

32. Smirnov, O., Kuznetsov, A., Stefanovych, O., Gorbenko, Y., Krasnobayev, V., Kuznetsova K. «Information Hiding Using 3D-Printing Technology», 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019; Metz; France; 18-21 September 2019. P.701-706.

33. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

34. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

35. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». Системи управління, навігації та зв'язку, 2023, вип. 2(72), С. 170-178.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		98

36. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». Сучасні інформаційні системи, 2023, том 7, № 2, С. 49-56.

37. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А. «Дослідження нормативної документації та стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». VI міжнародна науково-практична конференція «Інформаційна безпека та комп'ютерні технології», м. Кропивницький. 20-21 квітня 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 35-36.

38. Смірнов, О.А., Усік П.С., Полігенько О.О., Одарченко Р.С., Терещенко Л.Ю. «Інформаційна технологія та програмне забезпечення для підвищення ефективності планування підсистеми базових станцій стільникового зв'язку». Проблеми телекомунікацій. № 1(26). С. 83-96. 2020.

39. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» Вісник Черкаського державного технологічного університету. Технічні науки. №4. С. 103-110. 2020.

40. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.

41. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральнуукраїнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.

42. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan

D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

43. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

44. Смірнов О.А., Котелянець В.В. Стійкі до колізій стохастичні моделі функціонування безпроводових сенсорних мереж. Вісник інженерної академії України, №3, с. 145-152, 2018

45. O. Smirnov, O. Kovalenko, A. Kovalenko, S. Smirnov, V. Vialkova. The mathematical model of the testing technology for DOM XSS vulnerabilities. Scientific & practical cyber security journal (SPCSJ) Vol 2 Issue 1, 22-28 pp. [Электронный Журнал]. Georgia. Tbilisi: SCSA – 2018.

46. Oleksii Smirnov, Oleksandr Kovalenko, Jamil Al-Azzeh, Anna Kovalenko, Serhii Smirnov. Qualitative risk analysis of software development. Asian Journal of Information Technology. – Volume 17(3). – Medwell Journals. – 2018. – P. 218-230.

47. Смірнов О.А., Коваленко О.В., Коваленко А.С., Смірнов С.А. Розробка методу передтестової компіляції й розподілу доступу. Збірник наукових праць III міжнародної науково-практичної конференції “Інформаційна безпека та комп’ютерні технології”, м. Кропивницький. 19-20 квітня 2018р. – Кропивницький: ЦНТУ. – 2018. – С. 214-215

48. Smirnov Oleksii, Kovalenko Oleksandr, Kovalenko Anna, Smirnov Serhii. Method of testing the DOM XSS vulnerability. International Conference «Information technologies, systems and networks ITSН-2017». Chisinau, Republic of Moldova. 17 – 18 October 2017. – Chisinau: Academy of Sciences of Moldova, Military Academy of Armed Forces “Alexandru cel Bun”. 2017. P7.

					<b>ВКРМ-122.23.0026.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		<b>100</b>

49. Смірнов О.А., Смірнов С.А., Коваленко О.В., Коваленко А.С. Технологія тестування DOM XSS уразливості. Науково-практичний журнал кібер безпеки (SPCSJ) № 1. [Електронний журнал]. Грузія. Тбілісі: SCSA – 2017.

50. Смірнов О.А., Лисенко І.А. Інформаційна технологія проектування тестових наборів з урахуванням вимог до програмного забезпечення. Системи управління, навігації та зв'язку. – Випуск 4 (44). – Полтава: ПолтНТУ. – 2017. – С. 112-115.

КБГПЗ-2023

					ВКРМ-122.23.0026.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		101

Додаток А  
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					<b>ВКРМ-122.23.0026.00.00.ТЗ</b>		
Вим.	Арк.	№ документа	Підпис	Дата			
Розробив	Яценко Д.Р.				Літ.	Аркуш	Аркушів
Перевірів	Улічев О.С.						
Н. Контр.	Коваленко А.С.				ЦНТУ КН-22М-1		
Затв.	Смірнов О.А.						

## 1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи GPS навігатора на основі чипу серії SiRF Atlas V.

## 2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 32-13 від 04.08.2023 року).

## 3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація системи GPS навігатора на основі чипу серії SiRF Atlas V.

## 4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

## 5 Технічні вимоги

### 5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;

					ВКРМ-122.23.0026.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

## 5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи GPS навігатора на основі чипу серії SiRF Atlas V;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

## 5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

## 5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

## 5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					<b>ВКРМ-122.23.0026.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

## 5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

## 5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ архітектури IBM PC, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

## 5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

### 5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

### 5.8.2 Мова програмування

Середовище Embarcadero Delphi.

					ВКРМ-122.23.0026.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

### 5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

### 5.8.4 Вихідні дані

Робоча програма.

## 6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

## 7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати розрахунок витрат показників економічного ефекту з урахуванням цін на 3 вересня 2023 року.

## 8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинен бути розглянутий аналіз умов праці на робочому місці ІТ-фахівця.

					ВКРМ-122.23.0026.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

## 9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 101 аркуш.

## 10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Розрахунок з техніко-економічного обґрунтування.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

## 11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 10.12.2023 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 11.12.2023 р.

					<b>ВКРМ-122.23.0026.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б  
(обов'язковий)

**Міністерство освіти і науки України**  
**Центральноукраїнський національний технічний університет**

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за  
другим (магістерським) рівнем вищої освіти

\_\_\_\_\_ Улічев О.С.

*Дослідження та програмна реалізація  
системи GPS навігатора на основі чипу серії SiRF Atlas V*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 42

Літера: РП

Кропивницький – 2023 року

**Налагодження параметрів GPS\_SIRF\_ATLAS\_V.pas**

```

unit GPS_SIRF_ATLAS_V;

  {Федоров Богдан Сергійович, КН-22М-1}

interface
uses
SysUtils, WinTypes, WinProcs, Messages, Classes, Graphics, Controls,
Forms, Dialogs, ExtCtrls, buttons, grids, stdctrls, clipbrd, Printers;

type
TMouseEvent = procedure (Sender: TObject; Shift: TShiftState; Points: trect)
of object;
TGPS_SIRF_ATLAS_VDataArray=array[0..16]of double;

TGPS_SIRF_ATLAS_V = class(TPanel)
private
MAXELNUM:integer;
sel:TSELARRAY;
npolypoints:word;
MapLimits:trect;
Titulo: string;
Nbloqs:word;
viewpct,viewext,viewbmp:trect;
FLegendChartPos,FLegendBulletsPos,FLegendShadowsPos:trect;
FZoomBox:trect;
arrfname:string;
bitmap:tbitmap;
backbitmap:tbitmap;
logobitmap:tbitmap;
tometa:boolean;
bmpfname,polfname:string;
Fready:boolean;
BB_anchorx,BB_anchory,BB_curx,BB_cury:integer;
BB_drawing:boolean;
FPaintMode:integer;
BackLink:array[1..5] of string;
nBacks:integer;
fgrid:integer;
paleta:hpalette;
FOnMouseDown:TMouseEvent;
FOnMouseUp:TMouseEvent;
FOnMouseMove:TMouseMoveEvent;
FOnMouseLine:TMouseEvent;
FOnMouseBox:TMouseEvent;
procedure time(sender:tobject);
procedure replaceglis;
Procedure CalcChartRatios;
Procedure SetPBgColor(k:integer;c:tcolor);
function GetPBgColor(k:integer):tcolor;
Procedure SetPFgColor(k:integer;c:tcolor);
function GetPFgColor(k:integer):tcolor;
Procedure SetPPattern(k:integer;c:byte);
function GetPPattern(k:integer):byte;
Procedure SetChartLegend(k:word;s:string);
function GetChartLegend(k:word):string;
Procedure SetBitMap(fname:string);
Procedure DrawBitMap(h:hdc);
Procedure CleanHDC(h:hdc;col:tcolor;box:trect);
Procedure Meridians(h:hdc;box:trect);
Procedure SetHDCFrame(h:hdc;r:trect);
Procedure SetChartValue(element,item:word;value:real);
Function GetChartValue(element,item:word):real;
Procedure WMF_clip2file(wnd:HWND;fname:string);
procedure ClpWMF2Text(fname:string;textfname:string);

```

```

Function ClpWMFRect (fname:string):trect;
Procedure ScaleClpWMF2Text (fname,textfname:string;iso:boolean);
Procedure RectToPolyLine (r2:trect;var cuadro:array of tpoint);
procedure adaptlegrect (r,r1:trect;var r2:trect);
Procedure SetFont (i:integer;f:tfont);
Procedure MyMouseMove (Sender: TObject; Shift: TShiftState; X, Y: Integer);
Procedure WMLButtonDown (var msg:TWMLButtonDown); message WM_LBUTTONDOWN;
Procedure WMRButtonDown (var msg:TWMRButtonDown); message WM_RBUTTONDOWN;
Procedure WMLButtonUp (var msg:TWMLButtonUp); message WM_LBUTTONUP;
Procedure WMRButtonUp (var msg:TWMRButtonUp); message WM_RBUTTONUP;
Procedure WMMouseMove (var msg:TWMMouseMove); message WM_MOUSEMOVE;
Procedure Loaded;Override;

public
drw:tdraws;
np:TSPOLARRAY;
Arr:TSArrange;
InfoPanel,
CLegPanel,
BLegPanel,
SLegPanel,
ChartPanel:TGMovPan;
UserLine:record
    p:array[1..300] of tpoint;
    np:integer;
    drawing:boolean;
    dist:double;
end;
MyVersion:string;
gli:TGlyphList;
timer:ttimer;
Procedure LoadGlyphsBitmap (fname:string);
procedure addgli (p:tpoint);
procedure placegli (ix:integer;p:tpoint);
function GlyphCount:integer;
Procedure
GlyphSettings (ixglyph:integer;imageix:integer;animate:boolean;enable:boolean);
constructor create (sender:tcomponent);override;
destructor destroy;override;
Procedure paint;override;
Procedure CreatePanels;
Procedure FromFile (fname:string);
Procedure ToFile (fname:string);
Procedure GetPoint (elem,ix:integer;var c:tcitypoint);
Function RemovePoint (elem,ix:integer):boolean;
Function GetPointCount (elem:integer):integer;
Procedure ClearPoints (elem:integer);
Procedure SetSymbol (f:tfont;c:char);
Procedure ForwardLink (ix:integer);
Procedure BackwardLink;
Procedure LoadPaletteFromFile (filename:string);
Procedure LoadMosaicBitmap (fname:string);
Procedure SetMapOptions (var op:tmapop);
Procedure SetChartOptions (var op:tchop);
Procedure SetBulletOptions (var op:BullOptions);
Procedure SetShadowOptions (var op:tshop);
Procedure SetThemeShadowOptions (var op:TSHThem);
Procedure SetBackgroundOptions (var op:tbkop);
Procedure SetLayerOptions (var op:TLAYOP);
Procedure GetMapOptions (var op:tmapop);
Procedure GetChartOptions (var op:tchop);
Procedure GetBulletOptions (var op:BullOptions);
Procedure GetShadowOptions (var op:tshop);
Procedure GetThemeShadowOptions (var op:TSHThem);
Procedure GetBackgroundOptions (var op:tbkop);
Procedure GetLayerOptions (var op:TLAYOP);
Procedure Merge (fname:string;layeroffset:integer; x,y:integer);
Procedure Flip (horz,vert:boolean);
Procedure ScaleTo (r:trect);

```

```

Procedure AddPoly(closed:boolean);
Procedure DeleteElements;
Procedure JoinElements(orig,dest:integer);
Procedure ToFront(elem:integer);
Procedure ToBack(elem:integer);
Procedure ReMap(var m:array of integer);
Property canvas;
Property PaintMode:integer read FPaintMode write FPaintMode;
Property ElementCount:word read npolypoints;
Property ElementName[i:word]:string read getElmName write SetElmName;
Property ZoomBox:TREct read FZoomBox write FZoomBox;
Property Ready:boolean read Fready write fready;
Property NumBox:boolean read GetNumBox write SetNumBox;
Property Selected[i:word]:boolean read GetElSelected write SetElSelected;
Property Pol_Layer[i:word;j:word]:byte read GetPolLayer write SetPolLayer;
Property PolCount[i:word]:byte read GetPolCount;
Property PolNames[i:word;j:word]:string read GetPolName write SetPolName;
Property Links[i:word]:string read GetElmLink write SetElmLink;
Property Fonts[i:integer]:Tfont read GetFont write SetFont;
property PBgColor[i:integer]:tcolor read GetPBgColor write SetPBgColor;
property PFgColor[i:integer]:tcolor read GetPFgColor write SetPFgColor;

implementation

constructor TGPS_SIRF_ATLAS_V.create(sender:tcomponent);
begin
inherited create(sender);
ready:=false;
fgrid:=8;
MaxElNum:=TGPS_SIRF_ATLAS_V_MAXELEMENTS;
MyVersion:=TGPS_SIRF_ATLAS_V_Version;
BackLink[1]:='';
nbacks:=0;

FPaintMode:=0;
UserLine.drawing:=false;

npolypoints:=0;
Arr:=nil;
bitmap:=nil;
backbitmap:=tbitmap.create;
logobitmap:=tbitmap.create;

drw:=tdraws.create;
tometa:=false;
Align:=AlClient;

paleta:=GetDefaultPalette;

FLegendChartPos:=rect(1,70,30,99);
FLegendBulletsPos:=rect(31,70,60,99);
FLegendShadowsPos:=rect(61,70,98,99);
bmpfname:='';
arrfname:='';
polfname:='';
zoombox:=rect(1,1,100,100);
viewpct:=rect(1,1,99,99);

end;

Procedure TGPS_SIRF_ATLAS_V.Loaded;
begin
Inherited Loaded;
CreatePanels;

end;

Procedure TGPS_SIRF_ATLAS_V.CreatePanels;

```

```

procedure createmasks (bm:tbitmap);
var t:tcolor;
    x,y,h:integer;
begin
    h:=bm.height;
    bm.height:=h*2;
    t:=bm.canvas.pixels[0,0];
    for x:=0 to bm.width-1 do for y:=0 to h-1 do
        if bm.canvas.pixels[x,y]=t then bm.canvas.pixels[x,y+h]:=$00FFFFFF
        else bm.canvas.pixels[x,y+h]:=$00000000
    end;
end;

begin
if not (ComponentState=[]) then exit;
gli:=TGlyphList.create(canvas);

timer:=ttimer.create(self);
timer.interval:=500;
timer.ontimer:=time;
timer.enabled:=false;

InfoPanel:=TGMovPan.create(self);
infopanel.Left:= 32;    infopanel.width:= 64;
infopanel.top:=8;    infopanel.height:= 20;
infopanel.Align:= alnone;    infopanel.marge:=3;
infopanel.borderwidth:=0;    infopanel.bevelwidth:=1;
infopanel.BevelOuter:= bvLowered;    infopanel.BevelInner:= bvNone;
infopanel.Caption:= 'GPS_SIRF_ATLAS_V 2.0';    infopanel.visible:=true;
InfoPanel.parent:=self;
if bitmap<>nil then SetBitmap(bmpfname);
if polfname<>' then Fromfile(polfname);
end;

destructor TGPS_SIRF_ATLAS_V.destroy;
var i:integer;
begin
gli.Free;
for i:=1 to npolypoints do np[i].free;
npolypoints:=0;
arr.free;
bitmap.free;
backbitmap.free;
logobitmap.free;
drw.free;
DeleteObject(paleta);
timer.free;

InfoPanel.free;
CLegPanel.free;
BLegPanel.free;
SLegPanel.free;
ChartPanel.free;
inherited destroy;
end;

Procedure TGPS_SIRF_ATLAS_V.FromFile(fname:string);
var f:file;
i,j:integer;
ax:shortstring;
bloque:record
codigo:word;
nombre:string[50];
numpolis:word;
nummarcos:word;
end;
elemento:record
fillcolor,bordercolor:longint;

```

```

numpuntos:word;
layer:word;
other:longint;
nombre:str50;
end;
buffer: ^SPltSmallPoint;

tfn:tfontname;
rt:trect;
nupu,ipu:word;
c:tsapoint;

begin
if not fileexists(fname) then exit;
polfname:=fname;
if not (ComponentState=[]) then exit;

new(buffer);
assignfile(f,fname);
reset(f,1);

blockread(f,ax,100);
if ax<>MyVersion then begin
closefile(f);
dispose(buffer);
exit;
end;
screen.cursor:=crHourGlass;
for i:=1 to npolypoints do np[i].free;
npolypoints:=0;
ready:=false;
c:=tsapoint.create;
try
if nbloqs> MaxElNum then nbloqs:=maxelnum;
npolypoints:=nbloqs;
for i:=1 to npolypoints do begin
blockread(f,bloque,sizeof(bloque));
np[i]:=TSPolyPol.create(bloque.nummarcos,bloque.numpolis,bloque.nombre);
for j:=1 to bloque.nummarcos do begin
blockread(f,elemento,sizeof(elemento));
blockread(f,buffer^,sizeof(tSmallPoint)*elemento.numpuntos);
np[i].SetMValsSmall(elemento.fillcolor,elemento.bordercolor,j,elemento.numpuntos
,buffer^);
np[i].m[j].drawer:=drw;
np[i].m[j].layer:=elemento.layer;
np[i].m[j].nombre:=elemento.nombre;
end;

for j:=1 to bloque.numpolis do begin
blockread(f,elemento,sizeof(elemento));
blockread(f,buffer^,sizeof(tSmallPoint)*elemento.numpuntos);
np[i].SetPValsSmall(elemento.fillcolor,elemento.bordercolor,j,elemento.numpuntos
,buffer^);
np[i].p[j].drawer:=drw;
np[i].p[j].layer:=elemento.layer;
np[i].p[j].nombre:=elemento.nombre;
end;
blockread(f,nupu,sizeof(nupu));
np[i].npoints:=nupu;
for ipu:=1 to nupu do begin
blockread(f,c.c,sizeof(c.c));
np[i].points[ipu]:=c;
end;
blockread(f,np[i].eo,sizeof(np[i].eo));
blockread(f,np[i].link,sizeof(np[i].link));
blockread(f,np[i].values,sizeof(np[i].values));
np[i].CalcCaja;
np[i].drawer:=drw;
end;
end;

```

```

finally
closefile(f);
dispose(buffer);
c.free;
drw.setfonts;
arr.free;
arr:=TSArrange.create(2,npolypoints,'Normal');
if bmpfname<>' ' then setbitmap(bmpfname);
if arrfname<>' ' then ArrFromFile(arrfname);
CalcCaja;
ready:=true;
zoombox:=MapLimits;
infopanel.caption:=title;
adaptlegrect(rect(left,top,left+width,top+height),FLegendChartPos,rt);
CLegPanel.Left:= rt.left;      CLegPanel.width:= rt.right-rt.left;
CLegPanel.top:=rt.top;      CLegPanel.height:= rt.bottom-rt.top;
adaptlegrect(rect(left,top,left+width,top+height),FLegendBulletsPos,rt);
BLegPanel.Left:= rt.left;      BLegPanel.width:= rt.right-rt.left;
BLegPanel.top:=rt.top;      BLegPanel.height:= rt.bottom-rt.top;
adaptlegrect(rect(left,top,left+width,top+height),FLegendShadowsPos,rt);
SLegPanel.Left:= rt.left;      SLegPanel.width:= rt.right-rt.left;
SLegPanel.top:=rt.top;      SLegPanel.height:= rt.bottom-rt.top;
infopanel.Left:= left+4;
infopanel.top:=top+4;
repaint;
screen.cursor:=crdefault;
end;
end;
Procedure  TGPS_SIRF_ATLAS_V.ToFile(fname:string);
var f:file;
i,j:integer;
ax:string;
bloque:record
codigo:word;
nombre:string[50];
numpolis:word;
nummarcos:word;
end;
elemento:record
fillcolor,bordercolor:longint;
numpuntos:word;
layer:word;
other:longint;
nombre:str50;
end;
buffer:^SPltSmallPoint;
nupu,ipu:word;
c:tsapoint;
begin
new(buffer);
c:=tsapoint.create;
assignfile(f,fname);
rewrite(f,1);
fillchar(ax,100,20);
ax:=MyVersion;
i:=0;
nbloqs:=npolypoints;
blockwrite(f,nbloqs,sizeof(nbloqs));
for i:=1 to npolypoints do begin
bloque.nummarcos:=np[i].nmarcos;
bloque.numpolis:=np[i].npolis;
bloque.nombre:=np[i].nombre;
bloque.codigo:=i;
blockwrite(f,bloque,sizeof(bloque));
for j:=1 to bloque.nummarcos do begin
elemento.fillcolor:=np[i].m[j].bkcol;
elemento.bordercolor:=np[i].m[j].frcol;
elemento.numpuntos:=np[i].m[j].n;
elemento.layer:=np[i].m[j].layer;

```

```

elemento.other:=0;
elemento.nombre:=np[i].m[j].nombre;
blockwrite(f,elemento,sizeof(elemento));
Copy32To16(np[i].m[j].pt^,buffer^,elemento.numpuntos);
blockwrite(f,buffer^,sizeof(TSmallPoint)*elemento.numpuntos);
end;
for j:=1 to bloque.numpolis do begin
elemento.fillcolor:=np[i].p[j].bkcol;
elemento.bordercolor:=np[i].p[j].frcol;
elemento.numpuntos:=np[i].p[j].n;
elemento.layer:=np[i].p[j].layer;
elemento.other:=0;
elemento.nombre:=np[i].p[j].nombre;
blockwrite(f,elemento,sizeof(elemento));
Copy32To16(np[i].p[j].pt^,buffer^,elemento.numpuntos);
move( np[i].p[j].pt^,buffer^,elemento.numpuntos*sizeof(tpoint));
blockwrite(f,buffer^,sizeof(tsmallpoint)*elemento.numpuntos);
end;
nupu:=np[i].npoints;
blockwrite(f,nupu,sizeof(nupu));
for ipu:=1 to nupu do begin
c:=np[i].points[ipu];
blockwrite(f,c.c,sizeof(c.c));
end;
blockwrite(f,np[i].eo,sizeof(np[i].eo));
blockwrite(f,np[i].link,sizeof(np[i].link));
blockwrite(f,np[i].values,sizeof(np[i].values));
end;

closefile(f);
dispose(buffer);
c.free;
end;

Procedure TGPS_SIRF_ATLAS_V.ReadData(fname:string);
var f:file of TGPS_SIRF_ATLAS_VDataArray;
    TGPS_SIRF_ATLAS_VDataArray=array[0..16]of double;
ad:TGPS_SIRF_ATLAS_VDataArray;
i,j:integer;
begin
assignfile(f,fname);
reset(f);
i:=1;
try
while ((not eof(f)) and (i<elementcount)) do begin
read(f,ad);
for j:=0 to 16 do np[i].values[j]:=ad[j];
inc(i);
end;
finally
closefile(f);
end;
end;

Procedure TGPS_SIRF_ATLAS_V.SaveData(fname:string);
var f:file of TGPS_SIRF_ATLAS_VDataArray;
    TGPS_SIRF_ATLAS_VDataArray=array[0..16]of double;
ad:TGPS_SIRF_ATLAS_VDataArray;
i,j:integer;
begin
assignfile(f,fname);
rewrite(f);
try
for i:=1 to elementcount do begin
for j:=0 to 16 do ad[j]:=np[i].values[j];
write(f,ad);
end;
finally
closefile(f);
end;
end;

```

```

end;
end;

Function TGPS_SIRF_ATLAS_V.GetFont(i:integer):tfont;
begin
case i of
1:GetFont:=drw.font1;
3:GetFont:=drw.font3;
4:GetFont:=drw.font4;
5:GetFont:=drw.font5;
else GetFont:=drw.font2;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.ArrCalcBoxes;
var i,j:integer;
begin
for i:=1 to arr.n do begin
prepect(arr.caja[i]);
for j:=1 to arr.t do if arr.matr.bit[i,j] then begin
comprect(np[j].caja,arr.caja[i]);
end;end;end;

Procedure TGPS_SIRF_ATLAS_V.ArrFromFile(fname:string);
begin
arrfname:=fname;
if not fileexists(fname) then exit;
arr.fromfile(fname);
arrcalcboxes;
end;

Function TGPS_SIRF_ATLAS_V.getElmName(i:word):string;
begin
if ((i>0) and (i<=npolypoints)) then GetElmName:=np[i].nombre
else GetElmName:='';
end;

Procedure TGPS_SIRF_ATLAS_V.SetElmName(i:word;s:string);
begin
if ((i>0) and (i<=npolypoints)) then np[i].nombre:=s;
end;

Procedure TGPS_SIRF_ATLAS_V.SetSel(lb:tlistbox);
var i:integer;
begin
for i:=1 to lb.items.count do sel[i]:=(lb.selected[i-1]);
end;

Function TGPS_SIRF_ATLAS_V.IsArrSelected(father:word):boolean;
var i:integer;
begin
IsArrSelected:=true;
for i:=1 to npolypoints do if sel[i] then
if arr.Esta(father,i) then exit;
IsArrSelected:=false;
end;

Procedure DoChart(h:hdc;r:trect);
var kx,ky,i,j:integer;
ox,oy:integer;
v:integer;
x0,y0:integer;
bola:integer;
ax:string;
mx:integer;
maximovalor,total:real;
caxa:trect;
pu:tpoint;
suma:array[0..16] of integer;

```

```

ds:array[1..16] of double;
dd:double;
a:array[0..255] of char;
centro:tpoint;
begin
ChartPanel.font:=font;
chartpanel.color:=drw.co.boxbackcolor;
centro.x:=(r.left+r.right) div 2;
centro.y:=(r.top+r.bottom) div 2;
selectobject(h,getstockobject(BLACK_PEN));
for i:=1 to 16 do ds[i]:=0;
maximovalor:=0;
total:=0;
for i:=drw.co.nvalues downto 1 do
for j:=1 to elementcount do if sel[j] then begin
ds[i]:=ds[i]+np[j].values[i];
if ds[i]>maximovalor then maximovalor:=ds[i];
end;
if maximovalor=0 then maximovalor:=1;
for i:=1 to drw.co.nvalues do total:=total+ds[i];

SetBkMode(h,1);
setTextAlign(h,TA_CENTER+TA_TOP);
str(total:3:drw.co.ndecimals,ax);
ax:='Total: '+ax;
strcpy(a,ax);
textout(h,centro.x,r.top+2,a,length(ax));

caxa:=r;

ox:=(caxa.right-caxa.left) div 5;
oy:=(caxa.bottom-caxa.top) div 5;
kx:=(caxa.right-caxa.left-2*ox) div drw.co.nvalues;
ky:=3*oy;
bola:=4;
mx:=kx div 2;
if ((ox<5) or (oy<5)) then exit;

if drw.co.style<>4 then begin
movetoex(h,caxa.left+ox,caxa.bottom-oy,nil);
lineto(h,caxa.left+ox+drw.co.nvalues*kx,caxa.bottom-oy);
MoveToEx(h,caxa.left+ox,caxa.bottom-oy,nil);
lineto(h,caxa.left+ox,caxa.bottom-oy - ky);
end;

case drw.co.style of
0:begin
for i:=1 to drw.co.nvalues do begin
drw.getbrocha(h,drw.co.patt[i],drw.co.linecolor,drw.co.colors[i]);
v:=trunc(ds[i]*ky/maximovalor);
rectangle(h,
caxa.left+ox+(i-1)*kx,
caxa.bottom-oy,
caxa.left+ox+i*kx,
caxa.bottom-oy - v);
drw.releasebrocha;
end;
end;
3:begin
for i:=1 to drw.co.nvalues do begin
v:=trunc(ds[i]*ky/maximovalor);
drw.bar3d(h,
caxa.left+ox+(i-1)*kx,
caxa.bottom-oy,
caxa.left+ox+i*kx,
caxa.bottom-oy - v,i);
end;
end;
end;
end;

```

```

1,2:begin
  drw.getpen(h,3,drw.co.colors[1]);
  x0:=caxa.left+ox+mx;y0:=caxa.bottom-oy -trunc(ds[1]*ky/maximovalor);
  MoveToEx(h,x0,y0,nil);
  for i:=1 to drw.co.nvalues do begin
    v:=trunc(ds[i]*ky/maximovalor);
    lineto(h,caxa.left+ox+i*kx-mx,caxa.bottom-oy - v);
  end;
  drw.releasepen;

  if drw.co.style=2 then begin
    drw.getbrocha(h,drw.co.patt[2],drw.co.LineColor,drw.co.Colors[2]);
    for i:=1 to drw.co.nvalues do begin
      v:=trunc(ds[i]*ky/maximovalor);
    end;
    drw.releasebrocha;
  end;
end;
4:begin
  pu:=centro;
  dd:=0; for i:=1 to drw.co.nvalues do begin
    dd:=dd+ds[i];
    ds[i]:=dd;
  end;
  suma[0]:=0; v:=0;
  for i:=1 to drw.co.nvalues do begin
    v:=v+trunc(ds[i]*360/dd);
    suma[i]:=trunc(ds[i]*360/dd);
  end;
  drw.Queso(h,pu.x,pu.y,drw.co.PieSize,0,360,true,
    drw.co.colors[1],drw.co.linecolor,drw.co.Colors[1]);
  for i:=1 to drw.co.nvalues do
    drw.Queso(h,pu.x,pu.y,drw.co.PieSize,suma[i-1],suma[i],false,
    drw.co.colors[1],drw.co.Colors[i],drw.co.linecolor);
  end;
end;
if drw.co.style<>4 then begin
  ox:=ox- kx div 2;
  for i:=1 to drw.co.nvalues do begin
    if drw.co.ndecimals=0 then ax:=inttostr(trunc(ds[i]))
    else str(ds[i]:3:drw.co.ndecimals,ax);
    v:=trunc(ds[i]*ky/maximovalor);
    drw.ContrastText(h,caxa.left+ox+i*kx ,
      caxa.bottom-oy - v,ax,3,false);
  end;
end;
end;

procedure SelObjects(c:tcolor;f:tfont);
begin
  oldf:=selectobject(h,f.handle);
  settextcolor(h,f.color);
  drw.getbrocha(h,0,f.color,c);
  rectangle(h,rto.left,rto.top,rto.right,rto.bottom);
  drw.releasebrocha;
end;

var mm,ww:double;
begin
  chartpanel.visible:=true;
  if not tometa then begin
    h:=ChartPanel.canvas.handle;
    rto:=rect(1,1,ChartPanel.width,ChartPanel.height);
    SelObjects(ChartPanel.color,drw.font4);
    Dochart(h,rto);
  end
  else begin

```

```

mm:=(r.right-r.left)/width;
ww:=(r.bottom-r.top)/height;
rto.left:=trunc(r.left+ChartPanel.left*mm);
rto.top:=trunc(r.top+ChartPanel.top*ww);
rto.right:=trunc(r.left+(ChartPanel.left+ChartPanel.width)*mm);
rto.bottom:=trunc(r.top+(ChartPanel.top+ChartPanel.height)*ww);
DoChart(h,rto);
end;
end;

Procedure ChartLegend(h:hdc;r:trect);
var i:integer;
wy,wb,wm:integer;
a:array[0..255] of char;
begin
wy:=(r.bottom-r.top) div (drw.co.nvalues+1);
wb:=trunc(0.60*wy);
wm:=trunc(0.20*wy);
if abs(font.size) >wy then font.size:=wy;
SetBkMode(h,1);
setTextAlign(h,TA_CENTER+TA_TOP);
strcpy(a,drw.co.title);
textout(h,(r.left+r.right) div 2,r.top+2,a,length(drw.co.title));
setTextAlign(h,TA_LEFT+TA_BOTTOM);
for i:=1 to drw.co.nvalues do begin
Bar(h,r.left+wb,r.top+wy*i+wm,
r.left+wb+wb,r.top+wy*i+wm+wb,i);
strcpy(a,drw.co.LegTexts[i]);
textout(h,r.left+3*wb,r.top+wy*i+wm+wb,a,length(drw.co.LegTexts[i]));
end;
end;

Procedure BulletLegend(h:hdc;rl:trect);
var i,j,k,k0,k2:integer;
wy,wm,wb:integer;
a:array[0..255] of char;
q:real;
ax:string;
pc:pchar;
oldf:hfont;
h1,h2,h3:integer;
f:tfont;
fh1:integer;
begin
case drw.bo.applyAs of
0,2:begin
k0:=trunc(drw.bo.size);
wb:=k0 div drw.bo.legendpoints;
q:=drw.maxval0 / drw.bo.legendpoints;
wy:=(rl.bottom-rl.top) div (drw.bo.legendpoints+1);
wm:=trunc(0.20*wy);
if abs(font.size) >wy then font.size:=wy;
SetBkMode(h,1);
setTextAlign(h,TA_CENTER+TA_TOP);
strcpy(a,drw.co.LegTexts[drw.bo.BulletSource]);
textout(h,(rl.left+rl.right) div
2,rl.top+2,a,length(drw.co.LegTexts[drw.bo.BulletSource]));
setTextAlign(h,TA_LEFT+TA_BOTTOM);
for i:=1 to drw.bo.legendpoints do begin
j:=drw.bo.legendpoints-i+1;
drw.fontsymbol.size:=k*2;
k2:=k ;
pc:=@drw.bo.thesymbol;
settextcolor(h,drw.bo.linecolor);
setbkmode(h,1);
oldf:=selectobject(h,drw.fontsymbol.handle);
fh1:=abs(drw.fontsymbol.height) div 2 ;
setTextAlign(h,TA_LEFT+TA_TOP);
textout(h,rl.left+wm,rl.top+wy*i+wm-fh1,pc,1);

```

```

        selectobject (h,oldf);
        str(q*j:3:drw.mo.decimals,ax);
        strcpy(a,ax);
        textout(h,rl.left+2*k0+wm,rl.top+wy*i+wm,a,length(ax));
        if drw.bo.applyAs=0 then begin
            movetoex(h,rl.left+wm+k0 ,rl.top + wy * i ,nil);
            lineto(h,rl.left+2*k0+wm,rl.top + wy * i);
        end;
    end;
end;
end;
1:begin
    wy:=(rl.bottom-rl.top) div (drw.bo.legendpoints+1);
    wb:=trunc(0.80*wy);
    wm:=trunc(0.10*wy);

    k:=100 div drw.bo.legendpoints;
    q:=drw.maxval0 / drw.bo.legendpoints;

    if abs(font.size) >wy then font.size:=wy;
    SetBkMode(h,1);
    setTextAlign(h,TA_CENTER+TA_TOP);
    ax:=drw.co.LegTexts[drw.bo.BulletSource];
    strcpy(a,ax);
    textout(h,(rl.left+rl.right) div 2,rl.top+2,a,length(ax));
    setTextAlign(h,TA_LEFT+TA_BOTTOM);

    for i:=1 to drw.bo.legendpoints do begin
        drw.getbrocha(h,0,drw.bo.shadows[k*i],drw.bo.shadows[k*i]);
        ellipse(h,rl.left+wb,rl.top+wy*i+wm,rl.left+wb+wb,rl.top+wy*i+wm+wb);
        drw.releasebrocha;
        str(q*i:3:drw.mo.decimals,ax); strcpy(a,ax);
        textout(h,rl.left+3*wb,rl.top+wy*i+wm+wb,a,length(ax));
    end;

end;
5:begin
    rectangle(h,rl.left,rl.top,rl.right,rl.bottom);
    h1:=rl.top+font.size*3;
    h2:=rl.top+rl.bottom -h1;
    h2:=h2 div drw.bo.custom.nranges;

    setTextAlign(h,TA_CENTER+TA_TOP);
    strcpy(a,drw.co.LegTexts[drw.bo.BulletSource]);
    textout(h,(rl.left+rl.right) div
2,rl.top+2,a,length(drw.co.LegTexts[drw.bo.BulletSource]));
    setTextAlign(h,TA_LEFT+TA_TOP);

    for i:=1 to drw.bo.custom.nranges do begin
        strcpy(a,drw.bo.custom.ranges[i].text);
        textout(h,44,h1+(i-1)*h2 + (drw.bo.custom.ranges[i].size div
2),a,length(drw.bo.custom.ranges[i].text));
    end;
    settextrcolor(h,drw.bo.linecolor);
    f:=tfont.create;
    f.name:=drw.bo.custom.fontname;
    for i:=1 to drw.bo.custom.nranges do begin
        f.size:=drw.bo.custom.ranges[i].size;
        oldf:=selectobject(h,f.handle);
        pc:=@drw.bo.custom.ranges[i].symbol;
        textout(h,16,h1+(i-1)*h2 ,pc,1);
        selectobject(h,oldf);
    end;
    f.free;
end;
end;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.CalcChartRatios;
var i,j:integer;

```

```

mx,mx0,mx1:real;
begin
mx:=0;mx0:=0;mx1:=0;
for i:=1 to npolypoints do if sel[i] then begin
if np[i].values[drw.bo.bulletsource]>mx0 then
mx0:=np[i].values[drw.bo.bulletsource];
if np[i].values[drw.so.source]>mx1 then mx1:=np[i].values[drw.so.source];
for j:=1 to drw.co.nvalues do
if np[i].values[j]>mx then mx:=np[i].values[j];
end;
drw.maxval0:=mx0;
drw.maxvall:=mx1;
drw.maxvals:=mx;
end;

Procedure  TGPS_SIRF_ATLAS_V.SetHDCFrame(h:hdc;r:trect);
begin
viewext:=MapLimits;
drw._view:=fzoombox;
drw._back:=r;
adaptlegrect(r,viewpct,drw._frame);
if drw.mo.aspect=1 then begin
if drw._view.right-drw._view.left<drw._view.bottom-drw._view.top
then begin drw._view.right:=drw._view.left+drw._view.bottom-drw._view.top;end
else begin drw._view.bottom:=drw._view.top+drw._view.right-drw._view.left;end;
end;
if (drw._view.right-drw._view.left) >0 then
drw.Rx:=(drw._frame.right-drw._frame.left)/(drw._view.right-drw._view.left)
else drw.Rx:=1;
if (drw._view.bottom-drw._view.top)>0 then
drw.Ry:=(drw._frame.bottom-drw._frame.top)/(drw._view.bottom-drw._view.top)
else drw.Ry:=1;
if drw.rx=0 then drw.rx:=1;
if drw.ry=0 then drw.ry:=1;
end;

Procedure  TGPS_SIRF_ATLAS_V.FillNames(lb:tlistbox);
var i,j,k:integer;
begin
lb.clear;
for i:=1 to npolypoints do lb.items.add(inttostr(i)+'/'+np[i].nombre);
end;

Procedure  TGPS_SIRF_ATLAS_V.FillArranges(lb:tlistbox);
begin
if not ready then exit;
arr.alltolistbox(lb);
end;

procedure TGPS_SIRF_ATLAS_V.ArrSet(father,son:word;ok:boolean);
begin
if not ready then exit;
arr.Poner(father,son,ok);
end;

Procedure TGPS_SIRF_ATLAS_V.ArrFromListBox(lb:tlistbox;father:word);
begin
if not ready then exit;
arr.FromListBox(lb,father);
ArrCalcBoxes;
end;

Procedure TGPS_SIRF_ATLAS_V.ArrToListBox(lb:tlistbox;father:word);
begin
if not ready then exit;
arr.ToListBox(lb,father);
end;

```

```

Procedure TGPS_SIRF_ATLAS_V.ArrSetName(name:string;father:word);
begin
if not ready then exit;
if father>arr.n then exit;
arr.titulos[father]^:=name;
end;

Procedure TGPS_SIRF_ATLAS_V.ArrSetNumber(n:word);
begin
arr.setnumber(n);
end;

Function TGPS_SIRF_ATLAS_V.ArrGetNumber:word;
begin
ArrGetNumber:=arr.n;
end;

Procedure TGPS_SIRF_ATLAS_V.SelArrGroup(father:integer;ok:boolean);
var i:integer;
begin
if father>0 then
if father<=arr.n then
for i:=1 to arr.t do if arr.Matr.Bit[father,i] then sel[i]:=ok;
end;

Function TGPS_SIRF_ATLAS_V.GetElSelected(i:word):boolean;
begin
GetElSelected:=false;
if i>0 then if i<=npolypoints then GetElSelected:=Sel[i];
end;

Procedure TGPS_SIRF_ATLAS_V.SetElSelected(i:word;ok:boolean);
begin
if i>0 then if i <=npolypoints then Sel[i]:=ok;
end;

Procedure TGPS_SIRF_ATLAS_V.SetAllSelections(ok:boolean);
var i:integer;
begin
for i:=1 to elementcount do selected[i]:= ok;
end;

Procedure TGPS_SIRF_ATLAS_V.SetBitMap(fname:string);
begin
bmpfname:=fname;
bitmap.free;
bitmap:=nil;
if fileexists(fname) then begin
bitmap:=tbitmap.create;
bitmap.loadfromfile(fname);
drw.mo.mapimage:=2;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.DrawBitMap(h:hdc);
var ew,ow:longint;
w:trect;
vex,vey,vox,voy:integer;
rwb:record x,y:real;end;
wb:trect;
bm:tbitmap;
i,j:integer;
wx,wy:double;
wf:trect;

begin
if bitmap=nil then exit;
if drw.mo.mapimage=0 then exit;

```

```

if drw.mo.mapimage=2 then begin
  wx:=Bitmap.width/(viewext.right-viewext.left);
  wy:=Bitmap.height/(viewext.bottom-viewext.top);

  w:=drw._view;
  if w.left<viewext.left then w.left:=viewext.left;
  if w.top<viewext.top then w.top:=viewext.top;
  if w.right>viewext.right then w.right:=viewext.right;
  if w.bottom>viewext.bottom then w.bottom:=viewext.bottom;

  wb.left:=Trunc((w.left-viewext.left)*wx);
  wb.right:=Trunc((w.right-viewext.left)*wx);
  wb.top:=Trunc((w.top-viewext.top)*wy);
  wb.bottom:=Trunc((w.bottom-viewext.top)*wy);

  drw.ConvertaRect(w,wf);

  if tometa then begin
    bm:=tbitmap.create;
    bm.assign(bitmap);
    StretchBlt(bm.canvas.handle,0,0,wb.right-wb.left,wb.bottom-wb.top,
      bitmap.canvas.handle,wb.left,wb.top,wb.right-wb.left,wb.bottom-
wb.top,SRCCOPY);

    bm.width:=wb.right-wb.left;bm.height:=wb.bottom-wb.top;
    StretchBlt(h,wf.left,wf.top,wf.right-wf.left,wf.bottom-wf.top,
      bm.canvas.handle,0,0,bm.width,bm.height,SRCCOPY);
    bm.destroy;
  end
  else
  with wf do
    StretchBlt(h,left,top,right-left,bottom-top,
      bitmap.canvas.handle,wb.left,wb.top,wb.right-wb.left,wb.bottom-
wb.top,SRCCOPY);
  end
  else begin
  with drw._frame do
    StretchBlt(h,left,top,right-left,bottom-top,
      bitmap.canvas.handle,0,0,bitmap.width,bitmap.height,SRCCOPY);
  end;
end;

Procedure TGPS_SIRE_ATLAS_V.CleanHDC(h:hdc;col:tcolor;box:trect);
var b,oldb:hbrush;
p,oldp:hpen;
i,j,k,wy:integer;
gr:TGrArray;
w,hh,x,y:integer;
bm:tbitmap;
begin
CalcColGradients(drw.bko.mincolor,drw.bko.maxcolor,drw.bko.GradItems,gr);
if not tometa then box:=drw._back;
w:=(box.right-box.left) div drw.bko.GradItems;
hh:=(box.bottom-box.top) div drw.bko.GradItems;
b:=drw.GetSolidBrush(gr[drw.bko.GradItems]);
oldb:=selectobject(h,b);

fillrect(h,rect(box.left,box.top,box.right,box.bottom),b);
selectobject(h,oldb);deleteobject(b);

case drw.bko.effect of
1:for i:=1 to drw.bko.GradItems do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
fillrect(h,rect(box.left,box.top+(i-1)*hh,box.right,box.top+i*hh),b);
selectobject(h,oldb);deleteobject(b);
end;
end;

```

```

2:for i:=1 to drw.bko.GradItems do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
fillrect(h,rect(box.left+(i-1)*w,box.top,box.left+i*w,box.bottom),b);
selectobject(h,oldb);deleteobject(b);
end;
3:begin
x:=(box.right-box.left) div 2;
y:=(box.bottom-box.top) div 2;
w:=x div 20;hh:=y div 20;
for i:=drw.bko.GradItems downto 1 do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
p:=createpen(0,0,gr[i]);
oldp:=selectobject(h,p);
ellipse(h,box.left+x-w*i,box.top+y-hh*i,box.left+x+w*i,box.top+y+hh*i);
selectobject(h,oldp);deleteobject(p);
selectobject(h,oldb);deleteobject(b);
end;
end;
4:begin
x:=(box.right-box.left) div 2;
y:=(box.bottom-box.top) div 2;
w:=x div drw.bko.GradItems;hh:=y div drw.bko.GradItems;
for i:=drw.bko.GradItems downto 1 do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
fillrect(h,rect(box.left+x-w*i,box.top+y-hh*i,box.left+x+w*i,box.top+y+hh*i),b);
selectobject(h,oldb);deleteobject(b);
end;
end;
5:begin
if not BackBitmap.empty then begin
x:=(box.right-box.left) div BackBitmap.width;
y:=(box.bottom-box.top) div BackBitmap.height;
for i:=0 to x do for j:=0 to y do
BitBlt(h,box.left+i*BackBitmap.width,box.top+j*BackBitmap.height,BackBitmap.widht
h,BackBitmap.height,BackBitmap.canvas.handle,0,0 ,SRCCOPY);
end;
end;
end;
end;

Procedure  TGPS_SIRF_ATLAS_V.Meridians(h:hdc;box:trect);
var i,j,w,hh,x,y:integer;
p,oldp:hpen;
pix:tcolor;
bm:tbitmap;
begin
if drw.bko.lines then begin
if not tometa then box:=drw._back;
w:=drw.bko.pctw;w:=trunc( w* (box.right-box.left)/ 100.0);
hh:=drw.bko.pctv;hh:=trunc( hh * (box.bottom-box.top)/ 100.0);
x:=(box.right-box.left) div w; y:= (box.bottom -box.top) div hh;
p:=createpen(0,0,drw.bko.linecolor);
oldp:=selectobject(h,p);
for i:=1 to x do begin
movetoex(h,box.left+i*w,box.top,nil);
lineto(h,box.left+i*w,box.bottom);
end;
for i:=1 to y do begin
movetoex(h,box.left,box.top+i*hh,nil);
lineto(h,box.right,box.top+i*hh);
end;
selectobject(h,oldp);
deleteobject(p);
end;
if drw.bko.logoon then begin
i:=LogoBitmap.width div 2;

```

```

j:=LogoBitmap.height div 2;
box.left:=box.left+1;box.right:=box.right-
1;box.top:=box.top+1;box.bottom:=box.bottom-1;
case drw.bko.Logopos of
  1:begin x:=box.left;y:=box.top;end;
  2:begin x:=(box.left+box.right) div 2 - i;y:=box.top;end;
  3:begin x:=box.right-LogoBitmap.width;y:=box.top;end;
  4:begin x:=box.left;y:=(box.top+box.bottom) div 2 -j;end;
  5:begin x:=(box.left+box.right) div 2 - i;y:=(box.top+box.bottom) div 2 -
j;end;
  6:begin x:=box.right-LogoBitmap.width;y:=(box.top+box.bottom) div 2 -j;end;
  7:begin x:=box.left;y:=box.bottom-LogoBitmap.height;end;
  8:begin x:=(box.left+box.right) div 2 - i;y:=box.bottom-LogoBitmap.height;end;
  9:begin x:=box.right-LogoBitmap.width;y:=box.bottom-LogoBitmap.height;end;
  else begin x:=box.left;y:=box.top;end;
end;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.SetColors(t:byte;f,b:tcolor;ix:word);
begin
np[ix].setcolors(t,f,b);
end;

Procedure TGPS_SIRF_ATLAS_V.SetPBGColor(k:integer;c:tcolor);
var i:integer;
begin
for i:=1 to np[k].npolis do np[k].p[i].bkcol:=c;
for i:=1 to np[k].nmarcos do np[k].m[i].bkcol:=c;
end;

function TGPS_SIRF_ATLAS_V.GetPBGColor(k:integer):tcolor;
begin
if np[k].npolis>0 then
GetPBGColor:=np[k].p[1].bkcol
else if np[k].nmarcos>0 then
GetPBGColor:=np[k].m[1].bkcol
else GetPBGColor:=clwhite;
end;

Procedure TGPS_SIRF_ATLAS_V.SetPFColor(k:integer;c:tcolor);
var i:integer;
begin
for i:=1 to np[k].npolis do np[k].p[i].frcol:=c;
for i:=1 to np[k].nmarcos do np[k].m[i].frcol:=c;
end;

function TGPS_SIRF_ATLAS_V.GetPFColor(k:integer):tcolor;
begin
if np[k].npolis>0 then
GetPFColor:=np[k].p[1].FRcol
else if np[k].nmarcos>0 then
GetPFColor:=np[k].m[1].FRcol
else GetPFColor:=clBLACK;
end;

Procedure TGPS_SIRF_ATLAS_V.SetPPattern(k:integer;c:byte);
var i:integer;
begin
for i:=1 to np[k].npolis do np[k].p[i].trama:=c;
for i:=1 to np[k].nmarcos do np[k].m[i].trama:=c;
end;

function TGPS_SIRF_ATLAS_V.GetPPattern(k:integer):byte;
begin
if np[k].npolis>0 then
GetPPattern:=np[k].p[1].trama
else if np[k].nmarcos>0 then
GetPPattern:=np[k].m[1].trama

```

```

else GetPPattern:=0;
end;

Procedure TGPS_SIRF_ATLAS_V.SetChartLegend(k:word;s:string);
begin
if k>16 then exit;
drw.co.LegTexts[k]:=s;
end;

function TGPS_SIRF_ATLAS_V.GetChartLegend(k:word):string;
begin
if k>16 then exit;
GetChartLegend:=drw.co.LegTexts[k];
end;

Procedure TGPS_SIRF_ATLAS_V.SetShadows(a,b:tcolor);
begin
drw.calcshadows(a,b);
end;

Procedure TGPS_SIRF_ATLAS_V.Applyshadows;
var i,j:integer;
aux:double;
c:integer;
pcts:array[0..250] of record
    f,s:double;
    v:double;
end;

Procedure CalcFrecs;
var i,j,k:integer;
d,m:double;
s:double;
nump:integer;
begin
d:=drw.maxvall / 250;
fillchar(pcts,sizeof(pcts),0);
nump:=0;
for i:=1 to 250 do pcts[i].v:=i*drw.maxvall/250;
for i:=1 to npolypoints do if sel[i] then begin
k:=trunc(np[i].values[drw.so.source] *250/drw.maxvall);
if drw.sot.calc=1 then
pcts[k].f:=pcts[k].f+1
else pcts[k].f:=pcts[k].f+np[i].values[drw.so.source];
inc(nump);
end;
s:=0;
for i:=1 to 250 do begin
s:=s+pcts[i].f;
pcts[i].s:=s;
end;
if drw.sot.calc=1 then m:=nump / drw.so.legendpoints
else m:=s / drw.so.legendpoints;
for i:=1 to drw.so.legendpoints do
for j:=1 to 250 do
if pcts[j].s>= m*i then begin
drw.brkshadow[i]:=pcts[j].v;
break;
end;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
for i:=1 to npolypoints do if sel[i] then begin
m:=np[i].values[drw.so.source];
for j:=1 to drw.so.legendpoints do
if m<=drw.brkshadow[j] then begin
np[i].SetColors(0,0,drw.sot.colsf[j]);
break;
end;
begin
calcchartrratios;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;

```

```

case drw.sot.kind of
0:begin
drw.calcshadows(drw.so.mincolor,drw.so.maxcolor);
for i:=1 to drw.so.legendpoints do
drw.brkshadow[i]:=(drw.maxvall/drw.so.legendpoints)*i;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
for i:=1 to npolypoints do if sel[i] then begin
c:=trunc(np[i].values[drw.so.source]*100/drw.maxvall);
if c<1 then c:=1;if c>100 then c:=100;
np[i].SetColor(0,0,drw.shadow[c]);
end;
end;
1:begin
if drw.sot.calc>0 then calcfrecs
else begin
for i:=1 to drw.so.legendpoints do
drw.brkshadow[i]:=(drw.maxvall/drw.so.legendpoints)*i;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
for i:=1 to npolypoints do if sel[i] then begin
c:=trunc(np[i].values[drw.so.source]*drw.so.legendpoints/drw.maxvall);
if c<1 then c:=1;if c>drw.so.legendpoints then c:=drw.so.legendpoints;
np[i].SetColor(0,0,drw.sot.colsf[c]);
end;
end;
end;
2:begin
for i:=1 to npolypoints do if sel[i] then begin
aux:=np[i].values[drw.so.source];
for j:=1 to drw.so.legendpoints do
if ((aux>=drw.sot.vals[j].min) and (aux<=drw.sot.vals[j].max)) then begin
np[i].SetColor(0,0,drw.sot.cols[j]);
break;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.RectToPolyLine(r2:trect;var cuadro:array of tpoint);
begin
cuadro[0].x:=r2.left;cuadro[0].y:=r2.top;
cuadro[1].x:=r2.right;cuadro[1].y:=r2.top;
cuadro[2].x:=r2.right;cuadro[2].y:=r2.bottom;
cuadro[3].x:=r2.left;cuadro[3].y:=r2.bottom;
cuadro[4].x:=r2.left;cuadro[4].y:=r2.top;
end;

procedure TGPS_SIRF_ATLAS_V.adaptlegrect(r,r1:trect;var r2:trect);
begin
r2.left:=trunc((r1.left/100)*(r.right-r.left)+r.left);
r2.top:=trunc((r1.top/100)*(r.bottom-r.top)+r.top);
r2.right:=trunc((r1.right/100)*(r.right-r.left)+r.left);
r2.bottom:=trunc((r1.bottom/100)*(r.bottom-r.top)+r.top);
end;

Procedure TGPS_SIRF_ATLAS_V.ClipWMF;
var
hdcmeta:hdc;
hmeta:hmetafile;
r:trect;
twmf:tmetafile;
begin
tometa:=true;
hdcmeta:=createmetabase(nil);
r:=rect(0,0,width,height);
SetWindowExtEx(hdcmeta,width,height,nil);
SetWindowOrgEx(hdcmeta,0,0,nil);
SetViewPortOrgEx(hdcmeta,0,0,nil);
SetViewPortExtEx(hdcmeta,width,height,nil);
Draw(hdcmeta,r);
hmeta:=closemetafile(hdcmeta);
twmf:=tmetafile.create;

```

```

twmf.width:=width;
twmf.height:=height;
with TMetafileCanvas.Create(twmf, 0) do
try
PlayMetaFile(handle,hmeta);
finally
free;
end;
twmf.enhanced:=false;
clipboard.assign(twmf);
twmf.free;
tometa:=false;
end;
Procedure TGPS_SIRF_ATLAS_V.WMF_clip2file(wnd:HWND;fname:string);
var
hGMem:longint;
lpmfp:^tMETAFILEPICT ;
f:file;
a:array[0..250] of char;
begin
OpenClipboard(Wnd);
hGMem:= GetClipboardData(CF_METAFILEPICT) ;
if (hGMem<>0) then begin
lpmfp:= GlobalLock(hGMem);
strcpy(a,fname);
copymetafile(lpmfp^.hmf,a);
GlobalUnlock(hGMem);
end;
CloseClipboard;
end;

procedure TGPS_SIRF_ATLAS_V.ClpWMF2Text(fname:string;textfname:string);
var tm:tmetaheader;
f:file;
ft:textfile;
rs:longint;
cf:word;
i,r,j:word;
s:longint;
tlf:tlogfont;
n,np:word;
ltp:^SPltSmallPoint;
begin
new(ltp);
assignfile(f,fname);
reset(f,1);
blockread(f,tm,sizeof(tm));
n:=0;
while not eof(f) do begin
blockread(f,rs,4);
blockread(f,cf,2);
if cf=$0324 then inc(n);
for j:=1 to rs-3 do blockread(f,cf,2);
end;
closefile(f);
if n=0 then exit;

assignfile(f,fname);
reset(f,1);

assignfile(ft,textfname);
rewrite(ft);

blockread(f,tm,sizeof(tm));
I:=1;

while not eof(f) do begin

```

```

blockread(f,rs,4);
blockread(f,cf,2);
if ((cf=$0324) and (i<maxelnum)) then begin
blockread(f,np,2);
blockread(f,ltp^,np*sizeof(TSmallPoint));
writeln(ft,':Element '+inttostr(i));
writeln(ft,'+P1');
for j:=1 to np do
writeln(ft,inttostr(ltp^[j].x)+' , '+inttostr(ltp^[j].y));
writeln(ft);
inc(i);
end
else begin
for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
end;
closefile(f);
closefile(ft);
dispose(ltp);
end;

Function TGPS_SIRE_ATLAS_V.ClpWMFRect(fname:string):trect;
var tm:tmetaheader;
f:file;
rs:longint;
cf:word;
i,r,j:word;
s:longint;
tlf:tlogfont;
n,np:word;
ltp:^SPltSmallPoint;
re:trect;
begin
new(ltp);
assignfile(f,fname);
reset(f,1);
blockread(f,tm,sizeof(tm));
n:=0;
re.left:=32000;re.top:=32000;re.bottom:=-32000;re.right:=-32000;

while not eof(f) do begin
blockread(f,rs,4);
blockread(f,cf,2);
if cf=$0324 then inc(n);
for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
closefile(f);
if n=0 then exit;

assignfile(f,fname);
reset(f,1);
blockread(f,tm,sizeof(tm));
I:=1;

while not eof(f) do begin
blockread(f,rs,4);
blockread(f,cf,2);
if ((cf=$0324) and (i<maxelnum)) then begin
blockread(f,np,2);
blockread(f,ltp^,np*sizeof(TSmallPoint));
for j:=1 to np do begin
if ltp^[j].x<re.left then re.left:=ltp^[j].x;
if ltp^[j].y<re.top then re.top:=ltp^[j].y;
if ltp^[j].x>re.right then re.right:=ltp^[j].x;
if ltp^[j].y>re.bottom then re.bottom:=ltp^[j].y;
end;
inc(i);
end
end

```

```

else begin
    for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
end;
closefile(f);
dispose(ltp);
ClpWMFRect:=re;
end;

Procedure
TGPS_SIRF_ATLAS_V.ScaleClpWMF2Text(fname,textfname:string;iso:boolean);
var tm:tmetaheader;
f:file;
ft:textfile;
rs:longint;
cf:word;
i,r,j:word;
s:longint;
tlf:tlogfont;
n,np:word;
ltp:^SPltSmallPoint;
e,re:trect;
sx,sy:real;
lt:integer;
puntos:integer;
a:array[0..255] of char;
ax:string;
begin
e:=rect(0,0,10000,10000);
re:=ClpWMFRect(fname);

if (re.right-re.left)>(re.bottom-re.top) then begin
sx:=(e.right-e.left)/(re.right-re.left);
sy:=(e.bottom-e.top)/(re.right-re.left);
end
else begin
sx:=(e.right-e.left)/(re.bottom-re.top);
sy:=(e.bottom-e.top)/(re.bottom-re.top);
end;

new(ltp);
assignfile(f,fname);
reset(f,1);
blockread(f,tm,sizeof(tm));
n:=0;
while not eof(f) do begin
blockread(f,rs,4);
blockread(f,cf,2);
if cf=$0324 then inc(n);
if cf=$0325 then inc(n);
for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
closefile(f);
if n=0 then exit;

assignfile(f,fname);
reset(f,1);

assignfile(ft,textfname);
rewrite(ft);

blockread(f,tm,sizeof(tm));
I:=1;
puntos:=0;
while not eof(f) do begin
blockread(f,rs,4);
blockread(f,cf,2);
if ((cf=$0324) and (i<maxelnum)) then begin

```

```

blockread(f,np,2);
blockread(f,ltp^,np*sizeof(TSmallPoint));
writeln(ft,':Element '+inttostr(i));
writeln(ft,'+P'+inttostr(i)+' #2');
for j:=1 to np do begin
  ltp^[j].x:=trunc((ltp^[j].x - re.left)*sx + e.left);
  ltp^[j].y:=trunc((ltp^[j].y - re.top)*sy + e.top);
  writeln(ft,inttostr(ltp^[j].x)+ ' , '+inttostr(ltp^[j].y));
end;
writeln(ft);
inc(i);
puntos:=1;
end
else if ((cf=$0325) and (i<maxelnum)) then begin
blockread(f,np,2);
blockread(f,ltp^,np*sizeof(TSmallPoint));
writeln(ft,':Element '+inttostr(i));
writeln(ft,'_L'+inttostr(i)+' #2');
for j:=1 to np do begin
  ltp^[j].x:=trunc((ltp^[j].x - re.left)*sx + e.left);
  ltp^[j].y:=trunc((ltp^[j].y - re.top)*sy + e.top);
  writeln(ft,inttostr(ltp^[j].x)+ ' , '+inttostr(ltp^[j].y));
end;
writeln(ft);
inc(i);
end

else if ((cf=$0521) and (i<100)) then begin
blockread(f,lt,2);
blockread(f,a,lt); a[lt+1]:=#0; ax:=strupas(a);
writeln(ft,'/T'+inttostr(puntos)+' #3');
blockread(f,lt,2);
write(ft,inttostr(lt)+' , ');
blockread(f,lt,2);
writeln(ft,inttostr(lt)+' , '+ ax);
inc(puntos);
end
else begin
  for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
end;
closefile(f);
closefile(ft);
dispose(ltp);
end;

Procedure TGPS_SIRF_ATLAS_V.FromWMF(fname:string);
var tmf:tmetafile;
a:array[0..255] of char;
ax,bx:string;
begin
GetWindowsDirectory(a,200);
ax:=strupas(a)+'\temp.clp';
bx:=strupas(a)+'\temp.txt';

tmf:=tmetafile.create;
tmf.loadfromfile(fname);
clipboard.assign(tmf);
WMF_clip2file(parent.handle,ax);
tmf.destroy;
ScaleClpWMF2Text(ax,bx,true);
FromTextFile(bx);
end;

Procedure TGPS_SIRF_ATLAS_V.FromClipWMF;
var a:array[0..255] of char;
ax,bx:string;
begin

```

```

if not clipboard.hasformat(CF_metafilepict) then exit;
GetWindowsDirectory(a,200);
ax:=strpas(a)+'\temp.clp';
bx:=strpas(a)+'\temp.txt';
WMF_clip2file(parent.handle,ax);
ScaleClpWMF2Text(ax,bx,true);
FromTextFile(bx);
end;

```

```

Procedure TGPS_SIRF_ATLAS_V.SaveWMF(fname:string);
var tmf:tmetafile;
begin
ClipWMF;
tmf:=tmetafile.create;
tmf.assign(clipboard);
tmf.savetofile(fname);
tmf.destroy;
end;

```

```

Procedure TGPS_SIRF_ATLAS_V.ClipBmp;
var
r:trect;
bm01:tbitmap;
begin
if not ready then exit;
bm01:=tbitmap.create;
try
r:=rect(0,0,width,height);
bm01.width:=(r.right-r.left)+1;
bm01.height:=(r.bottom-r.top)+1;
bm01.canvas.copyrect(r,canvas,r);
clipboard.assign(bm01);
finally
bm01.free;
end;
end;

```

```

Procedure TGPS_SIRF_ATLAS_V.SaveBmp(fname:string);
var bm:tbitmap;
r:trect;
begin
if not ready then exit;
bm:=tbitmap.create;
r:=rect(0,0,width,height);
bm.width:=(r.right-r.left)+1;
bm.height:=(r.bottom-r.top)+1;
bm.canvas.copyrect(r,canvas,r);
bm.SaveToFile(fname);
bm.free;
end;

```

```

Procedure
TGPS_SIRF_ATLAS_V.SnapShot(destinationcanvas:tcanvas;destrect:trect;maprect:trect);
var r:trect;
begin
if not ready then exit;
drw.convertarect(maprect,r);
StretchBlt(destinationcanvas.handle,
destrect.left,
destrect.top,
destrect.right-destrect.left+1,
destrect.bottom-destrect.top+1,
canvas.handle,
r.left,
r.top,

```

```

    r.right-r.left+1,
    r.bottom-r.top+1,
    SRCCOPY);

end;

Procedure TGPS_SIRF_ATLAS_V.SetChartValue(element,item:word;value:real);
begin
if element>npolypoints then exit;
if element<1 then exit;
if item>drw.co.nvalues then exit;
np[element].values[item]:=value;
end;

Function  TGPS_SIRF_ATLAS_V.GetChartValue(element,item:word):real;
begin
GetChartValue:=0;
if element>npolypoints then exit;
if element<1 then exit;
if item>drw.co.nvalues then exit;
GetChartValue:=np[element].values[item];
end;

Procedure  TGPS_SIRF_ATLAS_V.ValuesToGrid(grid:tstringgrid);
var i,j:integer;
ax:string;
begin
with grid do begin
colcount:=drw.co.nvalues+2;
rowcount:=npolypoints+1;
fixedcols:=0;
fixeddrows:=0;
cells[0,0]:='';
cells[1,0]:='Bullet';
for i:=1 to drw.co.nvalues do cells[1+i,0]:=drw.co.LegTexts[i];
for i:=1 to npolypoints do cells[0,i]:=np[i].nombre;
for i:=1 to npolypoints do
cells[1,i]:=fcadd(np[i].values[0],3,drw.co.ndecimals);
for i:=1 to drw.co.nvalues do
for j:=1 to npolypoints do
cells[i+1,j]:=fcadd(np[j].values[i],3,drw.co.ndecimals);
end;
end;

Procedure  TGPS_SIRF_ATLAS_V.ValuesFromGrid(grid:tstringgrid);
var i,j:integer;
ax:string;
begin
with grid do begin
drw.co.nvalues:=colcount-2;
for i:=0 to drw.co.nvalues do drw.co.LegTexts[i]:=cells[1+i,0];
for i:=1 to rowcount-1 do np[i].nombre:=cells[0,i];
for i:=1 to rowcount-1 do np[i].values[0]:=fnum(cells[1,i]);
for i:=1 to drw.co.nvalues do
for j:=1 to rowcount-1 do
np[j].values[i]:=fnum(cells[i+1,j]);
end;
end;
CalcChartRatios;
end;

Procedure  TGPS_SIRF_ATLAS_V.ImportDataFromGrid(grid:tstringgrid);
var i,j,k,code:integer;
r:real;
ax:string;
begin

```

```

if grid.colcount>18 then grid.colcount:=18;
drw.co.nvalues:=grid.colcount-2;
for i:=1 to grid.rowcount-1 do begin
ax:=uppercase(grid.cells[0,i]);
for j:=1 to ElementCount do
if uppercase(ElementName[j])=ax then
for k:=1 to grid.colcount-1 do begin
val(grid.cells[k,i],r,code);
chartvalue[j,k-1]:=r;
end;
end;
for i:=1 to grid.colcount-1 do chartlegend[i-1]:=grid.cells[i,0];
end;

Function  TGPS_SIRF_ATLAS_V.GetPatternBits(i:word):string;
begin
if i>8 then exit;
if i=0 then GetPatternBits:='000000'
else GetPatternBits:=trama[i];
end;

Procedure TGPS_SIRF_ATLAS_V.SetPatternBits(i:word;s:string);
begin
if ((i<1) or (i>8)) then exit;
trama[i]:=copy(s,1,64);
end;

Procedure TGPS_SIRF_ATLAS_V.Bar(h:hdc;x1,y1,x2,y2:integer;pat:word);
begin
if pat>16 then exit;if pat<1 then exit;
drw.GetBrocha(h,drw.co.patt[pat],drw.co.linecolor,drw.co.colors[pat]);
rectangle(h,x1,y1,x2,y2);
drw.ReleaseBrocha;
end;

Function TGPS_SIRF_ATLAS_V.GetPolLayer(element,pol:word):byte;
begin
GetPolLayer:=1;
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
GetPolLayer:=np[element].p[pol].layer;
end;

Procedure TGPS_SIRF_ATLAS_V.SetPolLayer(element,pol:word;l:byte);
begin
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
np[element].p[pol].layer:=l;
end;

Function TGPS_SIRF_ATLAS_V.GetPolCount(element:word):byte;
begin
GetPolCount:=0;
if element=0 then getpolcount:=npolypoints;
if (element>npolypoints) then exit;
GetPolCount:=np[element].npolis;
end;

Function TGPS_SIRF_ATLAS_V.GetPolName(element,pol:word):string;
begin
GetPolName:='';
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;

```

```

if (pol>np[element].npolis) then exit;
GetPolName:=np[element].p[pol].nombre;
end;

Procedure TGPS_SIRF_ATLAS_V.SetPolName(element,pol:word;s:string);
begin
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
np[element].p[pol].nombre:=s;
end;

Procedure TGPS_SIRF_ATLAS_V.ShowPolNames(b:boolean);
begin
drw.showpols:=b;
end;

Function TGPS_SIRF_ATLAS_V.GetDistance(p1,p2:tpoint):double;
begin
GetDistance:=distancia(p1.x,p1.y,p2.x,p2.y)*drw.mo.scale;
end;

Procedure TGPS_SIRF_ATLAS_V.Setnumbox(b:boolean);
begin
drw.numbox:=b;
end;
Function TGPS_SIRF_ATLAS_V.GetNumBox:boolean;
begin
GetNumBox:=drw.numbox;
end;

Function TGPS_SIRF_ATLAS_V.ElementbyName(s:string):integer;
var i,k:integer;
begin
k:=1;
for i:=1 to elementcount do
if pos(uppercase(s),uppercase(elementname[i]))>0 then begin
k:=i;
break;
end;
ElementbyName:=k;
end;

Function TGPS_SIRF_ATLAS_V.Nearest(p:tpoint):integer;
var i,j,k:integer;
mybox:trect;
begin
k:=0;
Nearest:=0;
for i:=npolypoints downto 1 do
if np[i].npolis>0 then begin
if ((p.x>np[i].caja.left) and (p.x<np[i].caja.right)
and (p.y>np[i].caja.top) and (p.y<np[i].caja.bottom))
then begin Nearest:=i;exit;end;
end;
Nearest:=k;
end;

Function TGPS_SIRF_ATLAS_V.NearestElement(p:tpoint):integer;
var i,j,k:integer;
mybox:trect;
begin
k:=0;
NearestElement:=0;
for i:=npolypoints downto 1 do if sel[i] then
if np[i].npolis>0 then begin
if ((p.x>np[i].caja.left) and (p.x<np[i].caja.right)

```

```

    and (p.y>np[i].caja.top) and (p.y<np[i].caja.bottom))
    then begin NearestElement:=i;exit;end;
end;
NearestElement:=k;
end;

procedure muerte;
begin
showmessage('Fatal error reading '+fname);
halt;
end;
var i,j,k,im,ip,ipu:integer;
ax,bx,nom:string;
fin:boolean;
p:tpoint;
f:textfile;
nsp:^tnsp;
n:word;
buffer:^SPtpoint;
maxcaja:trect;
c:char;
thelayer:integer;
lm:longint;
begin
reset(f);
ax='';
for i:=1 to npolypoints do begin
while ((ax[1]<>':') and (not eof(f))) do begin
readln(f,ax);
ax:=ltrim(ax);
end;
if eof(f) then muerte;
infopanel.caption:=inttostr(i)+' of '+ inttostr(npolypoints);
infopanel.repaint;
ax:=rltrim(derc(ax,':'));
np[i]:=TSPolyPol.create(nsp^[i].m,nsp^[i].p,ax);
np[i].npoints:=nsp^[i].pu;
im:=0;ip:=0; ipu:=0;
for j:=1 to nsp^[i].m + nsp^[i].p + nsp^[i].pu do begin
repeat
readln(f,ax);
ax:=ltrim(ax);
until ((ax[1] in ['_','+','/']) or eof(f));
if eof(f) then muerte;
c:=ax[1];
nom:=rltrim(izqc(derc(ax,c),'#'));
ax:=rltrim(derc(ax,'#'));
if ax='' then ax:='1';
thelayer:=num(ax);
if c='_' then begin
inc(im);
np[i].SetMVals(clwhite,clblack,im,k,buffer^);
np[i].m[im].layer:=thelayer;
np[i].m[im].drawer:=drw;
np[i].m[im].nombre:=nom;
end
else
if c='+' then begin
inc(ip);
np[i].SetPVals(clwhite,clblack,ip,k,buffer^);
np[i].p[ip].layer:=thelayer;
np[i].p[ip].drawer:=drw;
np[i].p[ip].nombre:=nom;
end;
end;
np[i].CalcCaja;
comprect(np[i].caja,maxcaja);
np[i].drawer:=drw;
end;

```

```

closefile(f);
dispose(nsp);
dispose(buffer);
for i:=1 to npolypoints do sel[i]:=true;
if arr<>nil then arr.destroy;
arr:=TSArrange.create(2,npolypoints,'Normal');
ready:=true;
Titulo:=fname;
Nbloqs:=npolypoints;
MapLimits:=MaxCaja;
end;

Function  TGPS_SIRF_ATLAS_V.AddPoint(elem:integer;c:tcitypoint):integer;
var a:tsapoint;
begin
AddPoint:=0;
if elem>npolypoints then exit; if elem<1 then exit;
if np[elem].npoints>15 then exit;
a:=tsapoint.create;
a.c:=c;
np[elem].npoints:=np[elem].npoints+1;
np[elem].points[np[elem].npoints]:=a;
AddPoint:=np[elem].npoints;
a.free;
end;

Procedure TGPS_SIRF_ATLAS_V.GetPoint(elem,ix:integer;var c:tcitypoint);
begin
if elem>npolypoints then exit; if elem<1 then exit;
if ix<1 then exit;if ix> np[elem].npoints then exit;
c:=np[elem].points[ix].c;
end;

Function  TGPS_SIRF_ATLAS_V.RemovePoint(elem,ix:integer):boolean;
var i:integer;
begin
removepoint:=false;
if elem>npolypoints then exit; if elem<1 then exit;
if np[elem].npoints<1 then exit;
if ix<1 then exit;if ix> np[elem].npoints then exit;
for i:=ix to np[elem].npoints-1 do np[elem].points[i]:=np[elem].points[i+1];
np[elem].npoints:=np[elem].npoints-1;
Removepoint:=true;
end;

Function  TGPS_SIRF_ATLAS_V.GetPointCount(elem:integer):integer;
begin
GetPointCount:=0;
if elem>npolypoints then exit; if elem<1 then exit;
if np[elem].npoints<1 then exit;
GetPointCount:=np[elem].npoints;
end;

Procedure TGPS_SIRF_ATLAS_V.ClearPoints(elem:integer);
begin
if elem>npolypoints then exit; if elem<1 then exit;
np[elem].npoints:=0;
end;

Procedure TGPS_SIRF_ATLAS_V.SetSymbol(f:tfont;c:char);
begin
drw.fontsymbol.assign(f);
drw.asymbol:=c;
end;

procedure TGPS_SIRF_ATLAS_V.time(sender:tobject);
begin
replaceglis;

```

```

gli.paint(canvas);
end;

Procedure TGPS_SiRF_ATLAS_V.MyMouseUp(Sender: TObject; Button:
TMouseButton; Shift: TShiftState; X, Y: Integer);
var p:tpoint;
i,k:integer;
pu:tpoint;
pr:trect;
begin
case drw.mo.mousemode of
8:begin
    p.x:=x;p.y:=y;
    p:=PointInMap(p);
    if button=mbright then begin
        p.x:=(GPS_SiRF_Atlas_Vbox.right+GPS_SiRF_Atlas_Vbox.left) div 2;
        p.y:=(GPS_SiRF_Atlas_Vbox.top+GPS_SiRF_Atlas_Vbox.bottom) div 2;
    end;
    k:=(zoombox.right-zoombox.left) div 2;
    i:=(zoombox.bottom-zoombox.top) div 2;
    fzoombox.left:=p.x-k; fzoombox.right:=p.x+k;
    fzoombox.top:=p.y-i; fzoombox.bottom:=p.y+i;
    repaint;
end;

10:if bb_drawing then begin
    bb_drawing:=false;
    DrawFocusRect(canvas.handle,rect(bb_anchorx,bb_anchory,bb_curx,bb_cury));
    p:=PointInMap(point(bb_anchorx,bb_anchory));
    if button=mbright then SelFromBox(GPS_SiRF_Atlas_Vbox,true)
    else
    if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
    begin
        pr.left:=p.x;pr.top:=p.y;
        p:=PointInMap(point(bb_curx,bb_cury));
        pr.right:=p.x;pr.bottom:=p.y;
        if ssCtrl in Shift then SelFromBox(pr,false)
        else SelFromBox(pr,true);
    end
    else begin
        k:=Nearest(p);
        if k>0 then begin
            if ssCtrl in Shift then selected[k]:=true
            else selected[k]:= not selected[k];
        end;
    end;
    invalidate;
end;

1,9,12,15:if bb_drawing then begin
    bb_drawing:=false;
    DrawFocusRect(canvas.handle,rect(bb_anchorx,bb_anchory,bb_curx,bb_cury));

    if drw.mo.mousemode =9 then begin
        pr.left:=trunc((bb_anchorx-left)/(drw._back.right-drw._back.left)*100);
        pr.right:=trunc((bb_curx-left)/(drw._back.right-drw._back.left)*100);
        pr.top:=trunc((bb_anchory-top)/(drw._back.bottom-drw._back.top)*100);
        pr.bottom:=trunc((bb_cury-top)/(drw._back.bottom-drw._back.top)*100);
    end;

    case drw.mo.mousemode of
    9:
        if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
        begin
            viewpct:=pr;
            infopanel.caption:=inttostr(viewpct.left)+' ','+
            inttostr(viewpct.top)+' - '+'
            inttostr(viewpct.right)+' ','+
            inttostr(viewpct.bottom);
        end;
    end;
end;

```

```

end
else viewpct:=rect(1,1,99,99);

1:
if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
begin
p:=PointInMap(point(bb_anchorx,bb_anchory));
fzoombox.left:=p.x;fzoombox.top:=p.y;
p:=PointInMap(point(bb_curx,bb_cury));
fzoombox.right:=p.x;fzoombox.bottom:=p.y;
end
else

begin
p.x:=x;p.y:=y;
p:=PointInMap(p);

k:=(zoombox.right-zoombox.left) div 2;
i:=(zoombox.bottom-zoombox.top) div 2;
fzoombox.left:=p.x-k; fzoombox.right:=p.x+k;
fzoombox.top:=p.y-i; fzoombox.bottom:=p.y+i;

end;

12:if Assigned(FOnMouseBox) then begin
if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
begin
p:=PointInMap(point(bb_anchorx,bb_anchory));
pr.left:=p.x;pr.top:=p.y;
p:=PointInMap(point(bb_curx,bb_cury));
pr.right:=p.x;pr.bottom:=p.y;
end
else pr:=GPS_SiRF_Atlas_Vbox;
OnMouseBox(self,shift,pr);
end;
15:begin
if ((bb_curx-bb_anchorx>20) and (bb_cury-bb_anchory>20)) then
begin
p:=PointInMap(point(bb_anchorx,bb_anchory));
pr.left:=p.x;pr.top:=p.y;
p:=PointInMap(point(bb_curx,bb_cury));
pr.right:=p.x;pr.bottom:=p.y;
end
else pr:=GPS_SiRF_Atlas_Vbox;
scaleto(pr);
end;
end;
invalidate;
end;
3:begin
bb_drawing:=false;
invalidate;
end;
4,11:if bb_drawing then begin
bb_drawing:=false;
SetRop2(canvas.handle,r2_not);
movetoex(canvas.handle,drw.cvx(bb_anchorx),drw.cvy(bb_anchory),nil);
lineto(canvas.handle,drw.cvx(bb_curx),drw.cvy(bb_cury));
if drw.mo.mousemode=11 then if assigned(FOnMouseLine)
then OnMouseLine(self,shift,rect(bb_anchorx,bb_anchory,bb_curx,bb_cury));
end;
5:if bb_drawing then begin
bb_drawing:=false;
SetRop2(canvas.handle,r2_not);
p.x:=abs(bb_curx-bb_anchorx);
p.y:=p.x;

```

```

        k:=abs(drw.cvx(bb_curx)-drw.cvx(bb_anchorx));
        pu.x:=drw.cvx(bb_anchorx);
        pu.y:=drw.cvy(bb_anchory);
        Arc(canvas.handle,pu.x-k,pu.y-k,
            pu.x+k,pu.y+k,
            pu.x,pu.y,pu.x,pu.y);
end;
16:   if not userline.drawing then begin
        if button=mbright then invalidate;
    end;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.WMLButtonDown(var msg:TWMLButtonDown);
begin
    if not ready then exit;
    MyMouseDown(self,mbleft,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
    If Assigned(FOnMouseDown) then
        OnMouseDown(self,mbleft,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
end;

Procedure TGPS_SIRF_ATLAS_V.WMRButtonDown(var msg:TWMRButtonDown);
begin
    if ready then begin
        MyMouseDown(self,mbright,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
        If Assigned(FOnMouseDown) then
            OnMouseDown(self,mbright,KeysToShiftState(msg.keys),msg.xpos,msg.ypos)
        else if drw.mo.mousemode=0 then parent.dispatch(msg);
    end
    else inherited;
end;

Procedure TGPS_SIRF_ATLAS_V.WMLButtonUp(var msg:TWMLButtonUp);
begin
    if not ready then exit;
    MyMouseUp(self,mbleft,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
    if ready then If Assigned(fOnMouseUp) then
        OnMouseUp(self,mbleft,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
    inherited;
end;

Procedure TGPS_SIRF_ATLAS_V.WMRButtonUp(var msg:TWMRButtonUp);
begin
    if ready then begin
        MyMouseUp(self,mbright,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
        if ready then If Assigned(fOnMouseUp) then
            OnMouseUp(self,mbright,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
        end;
    end;
end;

Procedure TGPS_SIRF_ATLAS_V.WMMouseMove(var msg:TWMMouseMove);
begin
    if not ready then exit;
    MyMouseMove(self,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
    if ready then if Assigned(fOnMouseMove) then
        OnMouseMove(self,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
    end;
end;

Procedure TGPS_SIRF_ATLAS_V.ToFront(elem:integer);
var auxp:TSPolyPol;
    i:integer;
    auxs:boolean;
begin
    if ((elem<1) or (elem>npolypoints)) then exit;
    auxp:=np[elem];
    auxs:=selected[elem];
    for i:=elem to npolypoints-1 do begin
        np[i]:=np[i+1];
        selected[i]:=selected[i+1];
    end;
end;

```

```

end;
np[npolypoints]:=auxp;
selected[npolypoints]:=auxs;
end;

Procedure TGPS_SIRF_ATLAS_V.ToBack(elem:integer);
var auxp:TSPolyPol;
i:integer;
auxs:boolean;
begin
if ((elem<1) or (elem>npolypoints)) then exit;
auxp:=np[elem];
auxs:=selected[elem];
for i:=elem downto 2 do begin
np[i]:=np[i-1];
selected[i]:=selected[i-1];
end;
np[1]:=auxp;
selected[1]:=auxs;
end;

Procedure TGPS_SIRF_ATLAS_V.ReMap(var m:array of integer);
var sp:^TSPOLARRAY;
i:integer;
begin
new(sp);
for i:=1 to npolypoints do sp^[i]:=np[m[i]];
for i:=1 to npolypoints do np[i]:=sp^[i];
dispose(sp);
end;

Function TGPS_SIRF_ATLAS_V.IsInBox(ix:integer;r:trect):boolean;
var rr:trect;
begin
IntersectRect(rr,r,np[ix].caja);
IsInBox:=((rr.left<>0) or (rr.right<>0) or (rr.bottom<>0) or (rr.top<>0));
end;

Procedure TGPS_SIRF_ATLAS_V.SelFromBox(r:trect;cleanup:boolean);
var i:integer;
begin
if cleanup then begin
for i:=1 to elementcount do selected[i]:= IsInBox(i,r);
end
else
for i:=1 to elementcount do if IsInBox(i,r) then selected[i]:= true;
end;

Procedure TGPS_SIRF_ATLAS_V.ForwardLink(ix:integer);
var ax:string;
begin
if ix>0 then
if np[ix].link<>' ' then
if fileexists(np[ix].link) then begin
if nbacks<5 then inc(nbacks);
BackLink[nbacks]:=polfname;
ax:=np[ix].link;
FileName:=ax;
drw.mo.mousemode:=14;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.BackwardLink;
var ax:string;
begin
if nbacks>0 then begin
if fileexists(BackLink[nbacks]) then begin
ax:=BackLink[nbacks];
FileName:=ax;

```

```

drw.mo.mousemode:=14;
end;
dec(nbacks);
end;
end;

Procedure TGPS_SIRF_ATLAS_V.SetBackgroundOptions(var op:tbkop);
begin
drw.bko:=op;
if drw.bko.tilefname<>' ' then if fileexists(drw.bko.tilefname)
then backbitmap.loadfromfile(drw.bko.tilefname);
if drw.bko.logofname<>' ' then if fileexists(drw.bko.logofname)
then logobitmap.loadfromfile(drw.bko.logofname);
end;

Procedure TGPS_SIRF_ATLAS_V.CalcCaja;
var i:integer;
begin;
prepect(MapLimits);
for i:=1 to npolypoints do comprect(np[i].caja,MapLimits);
end;

Procedure TGPS_SIRF_ATLAS_V.Merge(fname:string;layeroffset:integer;
x,y:integer);
var aGPS_SiRF_Atlas_V:TGPS_SIRF_ATLAS_V;
i,j,k:integer;
begin
if not fileexists(fname) then exit;
ready:=false;
aGPS_SiRF_Atlas_V:=TGPS_SIRF_ATLAS_V.create(self);
aGPS_SiRF_Atlas_V.filename:=fname;
for i:=1 to aGPS_SiRF_Atlas_V.elementcount do
np[elementcount+i]:=aGPS_SiRF_Atlas_V.np[i];
npolypoints:=npolypoints+aGPS_SiRF_Atlas_V.elementcount;
aGPS_SiRF_Atlas_V.npolypoints:=0;
aGPS_SiRF_Atlas_V.free;
CalcCaja;
ready:=true;
zoombox:=MapLimits;
end;

Procedure TGPS_SIRF_ATLAS_V.ScaleTo(r:trect);
var sb,aux:trect;
kx,ky:double;
x0,y0,x1,y1:integer;
i,j:integer;
function newx(x:integer):integer;
begin
newx:=trunc(x1+(x-x0)*kx);
end;
function newy(y:integer):integer;
begin
newy:=trunc(y1+(y-y0)*ky);
end;

function newrect(t:trect):trect;
var tt:trect;
begin
tt.left:=newx(t.left);
tt.right:=newx(t.right);
tt.top:=newy(t.top);
tt.bottom:=newy(t.bottom);
newrect:=tt;
end;
begin
for i:=1 to elementcount do if selected[i] then
np[i].scaleto(newrect(np[i].caja));
end;

```

```

Procedure TGPS_SIRF_ATLAS_V.AddPoly(closed:boolean);
var i,j:integer;
var slt:^SPltpoint;
begin
if userline.np<2 then exit;
npolypoints:=npolypoints+1;
np[npolypoints]:=TSPolyPol.create(1,0,'New');
slt:=addr(userline.p);
with np[npolypoints] do begin
SetMVals(clred,clblue,1,userline.np,slt^);
m[1].drawer:=drw;
m[1].layer:=2;
m[1].nombre:='Line 1';
CalcCaja;
drawer:=drw;
end;
end;
calccaja;
selected[npolypoints]:=true;
end;

Procedure TGPS_SIRF_ATLAS_V.DeleteElements;
var i,j,k:integer;
ts:tspolypol;
begin
k:=npolypoints;
for j:=k downto 1 do if selected[j] then begin
if j<npolypoints then begin
ts:=np[j];
np[j]:=np[npolypoints];
selected[j]:=false;
ts.free;
end
else begin
np[npolypoints].free;
end;
npolypoints:=npolypoints -1;
end;
end;

Procedure TGPS_SIRF_ATLAS_V.JoinElements(orig,dest:integer);
var i,j,k:integer;
ts:tspolypol;
begin
if orig<npolypoints then begin
ts:=np[orig];
np[orig]:=np[npolypoints];
ts.free;
end
else begin
np[npolypoints].free;
end;
npolypoints:=npolypoints -1;

end;

procedure TGPS_SIRF_ATLAS_V.posterxN(Nx,Ny:integer);
VAR
P: TPosterPrint;
begin
if not ready then exit;
if (( nx<1) or (ny<1) or (nx>10) or (ny>10)) then exit;
screen.cursor:=crhourglass;
p:=TPosterPrint.create(self);
try
with p do begin
HorzPages:=nx;
VertPages:=ny;

```

```

    HorzMarge:=100;
    VertMarge:=100;
    Image:='temp.wmf';
    LabelPages:=true;
end;
SaveWMF('temp.wmf');
p.doprint;
finally
p.free;
screen.cursor:=crdefault;
end;
end;

Function TGPS_SIRF_ATLAS_V.GetArea (elem:integer):longint;
begin
GetArea:=trunc(np[elem].GetArea (drw.mo.scale,drw.mo.scale));
end;

Procedure TGPS_SIRF_ATLAS_V.LoadPaletteFromFile (filename:string);
begin
deleteobject (paleta);
paleta:=CreatePaletteFromFile (filename);
end;

Procedure TGPS_SIRF_ATLAS_V.LoadMosaicBitmap (fname:string);
begin
drw.mosaicbmp.loadfromfile (fname);
end;

procedure TGPS_SIRF_ATLAS_V.placegli (ix:integer;p:tpoint);
var pp:tpoint;
begin
if ((ix>0) and (ix<=gli.count)) then begin
gli.glyph[ix].mappos:=p;
drw.convertapoint (p,pp);
gli.glyph[ix].pos:=pp;
gli.glyph[ix].paint (canvas);
end;
end;

procedure TGPS_SIRF_ATLAS_V.addgli (p:tpoint);
begin
gli.add(0);
placegli (gli.count,p);
end;

procedure TGPS_SIRF_ATLAS_V.replaceglis;
var i:integer;
p,pp:tpoint;
begin
for i:=gli.count downto 1 do begin
drw.convertapoint (gli.glyph[i].mappos,p);
gli.glyph[i].pos:=p;
end;
end;

function TGPS_SIRF_ATLAS_V.GlyphCount:integer;
begin
GlyphCount:=gli.count;
end;

Procedure
TGPS_SIRF_ATLAS_V.GlyphSettings (ixglyph:integer;imageix:integer;animate:boolean;
enable:boolean);
begin
if gli.glyph[ixglyph]<>nil then with gli.glyph[ixglyph] do begin
if imageix<gli.gn then index:=imageix;
animated:=animate;

```

```
enabled:=enable;  
end;  
end;
```

```
Procedure TGPS_SIRF_ATLAS_V.LoadGlyphsBitmap(fname:string);  
begin  
gli.Loadfromfile(fname);  
end;  
  
end.
```

К6П3-2023

## Головний файл програми

```

program GPS_SIRF_ATLAS_V_NAVIGATION;
{Федоров Богдан Сергійович, КН-22М-1}
uses
  Forms,
  GPS_SIRF_ATLAS_V_About in 'GPS_SIRF_ATLAS_V_About.pas' {F2},
  GPS_SIRF_ATLAS_V_MainForm in 'GPS_SIRF_ATLAS_V_MainForm.pas' {F1},
  GPS_SIRF_ATLAS_V in 'GPS_SIRF_ATLAS_V.pas';
{$R *.RES}

procedure GetSubDirs (Folder: string; sList: TStringList);
var
  sr: TSearchRec;
begin
  if FindFirst (Folder + '.*', faDirectory, sr) = 0 then
    try
      repeat
        if (sr.Attr and faDirectory) = faDirectory then
          sList.Add (sr.Name);
        until FindNext(sr) 0;
      finally
        FindClose(sr);
      end;
    end;
end;

function GetAllFiles (Filter, Folder: string):TFilesList;
var
  sr: TSearchRec;
  sDirList, _FilesList, _LST: TStringList;
  i, j: Integer;
begin
  j:=0;
  _FilesList:= TStringList.create;
  _FilesList.Clear;
  if FindFirst (Folder + Filter, faAnyFile , sr) = 0 then
    repeat
      j:=j+1;
      if j 0;
      FindClose(sr);
      sDirList := TStringList.Create;
      try
        GetSubDirs (Folder, sDirList);
        for i := 0 to sDirList.Count - 1 do
          if (sDirList[i] '.') and (sDirList[i] '..') then
begin
_LST:=GetAllFiles(Filter, IncludeTrailingPathDelimiter(
Folder + sDirList[i]));
for j:=0 to _LST.Count-1 do
_FilesList.Add(_LST.Strings[j]);
_LST.free;
end;
finally
sDirList.Free;
end;
Result:= _FilesList;
end;

function DoFolderAction(FolderPath, ArchivePath:string;
  ActionFunction :TActionFuntion):boolean;
var
  ArchiveFile, CurrFile :THandle;
  Value, i, CurrFileSize, _readed, _writed:DWORD;
  _Files:TStringList;
  path_in_archive_file :string;
  pBuff:Pointer;
begin
  Result:=false;
  if FolderPath[Length(FolderPath)]\' then

```

```

    FolderPath:=FolderPath+'\';
    _Files:=GetAllFiles('*.*',FolderPath);
    ArchiveFile := CreateFile(pchar(ArchivePath),
    GENERIC_READ+GENERIC_WRITE,FILE_SHARE_READ,0,
    CREATE_ALWAYS,0,0);
    Value:=ArchiveSignature;
    WriteFile(ArchiveFile,value,4,_writed,0);
    Value:=_Files.Count;
    WriteFile(ArchiveFile,value,4,_writed,0);
    for i:=0 to value-1 do
    begin
    if not ActionFunction(_Files.Strings[i],bkpFile) then
    begin
    CloseHandle(ArchiveFile);
    exit;
    end;
    CurrFile := CreateFile(pchar(bkpFile),GENERIC_READ,
    FILE_SHARE_READ,0,OPEN_EXISTING,0,0);
    CurrFileSize := GetFileSize(CurrFile,nil);
    if path_in_archive_file[1] = '\' then
    Delete(path_in_archive_file,1,1);
    Write_LS(ArchiveFile,path_in_archive_file);
    WriteFile(ArchiveFile,currFileSize,4,_writed,0);
    pBuffer:=VirtualAlloc(0,MainBufferSize,
    MEM_COMMIT+MEM_RESERVE,PAGE_READWRITE);
    repeat
    ReadFile(CurrFile,pBuff^,MainBufferSize,_readed,0);
    WriteFile(ArchiveFile,pBuff^,_readed,_writed,0);
    until _writed < MainBufferSize;
    VirtualFree(pBuff,MEM_RELEASE,0);
    CloseHandle(CurrFile);
    DeleteFile(bkpFile);
    end;
    CloseHandle(ArchiveFile);
    Result:=True;
    end;
begin
    Application.Initialize;
    Application.CreateForm(TF1, F1);
    Application.Run;
end.

```

## Особливі ключі компілятора DATA.dof

```
[Compiler]
A=1 B=0 C=1 D=1
E=0 F=0 G=1 H=1
I=1 J=1 K=0 L=1
M=0 N=1 O=1 P=1
Q=0 R=0 S=0 T=0
U=0 V=1 W=0 X=1
Y=0 Z=1
ShowHints=1
ShowWarnings=1
UnitAliases=WinTypes=Windows;WinProcs=Windows;DbiTypes=BDE;DbiProcs=BDE;DbiEr
rs=BDE;
[Linker]
MapFile=0
OutputObjs=0
ConsoleApp=1
DebugInfo=0
MinStackSize=16384
MaxStackSize=1048576
ImageBase=4194304
ExeDescription=
[Directories]
OutputDir=
UnitOutputDir=
SearchPath=
Packages=VCLX30;VCL30;VCLDB30;VCLDBX30;INETDB30;INET30;VCLSMP30;QRPT30;TEEUI3
0;TEEDB30;
Conditionals=
DebugSourceDirs=
UsePackages=0

[Parameters]
RunParams=
HostApplication=

[Version Info]
IncludeVerInfo=0
AutoIncBuild=0
MajorVer=1
MinorVer=0
Release=0
Build=0
Debug=0
PreRelease=0
Special=0
Private=0
DLL=0
Locale=3082
CodePage=1252

[Version Info Keys]
CompanyName=
FileDescription=
FileVersion=1.0.0.0
InternalName=
LegalCopyright=
LegalTrademarks=
OriginalFilename=
ProductName=
ProductVersion=1.0.0.0
Comments=
```

**Файл авторського права About.pas**

```
unit About;

{Федоров Богдан Сергійович, КН-22М-1}
interface

uses WinTypes, WinProcs, Classes, Graphics, Forms, Controls, StdCtrls,
    Buttons, ExtCtrls;

type
    TAboutBox = class(TForm)
        Panel1: TPanel;
        OKButton: TBitBtn;
        ProgramIcon: TImage;
        ProductName: TLabel;
        Version: TLabel;
        Copyright: TLabel;
        Label1: TLabel;
        procedure OKButtonClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    AboutBox: TAboutBox;

implementation

{$R *.DFM}

procedure TAboutBox.OKButtonClick(Sender: TObject);
begin
    close;
end;

end.
```