

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Сумський державний університет

Класичний фаховий коледж

(повна назва інституту/факультету)

(повна назва кафедри)

«До захисту допущено»

(підпис) (Ім'я та ПРІЗВИЩЕ)

20__ р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня бакалавр

(бакалавр / магістр)

зі спеціальності 171 Електроніка

(код та назва)

освітньо-професійної програми Електронні інформаційні системи

(освітньо-професійної / освітньо-наукової)

(назва програми)

на тему: **Перешкодозахищена цифрова система телеуправління**

Здобувача групи ЕІ-91к

(шифр групи)
батькові)

Максима Дзекунова

(прізвище, ім'я, по

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

(підпис)

(Ім'я та ПРІЗВИЩЕ здобувача)

Керівник ст. викладач. к.т.н.

(посада, науковий ступінь, вчене звання,

В.І. Васильєв

Ім'я та ПРІЗВИЩЕ)

(підпис)

Консультант¹⁾

(посада, науковий ступінь, вчене звання Ім'я та ПРІЗВИЩЕ)

(підпис)

Конотоп – 2023

Примітки:

1) Зазначається за наявності

ЗМІСТ

	С.
ВСТУП	3
РОЗДІЛ 1 аналого-цифрові системи контролю руху	5
1.1 Кодування в технічних системах обробки і передачі інформації	5
1.2 АЦПеретворення переміщення і похідних в послідовний код	8
1.3 Застосування рефлексного кодування. Код Грея	9
РОЗДІЛ 2 АЛГОРИТМИ ПЕРешКОДОЗАХИЩЕНОСТІ	16
2.1 Алгоритм коду Геммингу на прикладі роботи	16
2.2 Перетворення переміщення в послідовний код. Вимірювання швидкості ...	18
РОЗДІЛ 3 ТЕХНІЧНІ РІШЕННЯ СИСТЕМИ ТЕЛЕУПРАВЛІННЯ	26
3.1 Шифратор і дешифратор команд.....	26
3.2 Робота пристрою	29
ВИСНОВКИ	32
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	34

АНОТАЦІЯ

Об'єктом дослідження кваліфікаційної роботи є аналіз відомих сучасних технічних рішень з контролю і керуванню складними електромеханічними системами, а також методів удосконалення технічних характеристик перетворювачів координат руху об'єкта в зручну для обробки в інформаційно-вимірювальної системі.

Мета роботи полягає в дослідженнях рівня техніки за поставленою темою, способів визначення координат руху об'єктів, а саме: переміщення, швидкості, прискорення/уповільнення, ривку тощо. Досліджувались визначення основних параметрів руху наземних об'єктів, їх особливості, джерела похибок при проведенні контролю і способи запобігання від промислових і природних завад.

При виконанні роботи використовувалися методи математичного й цифрового й комп'ютерного моделювання процесів і систем.

Досліджувались методи керування і контролю руху складних електромеханічних систем. Зокрема механічні датчики прискорення (акселерометри) і сучасні методи непрямого визначення диференційно залежних параметрів руху від основної координаті для багатокординатного способу керування.

Робота викладена на 34 сторінках, у тому числі включає 11 рисунків, 3 таблиць, список цитованої літератури з 20 джерел.

КЛЮЧОВІ СЛОВА: ПЕРЕТВОРЮВАЧІ, СИСТЕМИ АВТОМАТИЧНОГО КЕРУВАННЯ (САК), ВИМІРИ КООРДИНАТ РУХУ, ДІАПАЗОН ВИМІРЮВАННЯ, ЧУТЛИВІСТЬ, СТАТИЧНА, ДИНАМІЧНА ПОХИБКА.

ВСТУП

Прогрес машинобудування передбачає удосконалення сучасних технічних систем, а також розробку нових технічних рішень спрямованих в бік підвищення їх продуктивності, керованості, швидкодії, високої точності, надійності та безпеки.

Сучасні вимоги до технічних систем передбачають впровадження поширеної автоматизації операцій що виконують складні технічні системи і комплекси. Автоматизовані технології передбачають використання замкнутих систем автоматичного керування де використовують датчики і вимірювальні пристрої для контролю параметрів руху систем.

В залежності від умов існують певні особливості. Наприклад, основними параметрами руху є переміщення об'єкту і деякі похідні, наприклад, швидкість, прискорення.

Будь-який вимірювальний пристрій складається з чутливого елемента, що сприймає фізичну величину, і перетворювача, який перетворює переміщення чутливого елемента в іншу форму, зручну для подальшого використання, найчастіше електричний сигнал.

В якості вимірювачів переміщень використовують цифрові кодові системи, швидкості – тахогенератори або перетворюють шляхом диференціювання сигнали переміщень, прискорення - акселерометри або також диференціювання, кутів і кутових швидкостей використовуються гіроскопічні пристрої та системи.

Сучасні інформаційно-вимірювальні, а також комп'ютеризовані системи для сполучень з технологічними об'єктами широко використовують промислові і комп'ютерні інтерфейси.

РОЗДІЛ 1

АНАЛОГО-ЦИФРОВІ СИСТЕМИ КОНТРОЛЮ РУХУ

1.1 Кодування в технічних системах обробки і передачі інформації

Коригуючий код (також перешкодостійкий код) - код, призначений для виявлення та виправлення помилок.

Основна техніка надмірність – додавання при записі (передачі) в корисні дані спеціальним чином структурованої надлишкової інформації (наприклад, контрольного числа), а при читанні (прийомі) використання такої надлишкової інформації для виявлення та виправлення помилки. Число помилок, яке можна виправити, обмежене і залежить від конкретного коду, що застосовується.

Коди виявлення помилок (які можуть встановити факт помилки) належать до тих же класів кодів, що і коди, що виправляють помилки. Фактично будь-який код, що виправляє помилки, може бути використаний для виявлення помилок (при цьому він буде здатний виявити більшу кількість помилок, ніж був здатний виправити). Коди, що виправляють помилки, застосовуються в системах цифрового зв'язку, у тому числі: супутниковому, радіорелейному, стільниковому, передачі даних по телефонних каналах, а також у системах зберігання інформації, у тому числі магнітних та оптичних. Коди, що виявляють помилки, використовуються в мережевих протоколах різних рівнів.

За способом роботи з даними коди, що виправляють помилки, діляться на **блокові**, що ділять інформацію на фрагменти постійної довжини і обробляють кожен з них окремо, і **згорткові**, що працюють з даними як безперервним потоком.

Блоковий код, що розбиває інформацію на фрагменти довжиною біт і перетворює їх на кодові слова довжиною біт зазвичай позначають ; у своїй число називається швидкістю коду. Якщо вихідні біт код залишає незмінними, і додає перевірючих, такий код називається систематичним, інакше несистематичним.

Задати блоковий код можна по-різному, зокрема таблицею, де кожній сукупності з інформаційних біт зіставляється біт кодового слова. Однак хороший код повинен задовольняти щонайменше наступним критеріям:

- здатність виправляти якомога більше помилок,
- якомога менша надмірність,
- простота кодування та декодування.

Наведені вимоги у випадку суперечать одне одному, тому існує велика кількість кодів, кожен із яких придатний певного кола завдань. Практично всі коди, що використовуються, є лінійними, це пов'язано з тим, що нелінійні коди значно складніше досліджувати, і для них важко забезпечити прийнятну легкість кодування і декодування.

Лінійний блоковий код - такий код, що безліч його кодових слів утворює n -мірний лінійний підпростір n -мірному лінійному просторі, ізоморфний простору n -бітних векторів.

Це означає, що операція кодування відповідає множенню вихідного-бітного вектора на невироджену матрицю, що називається матрицею, що породжує.

Для ортогонального по відношенню до підпростору та матриці, що задає базис цього підпростору, і для будь-якого вектора справедливо:

Мінімальна відстань та коригуюча здатність

Відстанню Геммінга (метрика Геммінга) між двома кодovими словами і називається кількість відмінних біт на відповідних позиціях:

Мінімальна відстань Гемінга є важливою характеристикою лінійного блокового коду. Вона показує, наскільки "далеко" розташовані коди один від одного. Вона визначає іншу, не менш важливу характеристику — здатність, що коригує:

Коригуюча здатність визначає, скільки помилок передачі коду (типу) можна гарантовано виправити. Тобто навколо кожного кодового слова маємо околицю, яка складається з усіх можливих варіантів передачі кодового слова з числом помилок (t) не більше. Ніякі дві околиці двох будь-яких кодових слів не перетинаються один з одним, тому що відстань між кодovими словами (тобто центрами цих околиць) завжди більше двох їх радіусів.

Таким чином, отримавши спотворену кодову комбінацію з декодер приймає рішення, що вихідною була кодова комбінація , виправляючи тим самим не більше помилок.

Наприклад, за наявності всього двох кодових слів і з відстанню Геммінга між ними, що дорівнює 3, помилка в одному биті слова може бути виправлена, так як навіть у цьому випадку прийняте слово ближче до кодового слова, ніж до . Але якщо каналом були внесені помилки у двох бітах (у яких відрізнялося від), то результат помилкової передачі виявиться ближче до , ніж , і декодер ухвалить рішення, що передавалося слово .

Коди Геммінга - найпростіші лінійні коди з мінімальною відстанню 3, тобто здатні виправити одну помилку. Код Геммінга може бути представлений у такому вигляді, що синдром:

де — прийнятий вектор, дорівнюватиме номер позиції, в якій відбулася помилка. Ця властивість дозволяє зробити декодування дуже простим.

Загальний метод декодування лінійних кодів

Будь-який код (у тому числі нелінійний) можна декодувати за допомогою звичайної таблиці, де кожному значення прийнятого слова відповідає найбільш ймовірне передане слово. Однак даний метод вимагає застосування величезних таблиць для кодових слів порівняно невеликої довжини.

Для лінійних кодів цей спосіб можна значно спростити.

Лінійні циклічні коди

Циклічним кодом є лінійний код, що має таку властивість: якщо є кодовим словом, то його циклічна перестановка також є кодовим словом.

Слова циклічного коду зручно представляти як многочленов. Наприклад, кодове слово представляється як полінома. При цьому циклічний зсув кодового слова еквівалентний множенню многочлена на модуль.

Можна показати, що всі кодові слова конкретного циклічного коду кратні певному багаточлену, що породжує (генераторному). Багаточлен, що породжує, є дільником.

За допомогою багаточлена, що породжує, здійснюється кодування циклічним кодом. Зокрема:

- Несистематичне кодування здійснюється шляхом множення кодованого вектора на: ;
- Систематичне кодування здійснюється шляхом «дописування» до кодованого слова залишку від розподілу на , тобто.

1.2. АЦПеретворення переміщення і похідних в послідовний код.

При побудові цифрових перетворювачів переміщення в цифровий код використовуються тільки метод зчитування і метод послідовного розрахунку [3, 7]. Слід відмітити, що при побудові таких АЦП використовують різні фізичні явища, чим обумовлюється різноманітність елементів, використовуваних в перетворювачах. В більшості випадків кодується кутове переміщення, тому в багатьох випадках лінійні переміщення за допомогою різних приладів перетворюються в кутові. Тому в подальшому ми розглянемо тільки перетворювачі кутових переміщень.

Для виконання методу зчитування необхідно мати $2^n - 1$ еталонів з якими виконується порівняння перетвореної величини і стільки ж схем порівняння. Практично створюють кодові шкали, кількість яких рівна числу розрядів коду, а число інтервалів між поділками окремих шкал визначається як 2^i , де i – номер даної шкали, починаючи з нуля. Мінімальна ціна поділки при цьому буде визначатися кроком квантування.

Таким чином в складі перетворювача повинні бути як мінімум кодові шкали і пристрої порівняння. Реально в склад такого АЦП входить ряд пристроїв підсилення і перетворення сигналів, схеми зчитування і зберігання кодів.

Принцип дії перетворювачів кутового переміщення в код можна розглянути на прикладі перетворювача з контактним зніманням кодового сигналу. Розрізняють перетворювачі з натискним і ковзаючими контактами. Кодові шкали можуть бути дискові та барабанні. На рисунку 1.1 і 1.2 представлені кодуєчий диск і кодуєчий барабан чотирьох розрядного перетворювача.

На кодуєчому диску шкали представляють собою концентричні кола, які складаються з почергових провідних і невідних площадок. Кількість кругів рівне числу розрядів коду. Розміри площадок в кожному кільці однакові, а їх число визначається позицією даного розряду в кодї. Всї прохідні площадки електрично з'єднані між собою і спеціальним кільцем, яке являється струмопровідним елементом.

В перетворювачі з кодуєчим барабаном [7] шкали виготовляються у вигляді кільця, розташованих на поверхні барабану і складаються з почергових провідних і невідних ділянок, або фото- магніточутливих сенсорів (рис 1.2).

В якості чутливих елементів використовуються, наприклад, щітки – струмознімачі 1, які знаходяться над розрядними кільцями диску або барабана. При потраплянні щіток на провідні ділянки в відповідних розрядах формуються одиниці. Якщо щітка знаходиться над невідною ділянкою в даному розряді буде нуль.

1.3 Застосування рефлексного кодування. Код Грея

При використанні звичайних кодових шкал з'являється ймовірність виникнення хибних кодів при переході від одного числа до другого. Для запобігання цього явища необхідно щоб границя між числами була прямою лінією а струмознімачі знаходились точно вздовж цієї лінії [3]. В реальних умовах цього не можливо досягти в наслідок технологічних неточностей вставнолення струмознімачів та неточних розмірів диска чи барабану. Внаслідок цього всі струмоприймачі не будуть переходити бар'єр між числами.

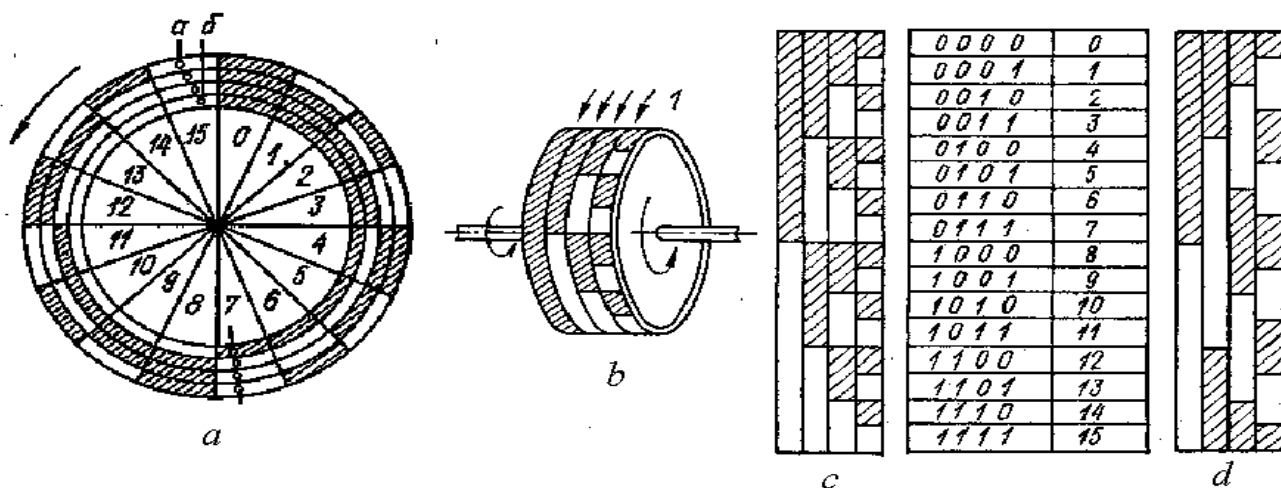


Рис 1.1 Кодуючий диск, двійковий і рефлексний код

Рефлексні коди (відбиті) є символічними кодами. Їх особливість полягає в тому, що два поряд знаки, що стоять, відрізняються тільки одним зміненим бітом, а знаки є взаємно симетричними (відбитими) до певних осей в одному, двох і більше бітах знака. Рефлексний код показаний у табл. 3.4 та на рис. 1.1.

Його можна отримати з натурального двійкового коду за певними правилами. Біти, яким передують 0, зберігаються; біти, яким передують 1, змінюються на зворотний сигнал.

При переході від рефлексного коду до натурального двійкового число попередніх одиниць, включаючи біт, що оцінюється, складається. Якщо результат непарний, на біті, що оцінюється, буде 1; якщо результат парний, на біті, що вбачається, буде 0.

Безпечні коди використовують надмірну інформацію контролю наданої інформації. Одним із безпечних кодів є код «2 з 5». Це двійковий код із 5 бітами в одному знаку. Загалом можна скласти $2^5 = 32$ знаки. Надлишковими є наступні знаки, що містять менше або більше 2 двійкових одиниць. При помилковому зчитуванні з'явиться більше або менше двійкових одиниць і відповідна система виявить помилку.

Подібним кодом є код «5 з 10». Іншим безпечним кодом є код "2,7". Кодовий знак заданий сімома бітами у двох групах: в одній групі є 2 біти, в іншій - 5. Тому цей код часто позначається як біквінарний. У кожній групі є по одній одиниці. Якщо їх більше, то лічені дані помилкові.

Коди "2 з 5", "5 з 10", "2,7" можуть виявити, але не усунути помилки.

Помилки в цьому знаку можуть бути знайдені та виправлені так званими коригуючими кодами. Це складні коди, в яких надмірність збільшена тим, що на знак вводяться додаткові допоміжні біти.

Нехай диск (рис 1.1) рухається в напрямку вказаному стрілкою, а струмознімачі розположені так, що їх крайні положення відповідають позиціям «а» і «б». В кінцевому стані всі струмознімачі знаходяться на провідних ділянках і вихідний код буде дорівнювати 1111. Наступне значення коду повинно бути 0000, але по мірі переміщення диску струмознімачі перейдуть границю між числами неодноразомно. Помилка при цьому може становити до 50%. Щоб цього уникнути використовують

спеціальні шкали дискретизації. В багатьох випадках використовують спеціальні види кодів, відрізняючись тим, що при переході від одного числа до другого змінюється тільки одиниця в одному розряді числа. До числа таких кодів відносяться так звані рефлексні коди з яких в перетворювачах кутових переміщень використовується рефлексний двійковий код, часто названий кодом Грея [1].

При виборі типу коду для перетворення переміщення в цифрову форму більш зручним є код Грея. Перевага коду Грея, якій також є позиційним, над звичайним двійковим в тому, що зміна контролюємого переміщення на одиницю завжди відповідає зміні кодової комбінації тільки в одному розряді. Код Грея будується на базі двійкового за наступним алгоритмом: старший розряд залишається без зміни; кожний наступний розряд інвертується, якщо попередній розряд вихідного двійкового коду дорівнює одиниці. Цей алгоритм побудови може бути формально представлений як результат складання по модулю 2 вихідних комбінації двійкового коду з такою ж комбінацією, але зрушеною на один розряд вправо. При цьому крайній правий розряд зрушеної комбінації відкидається.

Таким чином, код Грея є так званим однокроковим кодом [1], тому що при переході від одного числа до іншого завжди змінюється лише якийсь один біт. Похибка при зчитуванні інформації з механічного кодового диска при переході від одного числа до іншого призведе лише до того, що перехід від одного положення до іншого буде лише кілька зміщений за часом, проте видача абсолютно невірною значення кутового положення при переході від одного положення до іншого повністю виключається. Перевагою коду Грея є також його здатність дзеркального відображення інформації. Так, інвертувавши старший біт можна простим чином змінювати напрямок рахунку і, таким чином, підбиратись до фактичного (фізичного) напрямку обертання осі. Зміна напрямку рахунку може легко змінюватися, керуючи так званим входом "Complement". Значення що видається може бути зростаючим або спадаючим при одному і тому ж фізичному напрямку обертання осі.

Оскільки інформація в коді Грея, має чисто кодований характер що не несе реальної числової інформації, повинен він перед подальшою обробкою спершу перетворений в стандартний бінарний код. Здійснюється це за допомогою

перетворювача коду (декодера Грей-Бинар), який на щастя легко реалізується за допомогою ланцюга з логічних елементів «виключає або» як програмним, так і апаратним способом .

З таблиці 1 видно, що при переході від одного числа до іншого (сусіднього) лише один біт інформації змінює свій стан, якщо число представлено кодом Грея, в той час, як в двійковому коді можуть поміняти свій стан кілька біт одночасно. Код Грея - вихід, отже, він ніколи не має помилку зчитування і застосовується в багатьох абсолютних енкодерах.

Таблиця 1.1. Зміни біт в коді Грея

Десятковий код	Двійковий код				Код Грея			
	2^3	2^2	2^1	2^0				
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	0
5	0	1	0	1	0	1	1	1
6	0	1	1	0	0	1	0	1
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	1	0	0
9	1	0	0	1	1	1	0	1
10	1	0	1	0	1	1	1	1
11	1	0	1	1	1	1	1	0
12	1	1	0	0	1	0	1	0
13	1	1	0	1	1	0	1	1
14	1	1	1	0	1	0	0	1
15	1	1	1	1	1	0	0	0

Біти які змінюють своє значення, при переході від одного числа до іншого, позначені жирним шрифтом.

Видно на рис 1.2в, що на одній кодовій шкалі, побудованій за допомогою коду Грея, відсутні переходи, які характеризуються одночасною зміною одиниці в кількох розрядах. Крім того розміри елемента шкали молодшого розряду вдвічі більші, що дозволяє при однакових розмірах шкали реалізувати вдвічі більшу роздільну здатність.

Після зчитування код Грея перетворюється в звичайний двійковий код за допомогою спеціальних пристроїв. Фрагмент принципової схеми перетворювача зображено на рис. 1.3. Вона складається з елементів пам'яті на тригерах $T_{21} - T_{2n}$ T-типу і неповних суматорів виконаних на елементах “виключне АБО” (“складання за модулем 2”). Тригери $T_{21} - T_{2n}$ призначені для визначення і зберігання значень відповідних розрядів коду Грея. Суматори C_m виконують арифметичне сумування по модулю 2. На виходах відповідних суматорів формуються розряди двійкового коду.

Використання рефлексного коду не є єдиним способом усунення неоднозначності при кодуванні кутових переміщень. Відомі також інші методи, серед яких слід відмітити метод Баркера, який заключається у виборі елементів знімання на основі аналізу коду молодшого розряду, метод подвійної щітки та ряду інших.

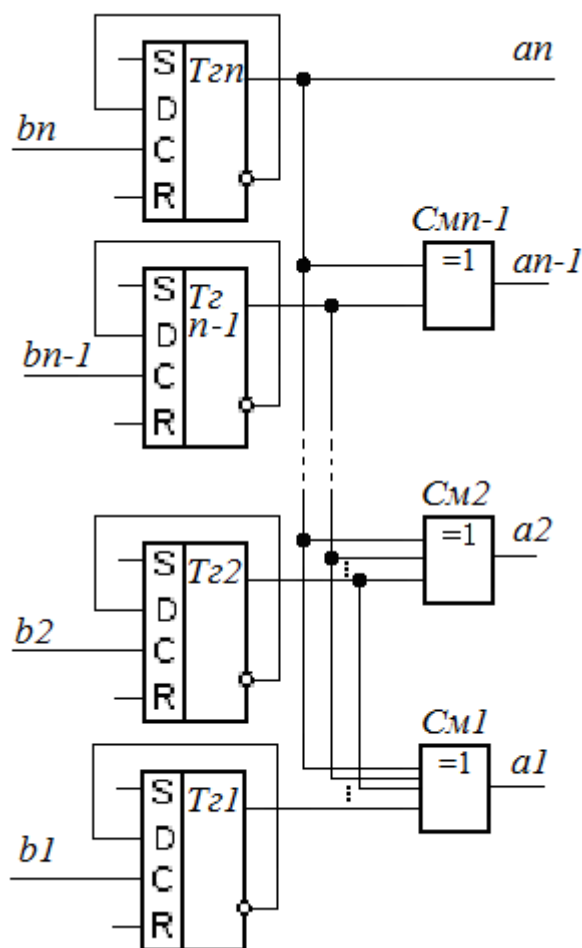


Рис 1.3 Цифрова електрична принципова схема апаратного перетворення коду Грея в двійковий код

Перетворення рефлексного коду на позиційний двійковий програмним шляхом

Дуже рідко може виникнути необхідність перетворення чисел, представлених у рефлексному коді (коді Грея) в позиційний код. У програмах на основі мікроконтролерів дані в коді Грея можуть надходити від деяких датчиків старого типу.

Для перетворення числа V , представленого в позиційному двійковому коді, число G записане в двійковому рефлексному коді необхідно провести операцію "що виключає АБО" між самим числом V і приватним, отриманим від поділу $V/2$:

$$G = V \wedge (V/2) = b_{n-1} * 2^{n-1} + (b_{n-2} \wedge b_{n-1}) * 2^{n-2} + \dots + (b_2 \wedge b_1) * 2^1 + b_1 \wedge b_0$$

Перетворення з рефлексного коду на позиційний (зводиться до послідовного визначення розрядів b_n двійкового числа). Як видно, старший коефіцієнт b_{n-1} у позиційному двійковому та рефлексному кодах завжди збігаються. Для визначення b_{n-2} необхідно зробити бітову операцію $b_{n-1} \wedge (b_{n-2} \wedge b_{n-1})$. Так само за відомим b_{n-2} визначається $b_{n-3} = b_{n-2} \wedge (b_{n-3} \wedge b_{n-2})$ і т.д. Підпрограма перетворення 16-розрядного числа в коді Грея на його двійковий еквівалент:

```

1   ; R17:R16 <- R17:R16
2   ; R17:R16 - число в рефлексном коде при входе
3   ; и в двоичном позиционном коде при выходе
4   ; R18,R19 - вспомогательные регистры
5
6   gr16_pos16:
7       ldi R18,0x80
8       ldi R19,16 ;инициализируем счётчик сдвигов
9       clc      ;обнуляем флаг C при входе в цикл
10  gr1:  rol R16      ;сдвигаем число R17:R16 на 1 разряд влево
11       rol R17
12       brcc PC+2    ;находим очередной разряд bn
13       eor  R17,R18
14       dec R19
15       brne gr1     ;повторяем операцию n раз
16       rol R16
17       rol R17
18       ret

```

РОЗДІЛ 2.

АЛГОРИТМИ ПЕРЕШКОДОЗАХИЩЕНОСТІ

2.1 Алгоритм коду Геммінгу на прикладі роботи [2,3,4]

Коди Хеммінга - найбільш відомі і, ймовірно, перші з кодів, що самоконтролюються і самокоректуються. Побудовані вони стосовно двійковій системі числення.

Іншими словами, це алгоритм, який дозволяє закодувати будь-яке інформаційне повідомлення певним чином і після передачі (наприклад по мережі) визначити чи з'явилася якась помилка в цьому повідомленні (наприклад через перешкоди) і, при можливості, відновити це повідомлення. Сьогодні я опишу найпростіший алгоритм Геммінгу, який може виправляти лише одну помилку.

Також варто відзначити, що існують досконаліші модифікації даного алгоритму, які дозволяють виявляти (і якщо можна виправляти) більшу кількість помилок.

Відразу варто сказати, що Код Геммінга складається із двох частин. Перша частина кодує вихідне повідомлення, вставляючи до нього певних місцях контрольні біти (обчислені особливим чином). Друга частина отримує вхідне повідомлення і заново обчислює контрольні біти (за тим самим алгоритмом, що перша частина). Якщо всі знову обчислені контрольні біти збігаються з отриманими, повідомлення отримано без помилок. В іншому випадку, виводиться повідомлення про помилку та при можливості помилка виправляється.

Щоб зрозуміти роботу даного алгоритму, розглянемо приклад [2,3].

Припустимо, повідомлення «habr», яке необхідно передати без помилок. Для цього спочатку потрібно наше повідомлення закодувати за допомогою коду Геммінга. Нам необхідно уявити його в бінарному вигляді.

Символ	ASCII код	Бінарне представлення
h	68	01000100
a	61	00111101
b	62	00111110
r	72	01001000

На цьому етапі варто визначитися з так званою довжиною інформаційного слова, тобто довжиною рядка з нулів та одиниць, які ми кодуватимемо. Припустимо, у нас довжина слова дорівнюватиме 16. Таким чином, нам необхідно розділити наше вихідне повідомлення («habr») на блоки по 16 біт, які ми потім кодуватимемо окремо один від одного. Так як один символ займає в пам'яті 8 біт, то в одне слово, що кодується, міститься рівно два ASCII символи. Отже, ми отримали два бінарні рядки по 16 біт:

h	a	i	h	a
01000100	00111101		00111110	01001000

Після цього процес кодування розпаралелюється, і дві частини повідомлення (ha і br) кодуються незалежно один від одного. Розглянемо, як це робиться на прикладі першої частини.

Насамперед, необхідно вставити контрольні біти. Вони вставляються в строго певних місцях - це позиції з номерами, рівними ступеням двійки. У нашому випадку (при довжині інформаційного слова 16 біт) це будуть позиції 1, 2, 4, 8, 16. Відповідно, у нас вийшло 5 контрольних біт (виділені):

Було:		Стало:	
h	a	h	a
01000100	00111101	0000 1000 0100	001 0 11101

Таким чином, довжина повідомлення збільшилася на 5 біт. До обчислення самих контрольних біт, ми надали їм значення «0».

Обчислення контрольних біт.

Тепер необхідно визначити значення кожного контрольного біта. Значення кожного контрольного біта залежить від значень інформаційних біт (як несподівано), але з усіх, лише від тих, які цей контрольний біт контролює. Для того, щоб зрозуміти, за які біти відповідає кожний контрольний біт, необхідно зрозуміти дуже просту закономірність: контрольний біт з номером N контролює всі наступні N біт через кожні N біт, починаючи з позиції N . Не дуже зрозуміло, але по картинці, думаю, стане ясніше:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	1	1	1	0	1	
x		x		x		x		x		x		x		x		x		x		x	1
	x	x			x	x			x	x			x	x			x	x			2
			x	x	x	x					x	x	x	x					x	x	4
							x	x	x	x	x	x	x	x							8
															x	x	x	x	x	x	16

Тут знаком «X» позначені ті біти, які контролює контрольний біт, номер якого праворуч. Тобто, наприклад, біт номер 12 контролюється бітами з номерами 4 і 8. Зрозуміло, щоб дізнатися якими бітами контролюється біт з номером N треба просто розкласти N за ступенями двійки.

Але як визначити значення кожного контрольного біта? Робиться це дуже просто: беремо кожен контрольний біт і дивимося скільки серед контрольованих ним біт одиниць, отримуємо деяке ціле число i , якщо воно парне, то ставимо нуль, інакше ставимо одиницю. От і все! Можна звісно й навпаки, якщо число парне, то ставимо одиницю, інакше, ставимо 0. Головне, щоб у «кодуєчій» і «декодуєчій» частинах алгоритм був однаковий. (застосуємо перший варіант).

Вирахувавши контрольні біти для нашого інформаційного слова отримуємо

наступне:		для другої частини:	
h	a	h	a
100110000100	001011101	100101101110	010101000

Декодування і виправлення помилок.

Тепер, якщо, наприклад, на іншому пристрої отримане закодоване першою частиною алгоритму повідомлення, але воно прийшло з помилкою. Наприклад, ми отримали що 11-ий біт передався неправильно):

h	a
100110000170	001011101

Вся друга частина алгоритму полягає в тому, що необхідно заново обчислити всі контрольні біти (як і в першій частині) і порівняти їх з контрольними бітами, які ми отримали. Так, поразивши контрольні біти з неправильним 11-м бітом ми отримаємо таку картину:

h	a
010110010170	001011101

Як бачимо, контрольні біти під номерами: 1, 2, 8 не збігаються з тими самими контрольними бітами, які ми отримали. Тепер просто склавши номери позицій неправильних контрольних біт ($1+2+8=11$) ми отримуємо позицію помилкового біта. Тепер просто інвертувавши його та відкинувши контрольні біти, ми отримаємо вихідне повідомлення у первозданному вигляді! Абсолютно аналогічно надаємо з другою частиною повідомлення.

У цьому прикладі, розглянуто повідомлення довжиною 16 біт, оскільки у мікроконтроллерной техніці вони досить популярні. Тобто 16 біт найоптимальніше для розгляду приклада. Але треба враховувати те що у цій простій версії алгоритму одне інформаційне слово можна виправити лише одну помилку.

2.2 Перетворення переміщення в послідовний код. Вимірювання швидкості

В якості датчика швидкості приймемо фотоімпульсний датчик, який на метр переміщення на виході формує 1000 імпульсів. [8] Серед промислових датчиків близьки характеристики має фотоімпульсний датчик ПДФ-3, де на оборот диску формується дві серії по 600 імпульсів, які зсунуті на 90 електричних градусів. Тому, використовуючі обидві серії можна отримати від 1200 або 2400 при використанні переднього та заднього фронту імпульсу. Відлік імпульсів датчика починається по передньому фронту сигналу, що управляє електронним ключем (ЕК), через який

надходить на лічильник контрольована послідовність. При закінченні сигналу, що управляє ЕК, тобто надходженні заднього фронту імпульсу або при перевищенні заданого інтервалу часу відлік зупиняється й виводиться на цифрове табло. При відхиленні результуючого значення коду лічильника від заданої межі на панелі відображення виводиться сигнал "відхилення параметру". У іншому випадку на панелі відображається дешифрований стан лічильника (або його розрядів) і величина обмірюваного значення послідовним кодом передається в пристрій обробки через струмову петлю.

Таким чином структурну схему складатимуть вузли (елементи):

1. Пристрій синхронізації й керування
2. Формувачі сигналів (гальванічні розв'язки і детектори фронтів)
3. Схема контролю параметрів чисельно-імпульсної послідовності (вузол лічильника)
4. Пристрій перетворення й відображення інформації (вузол індикації)
5. Пристрій перетворення паралельного коду в послідовний
6. Пристрій передачі сигналу через струмову петлю (лінійний підсилювач)

Функціональну схему пристрою виміру швидкості [8] представлено на рис 2.1. Основна частина складається з фільтра (Φ) електронного ключа (ЕК), лічильника й регістра (ЛІЧ), пристрою порівняння кодів (ППК), генератору тактових імпульсів (ГТІ), розподільнику імпульсів (РІ), оптоелектронного перетворювача (ОЕП) і лінійного підсилювача (ЛП). Електронний ключ виконаний на логічному елементі "Т" й керується сигналом дозволу рахування Т, що формується вузлом керування (ВК).

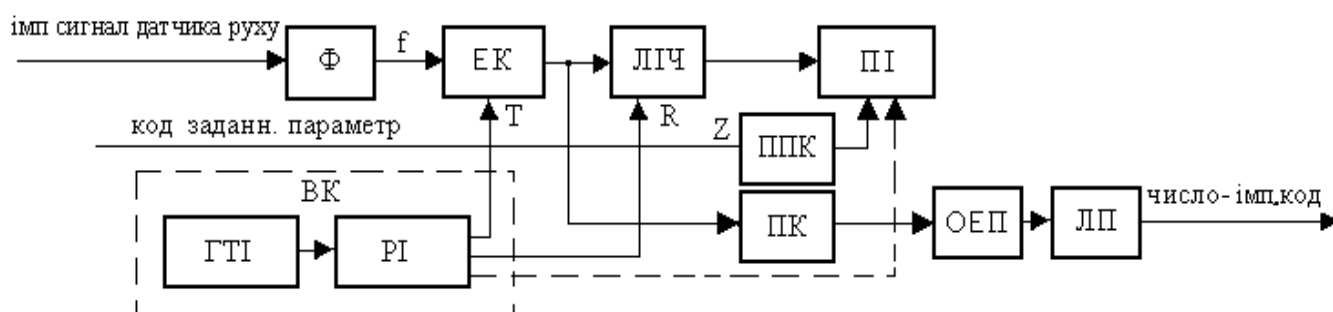


Рис. 2.1. Функціональна схема вимірювача швидкості

Промисловий датчик переміщення дискретний фотоелектричний (ПДФ-3 У2)

Загальні відомості. Датчик призначений для перетворення переміщення (кута повороту) робочих органів промислових механізмів в число імпульсів і кутової швидкості в частоту проходження імпульсів.

Технічні характеристики. Номінальна напруга живлення, 24В. Напруга вихідних сигналів, В - сигнал "1" – 24, сигнал "0" - 0,5

Максимальна частота обертання вхідного вала, об/хв 3000.

Число імпульсів на оборот вала 600.

Струм, мА не більше 300.

Габаритні розміри, мм 150 * 160 * 172.

Маса, кг не більше 4

Блок вимірювання швидкості веде рахунок кількості періодів вхідного сигналу в інтервалі часу за заданою дискретністю. На підставі початкових даних визначимо необхідну розрядність лічильника, а також визначимося з елементною базою. Найбільшу швидкість, що вимірюється беремо із запасом – 1.5 м/с. Точність вимірювань – 0.001 м/с. Таким чином необхідна розрядність лічильника визначається із співвідношення: $XXXX/Y = ZZZZ_{10} = (XXXX\ XXXX\ XXXX\ XXXX)_2$

Тобто для забезпечення виконання заданих умов лічильник повинен складатися або з n корпусів двійково-десяткових лічильників (наприклад, К561ИЕ14), або з n-1 корпусів двійкових лічильників (наприклад, К561ИЕ11).

Виходячи із цього обираємо варіант побудови схеми на підставі двійково-десяткових лічильників з попереднім встановленням (паралельним записом). Властивості паралельного запису будуть використатися для регістра зберігання інформації й безперервного виводу її на індикацію. Це дозволить більш зручне зчитування інформації, при постійному відновленні її після чергового виміру. В якості елементної бази доцільно скористатись цифровими мікросхемами КМОП-технології, які характеризуються мікро потужним споживанням і відзначаються високою надійністю роботи в пристроях промислової автоматики. На підставі функціональної схеми і з урахуванням розрахунку кількості цифрових елементів типова принципова схема блоку вимірювача швидкості [8] представлена фрагментом на рисунку 2.2.

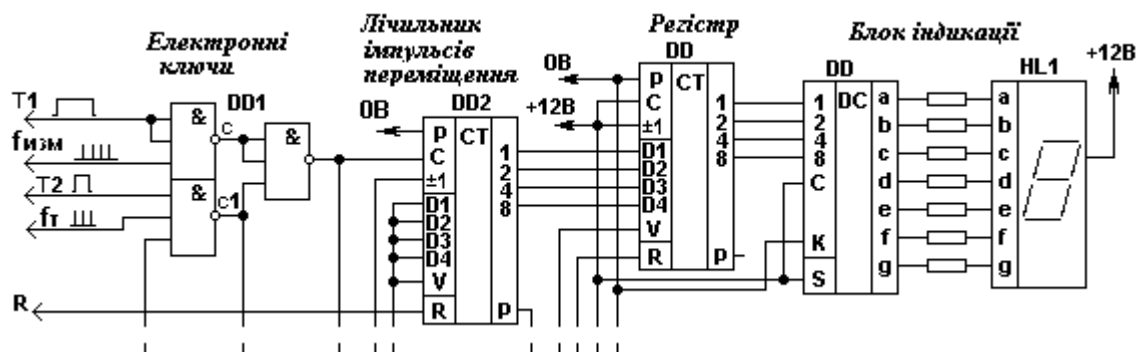


Рис.2.2. Фрагмент принципової електричної схеми цифрового блоку перетворення послідовного коду в паралельний для вимірювача швидкості.

Сучасні цифрові технології для оптимізації технічних рішень застосовують різні способи мінімізації елементної бази і зв'язків. Одним із принципів мінімізації є спосіб оснований на принципі розподілу обробки інформації в часі. Для створення принципу розподілу в часі роботи цифрових елементів: лічильників, регістрів, перетворювача паралельного коду в послідовний і засобів індикації необхідно щоб блок управління і синхронізації забезпечував [8] наступну часову діаграму сигналів, що представлено на рис. 2.3.

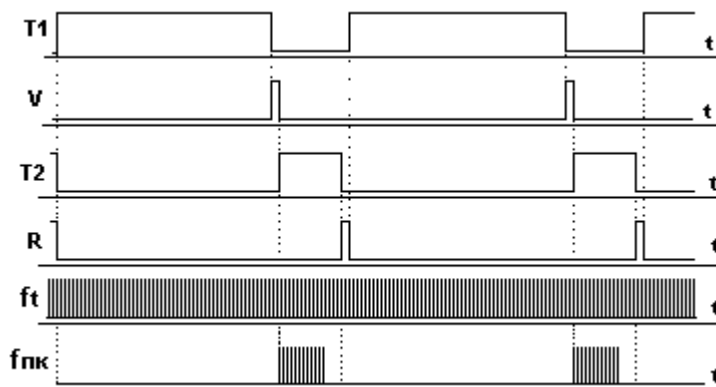


Рис. 2.3. Часові діаграми роботи лічильника-вимірювача та блоку управління і синхронізації

При визначенні тривалості відповідних імпульсів користуються критеріями заданої точності вимірювання ($T1$), часом повного заповнення ємності лічильника тактовими імпульсами ($T2$). Інші тривалості імпульсів (V , R) повинні відповідати паспортним даним мікросхем, що прийняти в якості елементної бази.

На діаграмах прийняти такі умовні позначки: T_1 – імпульси з часом вимірювання частоти; V – імпульси запису одержаної інформації в регістр; T_2 – імпульс часу перетворення паралельного коду в послідовний; R – сигнали встановлення лічильника в нульовий стан; f_t – імпульси генератора тактових імпульсів для передачі послідовним кодом; $f_{пк}$ – сигнали послідовного коду, що передаються по струмової петлі.

В залежності від конкретних умов і параметрів діапазону величин, що вимірюються уточнюються зазначені вище параметри тривалостей сигналів керування цифровими схемами, розраховуються стали часу RC-ланцюгів таймеру-генератору та ін. пристроїв.

2.3 Визначення похідних основної координати руху за допомогою нелінійного фільтра

При розробці високоякісних оптимальних за динамічністю сучасних систем виникає необхідність визначення і використання крім основної координати руху двох найближчих похідних. Це стосується способу багато-координатного керування складними за динамікою системами. Наприклад, шлях–швидкість–прискорення, або швидкість–уповільнення–ривок і т.п. де важливим параметром є уповільнення/прискорення, знак і величина якого несе інформацію про динамічні навантаження в системі і про ресурси безпеки системи. При вимірі цього параметру надо мати на увазі що цей параметр є похідної від швидкості і другої по відношенню до переміщення. Для вирішення даної технічної задачі існують електромеханічні датчики – акселерометри за допомогою яких в авіаційно-ракетних системах контролюють перевантаження.

Пристрій та принцип дії датчика лінійних прискорень розглянемо на прикладі однокомпонентного акселерометра з пружинним підвісом чутливого елемента (сейсмічної маси). Функціональна схема акселерометра представлена на рис. 2.4. На рисунку позначени складові перетворювача: чутливий елемент 3 підвішений в корпусі приладу 1 за допомогою пружин 2. Для демпфування власних коливань чутливого елемента використовується повітряний демпфер 4. Вихідний сигнал, пропорційний виміряному прискоренню, знімається з потенціометричного перетворювача 5. Осі

чутливості акселерометра є вісь OX . На тілі що рушиться вісь чутливості встановлюється паралельно до тієї осі тіла, щодо якої необхідно вимірювати прискорення. Основний недолік такого пристрою для керування по прискоренню є інерційністю і пов'язана з нею точність.

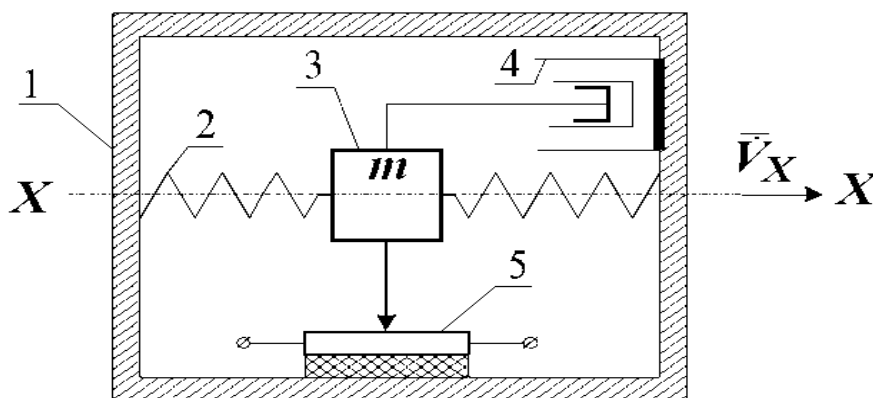


Рис 2.4 Функціональна схема акселерометра лінійних прискорень (уповільнень)
 1 – корпус; 2 – пружини підвісу; 3 - чутливий елемент - сейсмічна маса ;
 4 – повітряний демпфер; 5 – потенціометричний перетворювач.

Принцип дії акселерометра заснований практично на вимірі сили інерції, що розвивається сейсмічною масою під час її руху з прискоренням. Розглянемо сили, які прикладаються до сейсмічної маси під час руху тіла з прискоренням.

Визначення уповільнення шляхом диференціювання буде незадовільною, тому контроль даного параметру більш доцільно вимірювати на підставі нелінійних фільтрів, або взагалі комп'ютерними програмними алгоритмами, що реалізують модель, наприклад нелінійного фільтру другого порядку з використанням обмежень високих похідних узгодженими з динамікою приводу, або джерелом енергії. Тобто за допомогою нелінійного фільтру можна змоделювати умови за якими подавши на вхід НФ функцію сигналу на виходах отримаємо його похідні функції.

Пристрій, що виконує функції відображення одночасно декілька динамічних координат, наприклад переміщення, швидкість і прискорення/уповільнення можна реалізувати на підставі технічних рішень [12 - 14] представлено структурною схемою нелінійного фільтру другого порядку на рисунку 2.5. Цей пристрій з функцією синтезу

оптимальних за динамічністю керувань і визначення похідних основної функції руху. Перевагою його є менша інерційність і можливість використання для оптимального керування. Наявність нелінійного елемента з обмеженнями знижує вплив на пристрої перешкод. На його основі можна зробити вимірювач з аналоговим представленням інформації, або в табличній формі при роботі в комп'ютеризованих інформаційно-вимірювальних системах, принципи роботи яких описані в розділі 3 даної роботи.

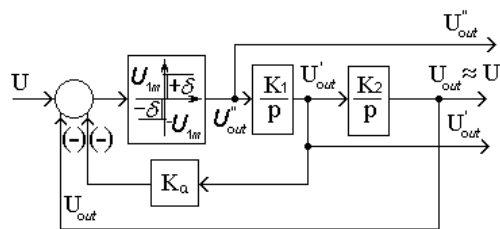


Рис. 2.5. Структурна схема пристрою вимірювача-індикатора похідних від основної функції руху.

РОЗДІЛ 3

ТЕХНІЧНІ РІШЕННЯ СИСТЕМИ ТЕЛЕУПРАВЛІННЯ

3.1. Шифратор і дешифратор команд

В наш час бурхливого розвитку безпілотних летальних апаратів (БПЛА) з'явилося багато технічних рішень вузлів дискретного радіо керування моделями [1, 2], що використовують різні способи кодування команд. Як правило кількість керуючих повідомлень на БПЛА обмежено, що спрощує вирішення проблем перешкодозахисту. Найбільш прийнятним для багатьох практичних випадків є цифровий спосіб. Проте і такі системи мають недостатню захищеність від імпульсних перешкод. Розглянемо принцип перешкодо- завадозахищеного керування системою з обмеженою кількістю команд до, 5-10, апаратними засобами. Як, наприклад у БПЛА.

Як відомо, джерелом імпульсних перешкод можуть бути не тільки грозові розряди, а й виконавчі двигуни моделі, а також різна апаратура, яка використовується в народному господарстві та медицині та працює на частотах, близьких до тих, що застосовують для телеуправління. Ці перешкоди, потрапляючи на вхід дешифратора, створюють з його виході хибний сигнал, і модель виконує хибну команду. Розглянута нижче система управління має підвищену захищеність від імпульсних перешкод завдяки особливій побудові дешифратора. Але використання контактних комутаційних елементів знижає її надійність. Розглянемо технічне рішення боротьби з перешкодами. У ній використано число-імпульсний принцип подачі команд.

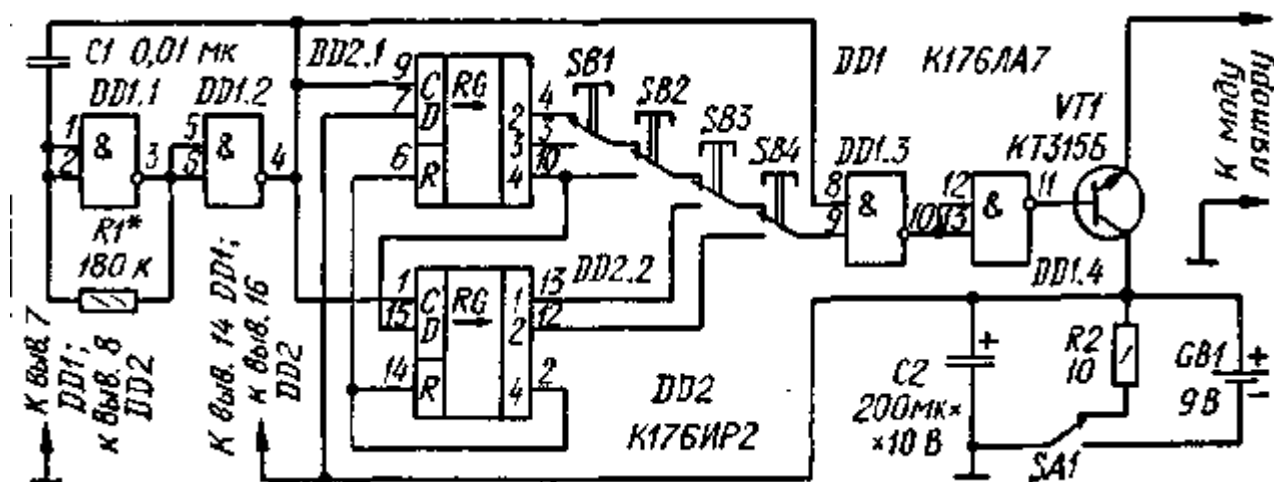


Рис. 3.1. Функціональна схема шифратора команд

На логічних елементах D01.1 и DD1.2 собрано тактовий генератор. Його частота залежить опір резистору R1 і ємності конденсатора C1. Вузол DD2.1, DD2.2 — восьмирозрядний зсувний реєстр. Транзистор — VT1 електронний ключ.

Розглянемо процес формування груп імпульсів з прикладу команди «Стоп». При подачі напруги живлення на шифратор тактовий генератор виробляє послідовність прямокутних імпульсів із частотою 200 Гц та шпаруватістю, що дорівнює двом (рис. 2, а). Ці імпульси одночасно надходять на лічильний вхід реєстр DD2.1 і DD2.2 і верхній за схемою вхід елемента DD1.3. Якщо командні кнопки SB1—SB4 перебувають у положенні, показаному на схемі, то нижньому вході цього елемента з'являтимуться імпульси тривалістю 30 мс (рис. 2, б). На виході інвертора DD1.4 буде сформовано групи імпульсів, розділені паузою (рис. 2, в). На час дії імпульсу транзистор VT) відкривається, і напруга джерела живлення GB1 надходить на модулятор передавача.

При вимиканні живлення перемикачем SA1 конденсатор C2 через резистор R2 швидко розряджається. Якщо його не розряджати, то при вимкненні живлення напруга на ньому буде зменшуватися повільно і антена передавача деякий час випромінюватиме в простір не командні групи, а послідовність імпульсів тактового генератора. Роботу дешифратора буде порушено. Як формуються групи імпульсів інших команд, легко зрозуміти, розглянувши таблицю.

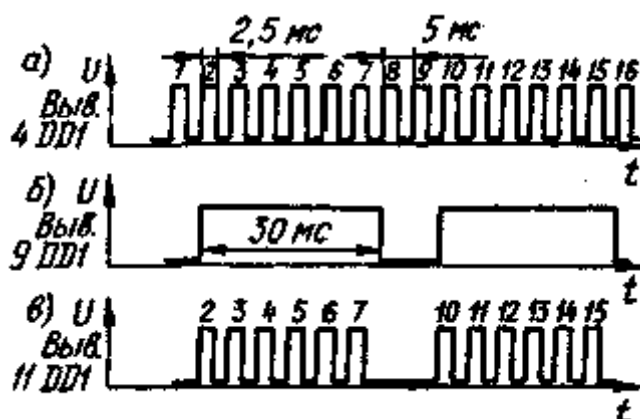


Рис.3.2. Группы импульсов

Команда	Нажата кнопка	Число імпульсів в групі
Стоп	—	Шість
Назад	SB1	Пять
Вперед	SB2	Четыре
Влево	SB3	Три
Вправо	SB4	Два

Щоб уникнути одночасної подачі двох і більше команд при випадковому натисканні на кілька кнопок, шифратор використовує кнопки з перемикаючими контактами [3].

Для правильної роботи пристрою захисту від імпульсів перешкод необхідно, щоб при переході від однієї команди до іншої кнопки SB1-SB4 хоча б на деякий час перебували у не натиснутому положенні. У цьому випадку послі кожної переданої команди модель виконуватиме команду "Стоп".

Принципова схема завод захищеного дешифратора показана на рис. 3.3

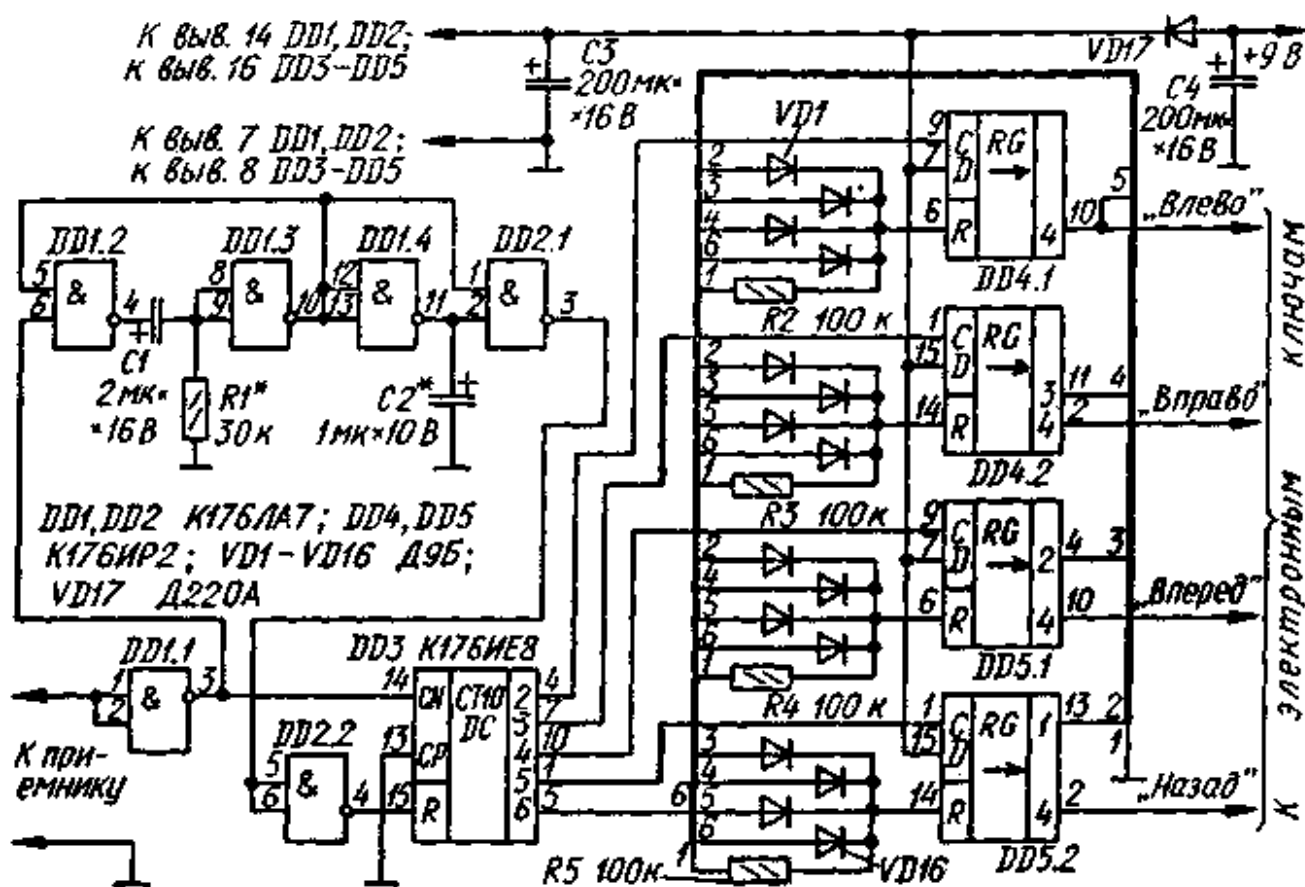


Рис.3.3 Принципова схема перешкодозахищеного дешифратора команд

Дешифратор складається з вузла, що визначає паузи між командними групами імпульсів – одинівбратор на логічних елементах DD1.2, DD1.3; формувача імпульсів обнулення на елементах DD1.4, DD2.1 та інверторі DD2.2; лічильника DD3 числа імпульсів команди в кожній групі та вузла захисту від імпульсів перешкод DD4, DD5, VD1—VD16, що вважає групи командних імпульсів. Регістр DD4.1 підраховує групи імпульсів команди "Вліво", DD4.2 - "Вправо", DD5.1 - "Вперед" і DD5.2 - "Назад". Діод VD17 перешкоджає проходженню ланцюга живлення негативних імпульсів перешкод, створюваних двигунами моделі. Конденсатори С3, С4 зменшують пульсації напруги, що виникають під час роботи моделі.

3.2 Робота пристрою

Розглянемо роботу дешифратора при команді «Стоп» без перешкод. Припустимо, що при подачі живлення на дешифратор лічильник DD3 і регістри DD4, DD5 встановлюються у вихідний стан, тобто на виході лічильника DD3 буде рівень 1, а на всіх виходах регістрів - рівень 0. Цей стан дешифратора вважають черговим, що встановлюється після включення спочатку живлення моделі, а через деякий час - передавача.

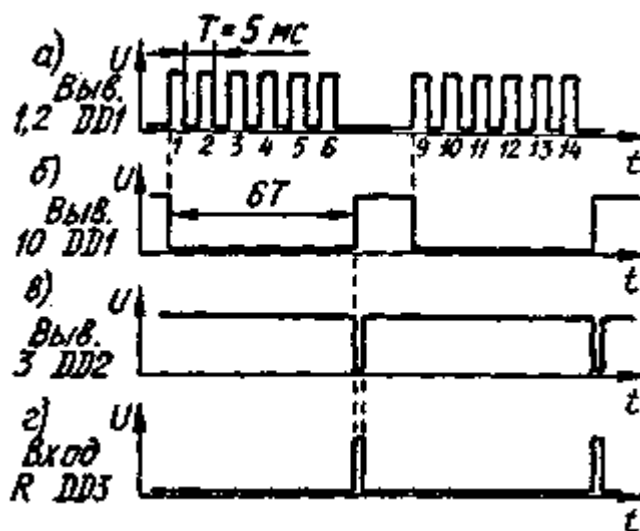


Рис.3.4. Групи імпульсів

Якщо тепер на вхід інвертора DD1.1 надійде перша група імпульсів команди «Стоп» (рис. 3.4 а), то фронт першого імпульсу запусить одинівбратор і на його виході (висновок 11 елемента DD1.4) з'явиться рівень 0 (рис. 3.4, б). Але імпульси

команди надійдуть також і на лічильний вхід лічильника DD3. З кожним імпульсом групи високий рівень буде переходити з одного виходу лічильника DD3 на інший у бік зростання їх номерів, і перші розряди регістрів DD4, DOS по черзі запишеться інформація з входу D.

По спаду шостого імпульсу групи рівень 1 з виходу лічильника 6 DD3 через відповідні діоди надійде на установочний вхід R всіх регістрів і підтвердить їх вихідний стан. Через проміжок часу, рівний $6T$ (його встановлюють, підбираючи резистор R1), на виході одновібратора з'явиться рівень 1, і на виході вузла формування імпульсів скидання (висновок елемента 4 DD2.1) сформується короткий імпульс негативної полярності (рис. 3.4, в). Тривалість імпульсу (близько 0.25 мс) встановлюють підбираючи конденсатор C2. З виходу інвертора DD2.2 імпульс (рис. 4, г) надійде на вхід R лічильника 003 та встановить його у вихідний стан. Потім на вхід дешифратора прийдуть друга, третя, четверта і т. д. групи, і розглянутий процес щоразу повторюватиметься.

Тепер вже легко буде зрозуміти роботу дешифратора під час прийому команди, наприклад, «Назад» у присутності перешкод. У кожній групі цієї команди по п'ять імпульсів тактового генератора. Припустимо, що у вхід дешифратора надходять групи імпульсів з перешкодами — у першій і третій групах міститься по одному імпульсу перешкоди, тобто, ці групи відповідатимуть групам імпульсів команди «Стоп».

В цьому випадку після закінчення першої групи регістр DD5.2 залишиться у вихідному стані. Наприкінці другої групи на виході цього регістру з'явиться рівень 1, який через відповідні діоди надійде на вхід R інших регістрів і заборонить запис інформації в них по входу D. Після третьої групи регістр DD5.2 повернеться у вихідний стан, а на входах R інших регістрів встановиться рівень 0.

Після закінчення четвертої групи імпульсів знову на виході 1 регістра DD5.2 з'явиться рівень 1. Потім після п'ятої, шостої та сьомої груп рівень 1 з'являтиметься відповідно на виходах 2, 3 і 4 регістра DD5.2. В результаті спрацює електронний ключ каналу "Назад" і модель виконає команду.

Якщо на вхід дешифратора надійде група імпульсів команди "Назад" з перешкодою, то всі регістри на дуже короткий час - 37,5 мс - повернуться у вихідний

стан, на виході "Назад" з'явиться рівень логічного нуля і електронний ключ закриється і знову відкриється. Навіть якщо виконавчий механізм моделі встигне спрацювати на цей час, положення моделі це практично не змінить.

Розглянемо ще один приклад - проходження команди "Вперед", коли на вхід дешифратора надходять групи імпульсів з перешкодами. У кожній групі цієї команди – по чотири імпульси. Допустимо, що лише до першої групи цієї команди додався один імпульс перешкоди. Тоді п'ятий імпульс переведе регістри у вихідний стан і подальший запис у них не відбуватиметься. Але оскільки друга та наступні групи імпульсів перешкод не містять, ні на одному з виходів дешифратора керуючої напруги команди не з'явиться (оскільки запис у регістр DD5.1 заборонено) і тоді оператор повинен буде на короткий час відпустити командну кнопку «Вперед» на передавачі та знову на неї натиснути. Іншими словами, хибна команда на вихід не пройде.

При налагодженні шифратора підбирають резистор R1 таким, щоб при частоті тактового генератора 200 Гц шпаруватість імпульсів дорівнювала двом. Підбираючи резистор R1 в дешифраторі, домагаються, щоб тривалість сигналу одновібратора дорівнювала 6Т. Струм, що споживається шифратором, в режимі команди «Стоп» - не більше 3 мА, а дешифратором - не більше 5 мА.

Описана перешкода захищена система телеуправління розрахована на п'ять команд. Проте їхню кількість неважко збільшити. Для отримання дев'яти команд необхідно в шифраторі використовувати дванадцятирозрядний регістр зсуву і додати чотири командні кнопки. У дешифраторі треба використовувати вільні виходи лічильника 003, додати відповідне число регістрів та діодно-резисторних вузлів, а також встановити тривалість вихідного імпульсу одновібратора, що дорівнює ЮТ.

ВИСНОВКИ

На основі проведеного в роботі аналізу сучасних технічних рішень надійному захисту складних електронних, електромеханічних, цифрових можна зробити такі висновки:

1. Основними перешкодами для надійного виконання апаратурою функцій які були надані дослідниками і конструкторами в процесі проектування, а саме, керування, вимірювання, обробки і передачі даних, тощо, є електромагнітні (ЕМ) випромінювання, природні і промислових властивостей в навколишньому середовищі. Для запобігання наслідків від перелічених перешкод використовують фізичні методи, метою яких є екранування корпусів, передавальних середовищ (дротових, кабельних) з обов'язковим заземленням екранного засобу, створення екіпотенціальної поверхні (екрану, лінії рівного потенціалу). Це поверхня, в усіх точках якої однаковий потенціал. Екіпотенціальні поверхні перпендикулярні до силових ліній поля.

2. Також важливим і чутливим для електронних пристроїв є захист від ЕМ завад через кола живлення, в т.ч. від мережі. Там використовуються спеціальні фільтри, і не тіки в блоках живлення, а також на блоках, модулях, платах.

3. При бурхливому розвитку цифрової схемотехніки з'явився новий від перешкод пов'язаний зі змаганнями логічних сигналів, брязкіт контактних перемикань при переключеннях. Таки проблеми вирішуються конструкторами при проектуванні. Але для контактних перемикань слабкострумових кіл існує проблема окислення контактів, яка також може вирішена переходом на електронні перемикання через RS-тригери і фільтруючі брязкіт фільтри. Для забезпечення ефективної розробки цифрових пристроїв проектувальник повинний вибрати найбільш раціональні варіанти рішення поставленої задачі на основі якісної математичної моделі.

4. В аналого-цифрових перетворювачах переміщення – код (паралельний або послідовний) використовуються рефлексії коди (код Грея). Також, використання зсунутих на 90 ел.градусів імпульсних серій, що дає можливість безпомилково визначати напрямок руху.

5. Для сполучення керуючого комп'ютера, або мікро-ЕОМ з будь якою вимірювальною апаратурою використовуються стандартні промислові інтерфейси. В залежності від вимог по відстані, швидкості передачі, можуть використовуватись, зокрема інтерфейси, існуючі: Centronics, RS-232, RS-485 і сучасні: USB-2, USB-3. Також в деяких випадках для мобільних технологій можуть використовуватись бездротові інтерфейси.

6. Особливе значення мають перешкоди, пов'язані з кодуванням сигналів в інформаційних цифрових і комп'ютерних системах. Використовують контроль парності та надмірність коду, що, крім виявлення помилок, може їх виправляти.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Абакумов В.Г. Электронные промышленные устройства. Киев, изд. "Вища школа", 1978, 376 с.
2. <https://ozlib.com/1056240/geografiya/refleksnyy>
3. <https://habr.com/ru/articles/140611/>
4. <http://users.cs.fiu.edu/~downeyt/cop3402/hamming.html>
5. <https://cxem.net/mc/book38.php>
6. Солодовников В.В. Основы теории и элементы систем автоматического регулирования / В.В. Солодовников, В.Н. Плотников, А.В. Яковлев // М.: Машиностроение, 1985. – с. 155.
7. <https://www.radioland.net.ua/sxemaid-175.html>
8. Методичні вказівки до виконання курсової роботи з дисципліни "Цифрова схемотехніка". КіСумДУ, 2006.
9. Бесекерский В.А. Теория систем автоматического регулирования. В.А. Бесекерский, Е.П. Попов // М., Наука, Гл. ред. ф.-м. литер. / 1975. –768 с.

10. <https://root-nation.com/news-ua/it-news-ua/ua-quantum-accelerometer/> дата доступа: 19.05.2023 г.
11. <https://www.atomic-energy.ru/news/2022/11/01/129859> дата доступа: 19.05.2023 г.
12. Бесекерский В.А. Динамический синтез гироскопической стабилизации / В.А. Бесекерский, Е.А.Фабрикант // – Л.: Судостроение, 1968. – 353 с.
13. Денисенко В. В. Компьютерное управление технологическим процессом, экспериментом, оборудованием. – М.: Горячая линия–Телеком, 2013. – 606 с.
14. Гук М. Интерфейсы ПК. Справочник: СПб: ЗАО «Издательство «Питер», 1999 416 с
15. Ан Пей. Сопряжение ПК с внешними устройствами: Пер. с англ. – М.: ДМК Пресс, 2001. – 320 с.: ил.
16. Смит Дж. Сопряжение компьютеров с внешними устройствами. Уроки реализации: Пер. с англ.- М.: Мир, 2000. – 266 с., ил.
17. Сопряжение датчиков и устройств ввода данных с компьютерами IBM PC: Пер. с англ./Под ред. У. Томпкинса, Дж. Уэбстера. – М.: Мир, 1992. – 592 с., ил. 1. В. Козлов. Узлы аппаратуры управления моделями.— Радио, 1983, № 4, с. 24, 25.
18. С. Алексеев. Квazисейсорные переключатели на микросхемах.— Радио, 1984. № 3, с. 26—29.
19. Васильев В.И. Пути снижения динамических нагрузок в канатах шахтных подъемных установок системами автоматически регулируемого предохранительного торможения / В.И. Васильев // Стальные канаты: Сб. науч. тр. МАИСК. – Одесса: “Астропринт”, 2010, №8. – С. 18-29.
20. РАДИО N 1, 1987 г.