Висновок: Підсумовуючи вище сказане розроблений верстак буде актуальний та простий в користуванні, реалізована модульність при реалізації дозволить полегшити ремонт та модернізацію пристрою за потреби. Також можливий розвиток концепції, за умови посилення конструкції та заміни двигунів на більш потужні можна замінити лазерний блок на малопотужну фрезу. Проте в такому випадку потрібно передбачити додавання третьої осі на портал. В результаті співвідношення ціна – якість буде на відносно високому рівні, що зробить даний пристрій доцільним для використання.

1. *https://www.nema.org/Standards/SecureDocuments/ICS16.pdf*
2. *http://homecnc.ru/mech/1-tips-cnc*
3. *https://www.trinamic.com/fileadmin/assets/Products/ICs_Documents/TMC2209_Datasheet_V103.pdf*
4. *https://www.farnell.com/datasheets/1682209.pdf*
5. *https://www.handsontec.com/dataspecs/cnc-3axis-shield.pdf*

Т. Голик

Науковий керівник – д.т.н., проф. Р. В. Кочан

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СЕРВЕРА ЛАЗЕРНОГО ГРАВЕРА З ЧИСЛОВИМ ПРОГРАМНИМ КЕРУВАННЯМ

Актуальність. Портативні верстати, не дивлячись на свої мініатюрні розміри, здатні виконувати якісну обробку безлічі матеріалів. Вони відрізняються великим діапазоном технічних характеристик, що дозволяє підібрати обладнання безпосередньо під ваші потреби, не переплачуючи за функціонал, який ви не будете використовувати.

Сучасні лазерні та фрезерні верстати управляються системою ЧПУ (спеціалізований контролер з відповідним ПО), яка забезпечує узгоджену роботу всіх приводів. Більшість портативних верстатів з ЧПУ мають управління, організоване за допомогою персонального комп'ютера. ЧПК повинен виконувати функції клієнта, а керувати ним буде серверна частина яка буде являти собою програмне забезпечення, встановлене на певну операційну систему. Це здешевлює апарат, так як окремий блок управління коштує досить дорого.

Основною метою розробки програмного забезпечення для лазерного гравера є бажання спростити та скоротити час для гравіювання будь якого зображення на твердій поверхні. І за допомогою декількох натискань на клавішу, ми зможемо вибрати, відредагувати та гравіювати.

Концепція розробки. Враховуючи вище сказане пропонується створення програми керування з можливістю обробки та редагування зображення.

Поставлене завдання:

1. Дослідити процес створення програмного забезпечення.
2. Проаналізувати аналоги: українські та зарубіжні зразки.
3. Виявити наявні переваги та недоліки.
4. Створити алгоритм написання прогами.
5. Розробити програму, протестувати, та описати отриманий результат.

Вирішення завдання. Результати дослідження та порівняння так званих аналогів даного програмного забезпечення свідчать про те, що деякі з них є або прості у використанні, але з малим асортиментом функцій, або ж навпаки. У зв'язку з очевидною конкурентоспроможністю цього програмного забезпечення в порівнянні з існуючими на сьогоднішній момент особливий інтерес представляє проблема функціоналу та вирішення поставлених задач, для яких було б ефективним містити в собі такі функції як:

1. Таймер.

Показує час початку процесу і приблизний час виконання роботи.

2. Відсотковий лічильник.

Вказує на скільки виконаний процес у відсотковому співвідношенні.

3. Кнопка «Завантажити».

Кнопка створена для зручного користування. Її функція полягає в тому, щоб легко та швидко можна було завантажити зображення в програму. При цьому програма автоматично конвертує фото в необхідний формат, та переводить його в чорно-білий ефект.

4. Редактор.

Функція, яка необхідна при виконанні даної роботи. Після завантаження зображення(яке буде гравіюватись) потрібно буде його відредагувати, а саме, виставити контраст(тони чорного, сірого, та білого відтінків), виділення контурів та країв.

5. Повзунки.

За допомогою так званих повзунків, користувач зможе регулювати швидкість руху лазера, що скорочує час виконання роботи(але це може вплинути на якість виконання роботи); потужність лазера(для різних типів поверхонь гравіювання); калібрування(для вираження явних переходів чи відтінків).

6. Кнопка «СТАРТ/СТОП».

Чітко виражена та примітна кнопка, яка подає команду Старту/Завершення процесу гравіювання.

Окрім зручності та простоти використання розробленої програми ще одним явною перевагою є створений алгоритм виконання роботи. Цей алгоритм оптимізує роботи пристрою завдяки його роботі не по периметру, а по вектору. Цей алгоритм забезпечує зменшення часу виконання завдання, чітко виділяє контури та переходи та оптимально використовує роботу лазера, що призводить до меншого зношення лазера, та запобігає перегріванню.

Висновок: У даній роботі проведено аналіз сучасного програмного забезпечення, для станків/граверів з ЧПК. Зроблено порівняння ПЗ, виявлено їх переваги та недоліки. Також було зроблено замітки по цих програмах. Після обґрунтування було створено висновки, та встановлено, що саме потрібно для розробки зручного та якісного ПЗ. Оскільки головною ідеєю для створення даного програмного забезпечення було зручність та простота використання, було створено перелік необхідних функцій, а також розроблено графічний інтерфейс програми.

1. *https://mirstankov.com/uk/yak-ce-pracyuye-chpu/*
2. *https://mirstankov.com/uk/catalog/nastolnyj-lazernyj-graver-msl4040-nezamenimyj-pomoshhnik-2/*
3. *https://prom.ua/p780861960-uluchshenyj-lazernyj-graver.html*

О. Мусянович
Науковий керівник – к.т.н., доц. Ю. В. Морозов

ПІДСИСТЕМА ЗАХИСТУ МУЛЬТИМЕДІЙНОЇ ІНФОРМАЦІЇ НА ВЕБ-САЙТІ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ ВІРТУАЛЬНИХ ДИСКІВ

Завдяки безперервному розвитку технологій, зростає частота нелегального використання мультимедійної інформації і, в наслідок чого, виникає необхідність посилення захисту даної інформації. Сучасні програмні засоби захисту, не завжди справляються з цим завданням, оскільки після розшифрування файлів, вони передають дані в вихідному вигляді стороннім програмам для відтворення інформації. Загальним недоліком такого підходу є те, що система захисту не завжди може бути впевнена в програмі, яка відтворюватиме розшифровану інформацію.

Отже, **актуальною є задача** розробка системи захисту мультимедійної інформації в поєднанні з відтворювачем даної інформації.

У роботі розглянуто розв’язок цієї задачі, за допомогою розробки підсистеми, на базі клієнт-серверної архітектури, що складається з