

Рисунок 4 – Зовнішній вигляд пьезодинаміка

Отже, мікрокомп'ютер Raspberry Pi є гарним пристроєм як для навчання, так і для виконання більш складних проєктів завдяки своїй багатофункціональності.

## РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЗА ДОПОМОГОЮ ВЕБ-СЕРВІСУ GitHub

*Смислова М.І.*

*Науковий керівник – Булаєнко М.В., канд. техн. наук, доцент*

GitHub — один з найбільших веб-сервісів для спільної розробки програмного забезпечення, з яким ми ознайомилися впродовж курсів навчання, що проводилися компанією СНІ Software. Базується веб-сервіс на системі керування версіями Git, яка розроблена Лінусом Торвальдсом. Сервіс безкоштовний для проєктів з відкритим вихідним кодом, з наданням користувачам усіх своїх можливостей (включаючи SSL), а для окремих індивідуальних проєктів пропонуються різні платні тарифні плани. Головна сторінка сервісу наведено на рисунку 1.

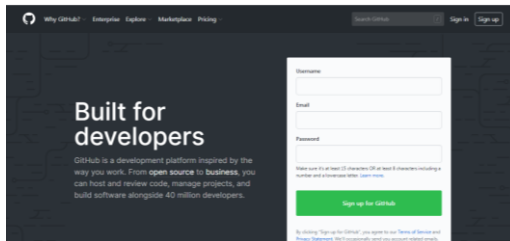


Рисунок 1 – Головна сторінка веб-сервісу GitHub

GitHub став популярний за рахунок того, що має великі можливості для соціальної взаємодії, фактично це соціальна мережа для розробки open source додатків. Окрім розміщення коду, учасники можуть спілкуватись, коментувати редагування один одного, а також слідкувати за новинами знайомих.

За допомогою широких можливостей Git програмісти можуть поєднувати свої репозиторії. GitHub дає зручний інтерфейс для цього і може показувати вклад кожного учасника в вигляді дерева.

Репозиторій Git – це каталог файлової системи, в якому знаходяться: файли конфігурації; файли журналів операцій, виконуваних над репозиторієм; індекс розташування файлів і сховище, що містить самі контрольовані файли (рис. 2).

Локальний репозиторій розташовано на локальному комп'ютері розробника в каталозі. Саме в ньому відбувається розробка і фіксація змін, які відправляються на віддалений репозиторій.

Віддалений репозиторій – це репозиторій, що знаходиться на віддаленому сервері. До нього приходять всі зміни і з нього забираються всі оновлення.

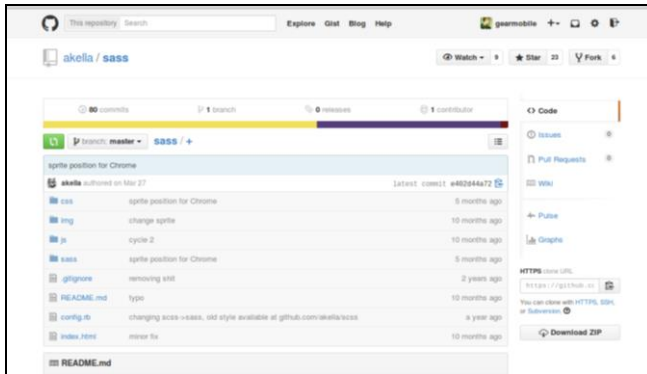


Рисунок 2 – Сторінка репозиторія GitHub

#### Можливості GitHub:

- Для проектів є особисті сторінки, невеликі Вікі та система відстеження помилок. Прямо на сайті можна дивитись файли проектів з підсвічуванням синтаксису для більшості мов програмування.
- На платних тарифних планах можна створювати приватні репозиторії, доступні обмеженому колу користувачів.
- Є можливість прямого додавання нових файлів в свій репозиторій через веб-інтерфейс сервісу.
- Код проектів можна не лише скопіювати через Git, але й завантажити у вигляді архіва.
- Окрім Git, сервіс підтримує отримання і редагування коду через SVN і Mercurial.
- На сайті є pastebin-сервіс для швидкої публікації фрагментів коду.

- Зберігання документації, включаючи автоматично відтворювані файли README у різних форматах файлів типу Markdown.
- Вкладені списки завдань у файлах.
- Візуалізація геопросторових даних.
- 3D-рендеринг файлів, які можна попередньо переглянути, використовуючи новий інтегрований переглядач файлів STL, який відображає файли на «3D canvas». Переглядач підтримує WebGL і Three.js.
- Внутрішній формат PSD для Photoshop з можливістю попередньо перегляду та порівняння з попередніми версіями того самого файлу.

Таким чином, GitHub – це гарний інструмент, який допоможе зробити процес написання коду більш простішим і приємнішим.

## УНІВЕРСАЛЬНИЙ АСИНХРОННИЙ ПРИЙМАЧ (UART)

*Рекель К.В.*

*Науковий керівник – Булаєнко М.В., канд. техн. наук, доцент*

Заняття, які проводилися компанією CHI Software, дали змогу познайомитися з універсальним асинхронним приймачем (Universal Asynchronous Receiver-Transmitter, UART).

Протокол UART – найстаріший і найпоширеніший на сьогоднішній день фізичний протокол передачі даних. Найбільш відомий з сімейства UART протокол RS-232, так званий COM-порт. Це, напевно, найстаріший комп'ютерний інтерфейс. Він дожив до наших днів і не втратив своєї актуальності. Основні робочі лінії – RXD та TXD, або просто TX та RX. Передаюча лінія - TXD (Transmitted Data), а порт RXD (Received Data) – приймаюча (рис.1).

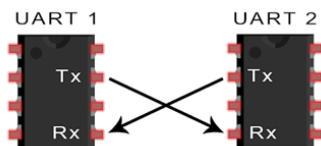


Рисунок 1 – Схема UART з сигналами TX та RX

Інтерфейс UART з'явився в США як засіб для передачі телеграфних повідомлень, і робочих біт там було всього п'ять (як в азбуці Морзе). Для передачі використовувалися механічні пристрої. Потім з'явилися комп'ютери, і коди ASCII, які вимагали сім біт. На початку 60-х на зміну прийшла 8-бітна таблиця ASCII, і тоді формат передачі став