

Ideja!

Omogućiti korištenje određenih ulaznih uređaja, virtualnih ili stvarnih(Kontrolera) na određenim sustavima posredno preko razvojnog računala.

Radi lakšeg shvaćanja ideje:

Neka je sustav kojeg se želi kontrolirati maketa automobila upravljana RPI mikroračunalom te se komande upravljanja zadaju tipkovnicom (W, S, A, D).

Prema tome:

Kontroler = tipkovnica

Sustav = RPI

Zašto?

- 1.) Mogućnost korištenje kontrolera bez direktne veze kontrolera i sustava kojim se kontrolira
- 2.) Mogućnost korištenja jednog kontrolera na više uređaja istovremeno
- 3.) Mogućnost korištenja ugrađeni kontrolera (npr. Tipkovnica na laptopu) na drugim uređajima
- 4.) Mogućnost korištenja neposotojećeg stvarnog kontrolera putem virtualnog kontrolera

Kako?

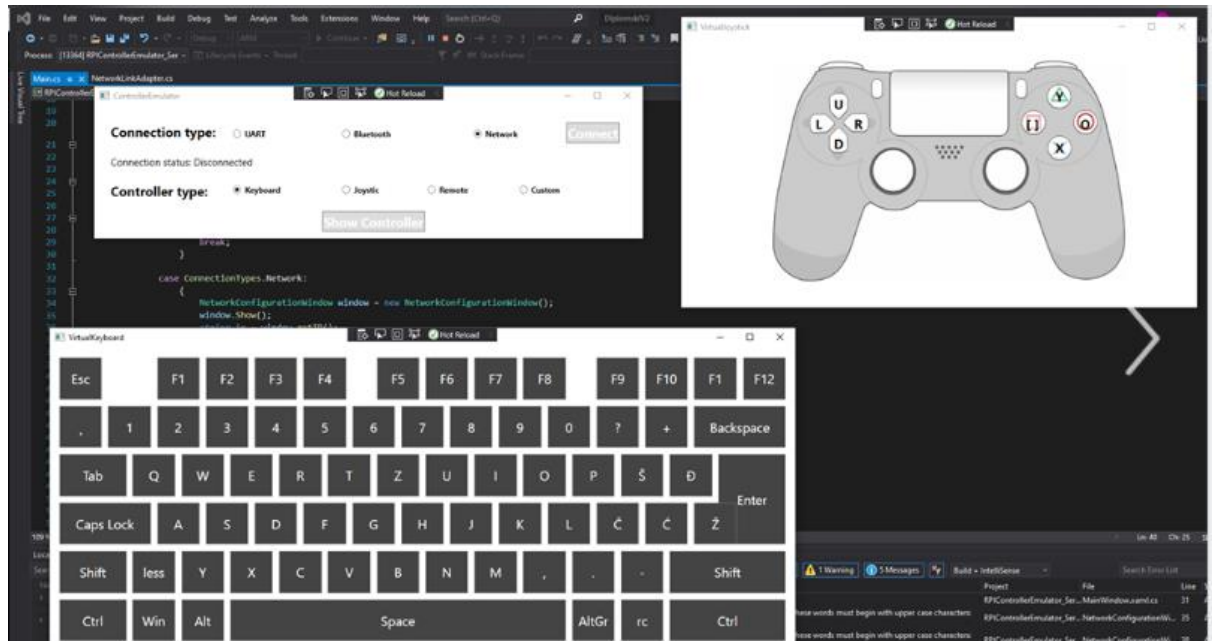
- 1.) Napisati driver za sustav kojeg se želi kontrolirati
- 2.) Napiasti aplikaciju namjenjenu razvojnem računalu
 - a. Napraviti virtualne uređaje
 - b. Povezati konkretne uređaje s razvojnim računalom, zatim s aplikacijom
- 3.) Povezati sustav kojeg se kontrolira sa razvojinim računalom određenom vezom(local area network, wireles local area network, bluetooth, uart).

Postoji li nešto slično?

- 1.) Putty – SSH remote connection
 - a. Jedina moguća veza je LAN ili WAN, nemoguće preko bluetootha
 - b. Ne postoji virtualni kontroler
 - c. Developer aplikacije koja koristi kontroler mora sam voditi računa o osluškivanju inputa
 - d. Nemogućnost pritiska dvije tipke istovremeno
- 2.) VNC / Remote desktop
 - e. Podrazumjeva da sustav podržava GUI
 - f. Ne postoji virtualni kontroler
- 3.) Bežični kontroler
 - g. Zahtjeva posjedovanje istog

Glavne cjeline projekta

- 1.) Windows desktop aplikacija za konfiguraciju veze između sustava i workstationa, unutar koje se nalaze virtualni kontroleri



- 2.) Driver za RPI koji bi razumio signale poslane bilo kojom ponuđenom vezom od strane working stationa
- 3.) Pokazni primjer u C jeziku – rpi projekt koji koristi ovakav tip kontrolera
- 4.) Pokazni primjer u PY jeziku – rpi projekt koji koristi ovakv tip kontrolera

Opcionalno:

- 5.) Driver za linux sustave
- 6.) Pokazni primjer na linux sustavu
- 7.) Setup & install link drivera i desktop aplikacije

Tehnologije, vještine i alati koje bi se primjenjivale:

- 1.) C#, C, Python, XAML, WPF, Visual Studio, Cross Compile, RPI, Ubuntu / Linux, Argument parser, TCP/IP, Bluetooth, UART, Kernel, Tredovi, MVVM

