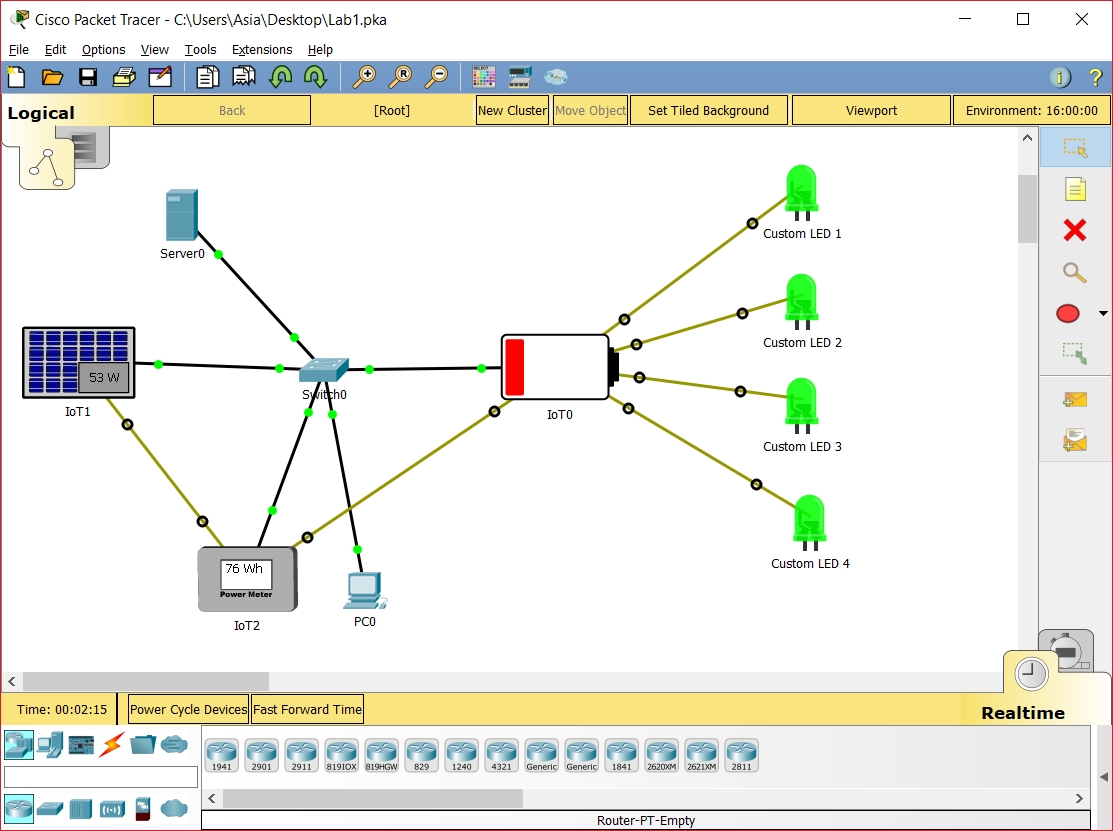
|  |
| --- |
| **Politechnika Świętokrzyska w Kielcach**  **Wydział Elektrotechniki, Automatyki i Informatyki** |

|  |
| --- |
| Technologie loT rozproszone sieci sensoryczne |

|  |  |
| --- | --- |
| Lab- Packet Tracer | Autor: **Joanna Gmyr**  Grupa: **3ID15B** |

|  |  |
| --- | --- |
| Numer laboratorium: **1** | Data wykonania: **4.11.2018** |

1. **Wyniki pracy laboratoryjnej.**

****

1. **GitHub**

* **Zastosowanie i opis systemu kontroli wersji.**
  + Z każdego miejsca mamy dostęp do naszych plików.
  + Ułatwia to pracę programistom nad wspólnym projektem.
* **Podstawowe komendy.**
  + **git init -** Inicjalizuje repozytorium GIT w danym katalogu
  + **git add [nazwa\_pliku] -** Dodaje zmiany we wskazanym pliku do commita
  + **git add . -** Dodaje wszystkie zmienione pliki do commita
  + **git add -p [nazwa\_pliku] -** Udostępnia możliwość dodania wybranych linii w zmodyfikowanym pliku do commita
  + **git commit -m "[treść\_commita]" -** Dodaje opis do commita. Dobrym zwyczajem jest opisanie co ta zmiana wprowadza do kodu w zakresie funkcjonalnym
  + **git add origin [adres\_repozytorium, np. https://github.com/username/moje-repozytorium.git] -** Ustawia konkretny adres zdalnego repozytorium jako główne repozytorium
  + **git push origin master -** Wysłanie zmian do branacha zdalnego
  + **git push -f -** Wysłanie zmian do zdalnego repozytorium ignorując konflikty, to znaczy, że jeśli wystapią konflikty to pliki zostaną nadpisane właśnie wysłaną wersją. Trzeba stosować to bardzo ostrożnie.
  + **git checkout [nazwa\_brancha] -** Zmienia aktywny branch na wybrany przez użytkownika
  + **git checkout [nazwa\_pliku] -** Usuwa zmiany w wybranym pliku
  + **git checkout . -** Usuwa zmiany we wszystkich zmienionych plikach
  + **git checkout -b [nazwa\_brancha] -** Tworzenie nowego brancha z aktywnego brancha i przełączenie się na niego
  + **git rebase master -** Zaciągnięcie zmian z brancha głównego do brancha aktywnego
  + **git push origin :[nazwa\_brancha] -** Usunięcie zdalnego brancha
  + **git branch -d [nazwa\_brancha] -** Usuwanie brancha lokalnie. Nie można usunąć w ten sposób aktywnego brancha
  + **git stash -** Dodanie zmienonych plików do pamięci/stosu i usunięcie ich z aktywnego brancha
  + **git pull –rebase -** Pobranie najnowszych zmian z aktywnego brancha zdalnego
  + **git stash pop -** Przywrócenie zmodyfikowanych plików z pamięci/stosu
  + **git stash clear -** Czyszczenie pamięci/stosu
  + **git remote prune origin -** Pobranie aktualizacji o usuniętych branchach zdalnych
  + **git fetch –all -** Pobranie listy zdalnych branchy
  + **git branch -** Wyświetlenie listy lokalnych branchy
  + **git branch -r -** Wyświetlenie listy zdalnych branchy
  + **git status -** Wyświetlenie listy zmienionych plików
  + **git diff [nazwa\_pliku] -** Szczegółowe wyświetlenie zmian w wybranym pliku
  + **git reset HEAD -** Resetowanie przygotowanych commitów (przed wysłaniem). Zmodyfikowane pliki są dostępne do ponownego dodania.
  + **git reset HEAD –hard -** usuwanie wszystkich zmian z brancha lokalnego i przywrócenie zmian z brancha zdalnego
  + **git reset HEAD^ --hard -** Usuwanie ostatniego commita z brancha
  + **git reset HEAD^^  
    git reset HEAD~2-** Obydwie komendy usuwają ostatnie 2 zmiany z brancha. Im więcej daszków (^) tym więcej commitów zostanie usuniętych.
  + **git rebase -i HEAD~3 -** Interaktywne zmienianie zawartości, opisów commitów. Commity mozna łączyć wtedy w jeden duży, zmienić jego opis, itd.

1. **Obserwacja.**

Bateria jest ładowana tylko w dzień.

Symulowany system posiada wiele możliwości rozbudowy. Można dodać inne źródła energii, np. turbinę wiatrową, która produkuje większą ilość energii. W każdym urządzeniu można modyfikować kod m.in. za pomocą Pythona.