

Płytki developerska Cyclone V GX Starter Kit

Sprzętowy hackathon on-line

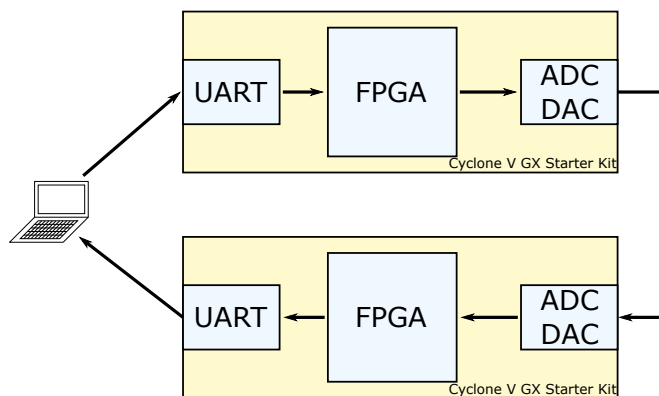


Nokia zajmuje się tworzeniem infrastruktury dla telefonii komórkowej. Nasi inżynierowie pracują nad tymi zagadnieniami już od ponad 25 lat, a w tym roku, w Centrum R&D Nokii w Krakowie będziemy świętować dziesięć lat rozwijania technologii LTE. Jednak nie spoczywamy na laurach. Obecnie intensywnie zajmujemy się technologią 5G, a już wkrótce także 6G. Szczególnie prędko rozwijającym się działem jest programowanie układów FPGA. Są one wyjątkowo przydatne przy przetwarzaniu dużych ilości danych, z jakimi spotykamy się w nowoczesnej telekomunikacji.

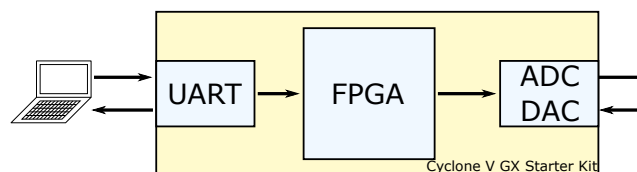
Aby dzielić się naszą wiedzą, w zeszłym roku zorganizowaliśmy konferencję oraz hackathon poświęcony tej tematyce. W artykule zaprezentuję kilka wspomnień tego wydarzenia. Jeżeli po lekturze, szanowny Czytelniku, zaczniesz żałować, że przegapiłeś to wydarzenie, to nic straconego. Już rozpoczęliśmy pracę nad drugą edycją, planowaną na listopad bieżącego roku. Wróćmy jednak do przeszłości...

Zadanie

Plan zrealizowania hackathonu FPGA pojawił się jeszcze w spokojnych czasach, gdy nikt nie słyszał o koronawirusie (czyli w połowie 2019 roku). Pierwotnie miał on odbyć się więc stacjonarnie, z początkiem kwietnia 2020 roku w Krakowie. Zadanie było planowane dla uczestników, którzy spotkają się na 24 godziny, a każda z drużyn miała otrzymać laptop z zainstalowanym oprogramowaniem oraz płytkę developerską Cyclone V GX Starter Kit[1], która została



Rysunek 1. Schemat zestawu do testowania rozwiązań



Rysunek 2. Stanowisko dla uczestnika

pokazana na **fotografii tytułowej**. Jest ona wyposażona w kodek audio SSM2603 oraz liniowe wejście i wyjście audio. Na potrzeby zadania zostały one połączone razem, co schematycznie pokazuje **rysunek 1**.

Celem było zaimplementowanie modemu wykorzystującego tak stworzone łącze do przesyłania danych z jak największą

przepustowością. Na obydwu płytkach znajdował się ten sam bitstream. Dane testowe są przesyłane z komputera poprzez port szeregowy. Układ FPGA musi zrealizować odpowiednią modulację i przekazać ją do kodeka audio. Następnie, na drugiej płytce, odebrane dane muszą zostać z powrotem odkodowane i przesłane do komputera. Na potrzeby testowania własnych pomysłów drużyny korzystały tylko z pojedynczej płytki ze zwartym wejściem i wyjściem (**rysunek 2**), natomiast zestaw z dwoma płytkami został użyty do oceniania rozwiązań.

Aby ułatwić rozpoczęcie pracy, uczestnicy otrzymali projekt startowy. Można go znaleźć w repozytorium [2]. Jak pokazano na **rysunku 3**, składa się on z trzech głównych części:

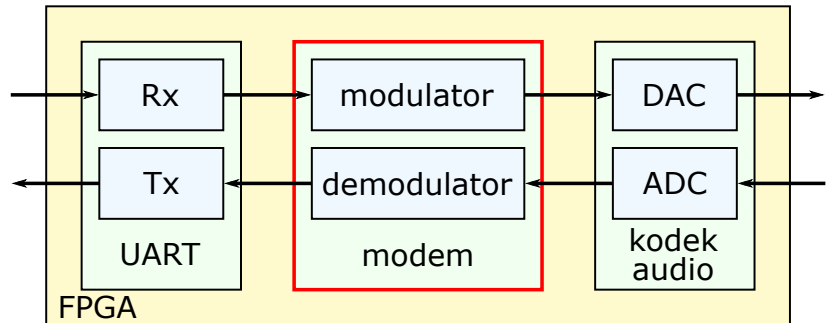
- obsługi portu szeregowego;
- modemu;
- interfejsu dla kodeka audio.

W dostarczonej wersji dostępna jest prosta implementacja modemu. Dzięki temu uczestnicy mogli zapoznać się z formatem danych wejściowych i wyjściowych. Mieli także dostępną bazę, na podstawie której mogli tworzyć własne rozwiązania.

Uczestnicy mieli do dyspozycji także dość spory zestaw narzędzi programistycznych. Głównym z nich było środowisko Quartus, umożliwiające generowanie wsadu oraz symulator Questa. Poza nimi można było też używać pakietu MATLAB&Simulink, Riviera-PRO oraz narzędzia do weryfikacji formalnej OneSpin.

Udostępnianie sprzętu

W połowie marca okazało się, że dwutygodniowy lock-down jednak zostanie z nami dużo dłużej. Kwietniowy termin był więc nie



Rysunek 3. Projekt startowy, uczestnicy modyfikują blok zaznaczony czerwoną ramką

do utrzymania. Z nadzieją, że sytuacja zdąży wrócić do normalności, nowy termin został ustalony na listopad. Szybko okazało się, że jeżeli chcemy, aby wydarzenie mogło się odbyć, to musimy przejść na formę on-line. Musieliśmy więc trochę zmodyfikować infrastrukturę. Zbudowane przez nas rozwiązanie zostało pokazane na **rysunku 4**. W naszym laboratorium w Krakowie uruchomiliśmy serwer, do którego podłączyliśmy poprzez USB programatory JTAG oraz porty szeregowo wszystkich dwudziestu płytek. Tak jak planowaliśmy wcześniej, zestawy dla drużyn miały podłączoną pętlę zwrotną. Natomiast wydzielone zestawy do oceniania składały się z dwóch połączonych razem płytek. Jedynym zadaniem serwera było zapewnienie dostępu do płytek za pomocą protokołu USBoverIP.

Dla każdej z drużyn przygotowaliśmy maszynę wirtualną z zainstalowanym oprogramowaniem. Dzięki udostępnieniu portów USB przez Internet maszyny wirtualne miały bezpośredni dostęp do sprzętu znajdującego się w laboratorium. Wszystkie narzędzia programistyczne łączyły się ze sprzętem w taki sposób, jakby był on fizycznie dołączony do gniazda USB komputera.

REKLAMA

ZAINTERESOWAŁ CIĘ POWYŻSZY ARTYKUŁ?

Dowiedz się więcej o aktualnych ofertach pracy w Nokii Kraków!

APLIKUJ



nokiakrakow.pl/oferty-pracy

CHCIAŁBYŚ POZNAĆ NAS LEPIEJ?

Dołącz do najbliższego Hackathonu i konferencji FPGA.

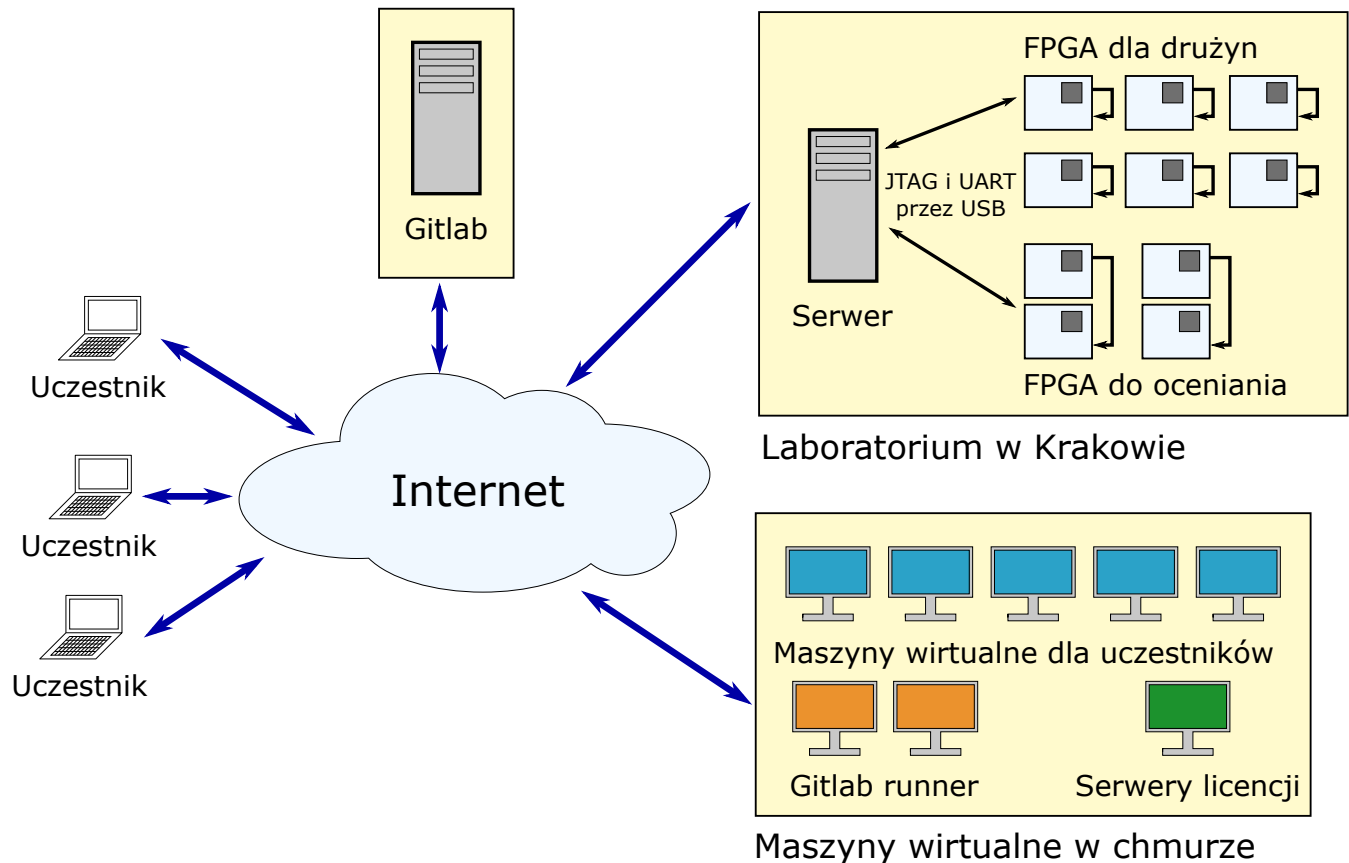
DOŁĄCZ



fpgahackathon.com

Nie tylko z zakresu FPGA!

NOKIA



Rysunek 4. Infrastruktura sieciowa hackathonu

Dysponując możliwością wirtualnego przepinania urządzeń pomiędzy maszynami, postanowiliśmy zwiększyć liczbę drużyn, które wzięły udział w zawodach, poprzez współdzielenie sprzętu: drużyny parzyste miały przydzielony sprzęt w godzinach parzystych, a nieparzyste w nieparzystych. Za przełączanie odpowiadał skrypt uruchomiony na serwerze.

Ocenianie rozwiązań

Do realizacji oceny rozwiązań użyliśmy narzędzi CI (*continuous integration* – ciągła integracja) udostępnianych przez platformę Gitlab. Każda z drużyn miała przypisane repozytorium, w którym umieszczala kolejne wersje swoich rozwiązań. Każdy commit powodował zakolejkowanie testu. Były one wykonywane po kolei, na jednym z dwóch zestawów. Obejmował on:

- skopiowanie rozwiązania na maszynę wirtualną,
- zaprogramowanie obydwóch układów FPGA tym samym projektem,
- uruchomienie skryptu mierzącego uzyskaną przepustowość,
- zwrócenie uzyskanego wyniku.

Wyniki były przekazywane do bazy danych i w czasie rzeczywistym prezentowane na stronie internetowej. Dzięki temu każda z drużyn mogła śledzić swoje postępy na tle pozostałych uczestników.

Przebieg konkursu

Zgodnie z początkowymi założeniami wydarzenie trwało 24 godziny. Publikacja zadania nastąpiła w sobotę o godzinie 12.00. W zawodach wzięło udział 28 drużyn. Dzięki zmianie formuły na wirtualną pojawili się zawodnicy z całego świata.

W czasie zawodów komunikacja odbywała się poprzez platformę Discord. Poza wspólnym kanałem każda z drużyn miała także swój własny pokój. W przypadku niejasności lub pytań technicznych zawodnicy mogli zaprosić do niego wolontariuszy z Nokii albo mentorów reprezentujących producentów zastosowanych narzędzi.

Po zaciętej rywalizacji, w niedzielę po południu, poznaliśmy wyniki. Zwyciężyła drużyna z Wojskowej Akademii Technicznej z Warszawy. Natomiast kolejne miejsca na podium zajęły ekipy z St. Petersburga oraz Kalifornii.

Co dalej?

Nasz krakowski zespół, bogatszy o zeszlóroczne doświadczenia, rozpoczął już przygotowania drugiej edycji. Wstępnie możemy zdradzić, że jest ona planowana na listopad bieżącego roku i tak jak poprzednio, odbędzie się przez internet. Więcej informacji już wkrótce pojawi się na naszej stronie: www.fgahackathon.com.

Zapraszamy wszystkich chętnych do sprawdzenia swoich sił w zawodach. Natomiast jeżeli chciałbyś już teraz dołączyć do naszego nokiowego zespołu, zachęcamy do sprawdzenia aktualnych ofert pracy: <https://nokiakrakow.pl/oferty-pracy/>.

NOKIA

Rafał Kozik
 FPGA design engineer
 Nokia Kraków
rafkozik@gmail.com

Odnośniki

- [1] <https://bit.ly/3yyBPY2>
- [2] <https://bit.ly/3vgGKBS>

- [3] <https://bit.ly/2T5bXJW>
- [4] <https://bit.ly/3ufJcr0>

<http://forum.ep.com.pl>