**Rafał Nazarko**

Koło Naukowe Elektroniki i Technologii Informacyjnych

**Sterowanie zestawem LEGO Mindstorms za pomocą komputera Raspberry Pi 4 przy użyciu języka Python**

**Streszczenie**

Artykuł opisuje zasadę działania komunikacji mikrokomputera Raspberry Pi 4 z jednostką sterującą zestawu LEGO Mindstorms NXT lub EV3 przy użyciu bardzo popularnego języka programowania Python.

**Słowa kluczowe:** układ elektroniczny, programowanie.

1. Wprowadzenie

Głównym celem przyświecającym stworzeniu oprogramowania była potrzeba zwiększenia elastyczności sterowania w robotach tworzonych przez członków koła naukowego. Podstawowe narzędzie do oprogramowania, dostarczone przez firmę LEGO, ma określony zestaw poleceń, który często okazuje się niewystarczający przy bardziej wymagających projektach.

Użycie zewnętrznego kontrolera pozwala zwiększyć moc obliczeniową, którą dysponuje tworzona maszyna – a co za tym idzie również prędkość działania, wykorzystać złożone algorytmy lub rozszerzyć komunikację o dodatkowe porty USB, połączenie internetowe lub urządzenia wejścia/wyjścia.

1. Komunikacja mikrokomputera Raspberry Pi 4 z jednostką sterującą LEGO Mindstorms

Jednostka sterująca LEGO Mindstorms zapewnia trzy rodzaje komunikacji: Bluetooth, Wi-Fi i USB. Wszystkie z nich pozwalają tworzyć aplikacje działające na dowolnym komputerze i komunikować się z jednostką sterującą LEGO Mindstorms. Aby zapewnić maksymalną prędkość transmisji danych, powinno się stosować połączenie za pomocą kabla USB.

Jednostka sterująca LEGO Mindstorms, podobnie jak inne współczesne komputery grupuje 8 bitów w jeden bajt i adresuje swoją pamięć po jedym bajcie. Sąsiadujące ze sobą bajty graficznie oddzielane są dwukropkami „ : ” lub pionowymi kreskami „ | ”. Z racji tego, że zapis binarny jest bardzo długi i zazwyczaj nieczytelny, przyjęło się stosować notację szesnastkową. Zapis w tej postaci jest zwarty a konwersja do i z systemu binarnego prosta, ponieważ jedna cyfra szesnastkowa oznacza pół bajtu.

Aby to dobrze zrozumieć, poniżej został przedstawiony przykład komendy, która pinguje jednostkę sterującą (zmusza ją do odpowiedzi):

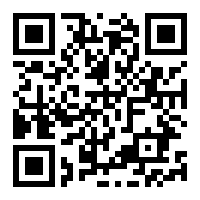
|  |
| --- |
| 0x|06:00|2A:00|00|00:00|01| |

Listing 1. Konstrukcja komendy ping

Jak widać komenda składa się 8 bajtów i znaku określającego liczbę szesnastkową *0x*. Dwa pierwsze bajty są częścią protokołów komunikacyjnych, którymi mogą być – jak wspomniano wcześniej – Bluetooth, USB lub Wi-Fi. Kolejne dwa bajty są licznikiem komunikatów, które pozwoli dopasować polecenie bezpośrednie oraz odpowiedź na nie. Piąty bajt określa czy nadawca tego polecenia oczekuje na odpowiedź czy też nie (|00| oznacza oczekiwanie na odpowiedź zaś |80| jej nie potrzebuje). Bajty 6 i 7 stanowią nagłówek komendy, który jest kombinacją dwóch liczb, które określają rozmiary polecenia bezpośredniego. Od bajtu 8 rozpoczynają się operacje, czyli dokładne informacje o działaniu, jakich oczekuje od jednostki sterującej mikrokomputer.

3. Podsumowanie

Aktualnie



Rysunek 3. Kod QR zawierający adres do repozytorium projektu

Źródło: opracowanie własne

Literatura

1. Lis M., *C#. Praktyczny kurs. Wydanie III*, Wydawnictwo Helion, 2016

Źródła internetowe

1. Stefan Jahn, Modified Nodal Analysis, http://qucs.sourceforge.net/tech/node14.html