

Ekran smartfona jako interfejs mobilny do amplitunera



czyli aplikacja mobilna poprzez UART w praktyce

W zeszłym roku, w kilku wydaniach „Elektroniki Praktycznej” (EP 4...7/20) ukazał się cykl artykułów na temat tworzenia aplikacji działającej na dowolnym smartfonie z użyciem języka JSON. Na smartfonie trzeba tylko zainstalować aplikację BBMobile, a cały design i funkcjonalność aplikacji realizowane są przez moduł sprzętowy.

Aplikacja BBmobile odbiera przez Bluetooth kod napisany w języku JSON i tworzy na ekranie smartfona interfejs użytkownika, a następnie cały czas odczytuje wszelkie polecenia z ekranu smartfona i przekazuje je zwrótnie do modułu BBMagic.

Moduł, w tym zastosowaniu potrzebny w wersji BBMobile, odbiera i nadaje dane poprzez łącze Bluetooth i przesyła je przez interfejs UART. Właściwy program trzeba napisać i zaimplementować w mikrokontrolerze (np. ARM) na własnej płytce drukowanej lub w dowolnym gotowym module (np. Arduino). Do komunikacji potrzebne są tylko dwa sygnały: TxD i RxD.

Komunikacja

Wyjście interfejsu UART naszego procesora oznaczone TxD należy podłączyć do wyprowadzenia nr 19 modułu, a wejście RxD

naszego procesora do wyprowadzenia nr 18 modułu. Na wyprowadzeniu 15 (oraz jego negacja na wyprowadzeniu 14) modułu BBMagic pojawia się stan zera logicznego, gdy moduł jest sparowany ze smartfonem.

Napięcie zasilania z zakresu 1,8...3,3 V można podłączyć do wyprowadzeń 7 i/lub 21 modułu, a masę do wyprowadzeń 1 i 20 i dodatkowo (nie jest to konieczne) 22. Po włączeniu zasilania moduł ustawia prędkość transmisji na 9600 bodów bez sprawdzania parzystości. Zestaw poprawnych poleceń rozpoznawanych przez moduł zależy do tego, czy jest w stanie połączenia BLE (Bluetooth), czy nie. Przed połączeniem BLE obecność modułu można sprawdzić, wysyłając komendę <hello zakończoną parą znaków \r\n. Wszystkie komendy kończymy w ten sposób (hexadecymalnie 0x0D, 0x0A).

W odpowiedzi moduł powinien wysłać odpowiedź >HI zakończoną tak, jak komendy. Podstawowe informacje o tym module były podane, we wspomnianych na wstępie, wydaniach EP. Jednak moim zdaniem zabrakło opisu kilku ciekawych poleceń. Mam na myśli polecenia do tworzenia tak zwanych kontrolek – EditText, Spinner oraz ProgBar.

Sprzet

Moduł BBMagic umieściłem w amplitunerze systemu 5.1 – AMPT-008 własnej konstrukcji. Sam amplituner został opisany w cyklu artykułów pt. „Amplituner kina domowego AMPL-008” w EP 5, EP 6 i EP 9 w 2013 roku. Amplituner ten jest wyposażony w procesor ARM typu LPC2387 oraz kodek typu WM8770 firmy Wolfson. Układ ten zawiera 1 przetwornik ADC STEREO (2 wejścia analogowe) oraz 4 przetworniki DAC STEREO (8 wyjść analogowych). Wejścia układu przełączane są za pomocą wbudowanego multiplexera analogowego – 8 wejść STEREO na 1 wyjście STEREO, które doprowadzone jest do układu PGA (regulacja wzmacnienia) i dalej na wejście przetwornika ADC.

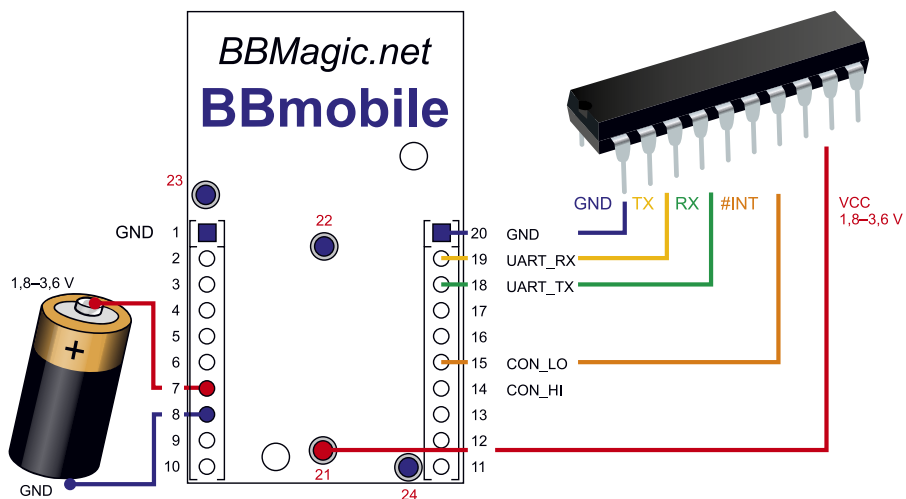
W moim amplitunerze umieściłem również cyfrowy dekodler Dolby Digital/DTS typu STA310 firmy STMicroelectronic. Układ jest wyposażony w wejście SPDIF i potrafi dekodować strumień danych w formacie PCM, Dolby Digital, DTS, MPEG i innych. Zawiera również procesor DSP, który może z 2 kanałów wejściowych (L, R) wygenerować 6 kanałów wyjściowych w formacie 5.1 (L, R, C, Sub, Ls, Rs). Procesor DSP może pracować w trybie Dolby Pro Logic lub Circle Surround.

Amplituner ma 5 wejść cyfrowych SPDIF (multiplexer cyfrowy doprowadza je na wejście układu STA310) i 4 wejścia analogowe STEREO (trzy z tyłu i jedno z przodu obudowy). Wyposażony jest w końcówki mocy typu D-Class. Dla kanałów głównych L i R zastosowałem układ typu TAS615, a dla kanałów C, Sub oraz Ls, Rs dwa układy typu TPA3123D2 produkcji firmy Texas Instruments. Jak widać z powyższego opisu, możliwości tej konstrukcji są dość szerokie. Amplituner wyposażyłem w wyświetlacz alfanumeryczny LCD 2x24 znaki. Jest to zdecydowanie małe „okno na świat” mojego amplitunera. Postanowiłem wykorzystać ekran smartfona jako alternatywny interfejs dla mojego amplitunera.

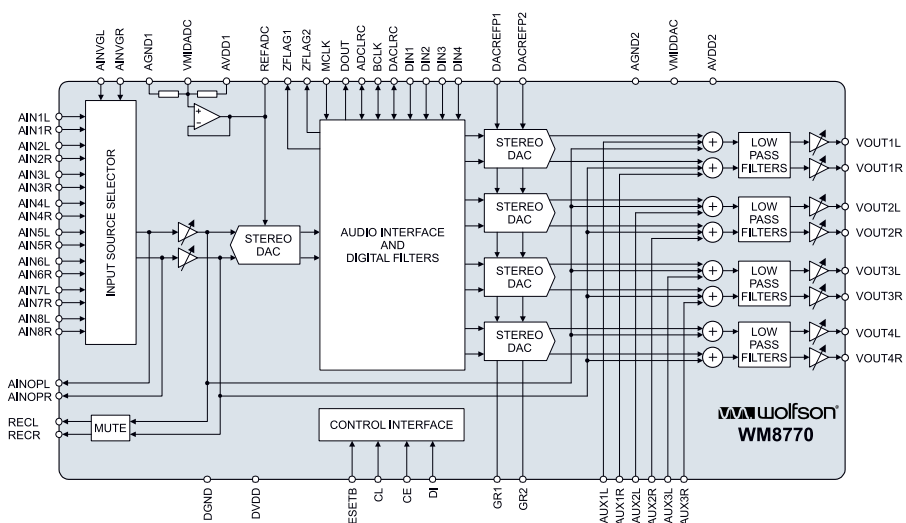
Aplikacja

Na wstępie chcę zwrócić uwagę na dwukierunkowość komunikacji pomiędzy smartfonem i amplitunerem. Moduł BBMagic podłączyłem do kanału UART2 procesora LPC2387. Sprawdzenie poprawności komunikacji można wykonać, wysyłając komendę <hello. Otrzymanie odpowiedzi >HI potwierdza prawidłowe połączenie i działanie modułu.

Na smartfonie (można do tego celu przeznaczyć nieużywany, starszy model) należy zainstalować aplikację BBMobile i włączyć Bluetooth. Po uruchomieniu aplikacji należy wybrać polecenie START, a następnie



Rysunek 1. Sposób podłączenia modułu BBMagic w wersji (z oprogramowaniem) BBMobile



Rysunek 2. Wewnętrzny schemat blokowy układu WM8770

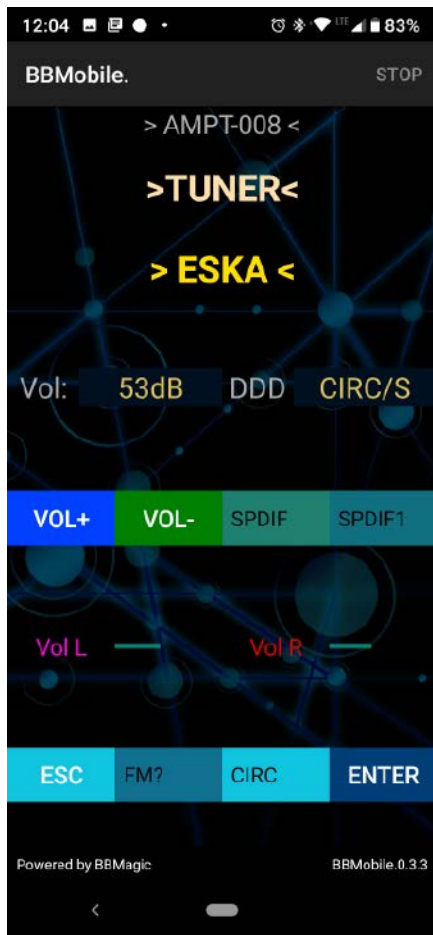
spośród znalezionych urządzeń (zwykle powinno być jedno) wybrać BBMobile. Od tego momentu aplikacja czeka na załadowanie poprzez Bluetooth tzw. Layoutu. W tym stanie w amplitunerze należy wywołać procedurę przesłania kodu Layout przez

interfejs UART2. W opisywanym zastosowaniu Layout jest definiowany kodem, który został pokazany na listingu 1.

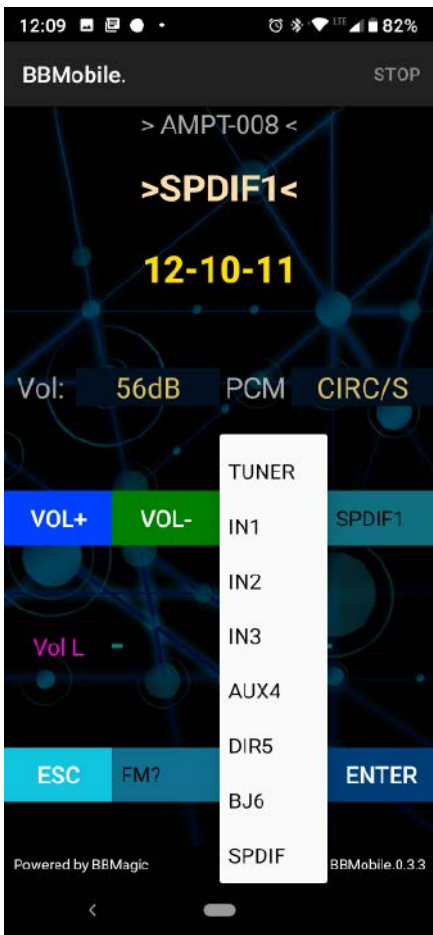
Główny blok Layout ma orientację pionową: "or": "V". Jako tło wybrałem image 6: "img": "6" (w programie zmieniam

Listing 1. Kod JSON definiujący Layout aplikacji mobilnej

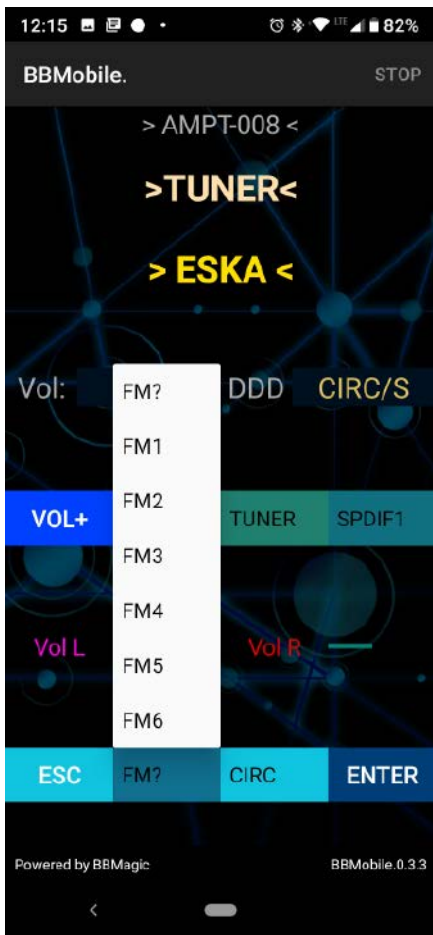
```
{
  "ty": "lout", "or": "V", "img": "6", "cs": [
    {
      "ty": "lout", "or": "V", "cs": [
        {
          "ty": "TextView", "n": "tv11", "te": "> AMPT-008 <", "tc": "174,174,174", "ts": "20", "w": "2"},
        {
          "ty": "TextView", "n": "tv12", "te": "INPUT", "tl": "bold", "tc": "255,224,174", "ts": "28", "w": "3"},
        {
          "ty": "TextView", "n": "tv13", "te": ">ABCDEFHG<", "tl": "bold", "ts": "28", "tc": "255,224,0", "w": "3"}
      ]
    },
    {
      "ty": "lout", "or": "H", "cs": [
        {
          "ty": "TextView", "n": "tv21", "te": "Vol:", "tc": "174,174,174", "ts": "18", "w": "1"},
          "ty": "TextView", "n": "tv22", "te": "99dB", "tc": "224,192,96", "bg": "0,16,32", "ts": "22", "w": "2"},
          "ty": "TextView", "n": "tv23", "te": "DEC:", "tc": "174,174,174", "ts": "18", "w": "1"},
          "ty": "TextView", "n": "tv24", "te": "PCM", "tc": "224,192,96", "bg": "0,16,32", "ts": "22", "w": "2"}
      ]
    },
    {
      "ty": "lout", "or": "H", "cs": [
        {
          "ty": "Button", "n": "b31", "te": "Vol+", "tc": "255,255,255", "bg": "0,64,255", "ts": "18"},
          "ty": "Button", "n": "b32", "te": "Vol-", "tc": "255,255,255", "bg": "0,128,0", "ts": "18"},
          "ty": "Spinner", "n": "sp33", "is": "IN?, TUNER, IN1, IN2, IN3, AUX4, DIR5, B36, SPDIF", "bg": "32,128,112", "ts": "22", "w": "1"},
          "ty": "Spinner", "n": "sp34", "is": "SPDIF?, SPDIF1, SPDIF2, SPDIF3, SPDIF4, SPDIF5", "bg": "16,112,128", "ts": "22", "w": "1"}
      ]
    },
    {
      "ty": "lout", "or": "H", "cs": [
        {
          "ty": "TextView", "n": "tv41", "te": "Vol L", "tc": "174,255,174", "ts": "18", "w": "1"},
          "ty": "ProgBar", "n": "pb1", "d": "1", "p": "25", "w": "1"},
          "ty": "TextView", "n": "tv42", "te": "Vol R", "tc": "174,255,174", "ts": "18", "w": "1"},
          "ty": "ProgBar", "n": "pb2", "d": "1", "p": "25", "w": "1"}
      ]
    },
    {
      "ty": "lout", "or": "H", "cs": [
        {
          "ty": "Button", "n": "b51", "te": "ESC", "tc": "255,255,255", "bg": "16,196,224", "ts": "18", "w": "1"},
          "ty": "Spinner", "n": "sp52", "is": "FM?, FM1, FM2, FM3, FM4, FM5, FM6", "bg": "16,112,144", "ts": "22", "w": "1"},
          "ty": "Spinner", "n": "sp53", "is": "DPL?, OFF, ProL, CIRC, DSTR0, DSLs, DSRs", "bg": "16,196,224", "ts": "22", "w": "1"},
          "ty": "Button", "n": "b53", "te": "ENTER", "tc": "255,255,255", "bg": "0,62,120", "ts": "18", "w": "2"}
      ]
    }
  ]
}
```



Rysunek 3. Ekran początkowy interfejsu mobilnego dla amplituneru AMPT-008



Rysunek 4. Ekran interfejsu mobilnego, wybór wejścia AUDIO



Rysunek 5. Ekran interfejsu mobilnego, wybór zapamiętanej stacji radiowej

pseudo-losowo cyfrę 6 na wartość z zakresu 1...6, tło image 7 pomijam, ponieważ jest bardzo jasne). W głównym bloku Layout zamkniętych jest 5 kolejnych Layoutów.

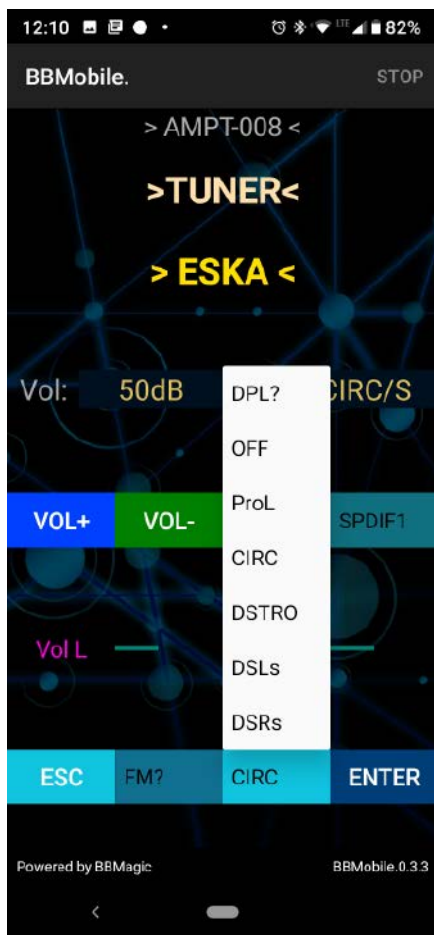
Pierwszy ma orientację pionową "or": "v" i zawiera 3 kontrolki typu TextView: "tv11", "tv12", "tv13". W trakcie pracy amplitunera pole tv12 będzie zawierało nazwę aktualnego wejścia np. >TUNER<, >SPDIF1< itp. Aby zmienić nazwę wyświetlanego aktualnego wejścia np. na >SPDIF1<, należy wysłać do modułu następującą sekwencję znaków: `$set,tv12:te=">SPDIF1<"\r\n`

Pole tv13 będzie używane do wyświetlania informacji RDS lub aktualnego czasu.

Drugi blok Layout ma orientację poziomą: "or": "h" i zawiera 4 kontrolki typu TextView: tv21...24. Szerokość pól dla tych napisów jest różna ("w": "1" ... "w": "2" ...) i wynosi odpowiednio: 1/6, 2/6, 1/6 i 2/6. Powoduje to, że szerokość zajmowana przez napisy tv22 i tv24 jest dwa razy większa od szerokości napisów tv21 i tv23. Pola tv21 – Vol: oraz tv23 – DEC pozostaną niezmienione, ponieważ są to pola opisowe. Pola tv22 oraz tv24 będą zawierały aktualne wartości, odpowiednio: głośności i wybranego dekodera. Aby wyświetlić np. wartość głośności równą 70 dB, należy wysłać do modułu następującą sekwencję znaków: `$set,tv22:te="70dB"/r/n`

Trzeci blok Layout również ma orientację poziomą i zawiera dwie kontrolki typu Button. Są to przyciski Vol+ oraz Vol- odpowiedzialne za regulację głośności oraz dwie kontrolki typu Spinner, które zawierają rozwijaną listę wyboru. Kontrolka Spinner sp33 pozwala wybrać wejście AUDIO: TUNER, IN1...SPDIF. Kontrolka Spinner sp34 umożliwia wybór konkretnego (domyślnego) wejścia SPDIF: SPDIF1, SPDIF2...5. Wyboru odpowiedniej pozycji z rozwijanej listy dokonujemy na ekranie smartfona dotykowo. Po wybraniu którejś linii rozwijana lista się zwiija. O tym, że dokonaliśmy jakiegoś wyboru, nie jest automatycznie przesyłana żadna informacja. Aby zatwierdzić i przesłać wybór, naciskamy dowolny przycisk – kontrolkę Button. Specjalnie do tego celu umieściłem w ostatnim rzędzie przycisk ENTER. Po naciśnięciu tego przycisku moduł BBMagic prześle przez UART taką informację: `$by,b53"1",sp33"TUNER",sp34"SPDIF1",sp52"FM?,sp53"DPL?"` standardowo zakończoną znakami `\r\n`. Oczywiście informację z kontrolki sp33, sp34, sp52 i sp53 są przesyłane po naciśnięciu dowolnego przycisku. Dla przycisku Vol+ przesłana informacja wygląda następująco: `$by,b31"1",sp33"TUNER",sp34"SPDIF1",sp52"FM?,sp53"DPL?"`

Tabela 1. Dokładny opis kontrolki typu Spinner - "ty": "Spinner"		
Nazwa pola	Mnemonik	Opis
Type	ty	Określa typ kontrolki. W tym przypadku jest to Spinner – pole umożliwiające wybór jednego elementu z wielu.
Name	n	Umożliwia nadanie kontrolce unikalnej nazwy służącej do jej identyfikacji w interfejsie, np. "s1".
Mode	m	0 – kontrolka będzie miała postać okna dialogowego; 1 lub brak parametru – kontrolka będzie miała postać listy.
Items	is	Lista zawierająca elementy wyboru rozdzielone przecinkami, np. „jeden, dwa, trzy, cztery”.
Weight	w	Definiuje wielkość kontrolki jako ułamek części całości widoku, gdzie licznik ułamka to wartość parametru 'w', a mianownik to suma wartości parametrów 'w' wszystkich kontrolkach znajdujących się w Layoutcie.



Rysunek 6. Ekran interfejsu mobilnego, wybór trybu pracy procesora audio

Program obsługi zaimplementowany w amplitunerze dla przycisków Vol+ i Vol- nie analizuje danych z kontrolki typu Spinner, ponieważ ich działanie ograniczam do regulacji głośności. Te dane są analizowane tylko po naciśnięciu przycisku ENTER. Dokładny opis kontrolki Spinner jest zawarty w tabeli 1.

Czwarty blok Layout zawiera dwie kontrolki TextView z napisami Vol L i Vol R oraz dwa razy nowy typ kontrolki: ProgBar. Służą one do wyświetlania paska postępu, tzw. *progress bar*. Postanowiłem zastosować te kontrolki do przedstawienia w sposób graficzny poziomu sygnału analogowego w kanale lewym i prawym amplitunera. Długość rysowanego paska należy podać w procentach (0...100).

Tabela 2. Dokładny opis kontrolki typu ProgBar – "ty": "ProgBar"		
Nazwa pola	Mnemonik	Opis
Type	ty	Określa typ kontrolki. W tym przypadku jest to ProgBar – pasek wizualizacji postępu.
Name	n	Umożliwia nadanie unikalnej nazwy kontrolce służącej do jej identyfikacji w interfejsie, np. "p1".
Determine	d	0 lub brak parametru – kontrolka sygnalizuje postęp bez określania stopnia wykonania; 1 – kontrolka pokazuje postęp określający stopień wykonania.
Progress	p	Aktualny postęp w procentach: przedział od 0 do 100.
Weight	w	Definiuje wielkość kontrolki jako ułamkową część całości widoku, gdzie licznik ułamka to wartość parametru 'w', a mianownik to suma wartości parametrów 'w' wszystkich kontrolki znajdujących się w Layoucie.

Dokładny opis kontrolki typu ProgBar jest zawarty w tabeli 2. Parametr *Determine* określa wygląd kontrolki ProgBar. Dla wartości 0 pasek postępu przyjmuje postać obracającego się wycinka okręgu i wygląda jak sygnał zajętości (zawieszenia się) aplikacji. Dla wartości 1 pasek postępu przyjmuje postać linii o długości proporcjonalnej do wartości parametru *Progress*. Nie mamy wpływu na kolor i grubość tej linii. Odświeżanie kilka razy na sekundę parametru *p* (*Progress*) pozwala wyświetlić zamiastkę analogowego wskaźnika poziomu sygnałów audio w lewym i prawym kanale. Aby sterować wyświetlaniem, ustawiamy aktualną wartość parametru *progress* przez wysłanie do modułu (dla chwilowej wartości 75%) następującej sekwencji znaków: `$set,pb1:p="75"/r/n`

W ostatnim bloku Layout znalazły się dwa przyciski: ESC i ENTER oraz dwie rozwijane listy typu Spinner. Pierwsza sp52 pozwala określić numer zapamiętanej częstotliwości dla 6 stacji radiowych: FM1, FM2...FM6. Druga lista pozwala wybrać tryb pracy cyfrowego procesora audio (STA310): OFF, ProL (Dolby Pro Logic), CIRC (Circle surround).

Można zauważyć, że początek każdej listy zaczyna się od neutralnych pozycji "IN?", "SPDIF?", "FM?" oraz "DPL?". Są one potrzebne, ponieważ zawsze otrzymujemy informację o wszystkich polach Spinner, a nie zawsze chcemy zmieniać cztery parametry. Dlatego jeśli np. chcemy zmienić tylko numer wejścia analogowego na np. trzecie, to ustawiamy pierwsze pole wyboru na IN3

a pozostałe zostawiamy (lub ustawiamy) na neutralne. Czyli odpowiednio na: „SPDIF?”, „FM?” oraz „DPL?”. Po naciśnięciu przycisku ENTER wiemy, że chodzi tylko o ustawienie numeru wejścia analogowego na IN3, a numer domyślnego wejścia SPDIF pozostanie niezmieniony. Analogicznie pozostawimy niezmieniony domyślny numer częstotliwości radia FM oraz nie zmienimy trybu pracy procesora audio.

Podsumowanie

Należy pamiętać, że po odwróceniu smartfona o 90° nasz Layout zostanie przerysowywany do pozycji horyzontalnej. Należy przygotować naszą aplikację także do takiej pozycji smartfona, ponieważ po odwróceniu niektóre napisy mogą stać się nieczytelne.

Chcę jeszcze wspomnieć o kontrolce EditText (której ostatecznie nie użyłem). Pozwala ona wyświetlić dowolny tekst – podobnie jak kontrolka Spinner, ale tekst należy wprowadzić. Informacja o wprowadzonym tekście zostanie przesłana dopiero po naciśnięciu na dowolny przycisk.

Trzeba przyznać, że dość tanim kosztem (moduł BBMagic można kupić za około 55 złotych) można prawie dowolnie urządzenie wyposażać w interfejs z ekranem dotykowym. Polecam użycie do tego celu „drugiego” smartfona. Zwykle każdy z nas zmienia co 2 lata swój smartfon na nowy. Warto stary, ale wciąż sprawny smartfon zastosować jako interfejs z aplikacją BBMobile.

Zbyszko Przybył
zbyszkejob@interia.pl

Chcesz czytać nasze najnowsze artykuły jeszcze przed wydrukowaniem w EP?

Zajrzyj na

www.ep.com.pl/EPwtoku