

# YAGARTO + DevCpp

1. Ściągamy dev-cpp ze strony:  
[http://sourceforge.net/projects/dev-cpp/files/Binaries/Dev-C%2B%2B%204.9.9.2/devcpp-4.9.9.2\\_setup.exe/download](http://sourceforge.net/projects/dev-cpp/files/Binaries/Dev-C%2B%2B%204.9.9.2/devcpp-4.9.9.2_setup.exe/download)
  2. Ściągamy YAGARTO GNU ARM toolchain ze strony:  
<http://sourceforge.net/projects/yagarto/files/YAGARTO%20for%20Windows/20091223/>
  3. Ściągamy FreeRTOS ze strony:  
<http://sourceforge.net/projects/freertos/files/FreeRTOS/>
  4. Instalujemy dev-cpp w katalogu C:\Dev-Cpp
  5. Instalujemy Yagarto UWAGA! koniecznie do katalogu C:\yagarto
  6. Rozpakowujemy FreeRTOS do katalogu C:\. Archiwum rozpakuje się do katalogu C:\FreeRTOSV6.0.3
  7. Otwieramy dev-cpp.  
Wybieramy Plik->Nowy->Projekt. W okienku wybieramy Pusty Projekt, C Projekt, nazwa niech zostanie Projekt1. Projekt zapisujemy w katalogu Demo mikrokontrolera którego programujemy ja programuje LCP2106 więc projekt zapisuje w katalogu C:\FreeRTOSV6.0.3\FreeRTOS\Demo\ARM7\_LPC2106\_GCC\  
8. Wybieramy Projekt->Dodaj do Projektu i wybieramy main.c
  9. Teraz dodamy YAGARTO jako kompilator naszego projektu. Wybieramy Narzędzia->Opcje kompilatora. W okienku które wyskoczyło klikamy "add new compiler set" (zielony krzyżyk +) i wpisujemy YAGARTO i klikamy OK. W tym samym okienku (Opcje kompilatora) klikamy zakładkę Katalogi, klikamy na ikonkę wyboru katalogu i wyszukujemy katalogu z plikami binarnymi YAGARTO C:\yagarto\bin. Klikamy OK. Następnie w oknie opcji kompilatora również klikamy OK. Teraz wchodzimy w Projekt->Opcje Projektu. Klikamy na zakładkę Kompilator i wybieramy z listy YAGARTO. Następnie klikamy na zakładkę Makefile zaznaczamy "Użyj Makefile użytkownika" i wyszukujemy Makefile z katalogu naszego mikrokontrolera np.  
  
C:\FreeRTOSV6.0.3\FreeRTOS\Demo\ARM7\_LPC2106\_GCC\Makefile klikamy OK.  
Następnie znowu OK w oknie opcji projektu.
10. W folderze naszego mikrokontrolera znajdują się pliki wsadowe, które ustawiają pewne parametry kompilacji i wywołują polecenie make. Np. plik rom\_arm.bat:

```
set USE_THUMB_MODE=NO
set DEBUG=
set OPTIM=-O3
set RUN_MODE=RUN_FROM_ROM
set LDSCRIPT=lpc2106-rom.ld
make
```

Musimy skopiować parametry:

```
set USE_THUMB_MODE=NO
set DEBUG=
set OPTIM=-O3
set RUN_MODE=RUN_FROM_ROM
set LDSCRIPT=lpc2106-rom.ld
```

Następnie wklejamy je na początku naszego Makefile. Teraz usuwamy "set" z każdej linijki, żeby wyglądało tak:

```
USE_THUMB_MODE=NO
DEBUG=
OPTIM=-O3
RUN_MODE=RUN_FROM_ROM
LDSCRIPT=lpc2106-rom.ld
```

11. Na koniec musimy dodać regułę all. Plikiem wynikowym naszej kompilacji będzie plik z rozszerzeniem .hex. W pliku Makefile napewno gdzieś ta nazwa występuje. W moim przypadku jedyna nazwa pliku z rozszerzeniem .hex jaką znalazłem w swoim Makefile to rtsdemo.hex. Aby dodać regułę all, na końcu pliku Makefile dopisujemy

```
all: rtsdemo.hex
```

12. Tak przygotowany projekt można kompilować w Dev-Cpp. Skompilowany plik hex nadaje się teraz do zapisania w pamięci naszego mikrokontrolera.

cegjuw@jadamspace.pl