

# **INSTALLATION AND USER GUIDE OF FFT FOR LAB@HOME**

**Tutor:** Prof. Dr. Stephen Bannwarth / **Student:** Guillermo Urizar

System Driven Hardware Design – SS2020

## FFT FOR LAB@HOME

The set of files on “fft\_application.zip” adds FFT calculation capabilities to the Lab@Home project using the PSoC without the need of Matlab. This guide focuses on describing the method input and output parameters of the function, how to use it, and how to install it in a PSoC Creator Project.

### 1. PARAMETERS

```
void fft_app(uint16 *adc_samples, int32 *fft_output, uint16 no_of_samples);
```

- **adc\_samples:** array from “main.c” that contains the time domain data from the ADC.
- **fft\_output:** array which will contain the result of the FFT calculation. It should be double the size of adc\_samples and declared on “main.c”. The result is downscaled by no\_of\_samples to prevent overflow.
- **no\_of\_samples:** number of ADC samples. It should be either 512 or 1024.

### 2. HOW TO USE

A simple sample code of how to use the function is included below. The code is compilable, but it will not produce any relevant information.

```
#include "project.h"
#include "fft_application.h"

#define SAMPLES 512

uint16 adc_array[SAMPLES];
int32  fft_array[2*SAMPLES];

int main()
{
    fft_app(adc_array, fft_array, SAMPLES);
}
```

It is important to note that the ADC samples must be obtained by an ADC component for the function to produce relevant results, although a test array might be also used to replace it as a testing or debugging step.

### 3. INSTALLATION IN PSOC CREATOR 4.3

**Step 1:** In the folder of the PSoC Project, copy and paste the folder which is inside of “fft\_application.zip”. Taking a look inside (Figure 3.1), there should be five files (four header files “.h” and one static library file “.a”).

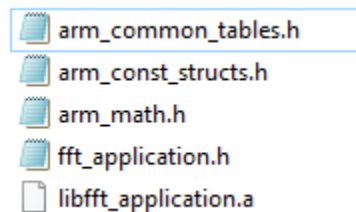


Figure 3.1

**Step 2:** Access the “Build Settings...” in the context menu of the PSoC project (Figure 3.2).

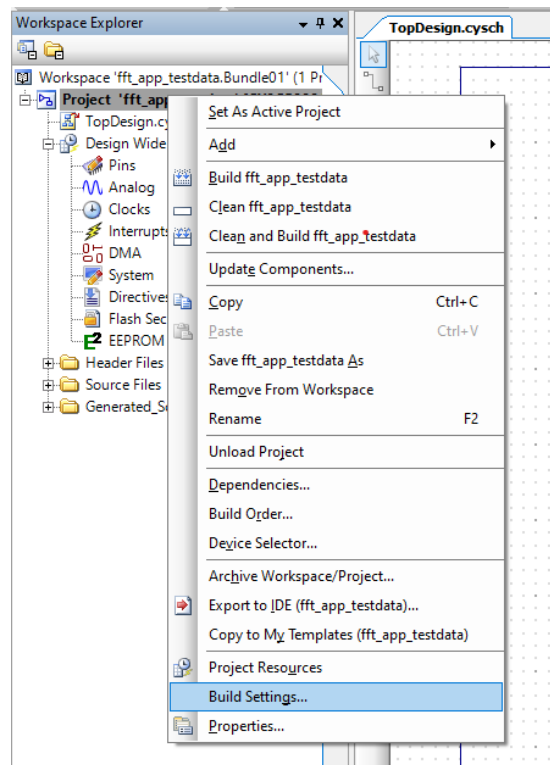


Figure 3.2

**Step 3:** On the “General” tab on the “Compiler” settings, add the directory of the folder added on “Step 1” on “Additional Include Directories” and the macro “ARM\_MATH\_CM3” on “Preprocessor Definitions” (Figure 3.3).

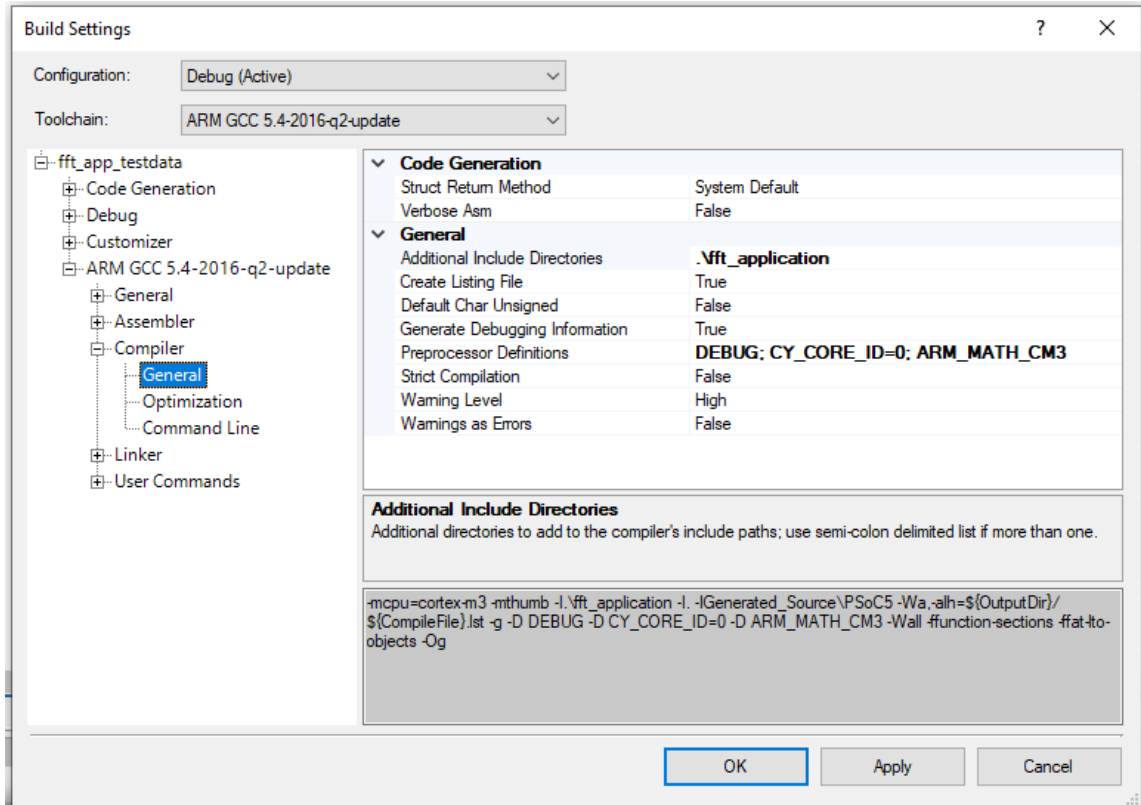


Figure 3.3

**Step 4:** On the “General” tab on the “Linker” settings, add the library “fft\_application” on “Additional Libraries” and the directory of the folder added on “Step 1” on “Additional Library Directories” (Figure 3.4). Apply the changes and close the “Build Settings” window.

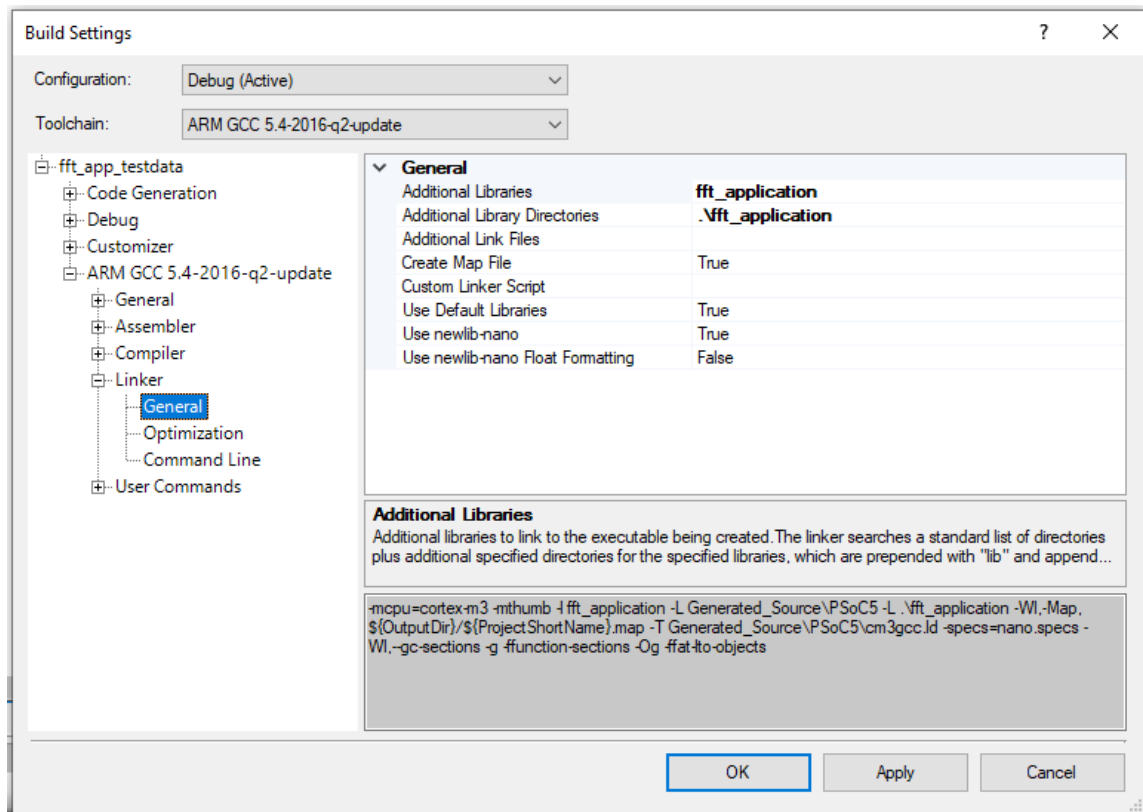


Figure 3.4

**Step 5:** On the “System” tab of the PSoC Project, increase the “Heap Size” to 0x2000 (Figure 3.5). **Note:** a possible reason for incoherent results could be an insufficient heap size, so increasing it might solve it.

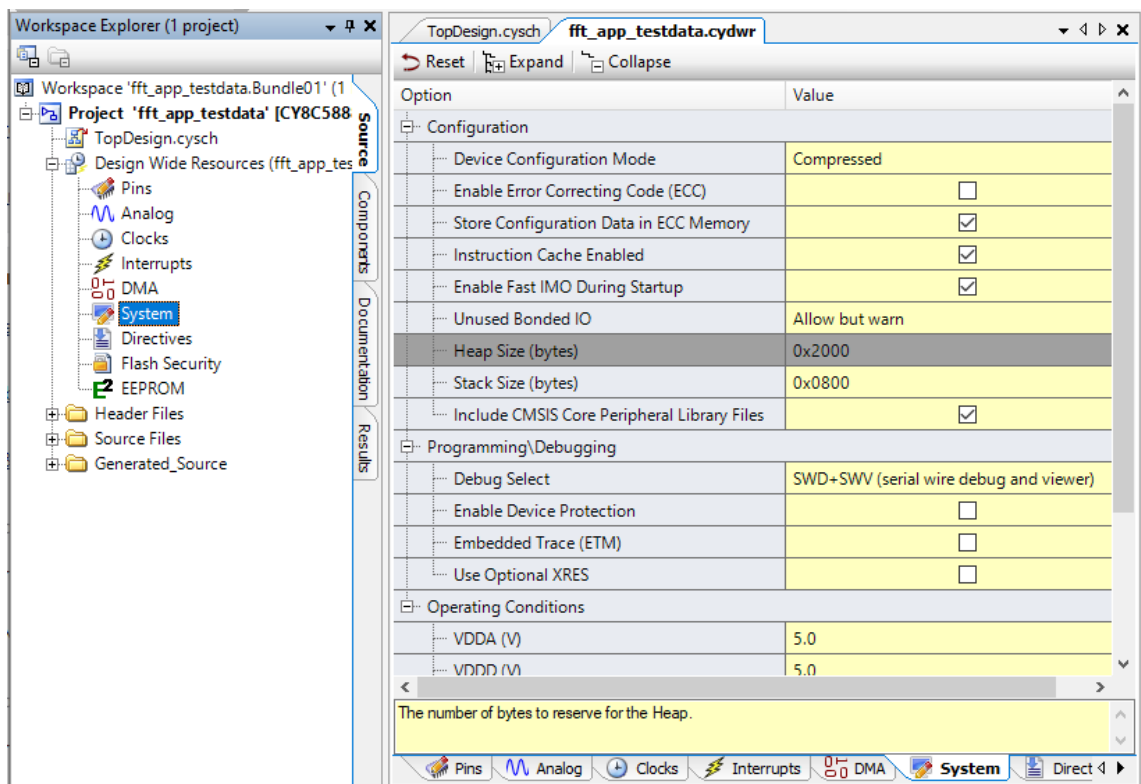


Figure 3.5

**Step 6:** As seen on the sample code on Section 2, to use the function, the header file "fft\_application.h" must be included on the code.