

# Configuração do DS-5 para programas em *assembly* ARMv8 e criação de primeiro programa

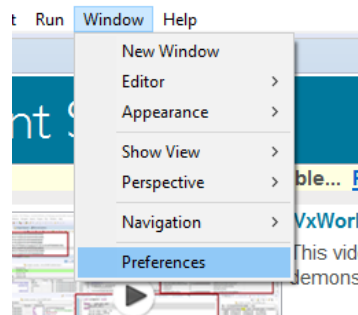
Para criar um projeto DS-5 para programas escritos em linguagem *assembly* ARM V8, pode proceder-se da forma indicada a seguir. Este tutorial assume que o computador já tem instalado o Arm DS-5 CE (disponível [aqui](#)) e que já foi descarregada para o computador a *toolchain* da Linaro (disponível [aqui](#)). Os computadores da FEUP já cumprem estes requisitos.

Um projeto já configurado está disponível [aqui](#) (em formato ZIP).

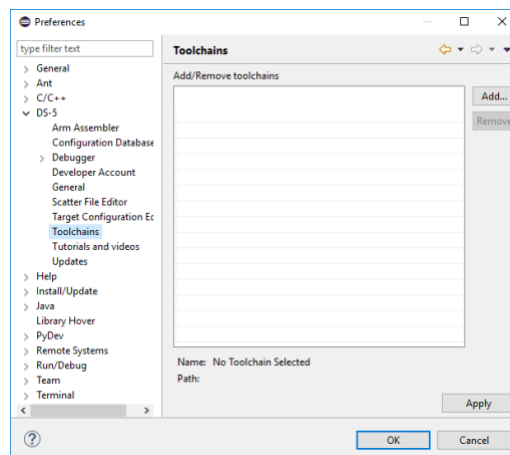
## 1. Adicionar *toolchain*

De forma a poder compilar um projeto é necessário adicionar um conjunto de ferramentas de compilação. Nesta unidade curricular vamos utilizar o *bare metal compiler* da Linaro.

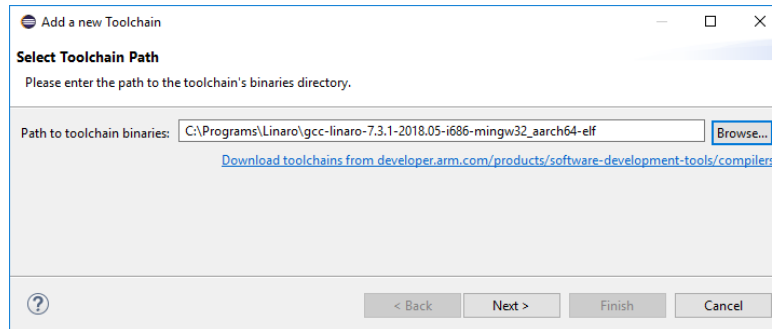
1.1 - Após iniciar DS-5 clicar em **Windows** -> **Preferences**.



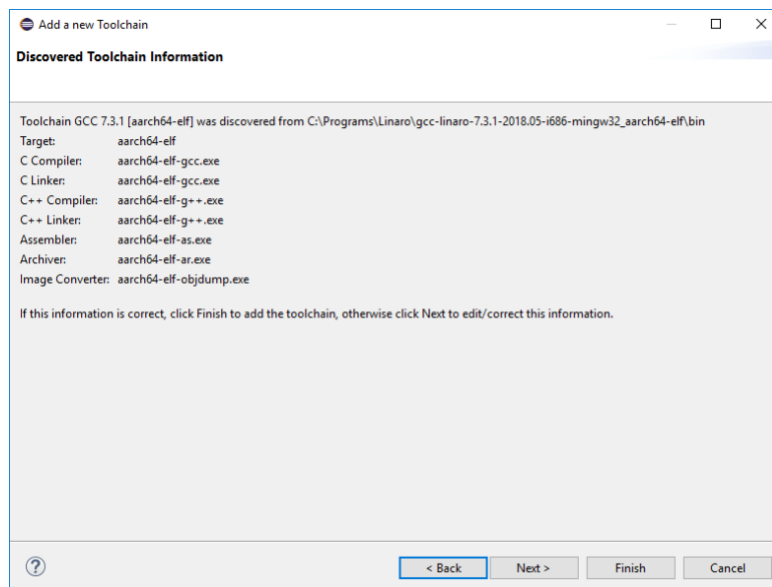
1.2 – Expandir a opção DS-5 e clicar em **Toolchains** -> **Add**.



- 1.3 – Clicar em **Browse...** e indicar o caminho onde se encontra a *toolchain*. (No caso dos computadores da FEUP a toolchain está no endereço [C:\Programs\Linaro\gcc-linaro-7.3.1-2018.05-i686-mingw32\\_aarch64-elf](#), se estiver a instalar no seu computador pessoal deverá indicar aqui o local para onde extraiu o ficheiro correspondente à *toolchain*).

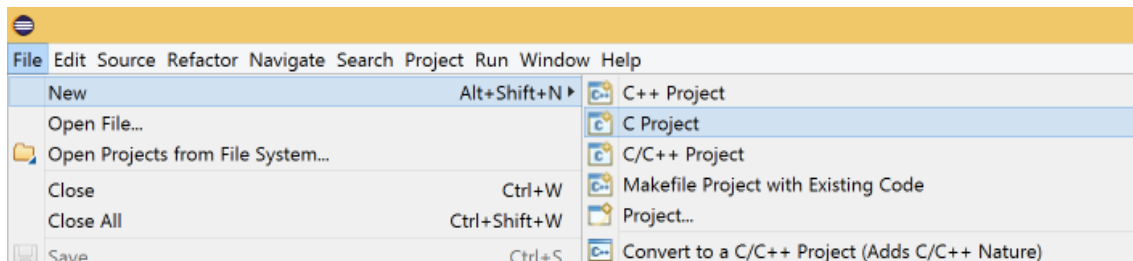


- 1.4 – Clicar em **Next** (se tudo correr bem o sistema irá detetar a toolchain como demonstra a imagem seguinte) e clicar em **Finish**.

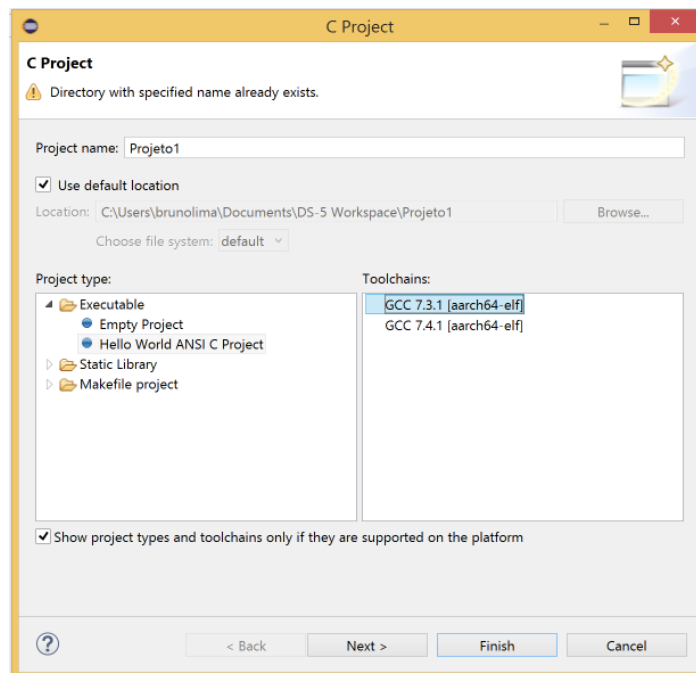


## 2. Criar um projeto novo

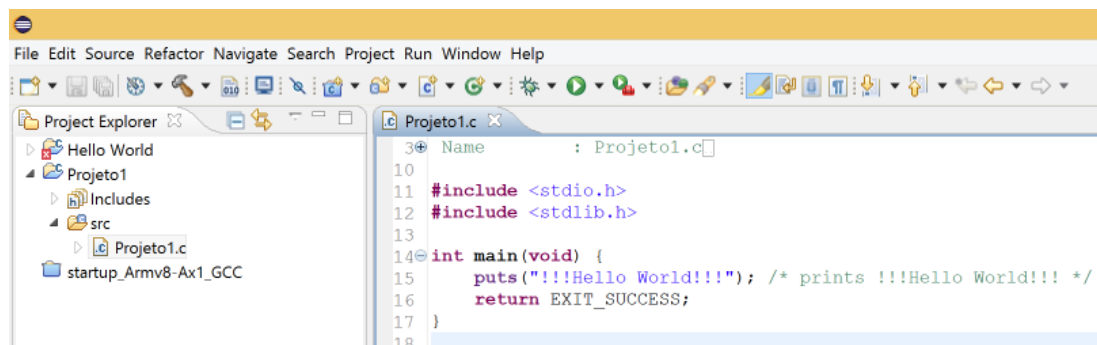
2.1 – No menu **File** do DS-5 selecione **New** -> **C Project**.



2.2 – Na janela seguinte introduza o nome do projeto “**Projeto1**”, selecione como tipo de projeto “**Hello World ANSI C Project**”, escolha a respetiva Toolchain “**GCC 7.3.1**” (ou uma versão mais recente que tenham instalado) e clique em **Finish**.

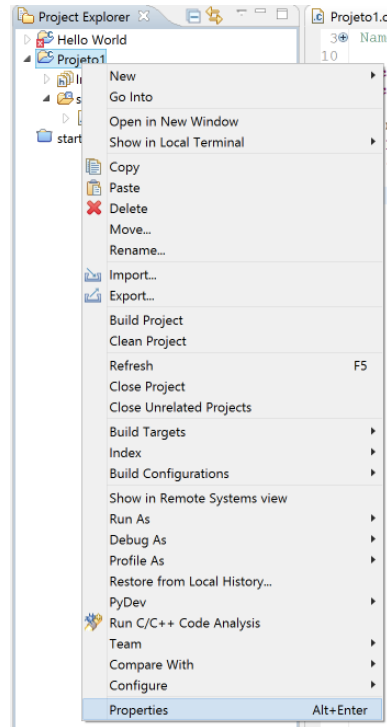


2.3 – Se tudo correu bem, o projeto deverá estar visível na janela “Project Explorer”

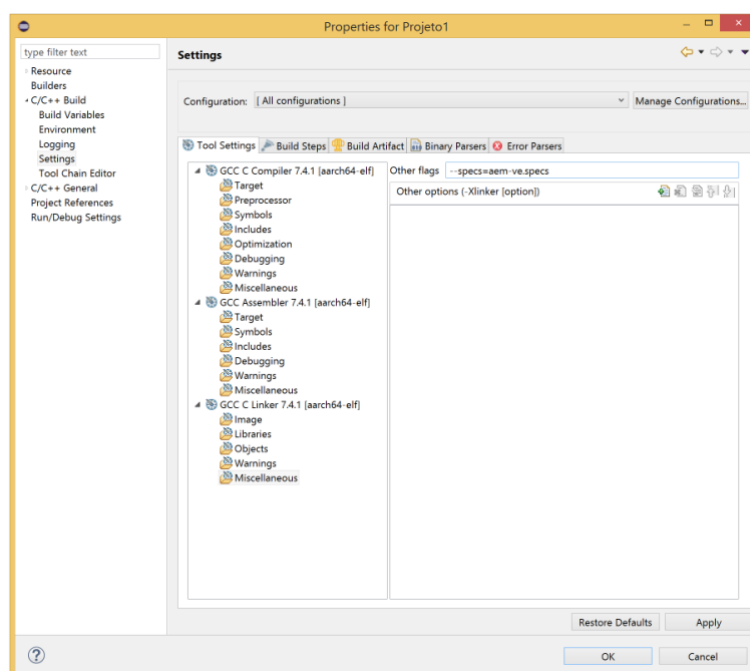


### 3. Modificar as opções de configuração do projeto

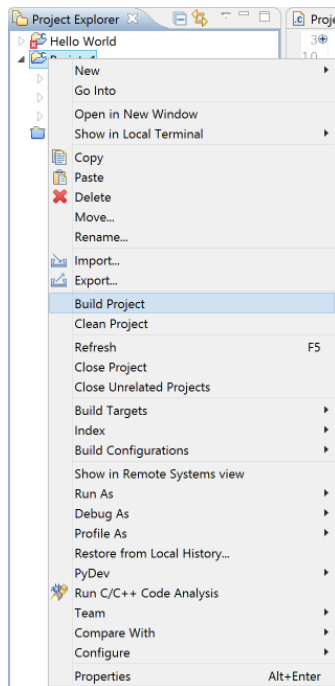
3.1 – Na janela **Project Explorer**, clique com o botão direito do rato sobre a pasta “Projeto1” e de seguida clique em **Properties**.



- 3.2 – a) Na janela seguinte selecione no menu **C/C++ Build** selecione a opção **Settings**.  
b) Em **Configuration** (no topo) selecione a opção **[ All configurations ]**.  
c) Em **GCC C Linker** clique em **Miscellaneous** e, na opção **Other flags**, introduza “--specs=aem-ve.specs”.  
d) Clique em **Ok**.

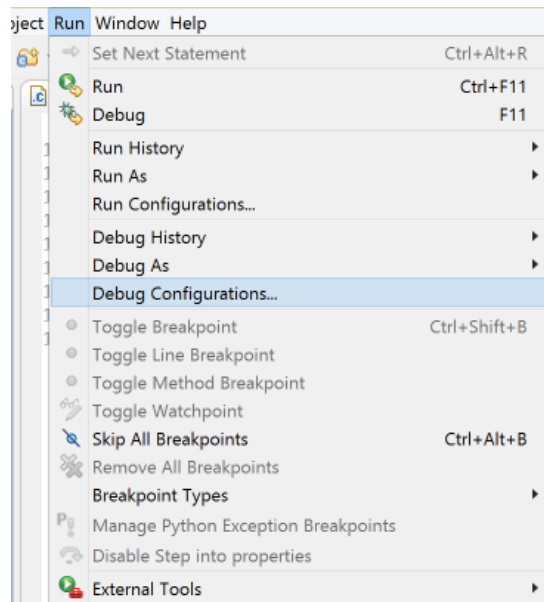


3.3 – Para confirmar que a configuração correu bem, deverá agora clicar com o botão direito do rato na pasta do projeto e seleccionar a opção **Build Project**. Após isso deverá ser automaticamente criada uma pasta **Debug** contendo, entre outros, o ficheiro **Projeto1.axf**.

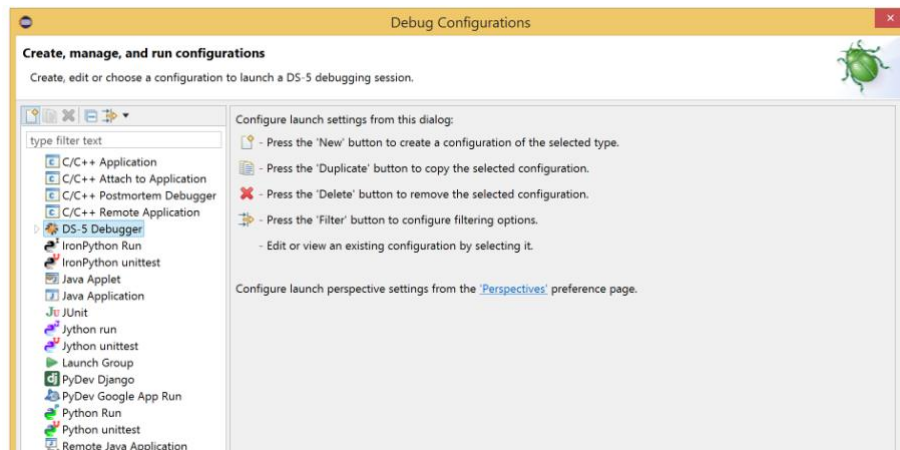


## 4. Criar uma configuração de Debug para o DS-5

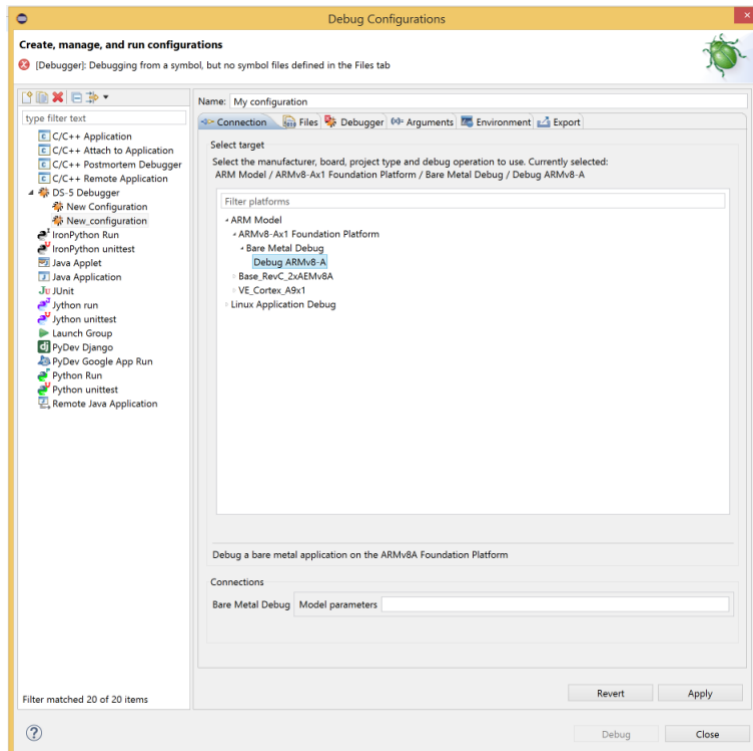
4.1 – No menu do DS-5 clique em **Run -> Debug Configurations**.



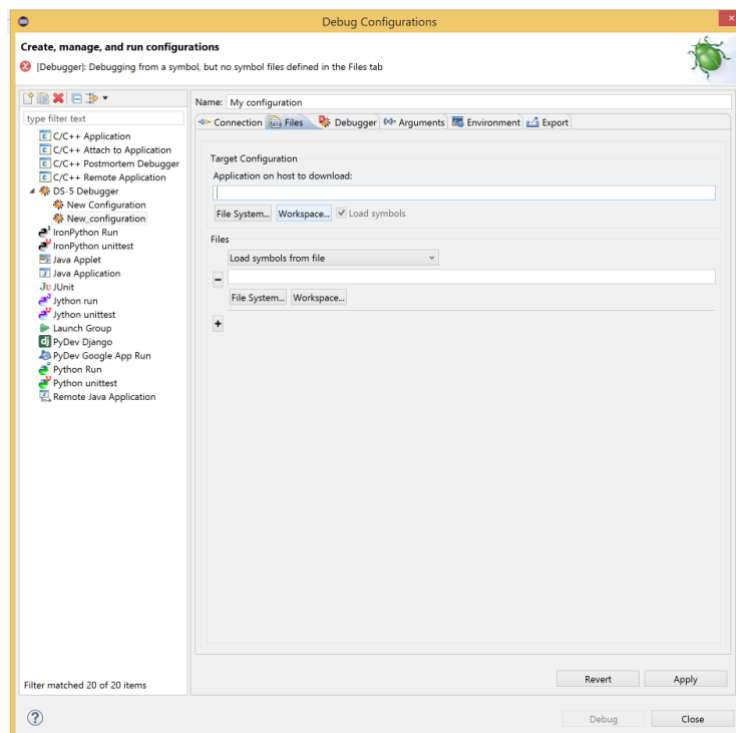
4.2 – Na janela de configuração expanda a opção **DS-5 Debugger** e clique em “**New configuration**”.



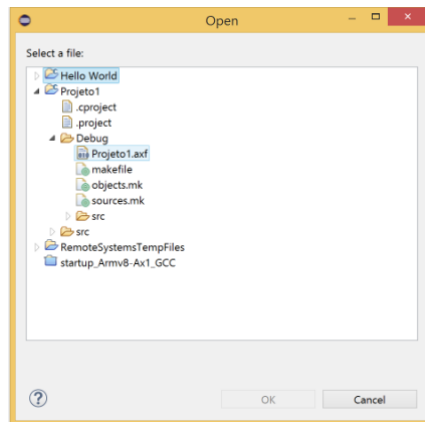
4.3 – Na janela da nova configuração indique um nome para a configuração, por exemplo “**MPCP\_configuration**”. No separador **Connection** procure e selecione a opção “**Arm Model > Armv8-Ax1 Foundation Platform > Bare Metal Debug > Debug Armv8-A**”.



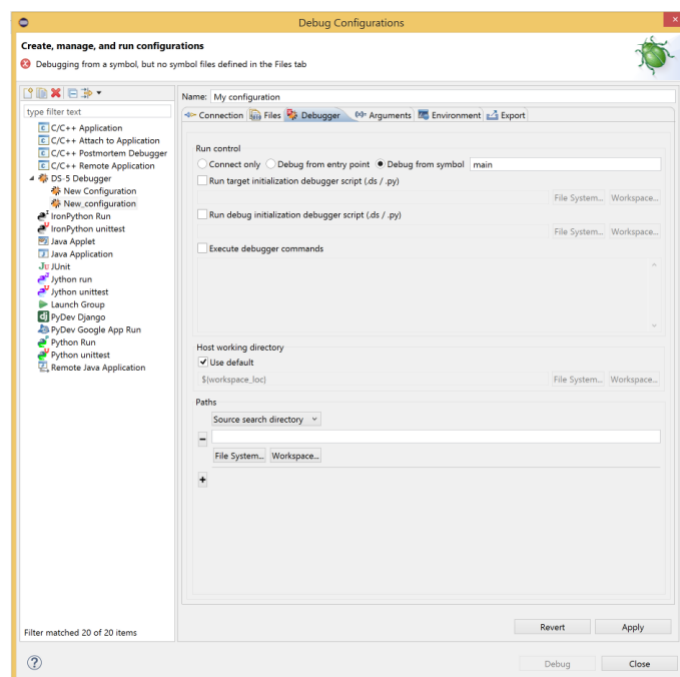
4.4 – No separador **Files** em **Target Configuration** clique em **Workspace**.



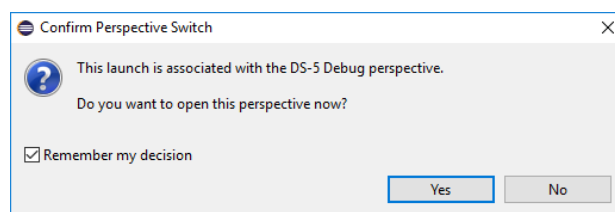
4.5 – Selecione o ficheiro **Projeto1 > Debug > Projeto1.axf** (criado no ponto 3.3).



4.6 – No separador **Debugger** confirme que a opção “**Debug from symbol**” está selecionada e que o seu valor é “**main**”.

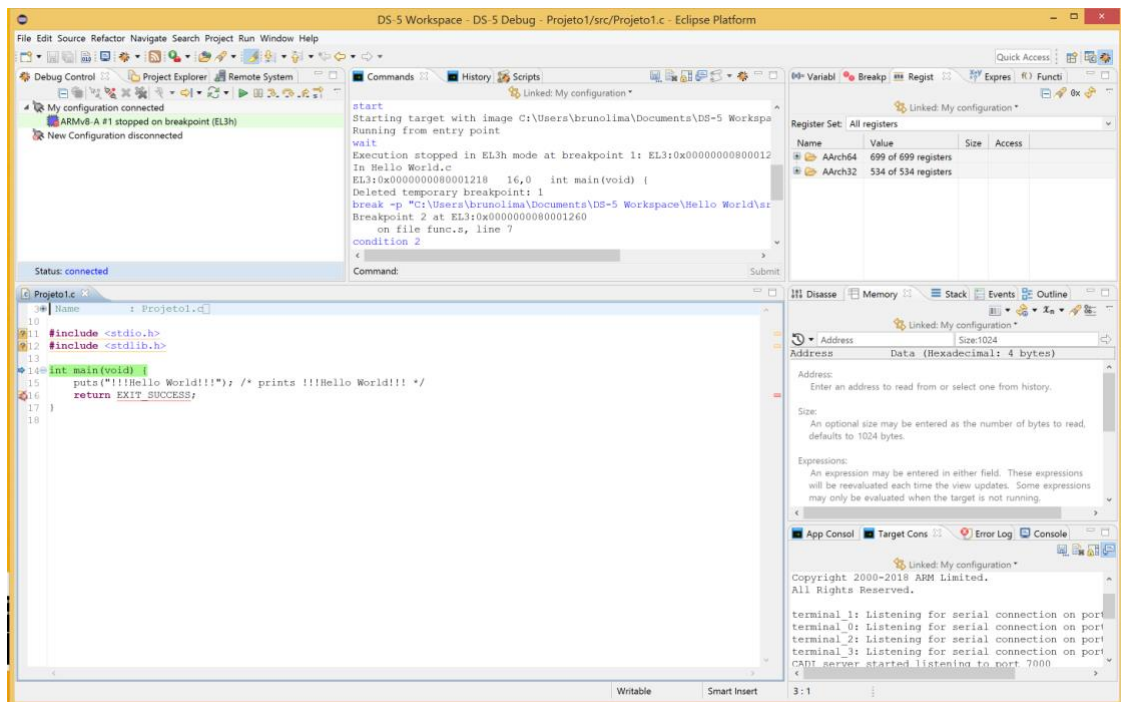


4.7 – Clique em **Debug** e na caixa de diálogo que surge responda **Yes**.





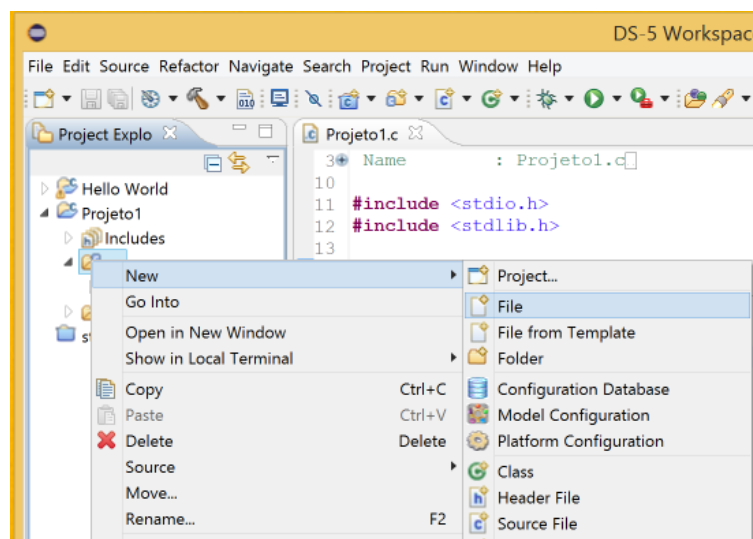
4.8 – Se tudo correu bem, irá ser apresentada a vista de Debug.



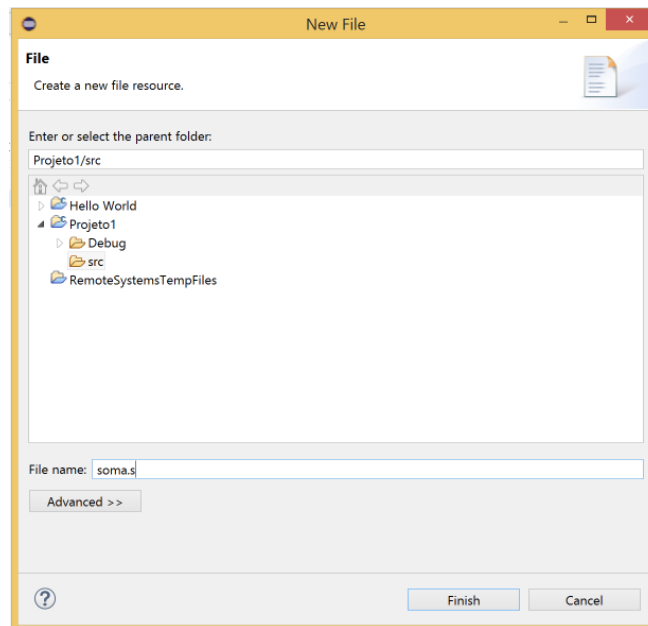
## 5. Criar a primeira sub-rotina em *assembly*

5.1 – Regressar à vista C/C++ clicando no respetivo botão no canto superior direito da *framework*.

5.2 – Na secção **Project Explorer** clicar com o botão direito do rato sobre a pasta “**src**” do projeto e seleccionar a opção **New > File**.



5.2 – Na janela seguinte deve dar um nome ao ficheiro *assembly*, por exemplo “soma.s” e clicar em **Finish**.



5.3 – No ficheiro **soma.s** vamos então criar a nossa primeira sub-rotina em *assembly*. A nossa sub-rotina **SOMA**, recebe dois argumentos e devolve o resultado da sua soma. Esta sub-rotina poderá posteriormente ser chamada a partir de código C. Para isso vamos colocar o seguinte código (não esquecer de gravar o ficheiro no final):

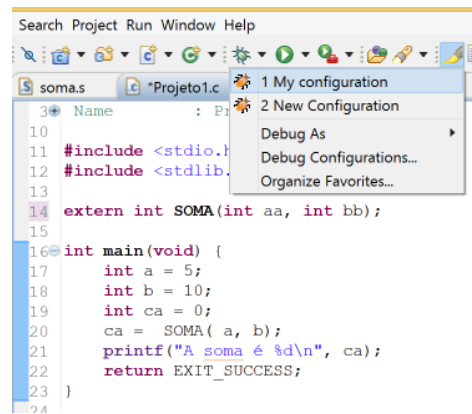
```
soma.s x Projeto1.c
1 .text
2 .global SOMA
3 .type SOMA,"function"
4
5 SOMA:  add    X0, X0, X1    // comentário
6        ret
```

5.4 – No ficheiro **Projeto1.c** vamos agora colocar o código que permitirá chamar a nossa sub-rotina e apresentar o resultado na consola (não esquecer de gravar o ficheiro no final).

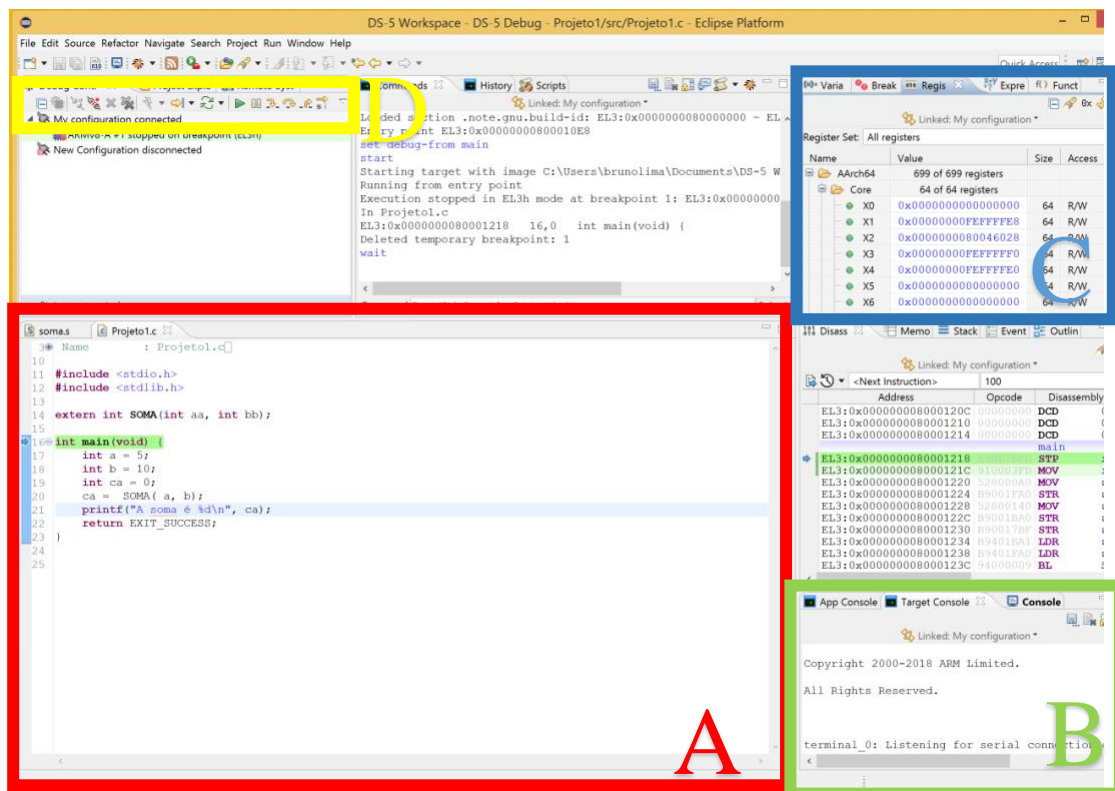
```
soma.s x *Projeto1.c x
3+ Name      : Projeto1.c
10
11 #include <stdio.h>
12 #include <stdlib.h>
13
14 extern int SOMA(int aa, int bb);
15
16 int main(void) {
17     int a = 5;
18     int b = 10;
19     int ca = 0;
20     ca = SOMA( a, b);
21     printf("A soma é %d\n", ca);
22     return EXIT_SUCCESS;
23 }
```

5.5 – Uma vez alterado o projeto é agora necessário voltar a compilar, para isso clicamos com o botão direito do rato na pasta do projeto e escolhemos a opção **Build Project** (à semelhança do que foi feito no passo 3.3).

5.6 – Uma vez compilado o projeto vamos executá-lo na vista de *debug*. Para isso clicamos no icon **Debug** e seleccionamos a opção “**MPCP\_configuration**”.



5.7 – Com o passo anterior voltamos à vista *debug* tal como apresentado na imagem seguinte:



Nesta vista temos quatro áreas principais que correspondem a:

- A - Código fonte
- B - Registos e seus valores atuais
- C - Consola
- D - Opções de *debug*

5.8 – Explore as diferentes opções de *debug*:

- a) Execução completa – Verifique que a mensagem é apresentada corretamente na consola.
- b) Execução passo a passo – Verifique os valores que os registos X0 e X1 assumem durante a execução e modifique os valores dos mesmos.