J5Eco Application Note

CortexM3 access

Calin Tamaian , June 2013

.1.Introduction

This document shows take M3 RTOS and M3 ISS out of reset. To the M3 SecSS core you should have access by default. Although this User Guide refers to J5Eco-GP, the same procedure is applied to J5Eco-HS.

.2. Requirements

J5Eco environment This document assumes that you have a working environment with J5Eco and that you are aware on how to make a Hello World project on a J5Eco board (Cortex-A8 core). If you are not familiar with this please follow the J5Eco Startup Guide.

JTAG Emulator In this guide XDS560V2 emulator was used but any emulator compatible with TI811x / TI814x will work.

Code Composer Studio In this guide CCS 5.3.0 was used. Other versions should work as well.

http://processors.wiki.ti.com/index.php/Download_CCS

.3.Procedure

In order to access the M3- ISS and M3-RTOS we need to enable the Ducati clocks. There are two ways that this can be achieved.

Procedure 1 – No GEL file

If no GEL file is used than we need to enable the Ducati clocks .This can be done by calling the following function when running the Cortex-A8 core:

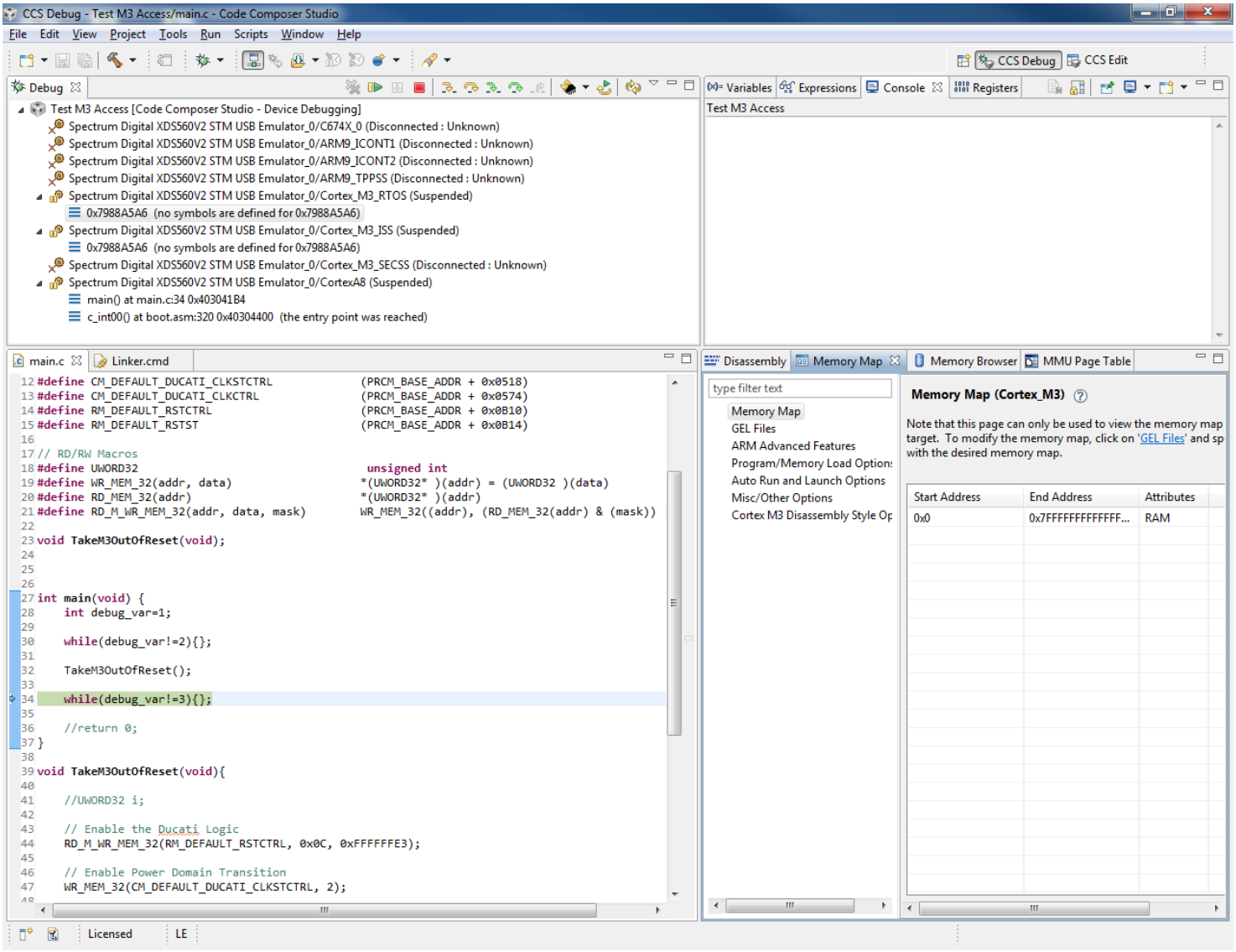
```
void TakeM3OutOfReset(void){

	// Enable the Ducati Logic
	RD_M_WR_MEM_32(RM_DEFAULT_RSTCTRL, 0x0C, 0xFFFFFFE3);

	// Enable Power Domain Transition
	WR_MEM_32(CM_DEFAULT_DUCATI_CLKSTCTRL, 2);

	// Enable Ducati Clocks
	WR_MEM_32(CM_DEFAULT_DUCATI_CLKCTRL, 2);

	// Check CLKIN200TR & CLKINTR are active
	while (((RD_MEM_32(CM_DEFAULT_DUCATI_CLKSTCTRL) & 0x300) >> 8) != 3) ;

	//Write Ducati IRAM Boot Image - address not accessible
	//WR_MEM_32(DUCATI_BASE_ADDR, 0x10000);
	//WR_MEM_32(DUCATI_BASE_ADDR + 0x4, 0x9);
	//WR_MEM_32(DUCATI_BASE_ADDR + 0x8, 0xE7FEE7FE);

	// Reset both CPU's
	RD_M_WR_MEM_32(RM_DEFAULT_RSTCTRL, 0x00, 0xFFFFFFE3);
	// Check for Ducati M3_0 & M3_1 out of Reset
	while (((RD_MEM_32(RM_DEFAULT_RSTST) & 0x1C)) != 0x1C) ;
	// Check Module is in Functional Mode
	while (((RD_MEM_32(CM_DEFAULT_DUCATI_CLKCTRL) & 0x30000) >> 16) != 0) ;
}
```

Once this function is executed on Cortex A8, you can connect to the M3 cores via debugger. Test example:

```
/*
 * This Software takes the M3 cores out of reset
 */

//Include files
#include <stdlib.h>
#include <stdint.h>
#include <stdbool.h>

#define PRCM_BASE_ADDR                                          0x48180000

#define CM_DEFAULT_DUCATI_CLKSTCTRL                         (PRCM_BASE_ADDR + 0x0518)
#define CM_DEFAULT_DUCATI_CLKCTRL                           (PRCM_BASE_ADDR + 0x0574)
#define RM_DEFAULT_RSTCTRL                                  (PRCM_BASE_ADDR + 0x0B10)
#define RM_DEFAULT_RSTST                                    (PRCM_BASE_ADDR + 0x0B14)

// RD/RW Macros
#define UWORD32                                                     unsigned int
#define WR_MEM_32(addr, data)                              *(UWORD32* )(addr) = (UWORD32 )(data)
#define RD_MEM_32(addr)                                          *(UWORD32* )(addr)
#define RD_M_WR_MEM_32(addr, data, mask)             WR_MEM_32((addr), (RD_MEM_32(addr) & (mask))
| (data))

void TakeM3OutOfReset(void);

int main(void) {
	int debug_var=1;

	while(debug_var!=2){};

	TakeM3OutOfReset();

	while(debug_var!=3){};

	//return 0;
}

void TakeM3OutOfReset(void){

	//UWORD32 i;

	// Enable the Ducati Logic
	RD_M_WR_MEM_32(RM_DEFAULT_RSTCTRL, 0x0C, 0xFFFFFFE3);

	// Enable Power Domain Transition
	WR_MEM_32(CM_DEFAULT_DUCATI_CLKSTCTRL, 2);

	// Enable Ducati Clocks
	WR_MEM_32(CM_DEFAULT_DUCATI_CLKCTRL, 2);

	// Check CLKIN200TR & CLKINTR are active
	while (((RD_MEM_32(CM_DEFAULT_DUCATI_CLKSTCTRL) & 0x300) >> 8) != 3) ;

	//Write Ducati IRAM Boot Image - address not accesible
	//WR_MEM_32(DUCATI_BASE_ADDR, 0x10000);
	//WR_MEM_32(DUCATI_BASE_ADDR + 0x4, 0x9);
	//WR_MEM_32(DUCATI_BASE_ADDR + 0x8, 0xE7FEE7FE);
```

```
        // Reset both CPU's
        RD_M_WR_MEM_32(RM_DEFAULT_RSTCTRL, 0x00, 0xFFFFFFE3);
        // Check for Ducati M3_0 & M3_1 out of Reset
        while (((RD_MEM_32(RM_DEFAULT_RSTST) & 0x1C)) != 0x1C) ;

        // Check Module is in Functional Mode
        while (((RD_MEM_32(CM_DEFAULT_DUCATI_CLKCTRL) & 0x30000) >> 16) != 0) ;
}
```

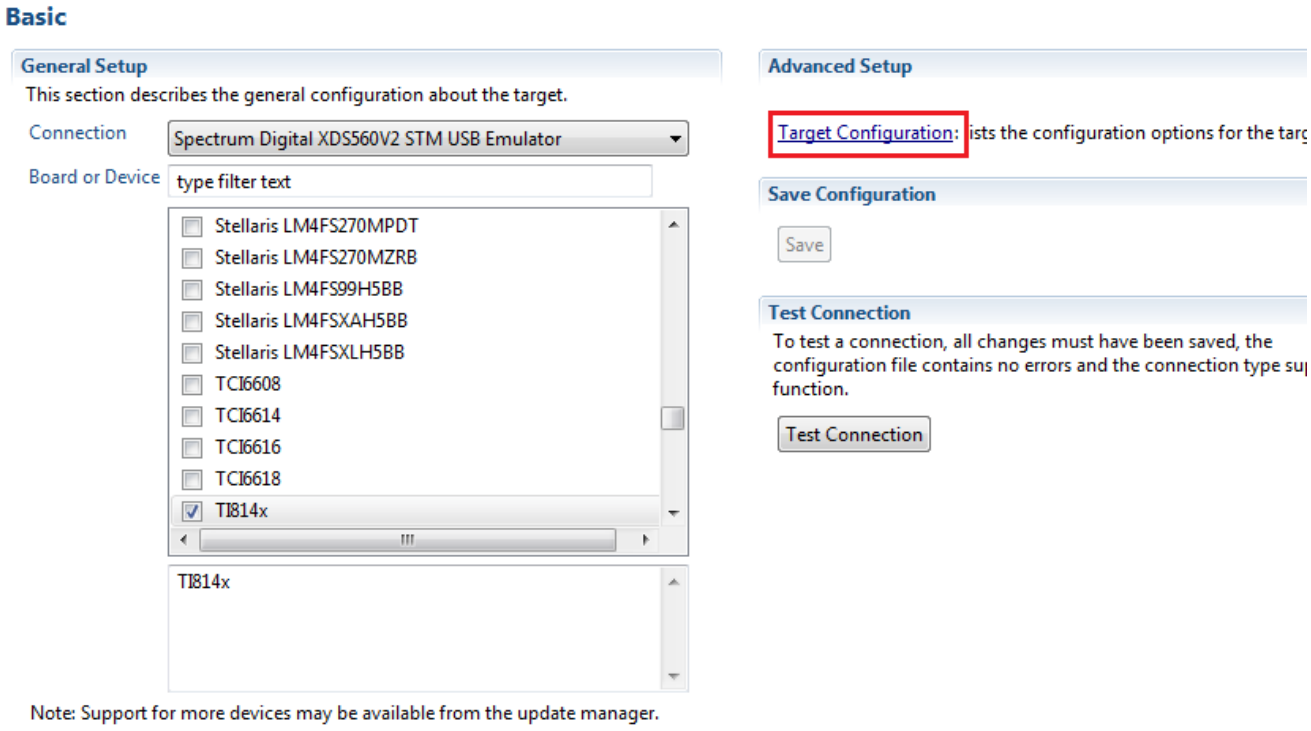
After we run the software the CPU will stop before executing the M3 access function. At this point we can check and see that we do not have access to ISS – M3 and RTOS – M3 cores. After we modify the debug_var to 2 we can run the software until the next while loop. At this point we should be able to access both M3 cores.

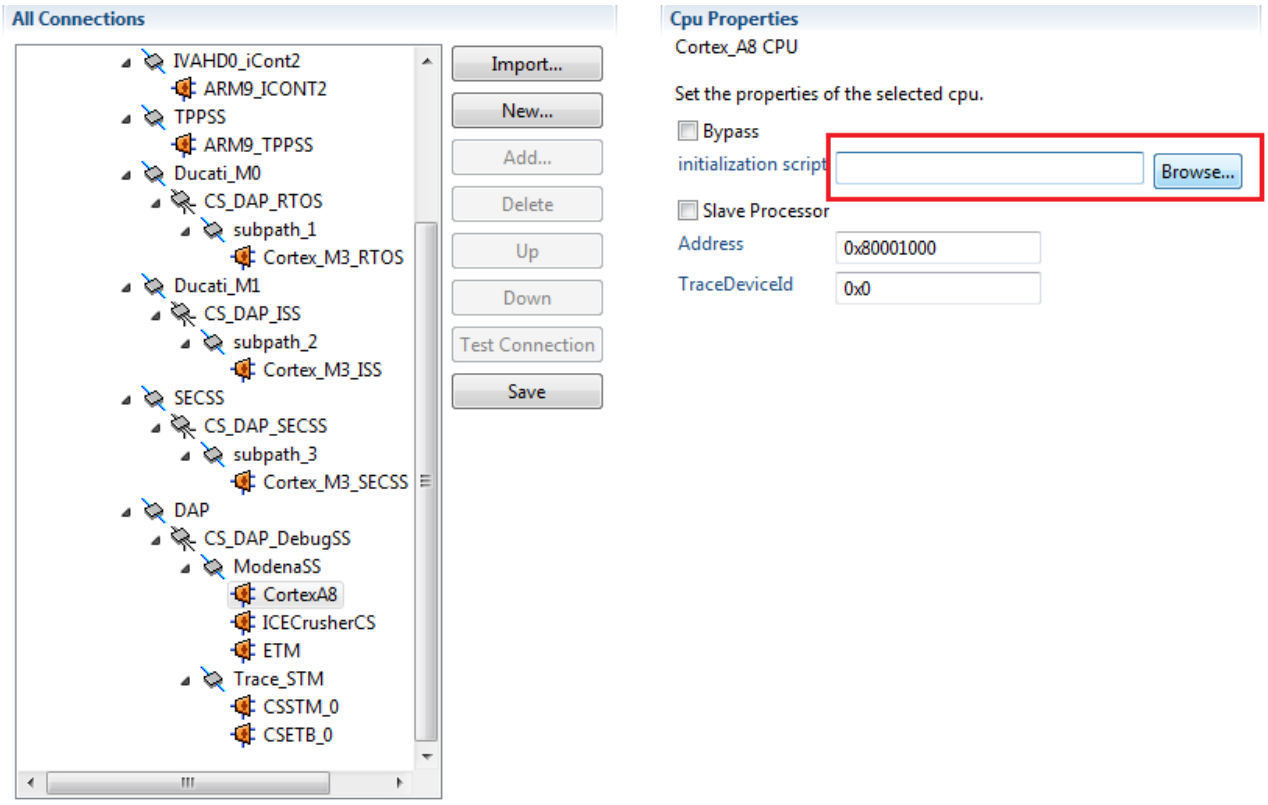


Procedure 2 – with GEL file

Another way to take the M3 cores out of reset is to use a predefined script of the GEL file provided by Spectrum Digital software package. The GEL file can be found here: http://support.spectrumdigital.com/boards/evm811x/revc/

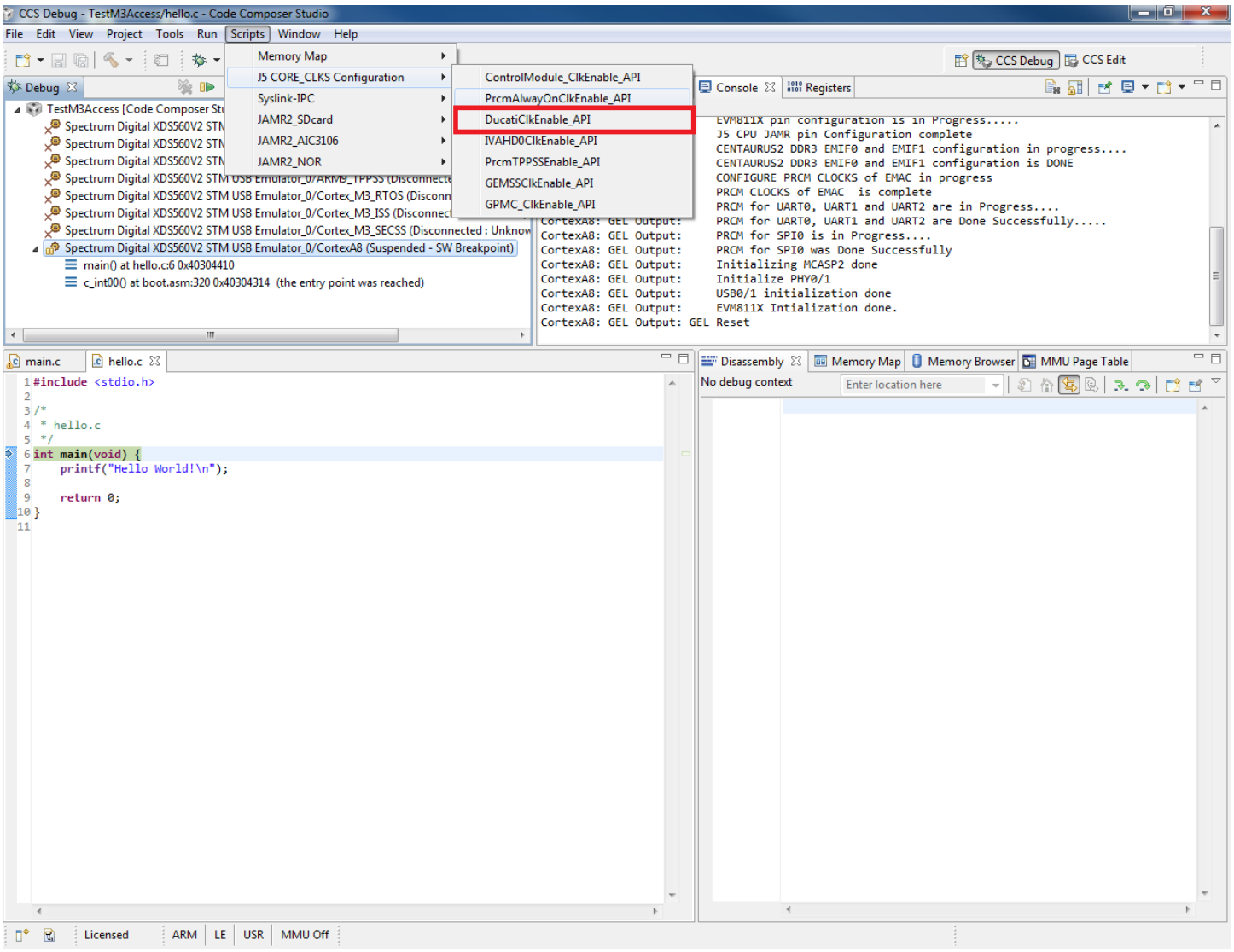
To add the GEL file to your Target Configuration, select Advance Setup and set the initialization script of Cortex A8:



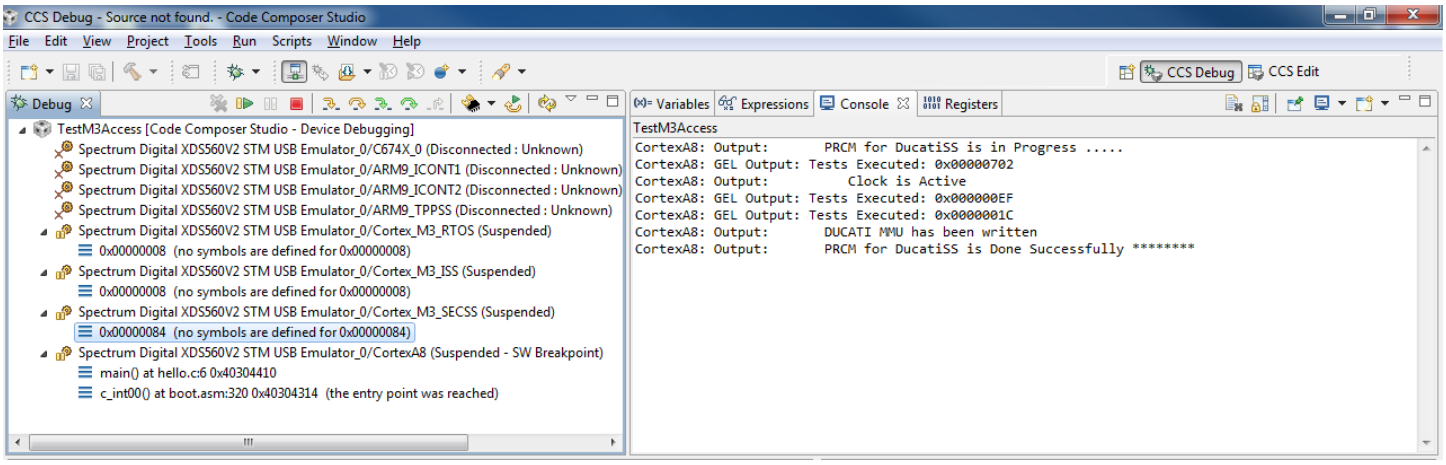


After you go in Debug Mode you connect to CortexA8 and use the DucatiClkEnable_API script:

CCS Upper Toolbar -> Scripts ->J5 Core_CLKS Configuration -> DucatiClkEnable_API



On the Console you should see the confirmation that your script is being executed.



Now we can access all M3 cores.