**Лабораторная работа № 5**

**Цель работы:** Изучение основных особенностей работы с аналого-цифровым преобразователем (АЦП) и считывание данных со встроенного датчика температуры.

**Приборы и материалы:**

1. Отладочная плата MDR1986VE91T Rev 4
2. Программатор J-Link ARM
3. Блок питания 5В, 1.4А
4. ПК с установленной средой программирования Keil uVision

**Ход работы**

1. Собрать аппаратную часть по рекомендациям лабораторной работы № 2
2. Открыть проект MDRProject в среде программирования Keil uVision.
3. Подключить к проекту библиотеку ADC, необходимую для работы с АЦП.

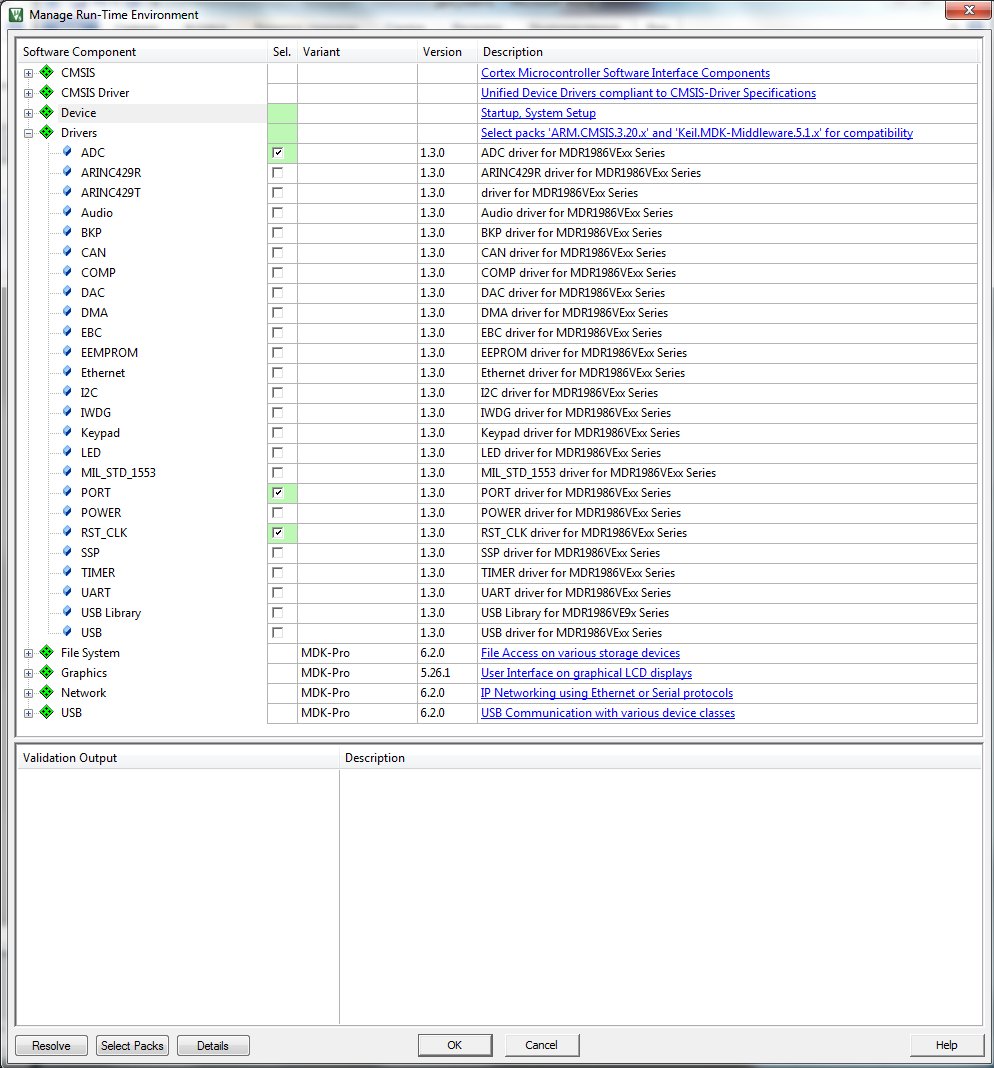


Рисунок 1 – Библиотека проекта для работы АЦП

1. Добавить в заголовок основного файла исходного кода *main.c* ссылки на заголовочные файлы библиотек:

#include <MDR32F9Qx\_port.h>

#include <MDR32F9Qx\_rst\_clk.h>

#include <MDR32F9Qx\_adc.h>

#include <MDR32F9Qx\_dac.h>

Массив двухбайтных беззнаковых чисел из десяти элементов

uint16\_t ADCConvertedValue[10];

Определение двух переменных для инициализации АЦП

ADC\_InitTypeDef ADC;

ADCx\_InitTypeDef ADC1;

Функция инициализации АЦП

void ADCInit(){

Включение тактирования АЦП и тактирования периферии

RST\_CLK\_PCLKcmd(RST\_CLK\_PCLK\_RST\_CLK | RST\_CLK\_PCLK\_ADC, ENABLE);

Начальная инициализация переменной ADC

ADC\_StructInit(&ADC);

Разрешение работы температурного сенсора

ADC.ADC\_TempSensor = ADC\_TEMP\_SENSOR\_Enable;

Разрешение преобразования сигнала температурного сенсора

ADC.ADC\_TempSensorConversion = ADC\_TEMP\_SENSOR\_CONVERSION\_Enable;

Вызываем инициализацию самого АЦП с указанными выше параметрами

ADC\_Init(&ADC);

Настройка АЦП1, начальная инициализация переменной ADC1

ADCx\_StructInit(&ADC1);

Выбор канала АЦП1 как канала температурного сенсора

ADC1.ADC\_ChannelNumber = ADC\_CH\_TEMP\_SENSOR;

Определение частоты преобразования АЦП1

ADC1.ADC\_Prescaler = ADC\_CLK\_div\_512;

Вызываем инициализацию самого АЦП1 с указанными выше параметрами

ADC1\_Init(&ADC1);

Разрешение прерывания АЦП

NVIC\_EnableIRQ(ADC\_IRQn);

Установка наивысшего приоритета прерывания

NVIC\_SetPriority(ADC\_IRQn, 0);

Конфигурация АЦП1 на прерывание по завершению преобразования

ADC1\_ITConfig(ADCx\_IT\_END\_OF\_CONVERSION, ENABLE);

Разрешение работы АЦП

ADC1\_Cmd(ENABLE);

}

Процедура обработки прервывания АЦП

uint8\_t result;

void ADC\_IRQHandler() {

Проверка статуса прерывания

if(ADC\_GetITStatus(ADC1\_IT\_END\_OF\_CONVERSION)){

В случае если преобразование АЦП завершено, результат считывается result = ADC1\_GetResult();

Очищается бит соответствующего прерывания.

NVIC\_ClearPendingIRQ(ADC\_IRQn);

}

}

Главный цикл программы:

int main() {

ADCInit();

ADC1\_Start();

while(1){

}

}

Весь код программы:

#include <MDR32F9Qx\_port.h>

#include <MDR32F9Qx\_rst\_clk.h>

#include <MDR32F9Qx\_adc.h>

#include <MDR32F9Qx\_dac.h>

uint16\_t ADCConvertedValue[10];

ADC\_InitTypeDef ADC;

ADCx\_InitTypeDef ADC1;

void ADCInit(){

RST\_CLK\_PCLKcmd(RST\_CLK\_PCLK\_RST\_CLK | RST\_CLK\_PCLK\_ADC, ENABLE);

ADC\_StructInit(&ADC);

ADC.ADC\_TempSensor = ADC\_TEMP\_SENSOR\_Enable;

ADC.ADC\_TempSensorConversion = ADC\_TEMP\_SENSOR\_CONVERSION\_Enable;

ADC\_Init(&ADC);

ADCx\_StructInit(&ADC1);

ADC1.ADC\_ChannelNumber = ADC\_CH\_TEMP\_SENSOR;

ADC1.ADC\_Prescaler = ADC\_CLK\_div\_512;

ADC1\_Init(&ADC1);

//Int

NVIC\_EnableIRQ(ADC\_IRQn);

NVIC\_SetPriority(ADC\_IRQn, 0);

ADC1\_ITConfig(ADCx\_IT\_END\_OF\_CONVERSION, ENABLE);

ADC1\_Cmd(ENABLE);

}

uint8\_t result;

void ADC\_IRQHandler() {

if(ADC\_GetITStatus(ADC1\_IT\_END\_OF\_CONVERSION)){

result = ADC1\_GetResult();

NVIC\_ClearPendingIRQ(ADC\_IRQn);

}

}

int main() {

ADCInit();

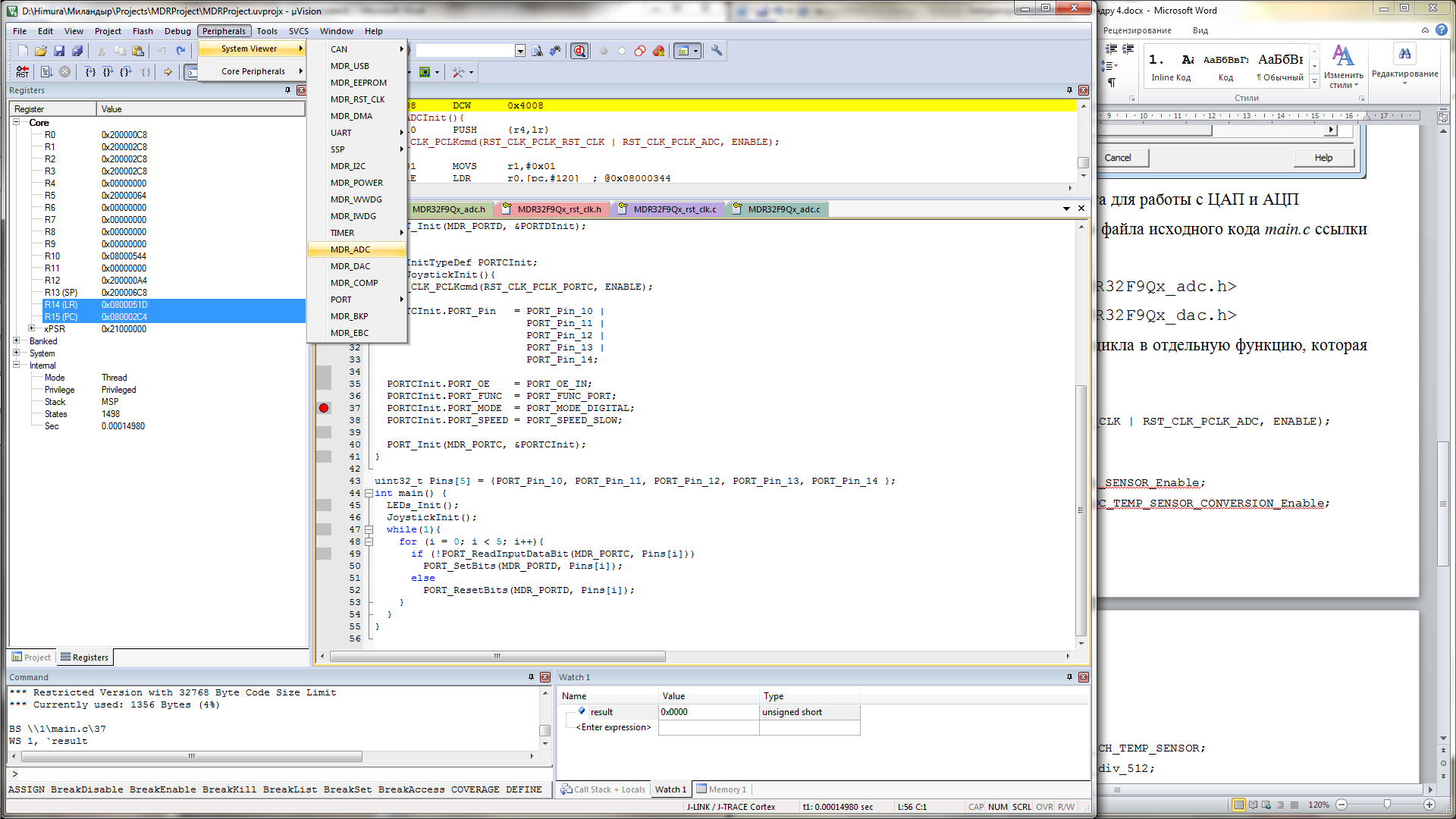
ADC1\_Start();

while(1){

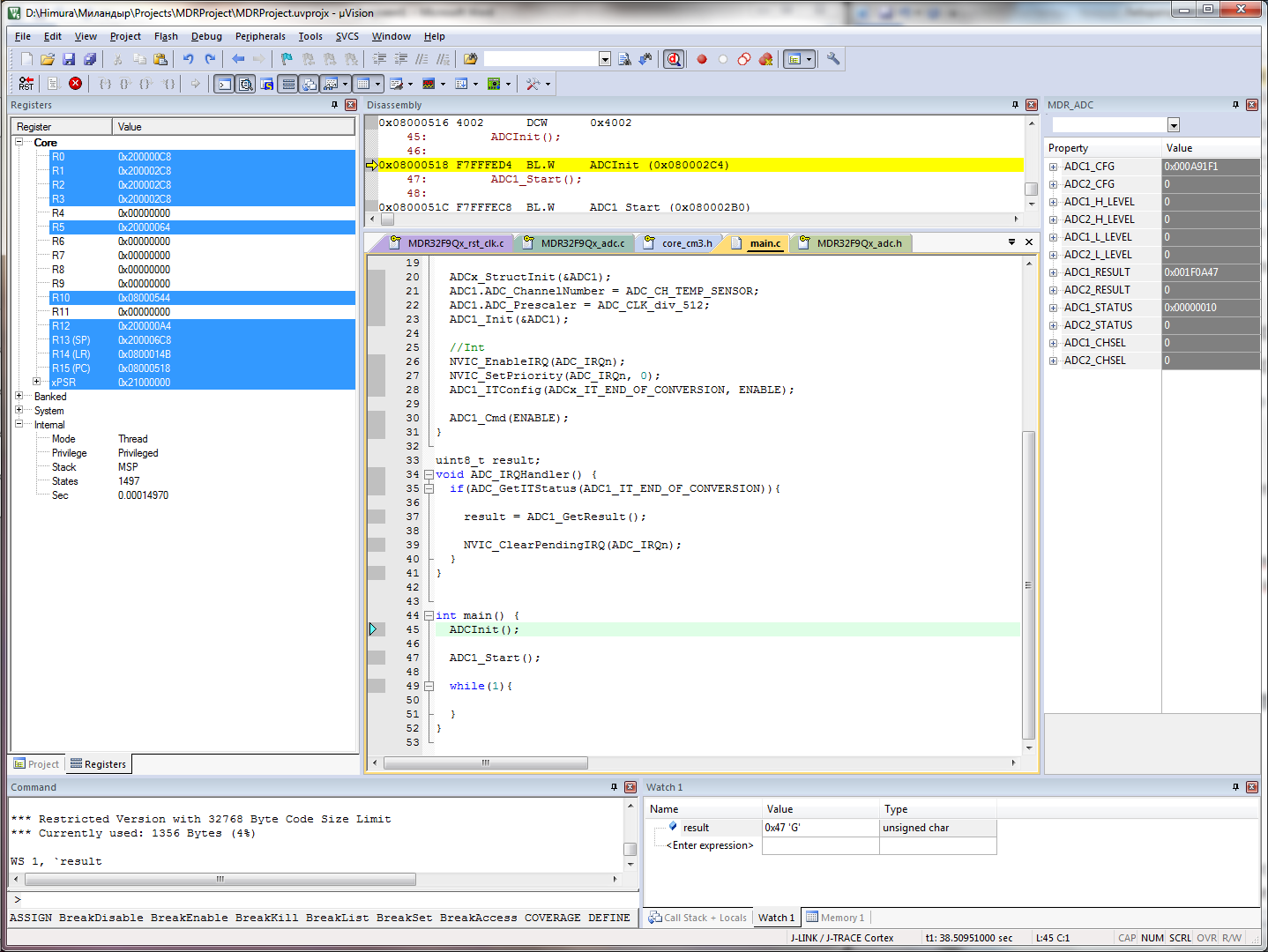
}

}

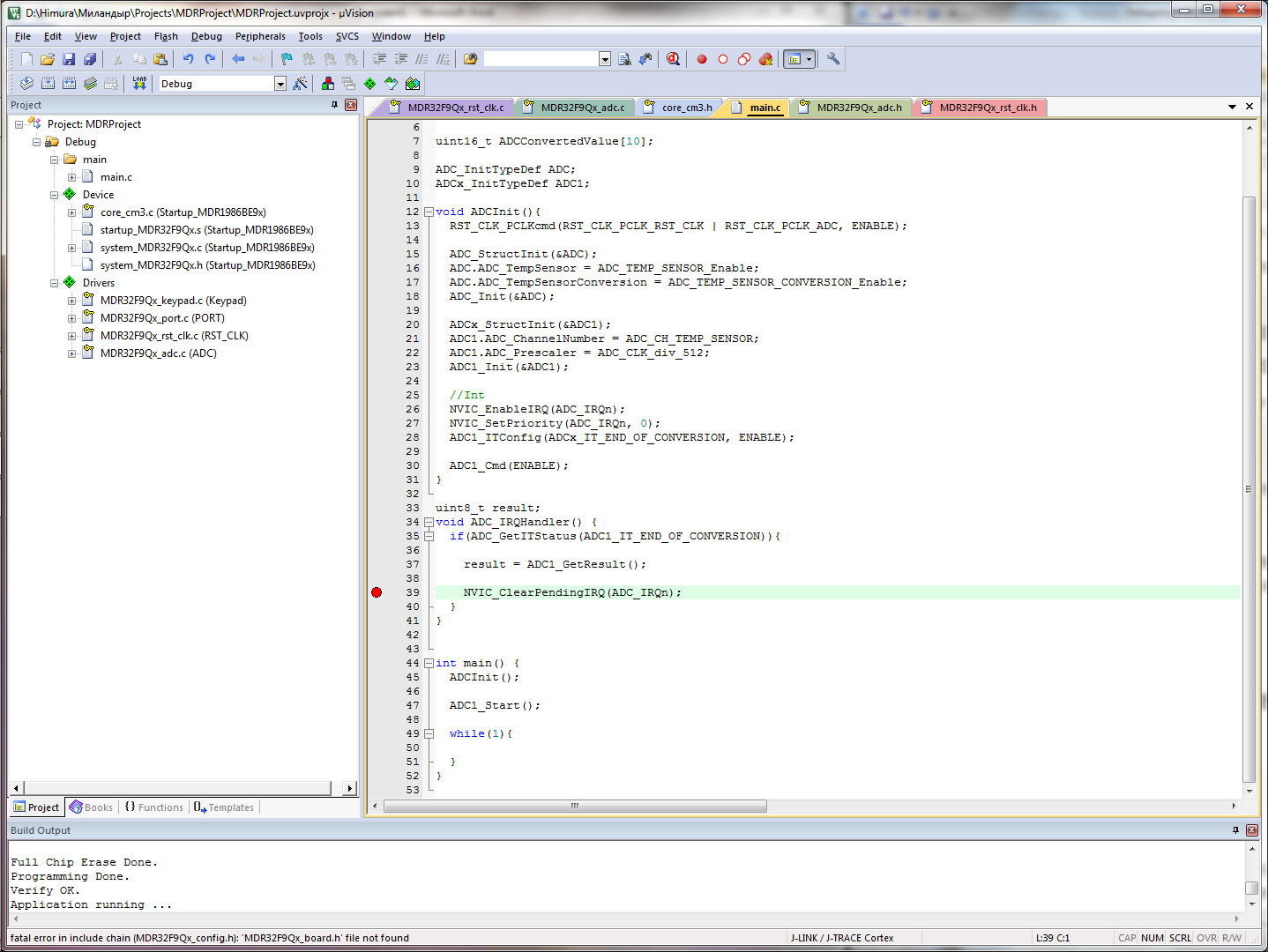
Откроем вкладку состояния регистров процессоров, нажав Peripherals>System Viewer>MDR\_ADC



Справа появилась вкладка, описывающая состояния регистров процессора, в поле ADC1\_RESULT можно увидеть показания термодатчика в шестнадцатеричном коде.



Поставим прерывание в строке 39, чтобы выполнить отладку до появления показаний термодатчика в регистре процессора, – для этого слева от номера нужной строки один раз кликнуть левой кнопкой мыши.



Для возможности наблюдения за состоянием конкретной переменной result нажмём на ней правой кнопкой мыши, далее проследуем по следующему пути выпадающего меню: Add ‘result’ to … > Watch 1

