

ВЕРСИЯ  
ДОКУМЕНТА  
**1.1**

# Отладочная набор TK-StarterKit\_S7G2

## РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ



# Содержание

Введение .....	3
Технические характеристики .....	3
Описание отладочного набора .....	4
Коммутационные таблицы .....	6
Создание проекта в среде e2_studio .....	7
Приложение 1. Схема электрическая принципиальная. ....	16

# Введение

Данный документ содержит описание отладочного набора TK-StarterKit\_S7G2 производства ООО «ТехноКом», предназначенного для изучения платформы Renesas Synergy Platform.

## Технические характеристики

Наименование параметра	Значение
Напряжение питания, В	8...60
Напряжение питания линий, разведенных на разъемы расширения, В	3.3 <sup>1</sup> 5.0 <sup>2</sup>
Температурный диапазон, °С	-40...+85
Габаритные размеры, не более, мм	154 x 120 x 30
Вес, не более, г	150
Средний срок службы, лет	10

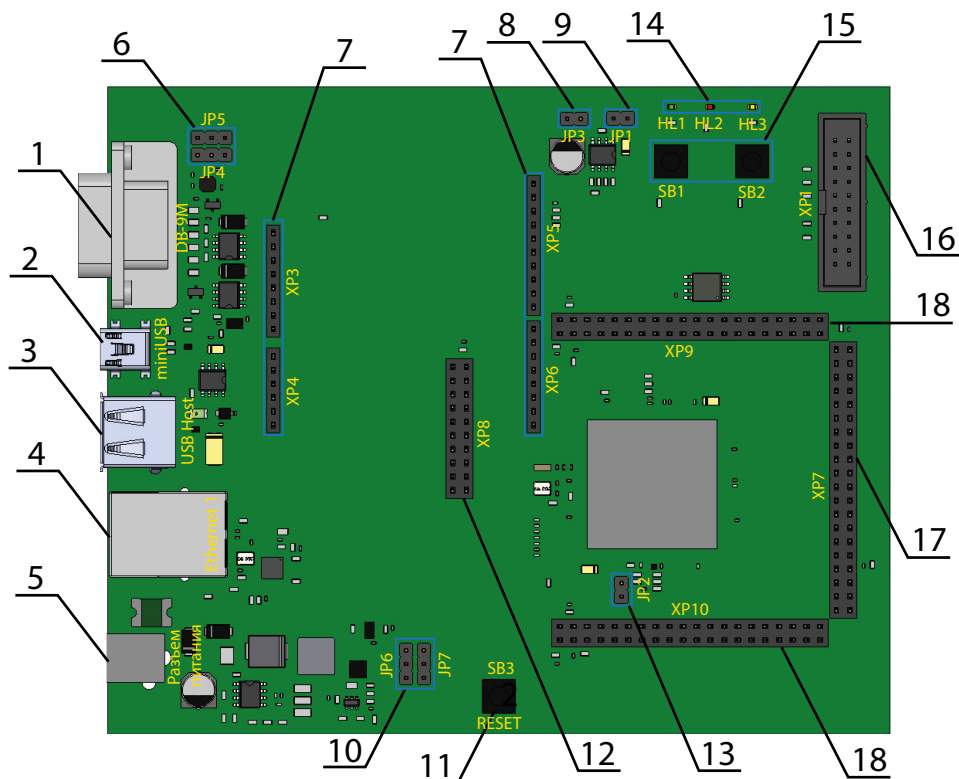
<sup>1</sup> Максимальный ток 3 А или 0.2 А, выбирается конфигурацией разъемов JP6 и JP7 на плате (см. раздел «Коммутационные таблицы»).

<sup>2</sup> Максимальный ток 5 А.

# Описание отладочного набора

Отладочный набор «TK-StarterKit\_S7G2» состоит из материнской платы и съемного модуля, на котором установлен GSM модем. Набор предназначен для изучения возможностей предоставляемых платформой Renesas Synergy Platform.

На **Рис.1** показан внешний вид материнской платы.



**Рис.1.** Материнская плата.

## НА МАТЕРИНСКОЙ ПЛАТЕ РАСПОЛОЖЕНЫ:

1. Разъем DB-9M, на который выведены интерфейсы RS232, RS485 и CAN.
2. Разъем miniUSB.
3. Разъем USB Host.
4. Разъем Ethernet.
5. Разъем питания. Центральный вывод разъема питания является плюсовым.
6. Разъемы JP4 и JP5. Предназначены для коммутации линии UART RXD и UART TXD соответственно. Коммутация осуществляется на интерфейс RS485 или RS232.

7. Разъемы XP3-XP6. Предназначены для подключения Arduino Shield.
8. Разъем JP3. К разъему JP3 выведен выход усилителя звука.
9. Разъем JP1. Разъем JP1 замыкается для включения функции MUTE усилителя звука.
10. Разъемы JP6 и JP7. Предназначены для выбора источника питания для платы расширения (см. схему).
11. Кнопка RESET – сброс микроконтроллера.
12. Разъем XP8 используется для подключения модуля камеры VGA, совместимого с Arduino.
13. Разъем JP2. Управление загрузчиком.
14. 3 светодиода, подключенные к выводам P6\_00 (HL1), P6\_01 (HL2) и P6\_02 (HL3), соответственно.
15. Две кнопки, подключенные к выводам на выводам P0\_05 (SB1) и P0\_06 (SB2), соответственно.
16. Разъем XP1. Предназначен для подключения разъема программатора.
17. Разъем XP7. Предназначен для подключения платы LCD дисплея DK-S7G2LCD.
18. Разъемы XP9, XP10. Предназначены для подключения платы расширения (например, SKGSM\_GPS).

На Рис.2 приведена материнская плата с установленной платой расширения «SKGSM\_GPS».

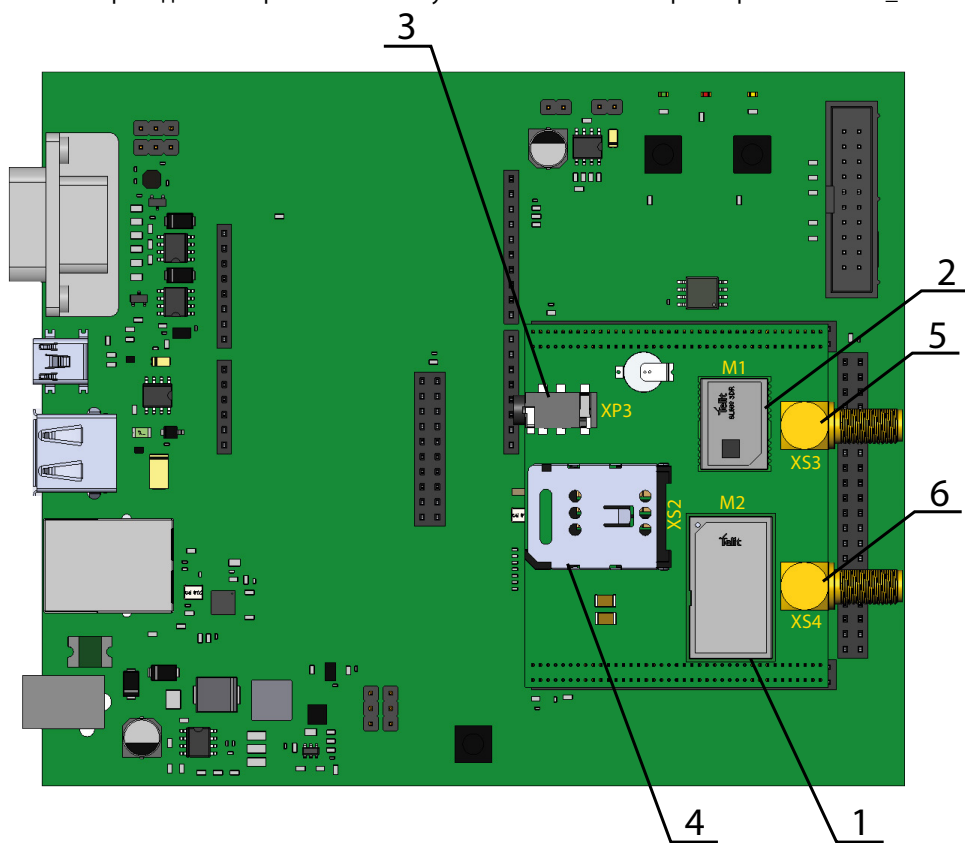


Рис.2. Отладочная плата (в сборе).

**НА ПЛАТЕ РАСШИРЕНИЯ SKGSM\_GPS РАСПОЛОЖЕНЫ:**

1. M2 – GSM Modem Telit UE866-EU.
2. M1 – GNSS приемник Telit SL869x.
3. XP3 – Разъем для подключения наушников.
4. XS2 – Держатель SIM-карты.
5. XS3 – Разъем для подключения антенны GPS/GLONASS.
6. XS4 – Разъем для подключения антенны GSM.

## Коммутационные таблицы

Ниже приведены коммутационные таблицы разъемов.

### JP1

Состояние разъема JP1	Функция MUTE усилителя звука
JP1 – замкнут	MUTE вкл
JP1 – разомкнут	MUTE выкл

### JP2

Состояние разъема JP2	Функция MUTE усилителя звука
JP2 – замкнут	USB загрузчик активен
JP2 – разомкнут	USB загрузчик НЕ активен

### JP4, JP5

Состояние разъема JP4	Состояние разъема JP5	Интерфейс
Замкнуты выводы 1 и 2	Замкнуты выводы 1 и 2	RS232
Замкнуты выводы 2 и 3	Замкнуты выводы 2 и 3	RS485

### JP6 (JP7)

Состояние разъема JP6(JP7)	Источник питания для выводов 4 и 37 разъема XP9 (XP10)	Интерфейс
Замкнуты выводы 1 и 2	TPS79333DBVR (3.3 В, 200 мА, LDO)	RS232
Замкнуты выводы 2 и 3	TPS563200 (3.3 В, 3 А)	RS485

# Создание проекта в среде e2\_studio

В качестве примера рассмотрим процесс создания проекта приложения, которое будет перенаправлять команды от ПК поступившие по USB на GSM модем, а ответ, полученный от модема, отправлять на ПК.

На вашем ПК должна быть установлена среда «e2\_studio» и программный пакет «SSP 1.2.0» (доступно на сайте <https://synergycastle.renesas.com/auth/login>). Подробное описание установки и настройки программного обеспечения, базовые знания о настройке и отладке проектов можно посмотреть в документе «Basics of the Renesas Synergy™ Platform», который доступен на сайте [www.renesas.com](http://www.renesas.com).

## ПРОЦЕСС РАЗРАБОТКИ:

1. Создать проект «Synergy C Project». Выбрать версию SSP 1.2.0, плату – S7G2 SK, процессор будет выбран автоматически – R7FS7G27H3A01CFC. В качестве «Project Template» выбрать «BSP».
2. Автоматически откроется перспектива «Synergy Configuration». Если перспектива не открылась, откройте её, выбрав Window -> Perspective -> Open Perspective -> Other..., далее в появившемся окне выберите «Synergy Configuration» и нажмите ОК.

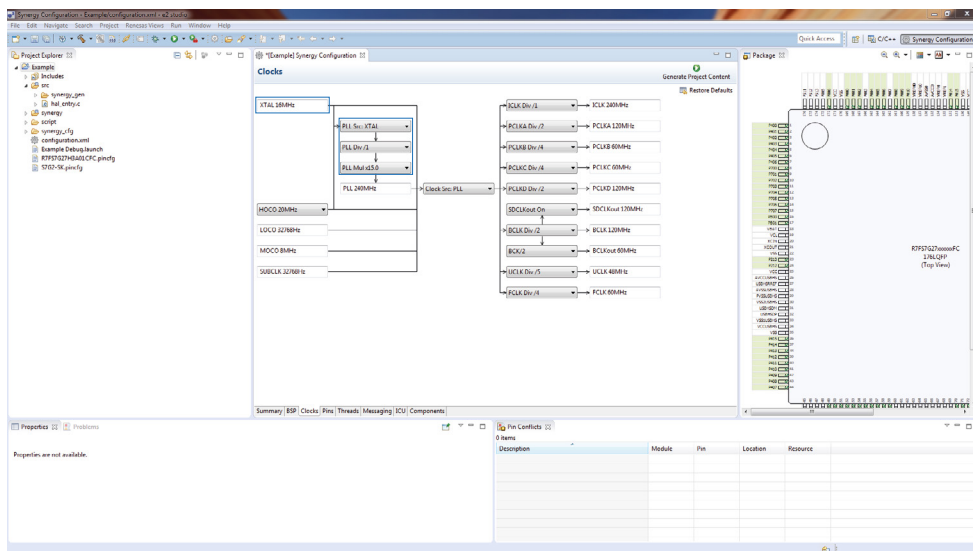


Рис.3. Вкладка «Clocks».

3. Перейдите на вкладку «Clocks». Измените параметр XTAL24 MHz, на 16 MHz и выберите его в качестве основного источника тактирования. В качестве PLL Div выберите 1, в качестве PLL Mul выберите x15.0 (Рис.3).
4. На вкладке «Threads» добавьте поток, который будет выполнять работу USB <-> UART преобразователя (Рис.4, п.1).

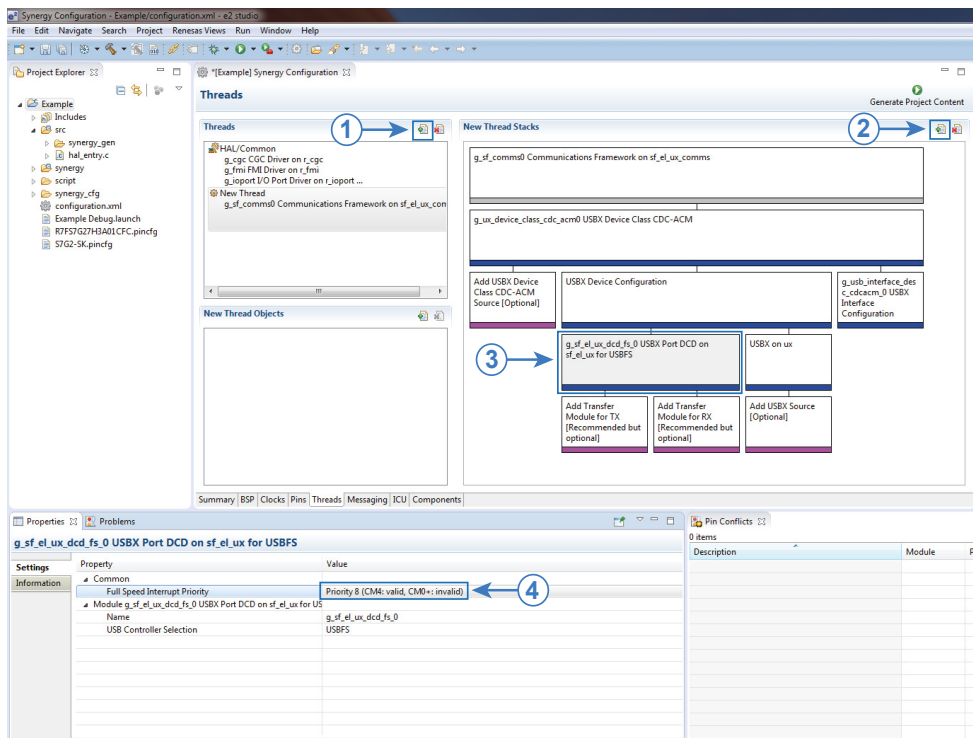


Рис.4. Добавление стека для UART-<-> UART преобразователя.

##### 5. Добавьте стек для ранее созданного потока, который будет работать с USB интерфейсом:

- нажмите кнопку «Добавить стек» (Рис.4, п.2);
- выберите Framework -> Connectivity -> Communications Framework on sf\_el\_ux\_comms;
- в появившемся стеке добавьте USBX Port DCD (Рис.4, п.3);
- в свойствах блока USBX Port DCD установите приоритет прерывания – выберите приоритет 8 (Рис.4, п.4);

##### 6. Добавьте стек для работы с UART интерфейсом (Рис.5):

- добавьте стек, выберите Framework -> Connectivity -> Communications Framework on sf\_uart\_comms;
- в свойствах стека нужно указать другое имя, т.к. g\_sf\_comms0 уже используется стеком USB;
- в свойствах блока g\_uart0 UART Driver on r\_sci\_uart нужно установить приоритет для прерываний, скорость обмена 115200, канал 6.



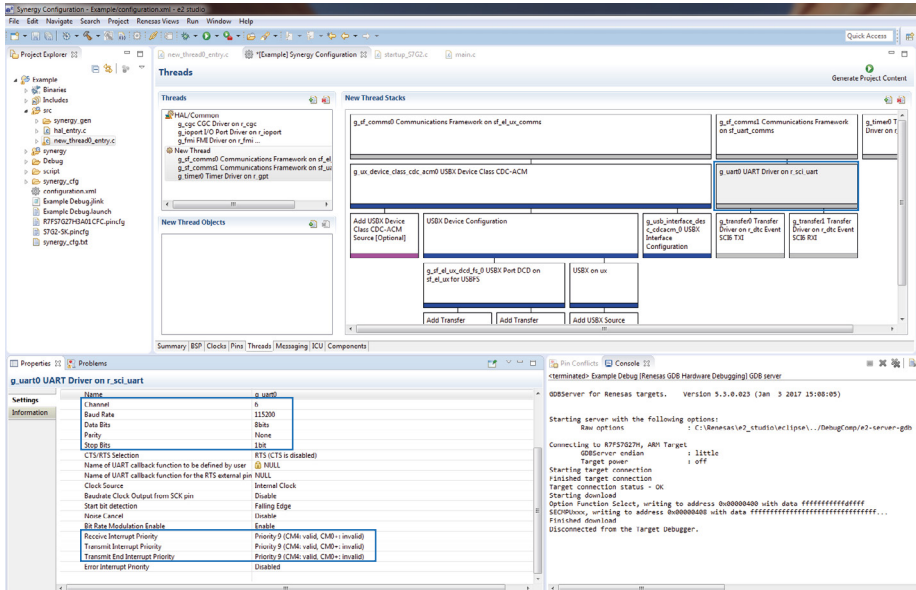


Рис.5. Добавление стека для работы с UART интерфейсом.

## 7. Добавьте стек таймера (Рис.6):

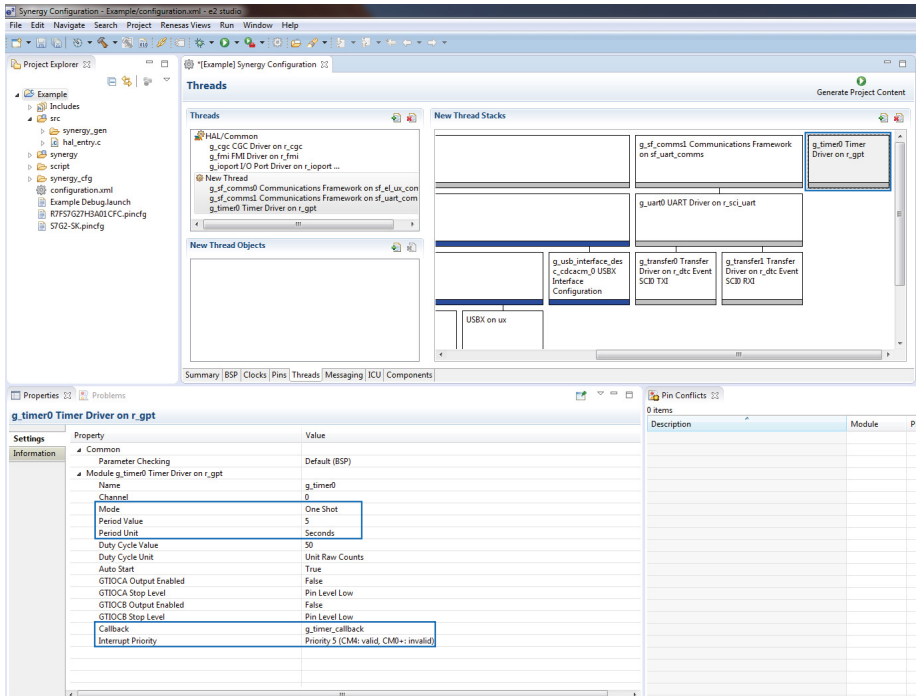
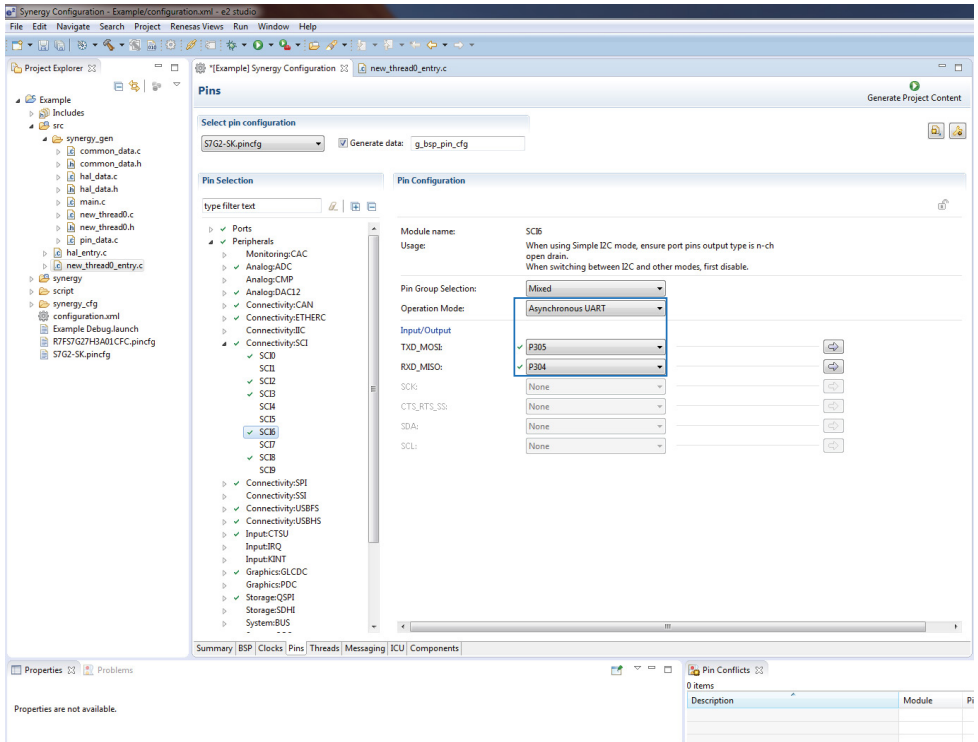


Рис.6. Добавление стека для таймера.

- Добавьте стек, выберите Driver -> Timers -> Timer Driver on r\_gpt;
- в свойствах выберите однократный режим, период 5 сек, приоритет прерывания и функцию обработчик g\_timer\_callback.

**8.** Перейдите на вкладку «Pins», далее выберите Peripherals->Connectivity:SCI и выберите канал SC16. По умолчанию в BSP для платы G7S2 SK этот канал настроен на работу с интерфейсом SPI. Необходимо настроить его на работу с интерфейсом UART. Установите соответствующие настройки (**Рис.7**).



**Рис.7.** Настройка канала SPI для работы с интерфейсом UART.

**9.** Выберите для настройки выводов Ports -> P3. Настройте выходы P311 и P312 на выход (**Рис.8, п.1**). Настройте Вывод P312, управляющий питанием GSM модема, на режим Output mode (Initial High). Настройте вывод P311, управляющий входом Reset GSM модема, на режим Output mode (Initial Low).

**10.** Настройка конфигурации закончена. Для того чтобы среда сгенерировала программный код нажмите «Generate Project Content» (**Рис.8, п.2**). Если генерация прошла успешно, в окне Project Explorer в папке src появятся необходимые файлы (**Рис.9**).

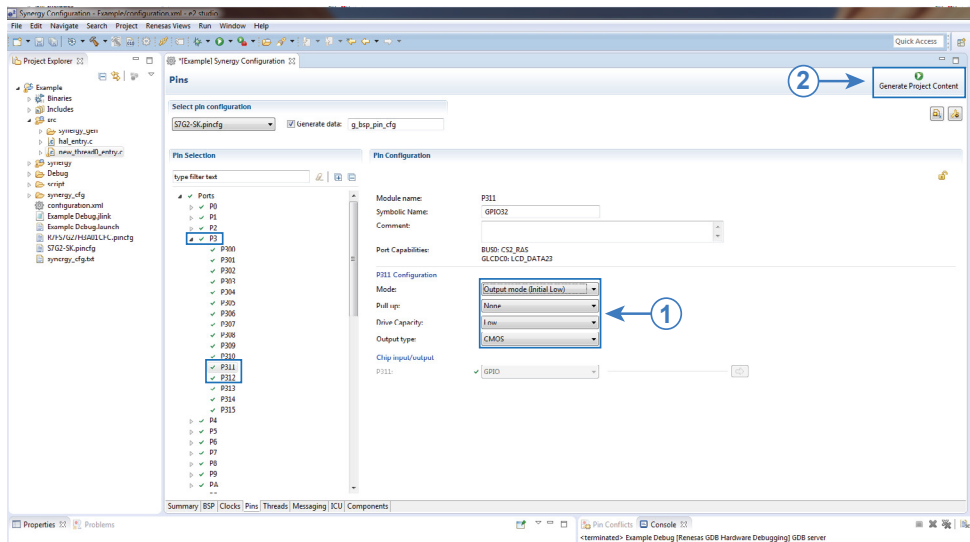


Рис.8. Настройка выводов и генерация программного кода.

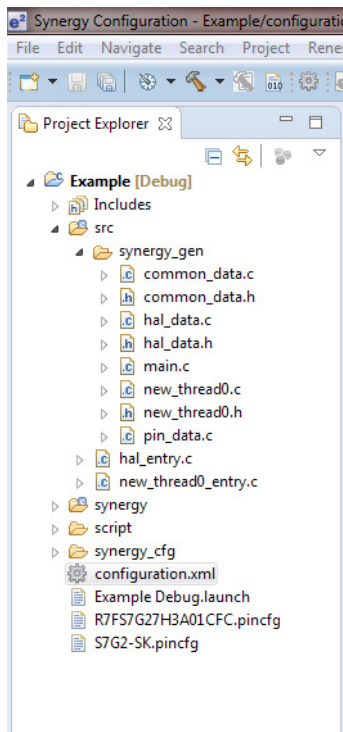


Рис.9. Project Explorer.

**11.** Откройте файл «new\_thread0\_entry.c», замените функцию «new\_thread0\_entry» на следующую:

```
void new_thread0_entry(void)
{
    uint8_t head_rx = 0;
    memset(rx_char, 0, sizeof(rx_char));
    g_timer0.p_api->open(g_timer0.p_ctrl, g_timer0.p_cfg);

    /* TODO: add your own code here */
    while (1)
    {
        if(g_sf_comms0.p_api->read(g_sf_comms0.p_ctrl, (uint8_t*)&rx_char[head_rx], 1, TX_WAIT_
FOREVER) == SSP_SUCCESS)    //read from USB
        {
            if(rx_char[head_rx] == '\r')
            {
                head_rx = 0;
                memset(tx_char, 0, sizeof(tx_char));
                g_sf_comms1.p_api->write(g_sf_comms1.p_ctrl, (uint8_t*)rx_char, strlen(rx_char), TX_NO_
WAIT);    //write to UART
                g_timer0.p_api->start(g_timer0.p_ctrl);
                g_sf_comms1.p_api->read(g_sf_comms1.p_ctrl, (uint8_t*)tx_char, sizeof(tx_char), TX_WAIT_
FOREVER);    //read from UART
                g_timer0.p_api->stop(g_timer0.p_ctrl);
                g_sf_comms0.p_api->write(g_sf_comms0.p_ctrl, (uint8_t*)tx_char, strlen(tx_char), TX_NO_
WAIT);    //write to USB
                memset(rx_char, 0, sizeof(rx_char));
            }
            else head_rx++;
        }
    }
}
```

Создайте массивы для приема и передачи данных:

```
char tx_char[512];
char rx_char[512];
```

Добавьте функцию для обработки прерывания от таймера:

```
extern TX_THREAD new_thread0;
void g_timer_callback(timer_callback_args_t *p_args)
{
    tx_thread_wait_abort(&new_thread0);
}
```

Полученный программный код ожидает команду от ПК, которая в соответствии со спецификацией AT команд GSM модема должна заканчиваться символом '\r'. Полученная команда отправляется по интерфейсу UART в GSM модем. Таймер g\_timer0 начинает работать, а функция g\_sf\_comms1.p\_api->read ожидает ответа от GSM модема. По истечении периода таймера полученный ответ от модема отправляется на ПК.

**12.** Для компиляции нажмите пиктограмму «Молот» (Рис.10, п.1). Если компиляция прошла успешно, то в окне Console появится сообщение об отсутствии ошибок (Рис.10, п.2).

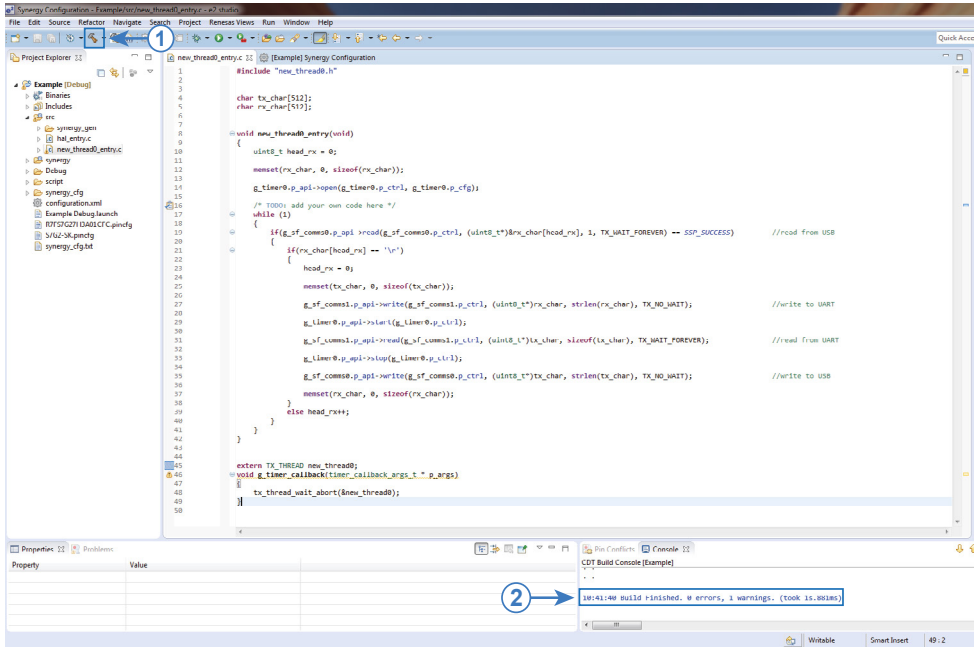


Рис.10. Компиляция.

**13.** Установите модуль с GSM модемом на плату TK-StarterKit\_S7G2. Подайте на плату напряжение питания. Подключите JTag программатор (например SEGGER j-Link Ultra+) к ПК и установите соответствующие драйвера. Подключите JTag разъем платы TK-StarterKit к программатору (см. описание отладочной платы).

**14.** Для загрузки программы в контроллер и отладки нажмите на треугольник рядом с пиктограммой Debug (Рис.11, п.1), выберите Debug Configurations... выберите Renesas GDB Hardware Debugging и наш проект Example Debug (Рис.11, п.2) и нажмите Debug (Рис.11, п.3).

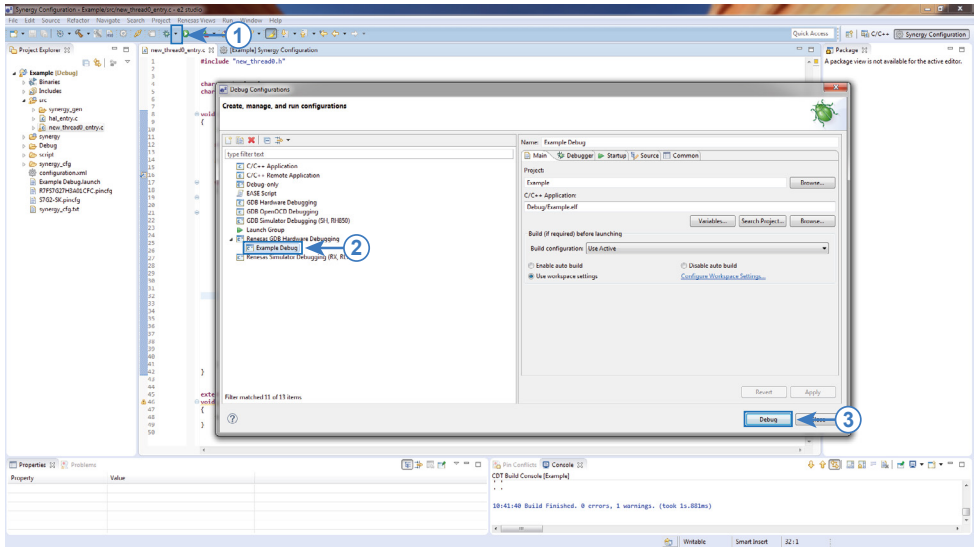


Рис.11. Загрузка программы в контроллер и отладка.

15. Открывается перспектива Debug. Для запуска приложения нажмите пиктограмму Start (Рис.12, п.1). Управление перейдет в функцию main, нажмите Start еще раз и программа продолжит выполняться.

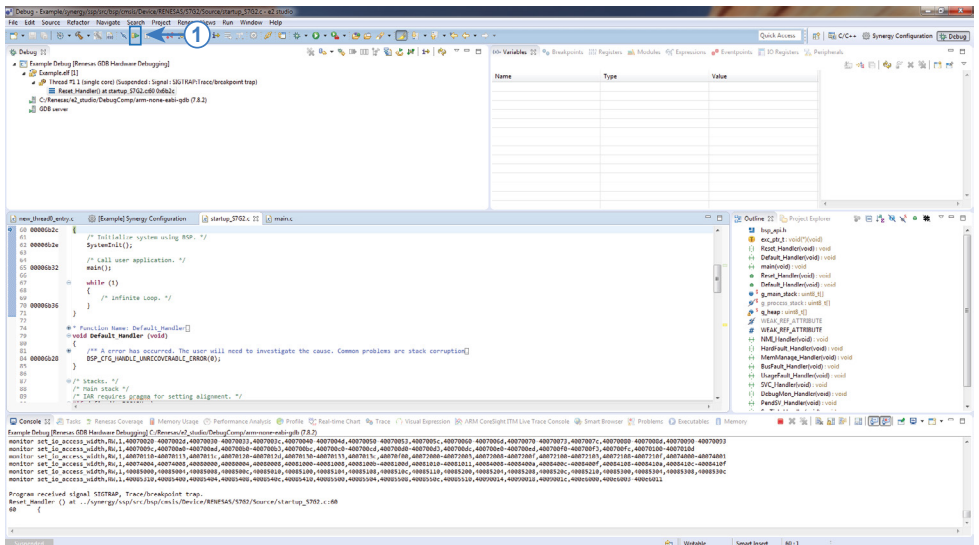
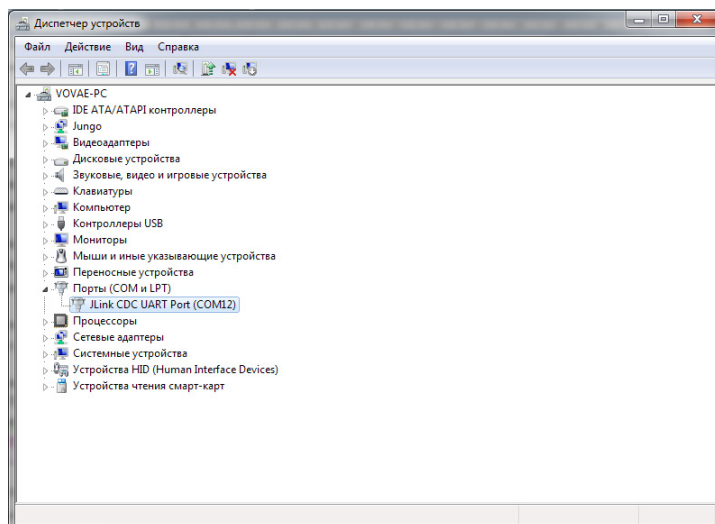


Рис.12. Запуск приложения.

**16.** Подключите плату к ПК кабелем miniUSB. Драйвера установятся автоматически. В диспетчере задач в разделе Порты (COM и LPT) появится устройство JLink CDC UART Port (Рис.13). Если устройство не установилось автоматически, обратитесь к базе знаний Renesas: [https://en-us.knowledgebase.renesas.com/English\\_Content/Renesas\\_Synergy™\\_Platform/Renesas\\_Synergy\\_Knowledge\\_Base/Installing\\_USB\\_CDC\\_Driver\\_on\\_Windows\\_10](https://en-us.knowledgebase.renesas.com/English_Content/Renesas_Synergy™_Platform/Renesas_Synergy_Knowledge_Base/Installing_USB_CDC_Driver_on_Windows_10).



**Рис.13.** Обнаружено устройство «JLink CDC UART Port».

**17.** Запустите любую терминальную программу, подключитесь к соответствующему COM порту и отправляйте команды на GSM модем.







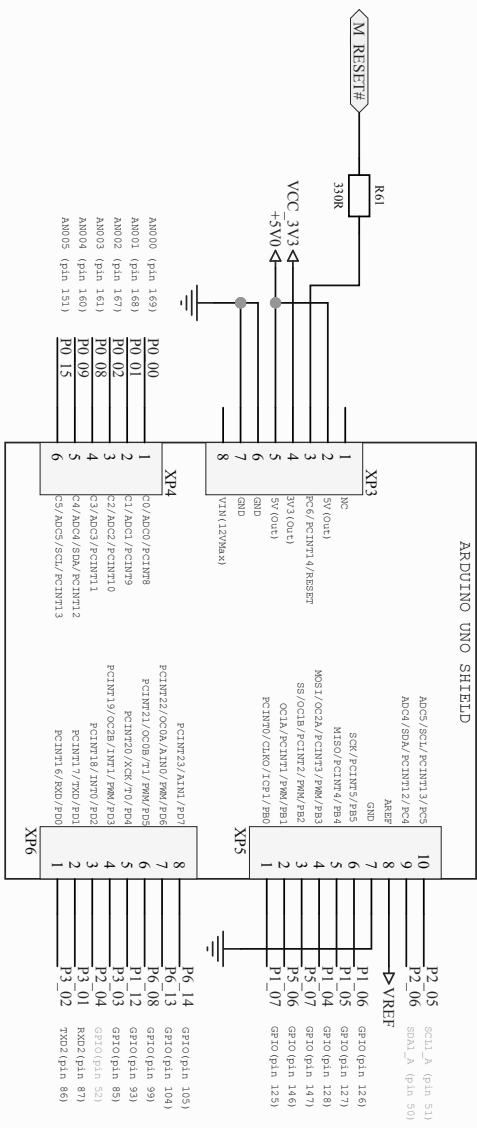






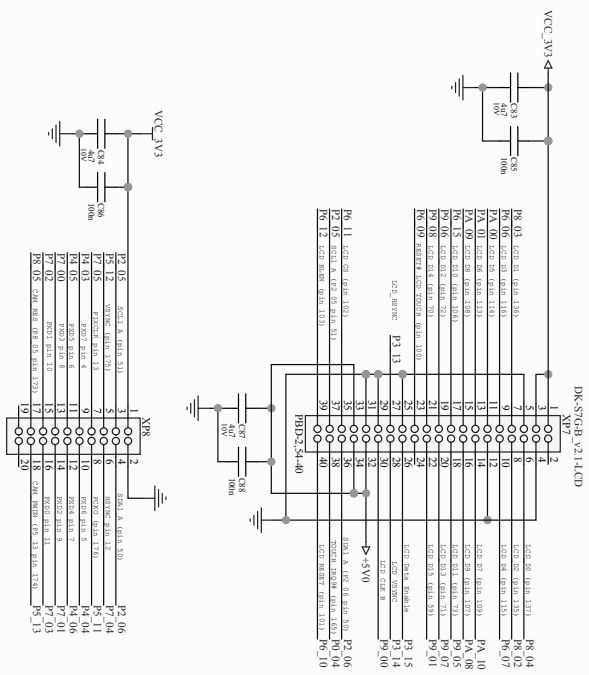


Име. № подл.	Подп. и дата	Взам. инв. №	Име. № дубл.	Подп. и дата



Име.	Лист	№ докум.	Подпись	Дата	Компьютер	Лист

Line No. no.:	Ro. u. data:	Blank. u. No.:	Line No. 9.6.:	Ro. u. data:



Proj. No.:	Proj. Ime:	Proj. Datum:	Proj. Status:	Proj. Verzi:	Proj. Datum:	Proj. Status:	Proj. Verzi:

Revizija: 02

Datum:





ООО «ТехноКом»

---

Все права защищены  
© Челябинск, 2017

[www.tk-nav.ru](http://www.tk-nav.ru)  
[mail@tk-chel.ru](mailto:mail@tk-chel.ru)