



Использование плагина для разработки программ на языке C для процессора Байкал-T1 в IDE Eclipse
Статус: Релиз
01.09.2016

Использование плагина для разработки программ на языке C для процессора Байкал-T1 в IDE Eclipse

Редактор: Михаил Бессонов

Доступ: Публичный

История изменений

Версия	Дата	Исполнитель	Описание
1.0	01.09.2016	Т. Яловая	Выполнен перенос раздела из документа «Комплект средств разработки ПО для микропроцессора Baikal-T» в новый документ.



Использование плагина для разработки программ на языке C для процессора Байкал-Т1 в IDE Eclipse
Статус: Релиз
01.09.2016

Содержание

1 Подготовка.....	3
2 Выбор типа проекта.....	3
3 Описание и использование разных типов проекта.....	5
3.1 Baikal-t Bare-metal for board.....	5
3.2 Baikal-t Bare-metal for QEMU.....	8
3.3 Baikal-t Target Application.....	10
3.4 Baikal-t User Application for QEMU.....	10

Список рисунков

Рисунок 1. Выбор проекта на языке C.....	4
Рисунок 2. Выбор типа и шаблона.....	4
Рисунок 3. Сборка проекта из исходников.....	6
Рисунок 4. Настройка инструментов сборки.....	6
Рисунок 5. Окно настройки отладчика для работы через Jtag via openocd.....	7
Рисунок 6. Выбор параметров симулятора терминала.....	8
Рисунок 7. Запуск симулятора QEMU.....	9
Рисунок 8. Окно настройки отладчика для работы через remote gdbserver.....	9



Использование плагина для разработки программ на языке C для процессора Байкал-T1 в IDE Eclipse
Статус: Релиз
01.09.2016

1 Подготовка

Интегрированная среда разработки (IDE) Eclipse расположена в каталоге

```
<path_to_SDK>/baikal/usr/eclipse/
```

Для запуска следует ввести в консоли команды:

```
cd <path_to_SDK>/baikal/usr/eclipse/  
./eclipse
```

Никаких дополнительных настроек не требуется.

После загрузки IDE откроется окно с предложением выбрать папку на локальном диске, в которой будут храниться исходники вашего проекта. Рекомендуется создать новую или выбрать пустую папку для корректной конфигурации проекта.

2 Выбор типа проекта

После выбора или создания папки для нового проекта откроется стандартное окно приветствия Eclipse. В меню **File** необходимо последовательно выбрать следующие пункты: **File** → **New** → **C Project** (рисунок 1).

Далее следует выбрать тип проекта (рисунок 2). Помимо стандартных шаблонов, здесь присутствуют шаблоны проектов для процессора Байкал-T1. В поле **Project name** необходимо ввести имя нового проекта (исходный файл с функцией main будет иметь такое имя), а также выбрать тип проекта и шаблон (**Empty Project** или **Hello world project**).

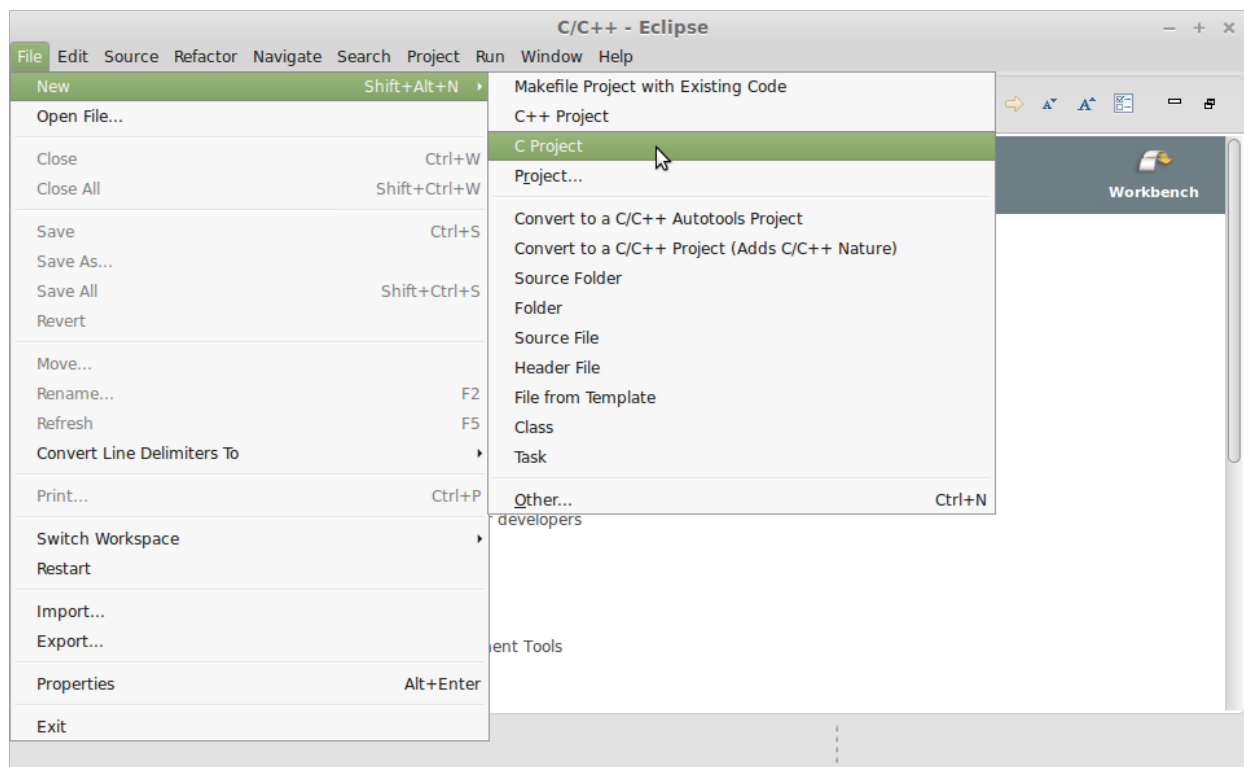


Рисунок 1. Выбор проекта на языке C

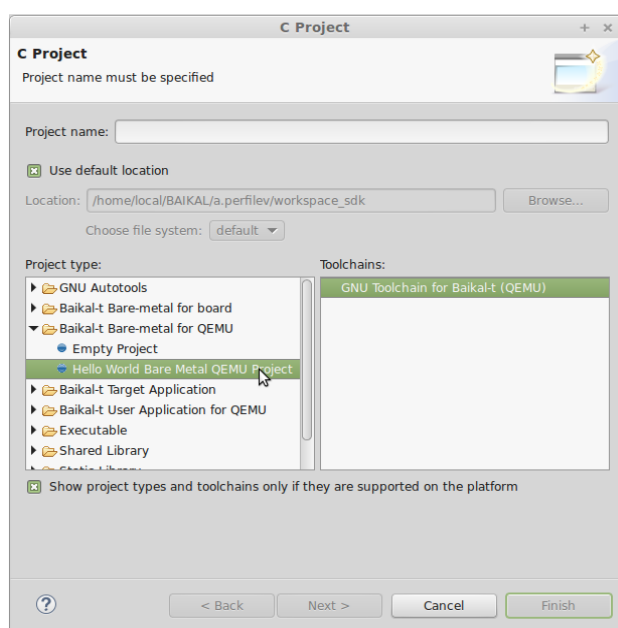


Рисунок 2. Выбор типа и шаблона



3 Описание и использование разных типов проекта

3.1 *Baikal-t Bare-metal for board*

Данный проект следует использовать для разработки Bare-metal программ для плат БФК. При выборе шаблона Hello world в папки проекта будут добавлены все необходимые файлы для инициализации процессора. Чтобы увидеть сакраментальную фразу: Hello world через интерфейс uart, необходимо:

1. Собрать проект. Для этого необходимо нажать кнопку **Build 'Debug' for project 'Board'** (имеет иконку в виде молотка) на панели инструментов, расположенной в верхней части окна приложения (рисунок 3). Никаких дополнительных настроек выполнять не надо, так как набор средств кросс-компиляции cross-toolchain лежит в папке SDK. Если же вы все-таки хотите добавить или просто просмотреть флаги, которые используются при компиляции, в окне **Project Explorer** необходимо нажать правой кнопкой мыши по папке с именем проекта, из контекстного меню выбрать пункт **Properties** и в открывшемся окне развернуть категорию C/C++ **Build** → **Settings** (рисунок 4). Здесь можно настроить все инструменты сборки, задать или убрать необходимые флаги. Если для какого-то инструмента не получается найти требуемый флаг, следует задать его вручную в категории **Miscellaneous**. Такая категория присутствует для каждого инструмента.
2. С помощью утилиты dfu-util, которая входит в состав BSP, следует прошить плату файлом с расширением *.text. Команда, которую нужно ввести в терминале, выглядит так (в одну строку):

```
<path_to_SDK>/baikal/bin/dfu-util -D  
<eclipse_workspace>/<project_name>/Debug/<project_name>.text -a 0 -d  
[Run-Time VENDOR]:[Run-Time PRODUCT]
```
3. Чтобы увидеть результат, необходимо подключиться к плате через интерфейс uart с помощью программы типа minicom со следующими параметрами для com: 115200, 8, N, 1.

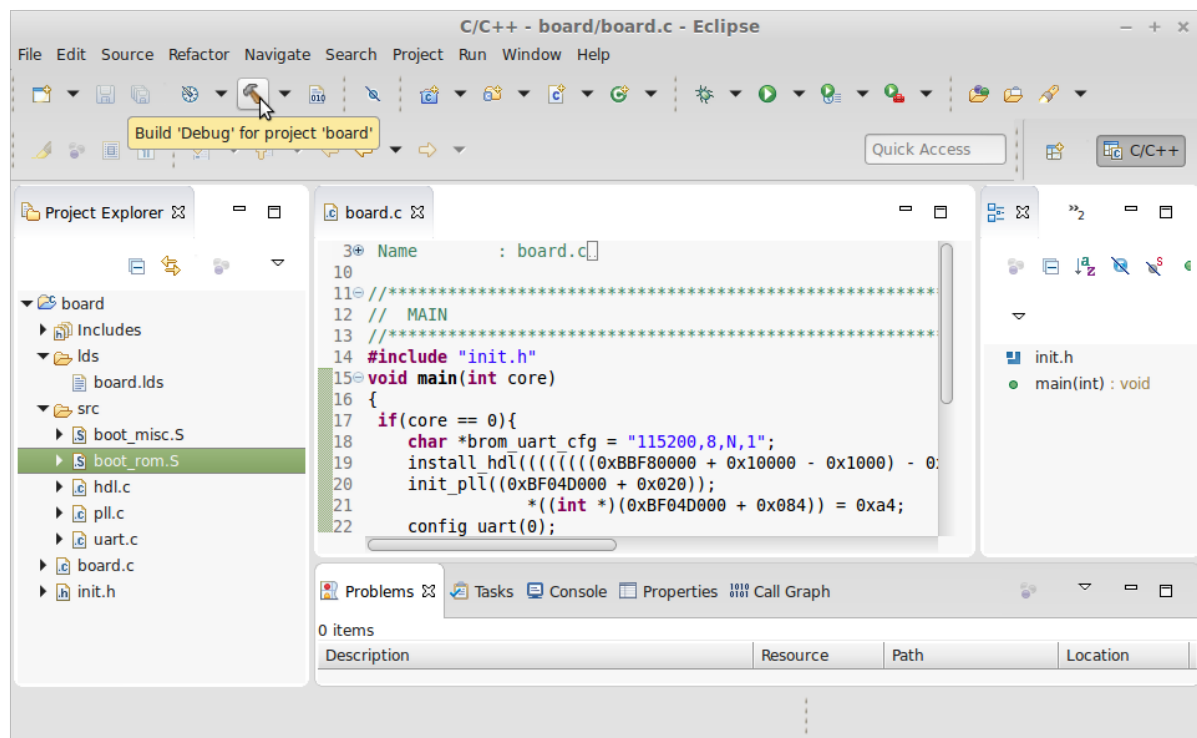


Рисунок 3. Сборка проекта из исходников

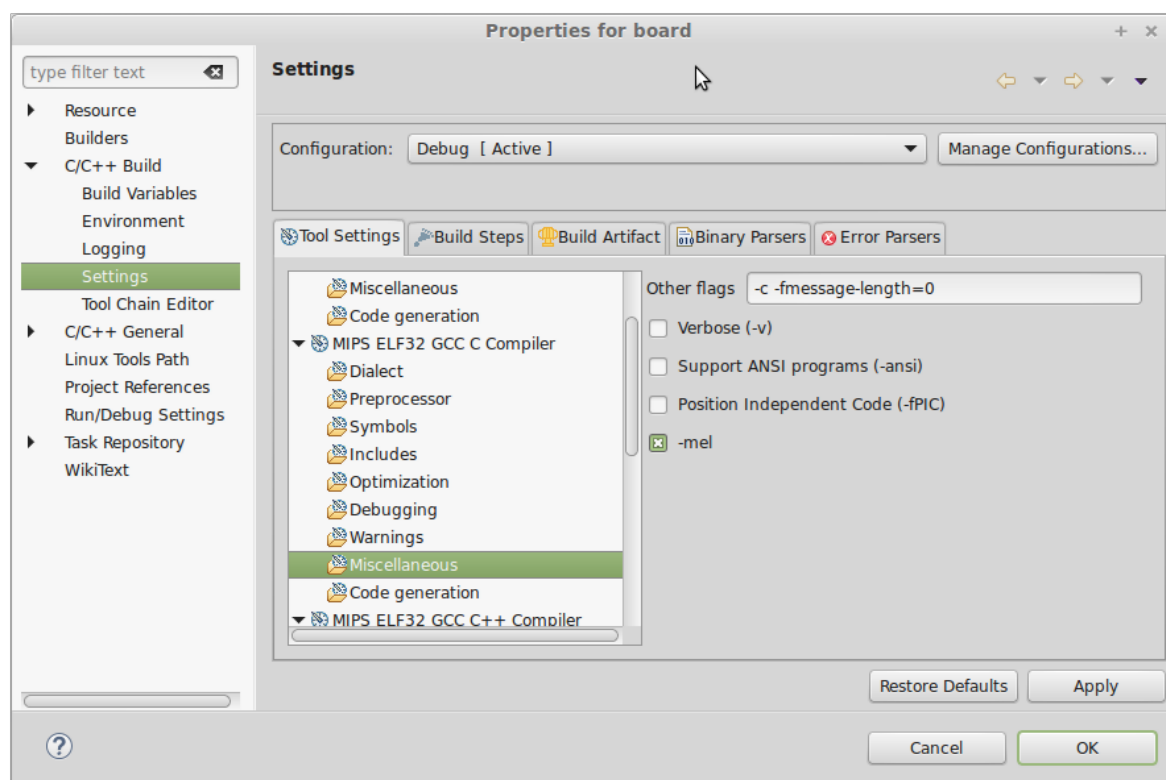


Рисунок 4. Настройка инструментов сборки

4. Отладка осуществляется через интерфейс Jtag via openocd. Чтобы перейти в режим отладчика в IDE, следует выбрать пункт меню **Run** → **debug configuration...** В открывшемся окне два раза щелкнуть левой кнопкой мыши по **MIPS C/C++ application**, после чего будет создана конфигурация для текущего проекта. Далее следует перейти на вкладку **Debugger**, в поле **Debugger Options** выбирать вкладку **Gdbserver Settings** (рисунок 5) и, если есть необходимость, изменить настройки.

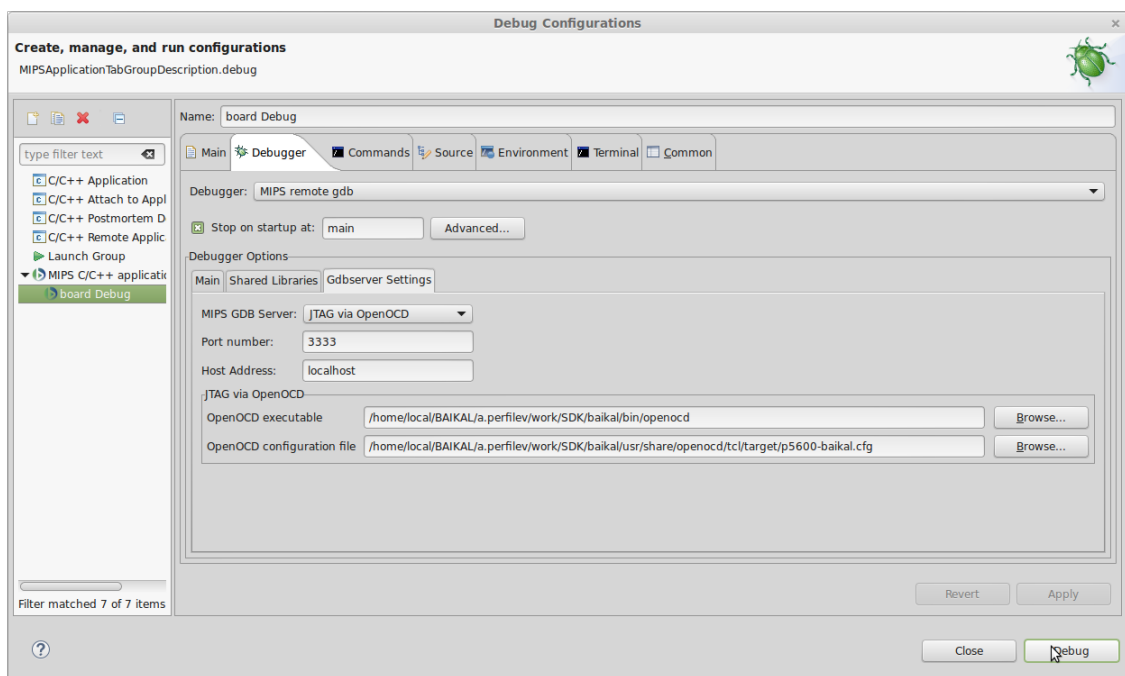


Рисунок 5. Окно настройки отладчика для работы через Jtag via openocd

5. По умолчанию все настройки выставлены на стандартную конфигурацию openocd. На вкладке **Terminal** можно выбрать, следует ли запускать эмулятор терминала, и при необходимости задать для него параметры. При этом плата должна быть подключена через uart порт (рисунок 6).

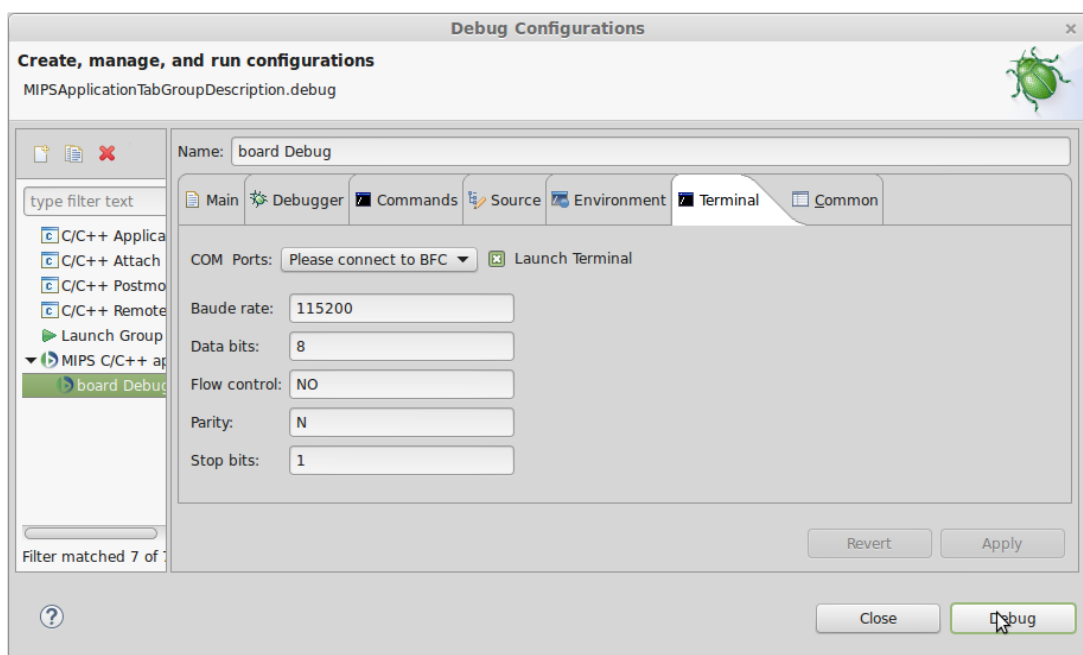


Рисунок 6. Выбор параметров симулятора терминала

3.2 Baikal-t Bare-metal for QEMU

Выбрав этот проект, можно собрать приложение для симулятора qemu-system-mipsel. Сборка и настройка осуществляется так же, как было описано в предыдущем подразделе.

Отладка приложения выполняется в два шага. Сначала запускается QEMU путем выбора следующих пунктов меню: **Run** → **External Tools** → **START_QEMU_DEBUG** (рисунок 7).

Затем в меню отладчика следует выбрать пункты **MIPS C/C++ application** →

<project_name> Debug. По умолчанию все работает в поставляемой конфигурации без дополнительной настройки. При необходимости можно настроить порт для прослушивания на вкладке **Debugger**, для этого следует в поле **Debugger Options** выбрать вкладку **Gdbserver Settings** (рисунок 8).

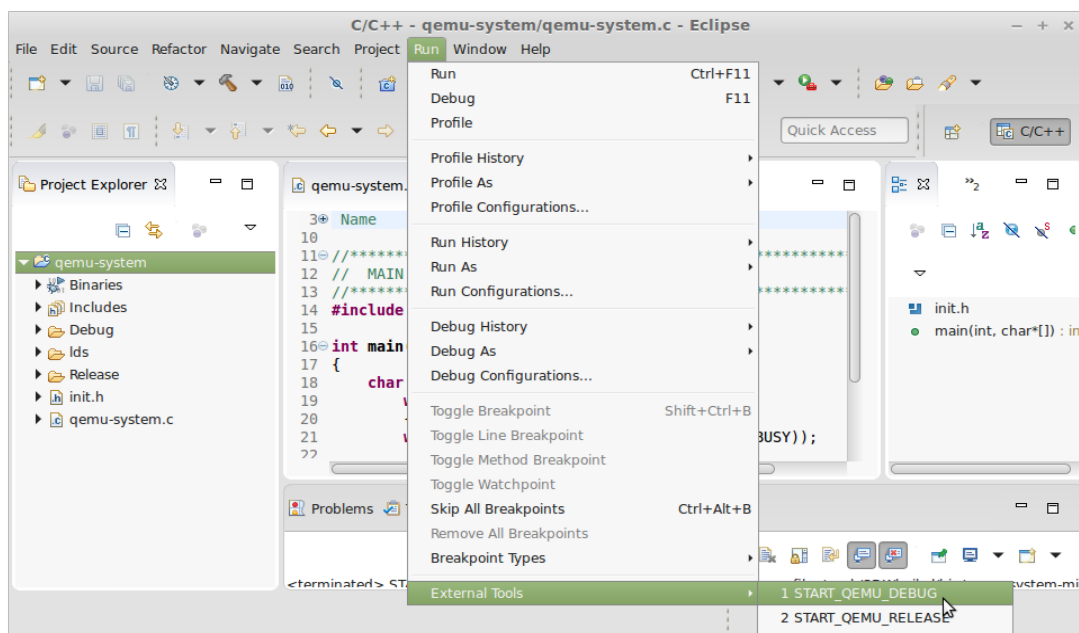


Рисунок 7. Запуск симулятора QEMU

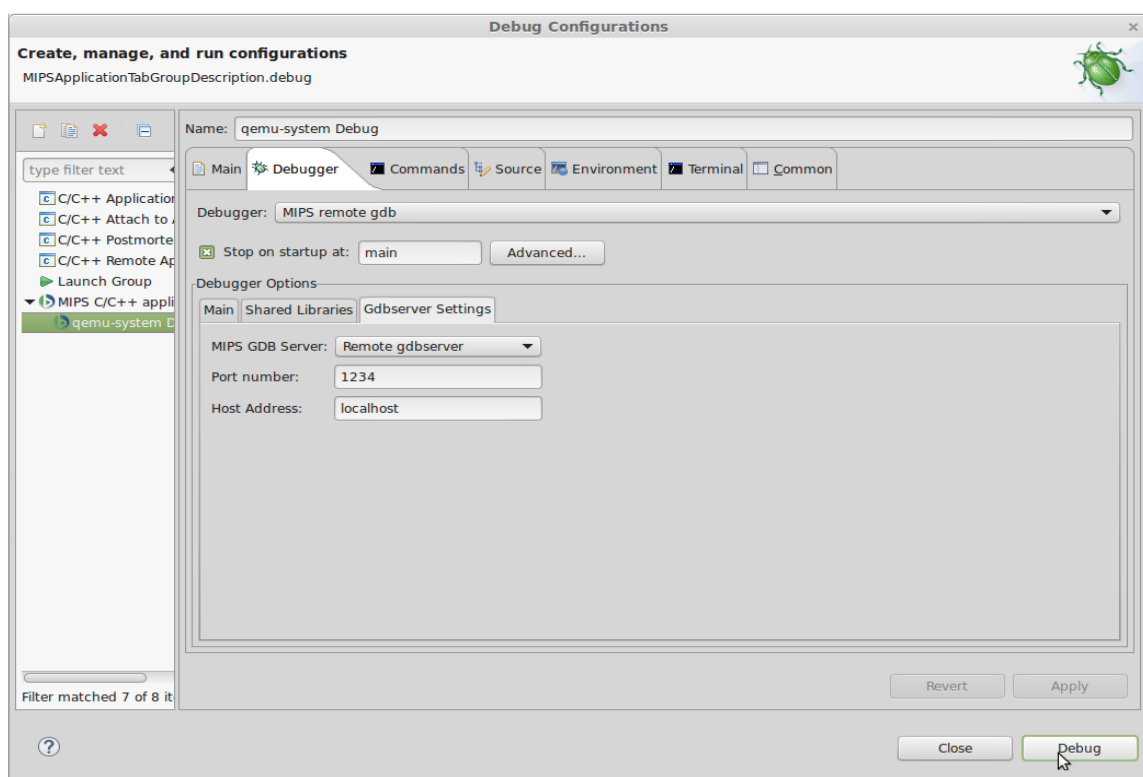


Рисунок 8. Окно настройки отладчика для работы через remote gdbserver



Использование плагина для разработки программ на языке C для процессора Байкал-T1 в IDE Eclipse
Статус: Релиз
01.09.2016

3.3 Baikal-t Target Application

Это обычное приложение, собранное кросс-компилятором для целевой машины на процессоре Baikal-T1. Сборка и настройка производятся аналогично перечисленным выше проектам. Отладка осуществляется через Remote gdbserver, который можно настроить, как показано на рисунке 8.

3.4 Baikal-t User Application for QEMU

Как и предыдущий, является обычным приложением, однако его можно запускать при помощи симулятора qemu-mipsel. Запуск, настройка и отладка производятся аналогично проекту Baikal-t Bare-metal for QEMU (см. п. 3.2).