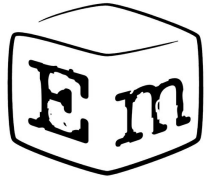


Embox российская ОС для цифровых устройств

Антон Бондарев

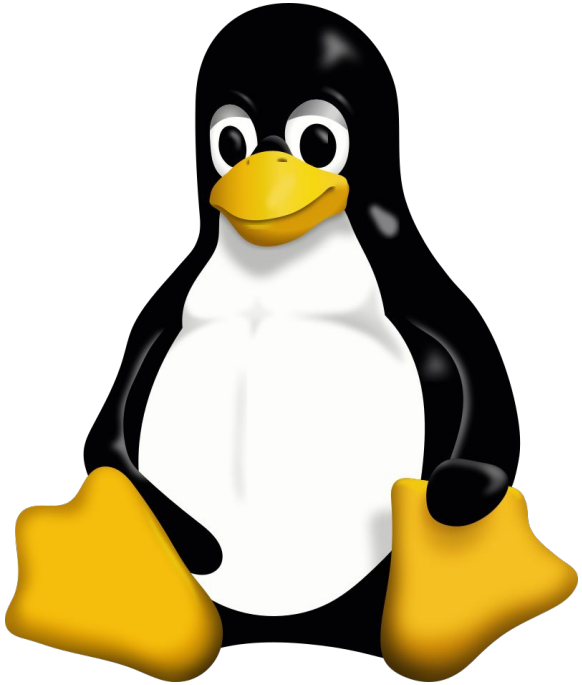
OS Day, 22 июня 2022

Embox

Embox — свободная операционная система реального времени (RTOS), разрабатываемая для встроенных систем.
Основная идея использование ПО **Линукс** в более безопасном и детерминированном, менее ресурсоемком и энергопотребляющем окружении

Linux

- Величайшая ОС в мире :)
- Огромное количество ПО на любой вкус
- Потрясающая поддержка аппаратуры



- Но Линукс универсален, поэтому:
  - Требует мощной аппаратуры
  - Трудно обеспечить реал-тайм
  - Подвержен вредоносному ПО

Embox

- “Недавно портирован за три недели проект, который разрабатывался два года для Линуха, потом с разными спотыкачами пытались портировать пол-года на NuttX и ещё на пару систем для встроенных решений, которые провозгласили POSIX-совместимость. Проект реализован на Си++.”

- “Для меня было принципиальным моментом когда мне показали обычное ПО на микроконтроллере”

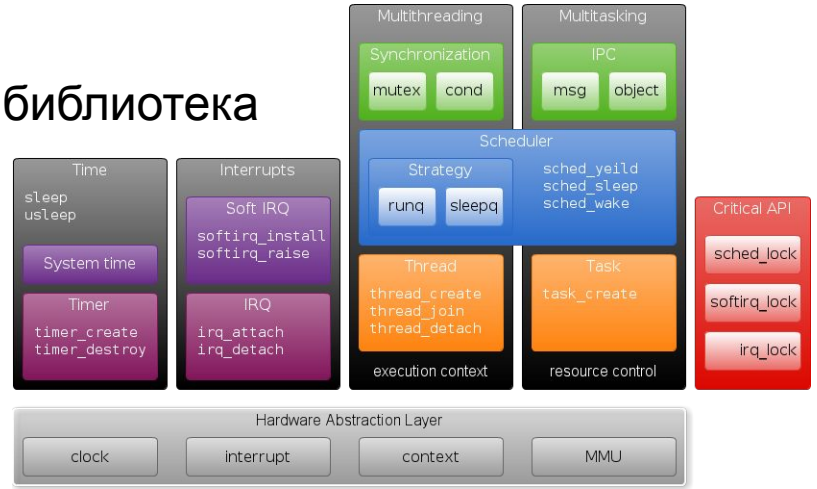
- “Ключевым моментом является, что не нужно сильно переучивать разработчиков”

Средства разработки

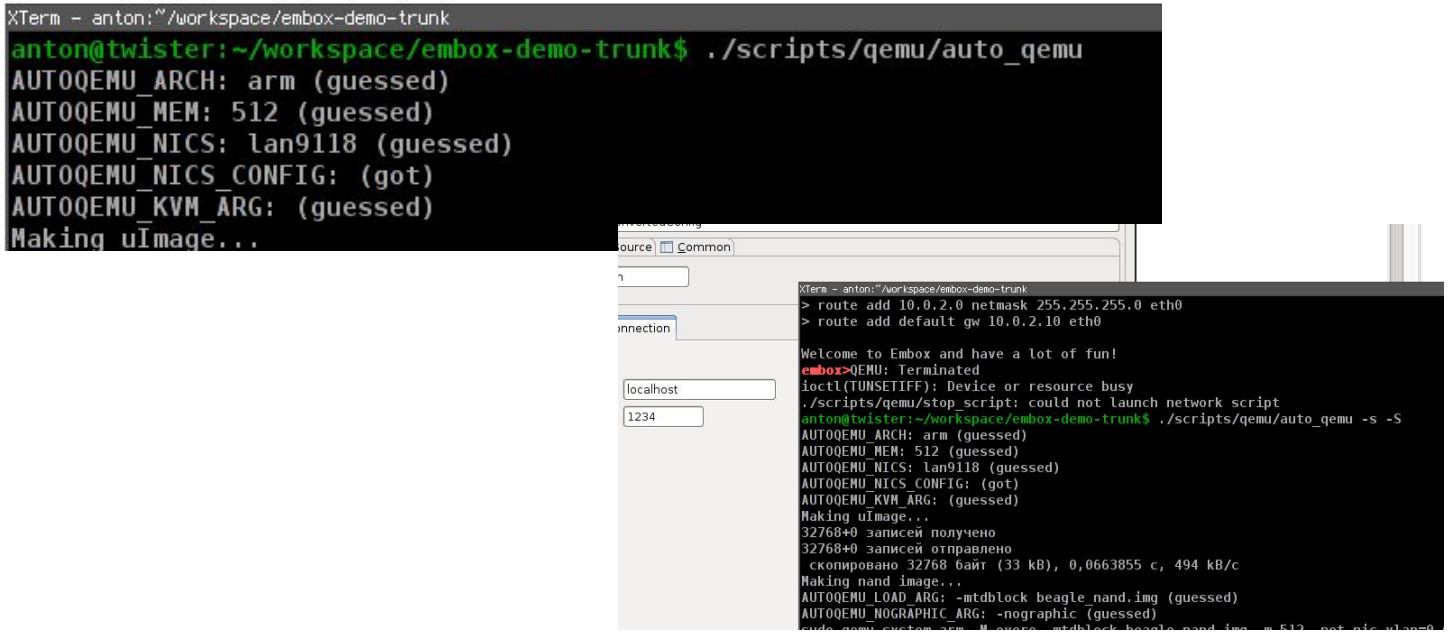
- Используются распространенные и открытые средства разработки (gcc, gdb, Eclipse, VSCode, openOCD, …)
- Ядро написано на **языке C** (gnu99).
- Используется собственная **стандартная библиотека**
- Поддержка распространенных языков: C++, Lua, JS, Tcl, Ruby, Perl, ...

Embox

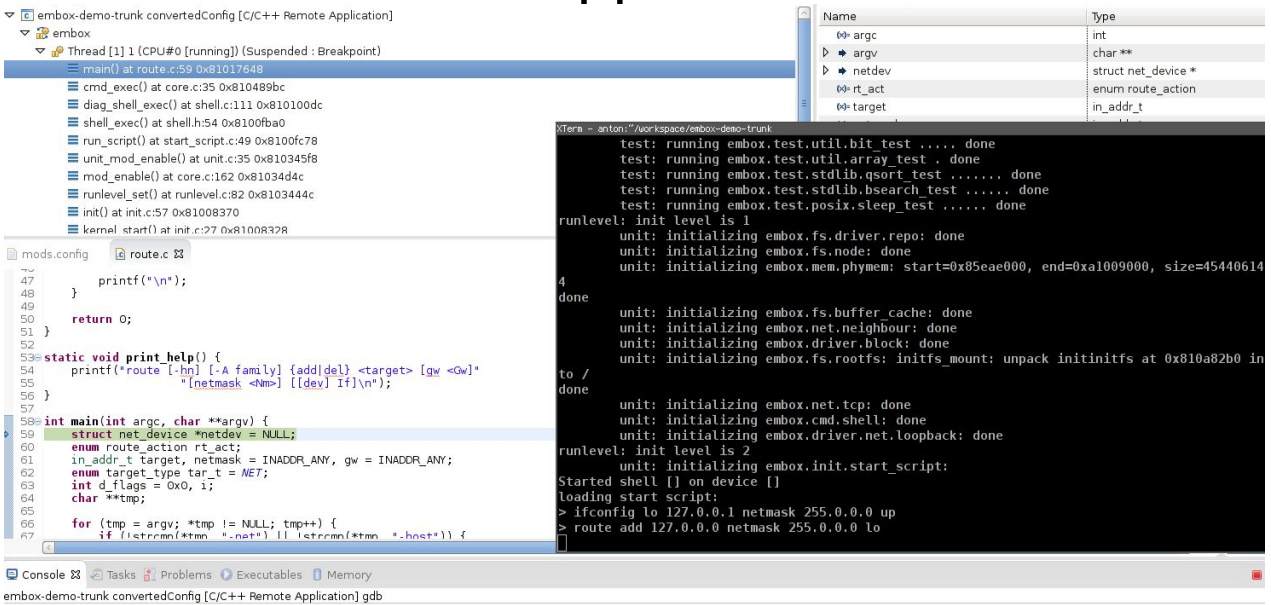
- Архитектуры ARM, x86, RISC-V, MIPS, SPARC, ...
- Полнофункциональная современная ОС
  - Ядро
    - Вытесняющая многозадачность
    - IPC
    - Управление памятью
    - …
  - Файловая система на VFS
  - Конфигурируемая стандартная библиотека
  - Собственный TCP/IP
  - Графическая подсистема
    - …



Простой запуск и отладка



Настольная отладка



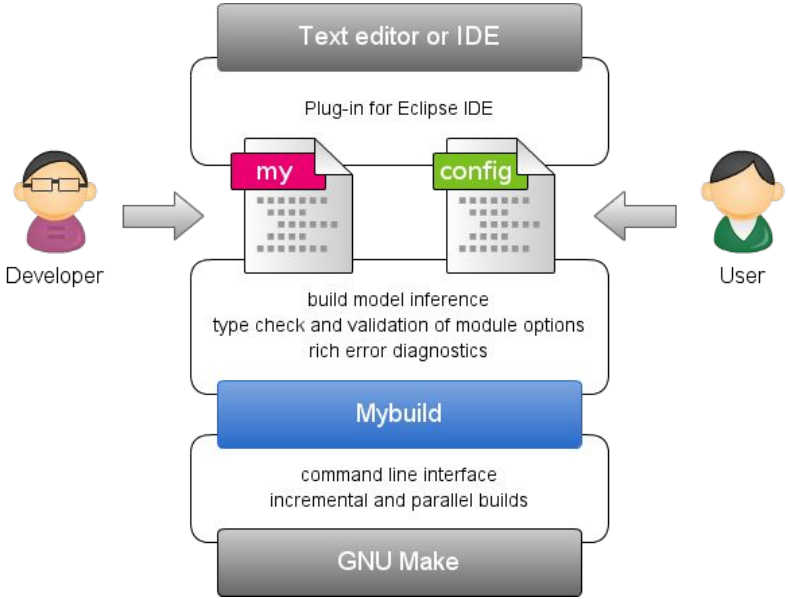
Embox

Операционная система под конкретную задачу, основанная на **специальном DSL языке** и использующая:

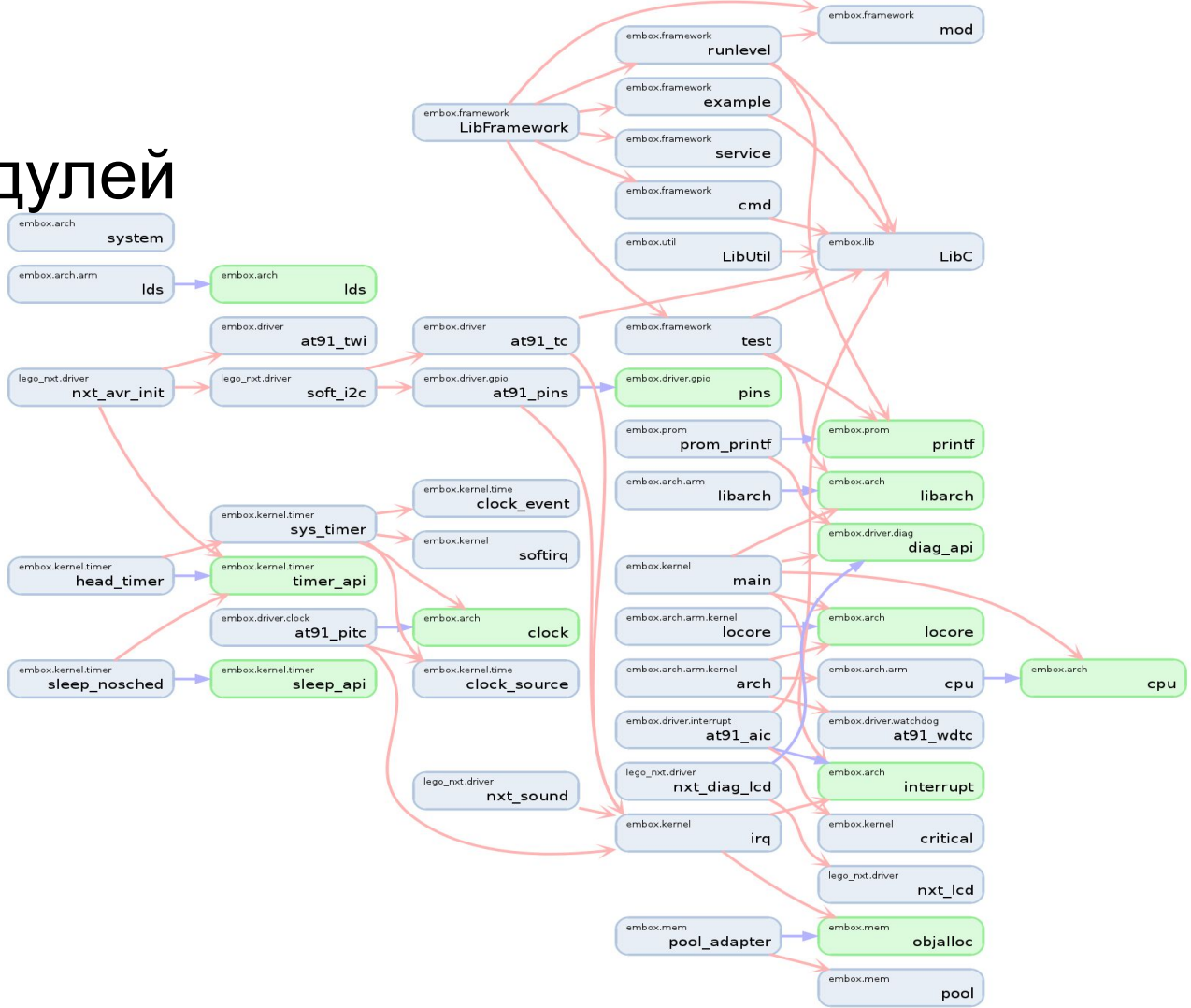
- Статическую информацию о задачах устройства
- Статическую конфигурацию системы
- Статический анализ зависимостей
- Статическую проверку параметров системы

Embox build system:
KBuild + OpenEmbedded + DevTree

Embox. Процесс сборки



Граф модулей



Embox: Hello world (source)

```
#include <stdio.h>


int main(int argc, char **argv) {
    printf("Hello, world!\n");
}
```

Embox: Hello world (Mybuild)

```
package embox.cmd


@AutoCmd
@Cmd(name = "hello_world", help="First Embox application")
module hello_world {
        source "hello_world.c"
}
```

Embox: Система Сборки (mods.conf)

```
include embox.kernel.task.resource.idesc_table(idesc_table_size=32)
include embox.kernel.task.resource.sig_table(sig_table_size=20)
include embox.kernel.task.resource.env(env_per_task=4,env_str_len=64)
include embox.kernel.thread.thread_local_none
include embox.kernel.thread.thread_cancel_disable
include embox.kernel.thread.signal.siginfoq(siginfo_pool_sz=8)
@Runlevel(2) include embox.kernel.sched.strategy.priority_based
…
include embox.cmd.hello_world
```

Embox. Окружение для разработки

- Кросс-компилятор (arm-none-eabi-(gcc))
- Стандартная библиотека (glibc, musl, newlib, …)
- Стартовый код
- Линкер скрипт
- Прикладные библиотеки

Embox: Mybuild (extbuild)

```
@Build(stage=2,script="$(EXTERNAL_MAKE)")
@Cmd(name = "nano",
…
...
    "")
module nano {
    source "^BUILD/extbld/^MOD_PATH/install/nano.o"
    @NoRuntime depends embox.compat.posix.regex
    depends embox.compat.posix.LibCurses
}
```

Embox: Mybuild (extbuild)

```
PKG_NAME := nano

PKG_VER  := 2.2.6

PKG_SOURCES := http://www.nano-editor.org/dist/v2.2/$(PKG_NAME)-$(PKG_VER).tar.gz

…

$(CONFIGURE) :

    export EMBOX_GCC_LINK=full; \

    cd $(PKG_SOURCE_DIR) && ( \

            ./configure --host=$(AUTOCONF_TARGET_TRIPLET) \

                    --target=$(AUTOCONF_TARGET_TRIPLET) \

           ...

    )
```

Embox: Mybuild (extbuild)

```
$(BUILD) :
    cd $(PKG_SOURCE_DIR) && ( \
     $(MAKE) MAKEFLAGS='$(EMBOX_IMPORTED_MAKEFLAGS)'; \
    )
    touch $@

$(INSTALL) :
    cp $(PKG_SOURCE_DIR)/src/nano $(PKG_INSTALL_DIR)/nano.o
    touch $@
```

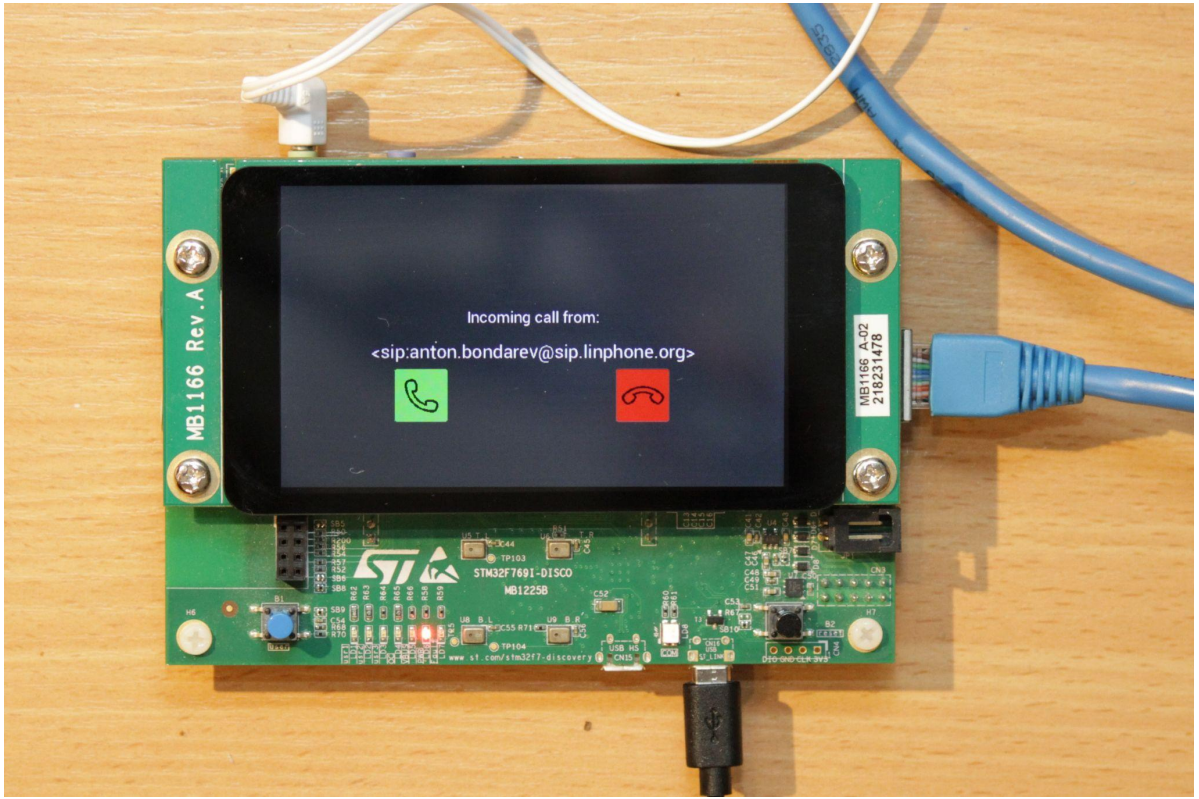
Embox: Mybuild (BuildDepends)

```
@BuildDepends(gcc_build)

@BuildArtifactPath(cppflags_before="-I$(abspath
$(EXTERNAL_BUILD_DIR))/third_party/gcc/gcc_build/install/_target/include/c++/_gcc_version
-I$(abspath
$(EXTERNAL_BUILD_DIR))/third_party/gcc/gcc_build/install/_target/include/c++/_gcc_version/_target
")

static module libstdcxx extends embox.lib.libstdcxx {

    @AddPrefix("^BUILD/extbld/third_party/gcc/gcc_build/install/libs")

    source "libstdc++.a"

    @NoRuntime depends gcc_build

}
```

Embox: Mybuild (BuildDepends)

```
@AutoCmd
@Cmd(name="stl_demo_sort1",
    help="example for stl algo sort",
    man="")
@BuildDepends(embox.lib.libstdcxx)
@Build(stage=2,script="true")
module stl_demo_sort1 {
    source "sort1.cpp"

    @NoRuntime depends embox.lib.libstdcxx
}
```

Embox: PJSIP


Incoming call from:
<sip:anton.bondarev@sip.linphone.org>
MB1166 Rev.A
STM32F769I-DISCO
MB1225B

PJSIP на Linux

```
$ ./configure \
        --prefix=$PREFIX \
        --disable-l16-codec   \
        --disable-ilbc-codec  \
        --disable-speex-codec \
        --disable-speex-aec   \
        --disable-gsm-codec   \
        --disable-g722-codec  \
        --disable-g7221-codec \
        --disable-libyuv \
        --disable-libwebrtc
$ make dep && make
```

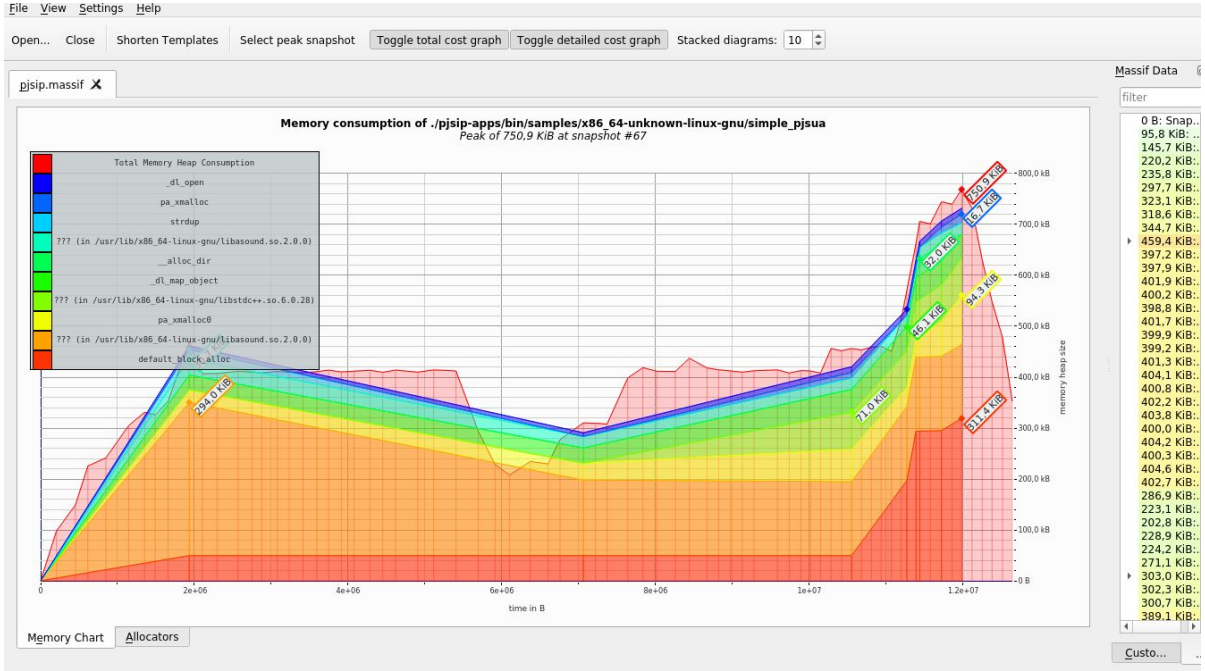
PJSIP на Linux

- Используем стандартный пример
- Модификация вынос параметров для аккаунта

```
#define SIP_DOMAIN      <sip_domain>
#define SIP_USER        <sip_user>
#define SIP_PASSWD      <sip_passwd>
```

valgrind on Linux

- valgrind --tool=massif --time-unit=B --massif-out-file=pjsip.massif ./pjsip-apps/bin/samples/x86_64-unknown-linux-gnu/simple_pjsua
- massif-visualizer pjsip.massif



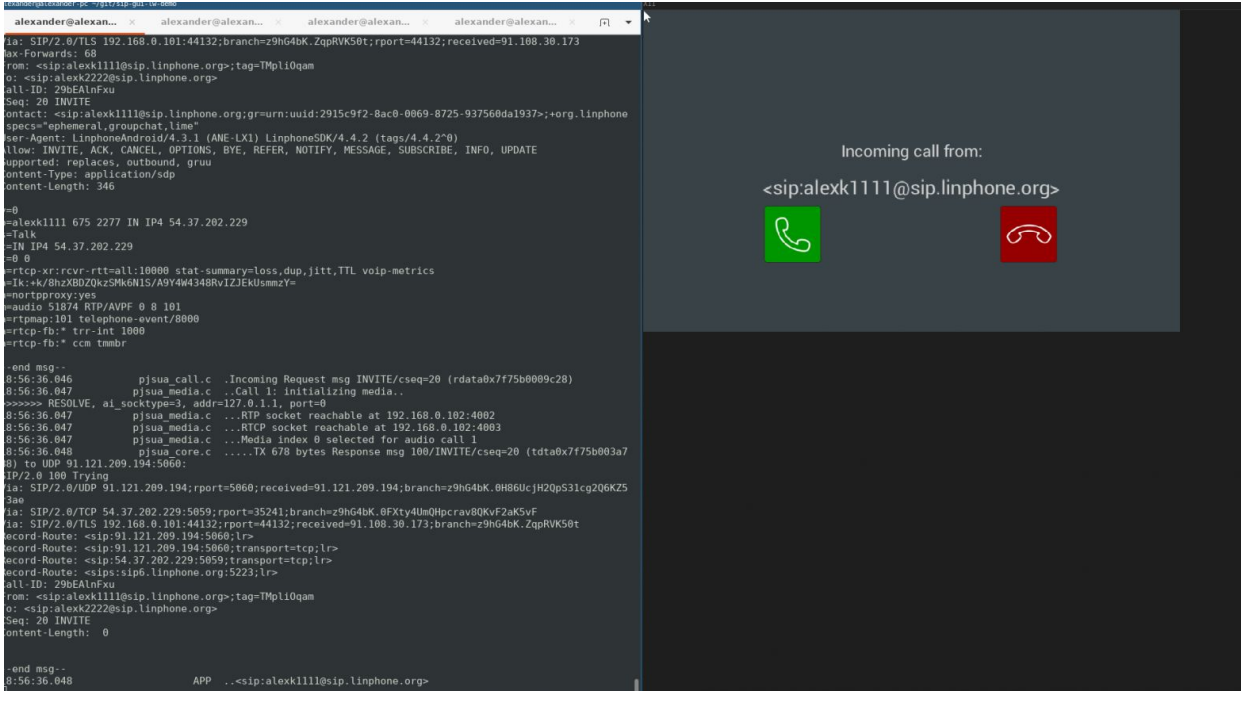
Embox: PJSIP

```
DISABLE_FEATURES := \
        l16-codec   \
        ilbc-codec  \
        speex-codec \
        speex-aec   \
        gsm-codec   \
        g722-codec  \
        g7221-codec \
        libyuv \
        libwebrtc
```

```
./configure \
CC=$(EMBOX_GCC) \
CXX=$(EMBOX_GXX) \
--host=$(AUTOCONF_TARGET_TRIPLET) \
--target=$(AUTOCONF_TARGET_TRIPLET) \
--prefix=$(PJSIP_INSTALL_DIR) \
$(DISABLE_FEATURES:%=--disable-%) \
--with-external-pa; \
)
```

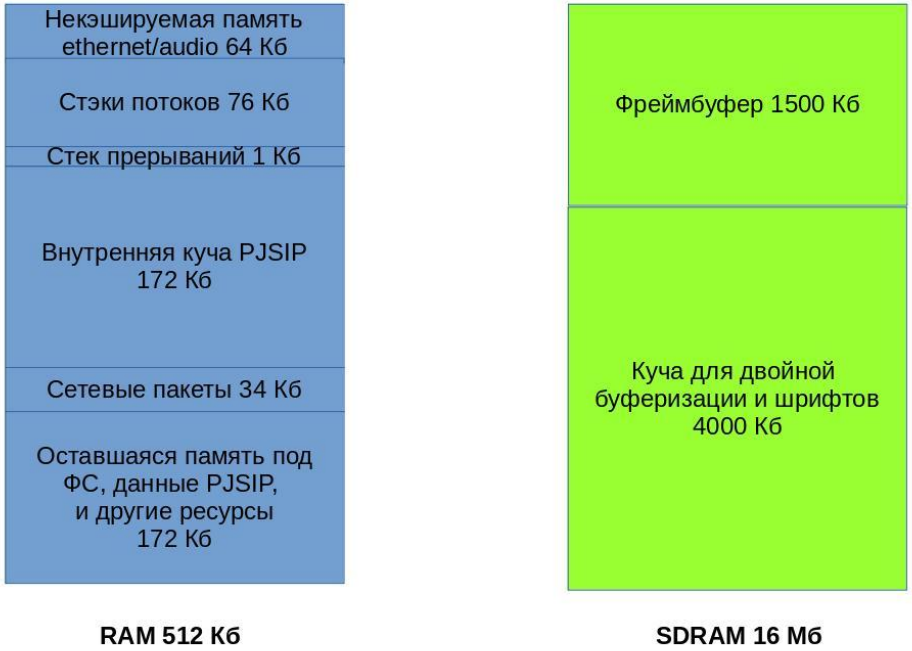
Embox: PJSIP on QEMU

- Добавили графический интерфейс
- Запустили на qemu

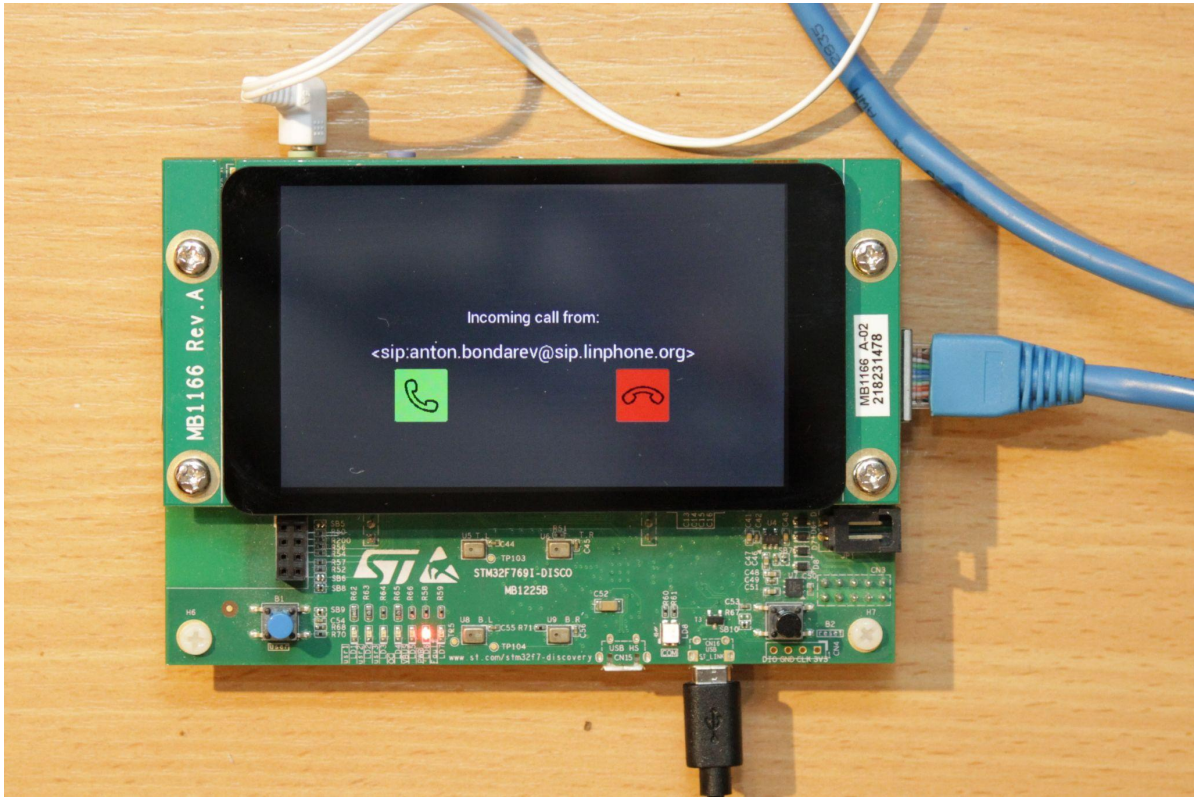


Embox: PJSIP на плате

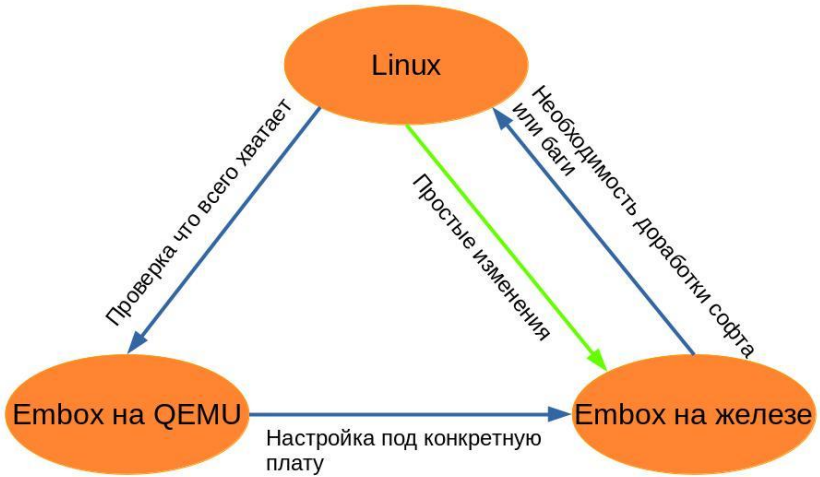
- карта памяти



Embox: PJSIP


Incoming call from:
<sip:anton.bondarev@sip.linphone.org>
MB1166 Rev.A
STM32F769I-DISCO
MB1225B

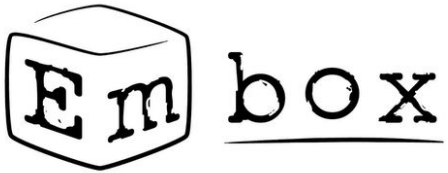
Процесс разработки под Embox



Заключение

- Наиболее развитая экосистема у ОС Linux
- Экосистема должна обеспечивать удобство, качество и скорость разработки под ОС.
- Embox обеспечивает эти свойства, за счет широкого использования экосистемы Linux

Contacts



Essential toolbox for embedded development

youtube

- **https://www.youtube.com/@embox-rtos**

Embox Project Homepage

- **http://embox.github.io/**

Telegram chat

- **https://t.me/embox_chat**