Санкт-Петербургский государственный университет

Кафедра системного программирования

Расширение функциональности ОС Embox для применения в АСУ ТП

Дмитрий Алексеевич Пилюк, группа 22.Б11-мм

Научный руководитель: к.ф.-м.н. Д.В. Луцив, доцент кафедры системного программирования
Консультант: А.В. Бондарев, генеральный директор ООО «Embox»

Санкт-Петербург
2025

Введение

Embox — операционная система реального времени. Её разработка начиналась, как студенческий проект на кафедре системного программировани Математико-механического факультета СПБГУ

- Модульная архитектура
- Поддержка POSIX
- Real time
- Маленький размер

Введение

- АСУТП — система технических и программных средств, предназначенных для автоматизации управления технологическим процессом
- IEC 61131-3 — раздел международного стандарта IEC 61131, описывающий языки программирования для программируемых логических контроллеров
- IEC 61499 — открытый стандарт распределенных систем управления и автоматизации. Стандарт описывает общую модель для распределенных систем управления и основан на стандарте IEC 61131
- Сейчас актуальна тема использования отечественного ПО для ПЛК на предприятиях

Существующие решения

- IEC 61131:
  - Beremiz
  - OpenPLC editor
  - Matiec
- IEC 61499:
  - Eclipse 4diac
    - 4diac IDE
    - 4diac forte

Постановка задачи

Целью работы является расширение возможностей ОСРВ Embox для использования в сфере АСУ ТП.
Задачи:

1. Предоставить среду разработки, поддерживающую создание и редактирование программ стандарта IEC 61499 для Embox
2. Добавить в Embox поддержку среды исполнения
3. Поставить эксперимент

Реализация

В данной работе было сделано:

- Добавлены необходимые зависимости и портирован 4diac forte в Embox
- Реализован механизм добавления собственных блоков
- Добавлены функциональные блоки управления светодиодами в Embox
- Запущен Embox с forte на плате STM nucleo-f429zi
- Добавлена инструкция использования

Создание собственных FB

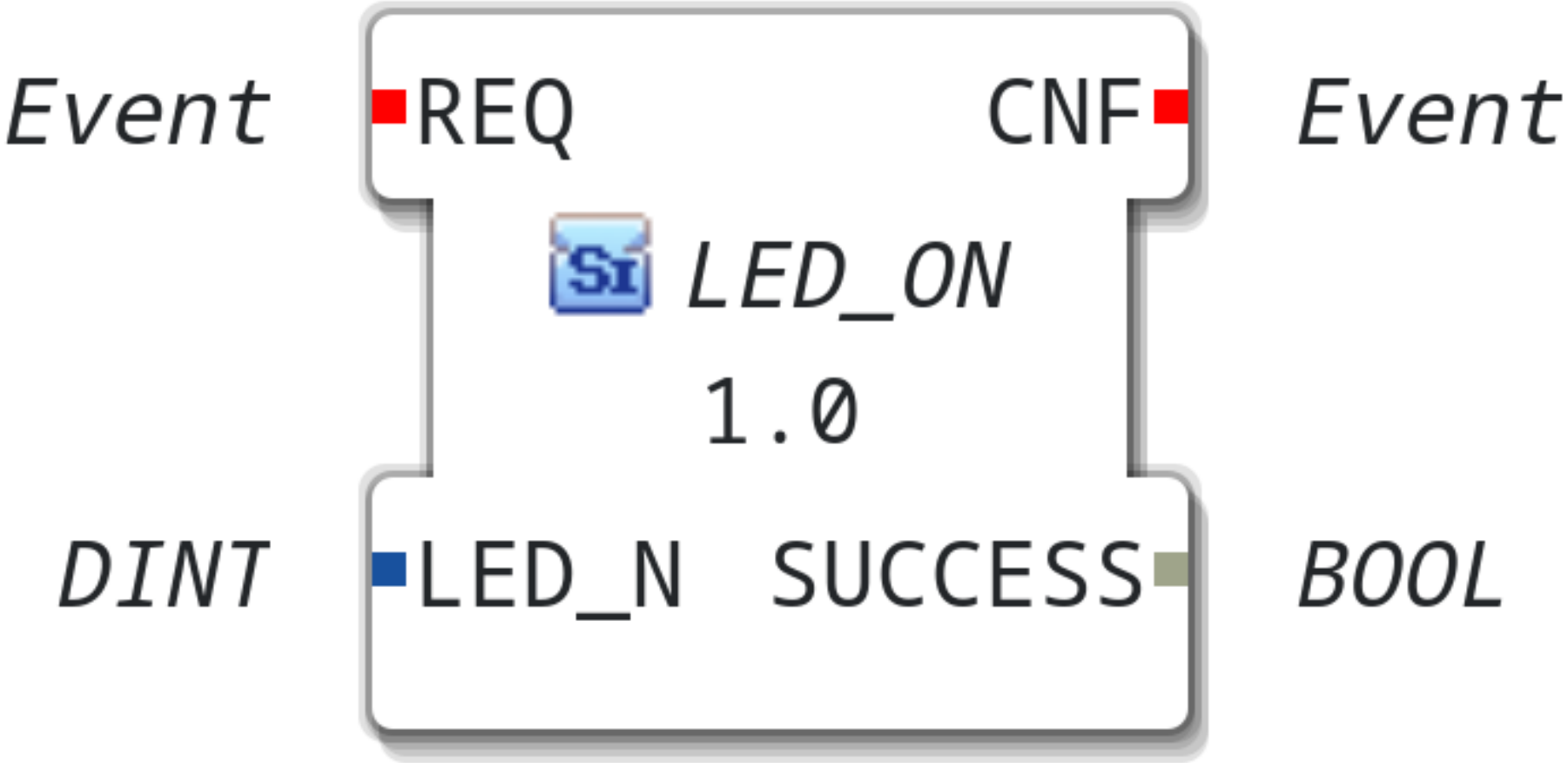

Рис.: Интерфейс FB

Создание собственных FB

```
12  #ifndef _LED_ON_H_
13  #define _LED_ON_H_
14
15  #include "funcbloc.h"
16  #include "forte_bool.h"
17  #include "forte_dint.h"
18  #include "forte_array_at.h"
19
20  #include <lib/leddrv.h>
21
22  class FORTE_LED_ON: public CFunctionBlock {
23    DECLARE_FIRMWARE_FB(FORTE_LED_ON)
24
25  private:
26    static const CStringDictionary::TStringId scm_anDataInputNames[];
27    static const CStringDictionary::TStringId scm_anDataInputTypeIds[];
28
29    static const CStringDictionary::TStringId scm_anDataOutputNames[];
30    static const CStringDictionary::TStringId scm_anDataOutputTypeIds[];
31
32    static const TEventID scm_nEventREQID = 0;
33
34    static const TForteInt16 scm_anEIWithIndexes[];
35    static const CStringDictionary::TStringId scm_anEventInputNames[];
36
37    static const TEventID scm_nEventCNFID = 0;
38
39    static const TForteInt16 scm_anEOWithIndexes[];
40    static const CStringDictionary::TStringId scm_anEventOutputNames[];
41
42
43    static const SFBInterfaceSpec scm_stFBInterfaceSpec;
44
45    CIEC_DINT &st_LED_N() {
46      return *static_cast<CIEC_DINT*>(getDI(0));
47    }
48
49    CIEC_BOOL &st_SUCCESS() {
50      return *static_cast<CIEC_BOOL*>(getDO(0));
51    }
52
53
54    FORTE_FB_DATA_ARRAY(1, 1, 1, 0);
55
56    void executeEvent(int pa_nEIID);
57
58  public:
59     FORTE_LED_ON(const CStringDictionary::TStringId pa_nInstanceNameId, CResource *pa_poSrcRes) :
60         CFunctionBlock( pa_poSrcRes, &scm_stFBInterfaceSpec, pa_nInstanceNameId, m_anFBConnData, m_anFBVarsData) {
61     };
62
63    virtual ~FORTE_LED_ON() = default;
64  };
65
66  #endif // _LED_ON_H_
```

Рис.: LED_ON.h

Создание собственных FB

```
12  #include "LED_ON.h"
13  #ifdef FORTE_ENABLE_GENERATED_SOURCE_CPP
14  #include "LED_ON_gen.cpp"
15  #endif
16
17
18  DEFINE_FIRMWARE_FB(FORTE_LED_ON, g_nStringIdLED_ON)
19
20  const CStringDictionary::TStringId FORTE_LED_ON::scm_anDataInputNames[] = {g_nStringIdLED_N};
21
22  const CStringDictionary::TStringId FORTE_LED_ON::scm_anDataInputTypeIds[] = {g_nStringIdDINT};
23
24  const CStringDictionary::TStringId FORTE_LED_ON::scm_anDataOutputNames[] = {g_nStringIdSUCCESS};
25
26  const CStringDictionary::TStringId FORTE_LED_ON::scm_anDataOutputTypeIds[] = {g_nStringIdBOOL};
27
28  const TForteInt16 FORTE_LED_ON::scm_anEIWithIndexes[] = {-1};
29  const CStringDictionary::TStringId FORTE_LED_ON::scm_anEventInputNames[] = {g_nStringIdREQ};
30
31  const TForteInt16 FORTE_LED_ON::scm_anEOWithIndexes[] = {-1};
32  const CStringDictionary::TStringId FORTE_LED_ON::scm_anEventOutputNames[] = {g_nStringIdCNF};
33
34
35  const SFBInterfaceSpec FORTE_LED_ON::scm_stFBInterfaceSpec = {
36    1, scm_anEventInputNames, nullptr, scm_anEIWithIndexes,
37    1, scm_anEventOutputNames, nullptr, scm_anEOWithIndexes,
38    1, scm_anDataInputNames, scm_anDataInputTypeIds,
39    1, scm_anDataOutputNames, scm_anDataOutputTypeIds,
40    0, nullptr
41  };
42
43  void FORTE_LED_ON::executeEvent(int pa_nEIID) {
44    if (pa_nEIID == scm_nEventREQID) {
45        int err = leddrv_led_on(!st_LED_N());
46        st_SUCCESS() = err? false : true;
47        sendOutputEvent(scm_nEventCNFID);
48    }
49  }
```

Рис.: LED_ON.cpp

Создание собственных FB

```
43  void FORTE_LED_ON::executeEvent(int pa_nEIID) {
44    if (pa_nEIID == scm_nEventREQID) {
45      int err = leddrv_led_on(!st_LED_N());
46      st_SUCCESS() = err? false : true;
47      sendOutputEvent(scm_nEventCNFID);
48    }
49  }
```

Рис.: LED_ON.cpp

Эксперимент

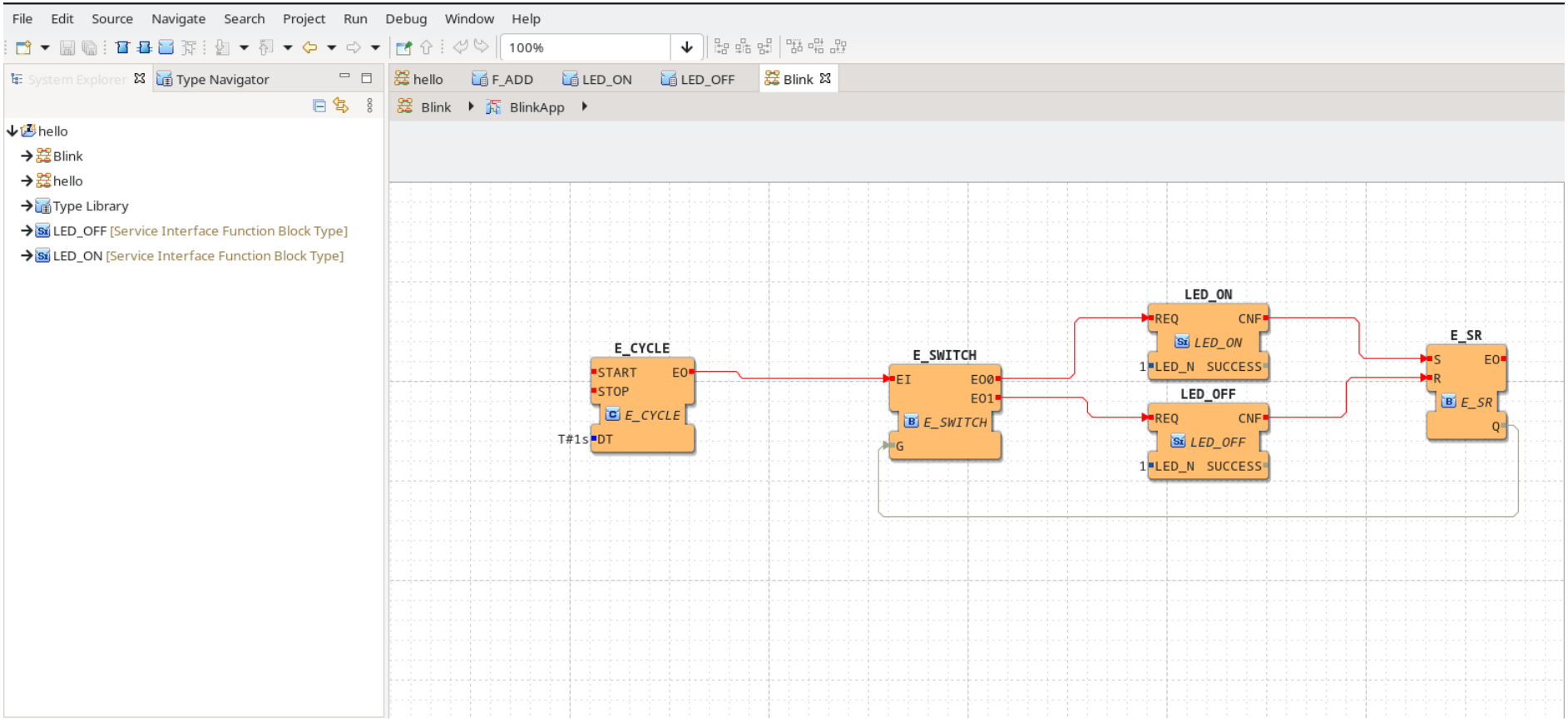

Рис.: Программа в 4diac IDE

Эксперимент

```
>netmanager
>tish
root@embox:/#forte -f Blink_Embox_qemu.fboot
INFO: T#12509014260: Using provided bootfile location set in the command line: Blink_Embox_qemu.fboot
INFO: T#12512470140: Boot file Blink_Embox_qemu.fboot opened
INFO: T#12521066200: FORTE is up and running
INFO: T#12523141340: Bootfile correctly loaded
INFO: T#12524372640: Closing bootfile
[info] (leddrv_stub) LED[0] := 1
[info] (leddrv_stub) LED[0] := 0
[info] (leddrv_stub) LED[0] := 1
[info] (leddrv_stub) LED[0] := 0
[info] (leddrv_stub) LED[0] := 1
[info] (leddrv_stub) LED[0] := 0
[info] (leddrv_stub) LED[0] := 1
[info] (leddrv_stub) LED[0] := 0
[info] (leddrv_stub) LED[0] := 1
```

Рис.: Пример. Запуск forte на Embox

Результаты

- Выбран проект Eclipse 4diac для создания программ. Реализована возможность добавления собственных функциональных блоков в Embox
- Добавлена среда исполнения 4diac forte в Embox
- Проведен эксперимент