Описание файлов исходного текста программ, генерируемых кодогенератором Си, входящим в состав программного обеспечения SimInTech

Кодогенератор генерирует 4 файла:

<имя алгоритма>.h - заголовочный файл с описанием привязки переменных к рабочим массивам

Структура файла:

/* Шапка сгенерированной программы */

/* ----------

Routine name: my_diagram

Description:

Project file: Simple.prt

---------- */

/* Определение использованных типов данных */

/* --- Base generator data types --- */

/* Real data type */

typedef double g_real_type;

/* Integer data type */

typedef int g_int_type;

/* Boolean data type */

typedef char g_boolean_type;

/* Complex data type */

typedef complex_64 g_complex_type;

/* Default initialization values */

/* Структурные идентификаторы модели */

/* Project signal database hash */

const unsigned int sp_database_hash_32=0;

/* Project sheme structure hash */

```
const unsigned int sp_sheme_hash_32=1564302271;

/* Константы с начальными значениями переменных модели */

const double xbc8_default=0;

const char xta9_default=0;

const double my_diagramv1_out_0_default=0;

const char my_diagramv0_out_0_default=0;

const double my_diagramv1_k_default=1;

/* Общие размерностные параметры модели */

/* Main structures defines */

/* External variables count */

#define ext_vars_count 2

/* Dynamical variables count */

#define din_vars_count 1

/* Internal state variables count */

#define state_vars_count 1

/* Constants count */

#define const_count 1

/* Некоторые настройки исходной модели, которые могут быть использованы в сгенерированной программе */

/*  --- Source model preferences --- */

/* Minimum integration step */

#define INTEGRATION_MIN_STEP 1E-5

/* Maximum integration step */

#define INTEGRATION_MAX_STEP 1

/* Integration synchronization step */
```

```
#define INTEGRATION_SYNC_STEP 0

/* Model integration method */

#define INTEGRATION_METHOD 5

/* Model relative error */

#define INTEGRATION_RELATIVE_ERROR 0.0001

/* Model absolute error */

#define INTEGRATION_ABSOLUTE_ERROR 1

/* Model end time */

#define INTEGRATION_END_TIME 10

/* Model maximum iteration count */

#define INTEGRATION_MAX_LOOP_ITER_COUNT 10


/* Таблица имён и размерностей внешних переменных модели */

const ext_var_info_record ext_vars_names[ext_vars_count] = {

{"xbc8",   vt_double,   {1}, 0,dir_out,"", (void*)&xbc8_default, sizeof(double)} ,

{"xta9",   vt_bool,   {1}, 1,dir_input,"", (void*)&xta9_default, sizeof(char)}

};

/* Определения привязки внешних переменных к массиву указателей */

#define xbc8 (*(double*)(ext_vars_addr[0]))

#define xta9 (*(char*)(ext_vars_addr[1]))

/* Таблица имён и размерностей динамических переменных модели */

const ext_var_info_record din_vars_names[din_vars_count] = {

{"my_diagramv1_out_0",   vt_double,   {1}, 0, dir_inout,"TIntegrator integrator output",
(void*)&my_diagramv1_out_0_default, sizeof(double)}

};

/* Определения привязки динамических переменных к общему массиву */

#define din_vars_dimension 1

#define my_diagramv1_out_0 din_vars[0]

#define my_diagramv1_out_0_deri derivates[0]
```

```
/* Таблица имён и размерностей переменных состояния модели */

const ext_var_info_record state_vars_names[state_vars_count] = {

{"my_diagramv0_out_0",  vt_bool,  {1}, 0, dir_inout,"Input pin state variable",
(void*)&my_diagramv0_out_0_default, sizeof(char)}

};

/* Определение структуры с переменными состояния модели */

  typedef struct {

char my_diagramv0_out_0_;

  } t_state_vars;

/* Таблица имён и размерностей констант модели */

const ext_var_info_record const_names[const_count] = {

{"my_diagramv1_k",  vt_double,  {1}, 0, dir_input,"TIntegrator gain",
(void*)&my_diagramv1_k_default, sizeof(double)}

};

/* Определение структуры с константами модели */

  typedef struct {

double my_diagramv1_k_;

  } t_consts;

 typedef char t_local;
```

<имя алгоритма>.inc – текст основной программы на языке Си (без заголовка функции), которая выполняется на каждом шаге расчёта

Структура файла:

/* Шапка файла программы */

```
/*  ----------------------------------------------------

    Routine name:  my_diagram

    Description:

    Project file:  Simple.prt


---------------------------------------------------  */
```

/* Определение зарезервированных локальных (стековых) переменных */

```
/*       Local stack variables                */
int i;
int j;
int c;
int itmp1;
int itmp2;
double tmp1;
double tmp2;
double tmp3;
double tmp4;
double tmp5;
double tmp6;
double tmp7;
char f;
char tmp_f_1;
char u_s;
char u_r;
ret = 0;
```

/* Селектор выполняемой секции кода в зависимости от стадии расчёта*/

switch (action){

case f_Stop:{

/* Код выполняемый при остановке расчёта */

};break;

case f_GetDeri:{

/* Код выполняемый на стадии вычисления значений производных динамических переменных */

/* Index=1

UID=1

GeneratorClassName=TIntegrator

Name=Integrator4

Type=Интегратор */

my_diagramv1_out_0_deri = consts->my_diagramv1_k_*state_vars->my_diagramv0_out_0_;

};break;

case f_GetAlgFun:{

/* Код выполняемый на стадии вычисления значений функций алгебраических переменных */

};break;

default:{

/* Код выполняемый на стадии вычисления выходов всех блоков основного или промежуточного шага */

/* Index=0

UID=0

GeneratorClassName=TInputPin

```
   Name=Const_source10
   Type=Входной контакт s3 */

state_vars->my_diagramv0_out_0_ = xta9;

/* Index=2
   UID=2
   GeneratorClassName=TOutPin
   Name=OutPin7
   Type=Выходной контакт s3 */

xbc8 = my_diagramv1_out_0;
};break;
};
```

<имя алгоритма>_init.inc – текст программы на языке Си, которая выполняется для инициализации начальных значений переменных

Структура файла:

/* Шапка файла программы */

/* --

Routine name: my_diagram

Description:

Project file: Simple.prt

-- */

/* Определение зарезервированных локальных (стековых) переменных */

int i;

int j;

int c;

ret = 0;

/* Копирование данных из констант в рабочие массивы – текст программы на языке Си */

memcpy(&my_diagramv1_out_0,&my_diagramv1_out_0_default,sizeof(my_diagramv1_out_0_default));

memcpy(&state_vars->my_diagramv0_out_0_,&my_diagramv0_out_0_default,sizeof(my_diagramv0_out_0_default));

memcpy(&consts->my_diagramv1_k_,&my_diagramv1_k_default,sizeof(my_diagramv1_k_default));

<имя алгоритма>_state.inc – текст программы на языке Си, которая выполняется после после основной программы для запоминания состояний задержек на шаг интегрирования и некоторых других блоков, имеющих дискретные состояния с приоритетной сортировкой.

/* Шапка файла программы */

/* --

Routine name: my_diagram

Description:

Project file: Simple.prt

---------- */

/* Определение зарезервированных локальных (стековых) переменных */

/* Local stack variables */

int i;

int j;

int c;

int itmp1;

int itmp2;

double tmp1;

double tmp2;

double tmp3;

double tmp4;

double tmp5;

double tmp6;

double tmp7;

char f;

char tmp_f_1;

char u_s;

char u_r;

ret = 0;

/* Дальше идёт текст программ на языке Си, который вызывается на стадии обновления выходов (промежуточной или конечной) шага интегрирования после основной программы – запоминание состояний блоков задержки на шаг */