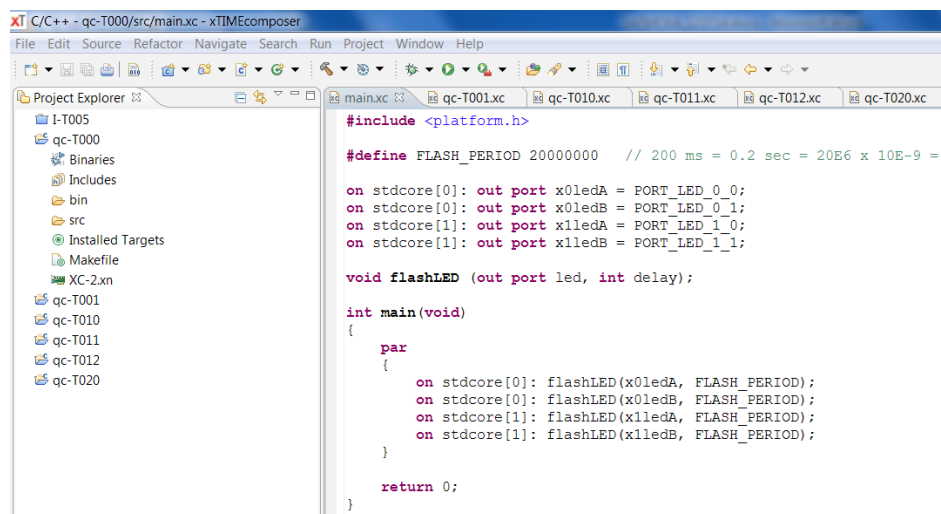


3. ЕКСПЕРИМЕНТАЛНА ОЦЕНКА НА ПРЕДЛОЖЕНИТЕ МОДЕЛИ

Разделът е посветен на основните моменти от експерименталната оценка на предложените паралелни структурни модели. Изборът на разглежданите моменти от една страна се определя от изискванията на използваната изпълнителна среда и развойни средства, а от друга - от спецификата на самите модели.

Опитната постановка за тестването и изпълнението на моделите се основава на типичната за вградените системи технология *инструментална-целева машина (host-target)*. На инструменталната машина се изпълнява развойната среда *xTIMEComposer Studio*¹ на фирмата XMOS [Б.12, Б.13, Б.54]. Със средствата на развойната



Фиг. 3. 1. Основният екран на развойната среда с разработените проекти

¹ По време на този етап от изследванията развойната среда на XMOS премина през няколко версии. Стремехът бе винаги да се работи с последната актуална версия. В заключителната част от работите това бе версия 12.2.0. През ноември 2013 г. успешни тестове бяха направени с новата версия 13 във варианта *xTIMEComposer Community*.

среда моделите се програмират на паралелния език XC, изпълнимият им код се зарежда в целевата машина и стартира в тестови режим или в режим на цялостно изпълнение.

Паралелната изпълнителна среда се осигурява от целевата машина XC-2 и съдържа 4-ядрен SMT/TLP процесор XS1-G4 от типа XCORE. В нея могат да се изпълняват едновременно до 32 паралелни процеса, максималният брой на физическите комуникационни канали в системата е 64 (т. 1.2).

На фиг. 3.1 е показан основният екран на развойната среда *xTIMEComposer Studio* с разработените проекти, разгледани в т. 2. Отворени са и петте разработени проекта, както и техните основни изходни модули. В работата са използвани и част от резултатите от проекта *I-T005*, посветен на методите за генериране на случайни последователности чрез използваната паралелна изпълнителна среда XCORE/XC [A.4].

Последователно ще се спрем на изпълнението на предложените три паралелни модела, като се концентрираме само върху основните моменти от това изпълнение. За целите на експерименталната оценка е необходимо да се достъпват вътрешни за моделите променливи. Оттук и необходимостта от тестовия (*Debug*) режим на работа на развойната среда.

+++

За да се стартира паралелният модел на поляризацията от т. 2.2, се отваря проектът *qc-T001* и главният му изходен модул *qc-T001.xc*. Задава се естествена поляризация и типа на генератора на случайни последователности

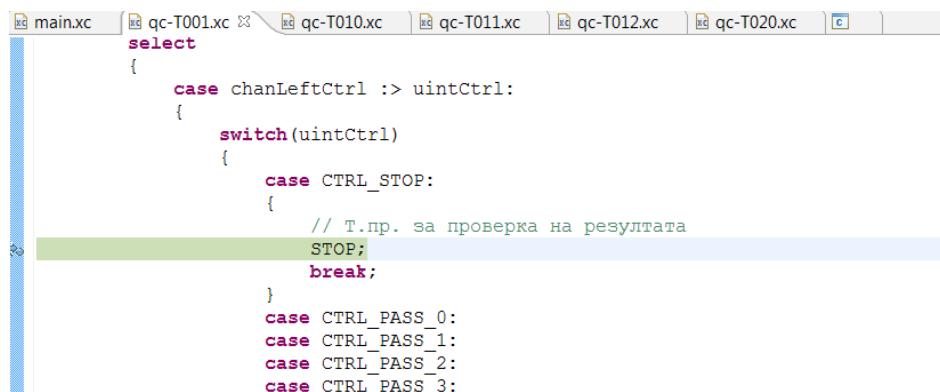
```
// ТИП ПОЛЯРИЗАЦИЯ НА ИЗТОЧНИКА
#define LE_POLAR          0
// 0 - POLAR_NATURAL
// 1 - POLAR_HORIZONTAL
// 2 - POLAR_VERTICAL
// 3 - POLAR_ANGULAR_PLUS
// 4 - POLAR_ANGULAR_MINUS

#define RNG_TYPE           1
// 1 - Генератор тип RNG с примитивна функция src32 на
изпълнителната среда;
// 2 - Генератор тип TRNG с кръговите осцилатори от
изпълнителната среда
// поддържа се при XS1-L, но не и при XS1-G4.
```

Опитната постановка предполага използването само на естествена поляризация. Другите възможности са добавени с изследователска цел. Използвани са и по време на настройката на модела.

Проектът се компилира във вариант за настройка (Debug). На реда на оператора `STOP` в главната функция `taskLightDetector()` на фотодетектора *LD* се поставя точка на прекъсване. Това е моментът, непосредствено преди приключването на работата на модела.

Моделът се стартира в тестови режим, например чрез клавиша *F11*. При това, изпълнимият код на модела се зарежда и стартира в целевата машина. Фактическото му изпълнение е около секунда, след което се достига точката на прекъсване (фиг. 3.2).



Фиг. 3. 2. Точка на прекъсване в края на изпълнението на модела на поляризацията

От екрана на дебъгера на развойната среда, в полето на променливите, се вземат стойностите на броячите на генерираните от източника *LE* и регистрираните от фотодетектора *LD* фотони (фиг. 3.3).

Приведени в таблицата от фиг. 2.12. резултати са получени по този начин чрез десет последователни стартирания на модела². Горните четири целочислени променливи от фиг. 3.3 са броячите на генерираните от източника *LE* фотони с дадена поляризация за всички четири опита.

Масивът `intCounter[]` съдържа стойностите на броячите на регистрираните от фотодетектора *LD* фотони за всеки един от опитите – 0 (без поляризатори), 1, 2 и 3³.

2 Разбира се, могат да се проведат и значително по-голям обем опити, за да се спазят формалните изисквания за статистическа достоверност, но това принципно няма да промени нищо в случая.

3 Може да се проследи по индекса в масива `intCounter[]` на фиг. 3.3.

Name	Value
intCounterHORIZONTAL	8225
intCounterVERTICAL	8065
intCounterANGULAR_PLUS	8164
intCounterANGULAR_MINUS	8314
chanLeftCtrl	770
chanLeft	1026
uintCtrl	4294967295
intCounter	0x0001fe30
intCounter[0]	8192
intCounter[1]	4144
intCounter[2]	0
intCounter[3]	1054
intInx	3
phLeft	{...}

Фиг. 3. 3. Количествени резултати от изпълнението на модела на поляризацията

След точката на прекъсване изпълнението на модела може да се спре принудително. Интересно е обаче изпълнението да се възобнови, например чрез клавиша *F8*. Така може да се проследи дали изпълнението ще се терминира автоматично, което е признак за съгласуваното приключване и на петте паралелни процеса⁴.

Работата с останалите два модела протича по същата схема. Единствената разлика е в избора на точките на прекъсване, специфичен за конкретния модел. В зависимост от поставената цел може да се използват и повече от една точки на прекъсване.

+++

За да се стартира паралелният модел на суперпозицията от т. 2.3, се отваря един от вариантите му - проектът *qc-T010*, *qc-T012* или *qc-T012*. Ще изберем проекта *qc-T012*, който е краен резултат от постепенното развитие на модела. Отваряме и главният му изходен модул *qc-T012.xc*.

Задава се началното квантово състояние на генерираните от

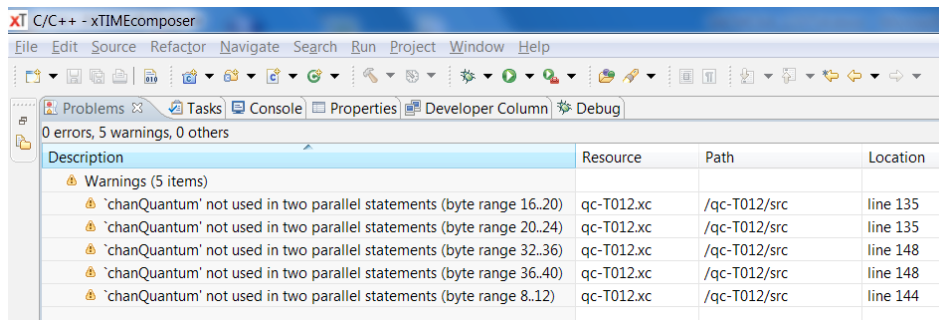
⁴ Паралелната конструкция *par* не завършва, докато не приключат всички съставлящи я процеси.

източника LE фотони

```
// НАЧАЛНО КВАНТОВО СЪСТОЯНИЕ НА ФОТОНА,  
// ГЕНЕРИРАН ОТ ИЗТОЧНИКА LE  
#define LE_QS 0  
// 0 - квантово състояние QS_0 с кет-вектор  $|0\rangle$   
// 1 - квантово състояние QS_1 с кет-вектор  $|1\rangle$ 
```

Началното състояние с кет-вектора $|0\rangle$ отговаря на означенията на фиг. 2.17. Моделът, разбира се, е инвариантен по отношение на избраното начално базисно състояние. Фотодетекторът, разположен в първоначалното направление на лъча⁵, ще регистрира 100% от излъчените фотони. Докато фотодетекторът, разположен ортогонално на първоначалното направление на лъча⁶, ще регистрира 0% фотони.

Проектът се компилира във вариант за настройка (*Debug*). Както бе посочено в т. 2.3, заради универсалността на главната функция `taskMirror()` на огледалата, се налага да се резервират по-голям брой канали от реално необходимите. Компиляторът извежда предупреждения за неизползваните канали (фиг. 3.4), които се игнорират.

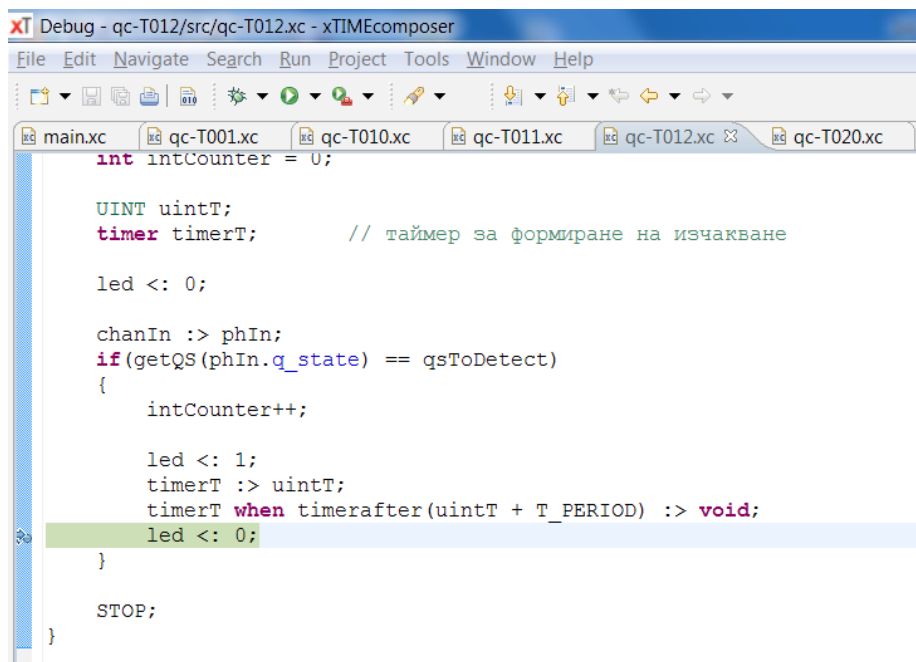


Фиг. 3. 4. Предупреждения за неизползваните канали от `taskMirror()`

Точката на прекъсване трябва да се постави на реда на оператора `led <: 0` в главната функция `taskLightDetector()` на фотодетектора LD , непосредствено след изтичане задръжката на таймера от 5 сек. Това е моментът преди приключването на работата на модела, когато светодиодът на фотодетектора, регистрирал фотона, се гаси.

Моделът се стартира в тестови режим, например чрез клавиша F11. При това, изпълнимият код на модела се зарежда и стартира

- 5 При кет-вектор $|0\rangle$ това е LD_0 .
- 6 При кет-вектор $|1\rangle$ това е LD_1 .



Фиг. 3. 5. Точка на прекъсване в края на изпълнението на модела на квантовата суперпозиция

Name	Value
chanIn	197122
led	68096
qsToDetect	QS_0
anon.input.dest <mi_cmd_var_create: unable to create variable object>	
phIn	{...}
id	1
q_state	{...}
common_factor	1.0000000000000002
c_0	{...}
r_part	1.0
i_part	0.0
c_1	{...}
r_part	0.0
i_part	0.0
intCounter	1

Фиг. 3. 6. Вътрешни променливи на фотодетектора, регистрирал фотона

в целевата машина. Фактическото му изпълнение е около секунда. Тъй като точката на прекъсване е разположена след сработването на таймера, тя се достига след изтичането на зададената задръжка от 5 sec (фиг. 3.5).

Избраната точката на прекъсване е в кода на главната функция `taskLightDetector()` на фотодетектора LD . Това позволява да се провери кой от фотодетекторите е регистрирал фотона. Ако началното състояние на фотона е кет-вектора $|0\rangle$, това трябва да бъде LD_0 . Обратно, ако началното състояние на фотона е кет-вектора $|1\rangle$, това трябва да бъде LD_1 . Проверката се извършва чрез вътрешните променливи на фотодетектора (фиг. 3.6).

И двата фотодетектора – процесите LD_0 и LD_1 използват копие на една и съща главна функция - `taskLightDetector()`, така както бе посочено в т. 2.1. Тъй като точката на прекъсване е разположена в тази главна функция, в нея трябва да се спира всеки път, щом единият или другият процес регистрира фотон.

Параметърът, по който тези процеси се различават е `qsToDetect`. Той определя какво трябва да бъде квантовото състояние на фотона, за да се регистрира от съответния процес. При избраната стойност на квантовото състояние на излъчвания фотон `qsToDetect = QS_0`, фотодетекторът регистрира същия фотон – идентификаторът и квантовото му състояние съвпадат с началните (фиг. 3.6).

Забелязва се, че общият множител `common_factor` на двете комплексни величини в квантовото състояние `phIn.q_state` се различава в 16-тия десетичен разряд след запетаята. Това се обяснява с грешката при обработката на числа с плаваща запетая и двойна точност⁷. Ако тази изчислителна грешка се игнорира, регистрираното квантово състояние отговаря на първоначално зададеното с кет-вектора $|0\rangle$.

След точката на прекъсване изпълнението трябва да се възобнови, например чрез клавиша $F8$. При това изпълнението би трябвало да се терминира автоматично, без да се попада втори път в точката на прекъсване. Това е признак за съгласуваното приключване работата на седемте паралелни процеса от модела⁸. В същия момент, това е и част от доказателството за адекватността на модела – само единият фотодетектор трябва да регистрира фотон⁹. И това трябва да бъде фотодетекторът, разположен в първоначалното направление на лъча¹⁰. Резултатите от изпълнението на предложения модел на

7 Съгласно стандарта IEEE 754.

8 Условие за *съгласуваност* - паралелната конструкция не завършва, докато не приключат всички съставлящи я процеси.

9 Условие за *единственост*.

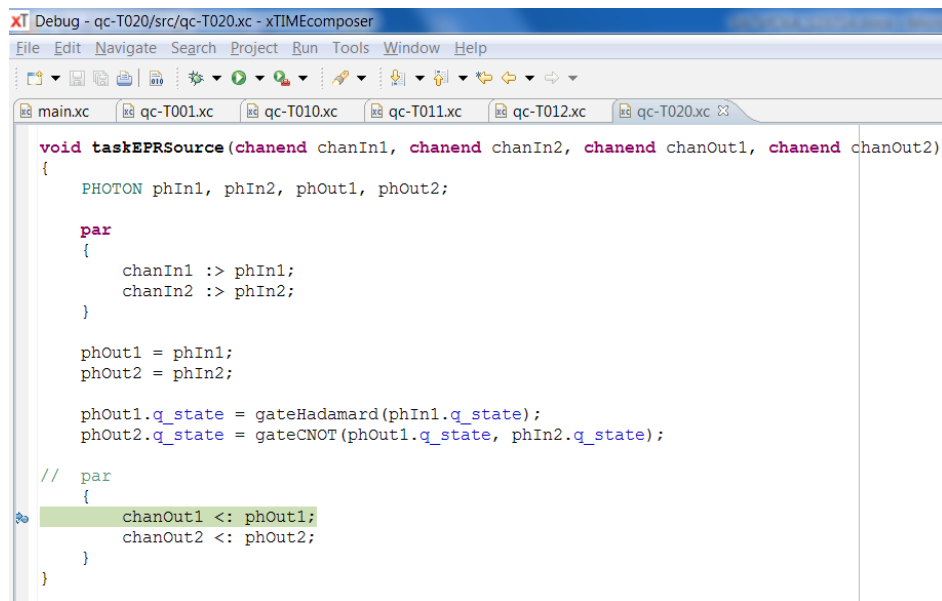
10 Условие за *тъждественост*.

суперпозицията доказват неговата адекватност - и трите критерия за адекватност се изпълняват.

+++

За да се стартира паралелният модел на квантовата телепортация от т. 2.4, се отваря проектът *qc-T020* и главният му изходен модул *qc-T020.xc*. Задава се квантовото състояние, което трябва да се телепортира

```
// КВАНТОВО СЪСТОЯНИЕ НА ФОТОНА,  
// КОЕТО ТРЯБВА ДА СЕ ТЕЛЕПОРТИРА  
#define LE_QS 0  
// 0 - квантово състояние QS_0 с кет-вектор |0>  
// 1 - квантово състояние QS_1 с кет-вектор |1>  
  
#define RNG_TYPE 1  
// 1 - Генератор тип RNG с примитивна функция crc32 на  
изпълнителната среда;  
// 2 - Генератор тип TRNG с кръговите осцилатори от  
изпълнителната среда  
// поддържа се при XS1-L, но не и при XS1-G4.
```



Фиг. 3. 7. Точка на прекъсване на изхода на *EPR* източника

Name	Value
$ 0\rangle$ chanIn1	66562
$ 0\rangle$ chanIn2	66306
$ 0\rangle$ chanOut1	66050
$ 0\rangle$ chanOut2	65794
phIn1	{...}
$ 0\rangle$ id	2
q_state	{...}
$ 0\rangle$ common_factor	1.0
c_0	{...}
$ 0\rangle$ r_part	1.0
$ 0\rangle$ i_part	0.0
c_1	{...}
$ 0\rangle$ r_part	0.0
$ 0\rangle$ i_part	0.0
phIn2	{...}
$ 0\rangle$ id	3
q_state	{...}
$ 0\rangle$ common_factor	1.0
c_0	{...}
$ 0\rangle$ r_part	1.0
$ 0\rangle$ i_part	0.0
c_1	{...}
$ 0\rangle$ r_part	0.0
$ 0\rangle$ i_part	0.0
phOut1	{...}
phOut2	{...}

Фиг. 3. 8. Квантови състояния на входа на *EPR* източника

Проектът се компилира във вариант за настройка (Debug). Поради предварителния характер на модела, ще проследим единствено работата на *EPR* източника.

Процесът *EPR* получава по входните си квантови канали chanIn1 и chanIn2 два фотона phIn1 и phIn2 в базисното състояние $|0\rangle$. Съгласно квантовата верига от фиг. 2.20, на 2-qubit изхода на *EPR* източника би трябвало да се генерира сплетената двойка фотони phIn1 и phIn2.

Затова точката на прекъсване е удачно да се постави на изхода на главната функция taskEPRSource() на процеса *EPR*. Това налага

временното редуциране на паралелното изпращане по изходните канали в последователно.

Моделът се стартира в тестови режим, например чрез клавиша *F11*. При това, изпълнимият код на модела се зарежда и стартира в целевата машина, след което би трябвало да се достигне зададената точка на прекъсване (фиг. 3.7).

От тази точка на прекъсване са достъпни вътрешните променливи на главната функция `taskEPRSource()` на процеса *EPR*.

Първо се проверяват квантовите състояния на входа, т.е. състоянията на входните фотони `phIn1` и `phIn2`. От фиг. 3.8 се вижда, че това са фотони с идентификатори 2 и 3. Квантовите състояния и на двата фотона отговарят на кет-вектора $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, както се изисква

Name	Value
chanIn1	66562
chanIn2	66306
chanOut1	66050
chanOut2	65794
phIn1	{...}
phIn2	{...}
phOut1	{...}
id	2
q_state	{...}
common_factor	0.7071067811865476
c_0	{...}
r_part	1.0
i_part	0.0
c_1	{...}
r_part	1.0
i_part	0.0
phOut2	{...}
id	3
q_state	{...}
common_factor	0.7071067811865476
c_0	{...}
r_part	1.0
i_part	0.0
c_1	{...}
r_part	1.0
i_part	0.0

Фиг. 3. 9. Квантови състояния на изхода на *EPR* източника

от алгоритъма.

На фиг. 3.9 са показани получените изходни квантови състояния, т.е. квантовите състояния на изходните фотони `phOut1` и `phOut2`.

И двете изходни квантови състояния са в суперпозиция от вида $\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$, което обаче не е достатъчно да бъдат сплетени, т.е. да са в едно от състоянията на Бел - $|\Phi^+\rangle$, $|\Psi^+\rangle$, $|\Phi^-\rangle$ или $|\Psi^-\rangle$, разгледани в т. 2.4.

Това е следствие на опита вентилът *CNOT* да се изпълни чрез *1-qubit* вентили, което е принципно възможно, но не и ако двата квантови бита са сплетени. Независимо от предварителния характер на третия модел, той е добра основа за осмисляне на моделираните явления, както и за развитието на предложения в работата подход.