Практическая работа № 3

Отладка многопоточного приложения с использованием окна стеков

На практическом занятии, будет рассмотрен пример отладки параллельного приложения, с использованием окна вызова стеков.

1. Создадим новое консольное приложение - "ParallelStackApplication":

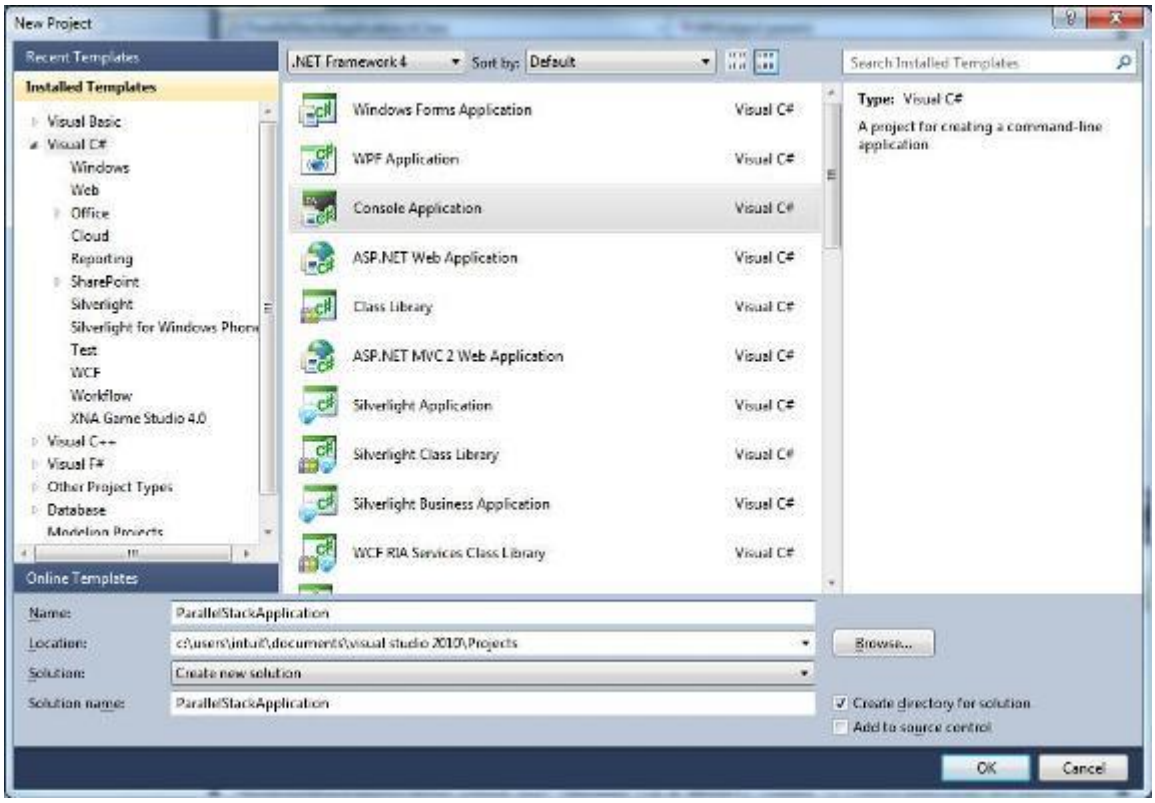

Рис. 1.

2. Напишем код в созданный проект:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading;
using System.Threading.Tasks;
using System.Diagnostics;
namespace ParallelStacksApplication
{
    class XClass
    {
        public void MA(object param) { MB(param); }
        public void MB(object param) { MC(param); }
        object s1 = new object();
        object s2 = new object();
        public void MC(object param)
        {
            if (param == "1")
            {
                MD();
            }
```

```
24.             if (param == "2")
25.             {
26.                 ME();
27.             }
28.             if (param == "3")
29.             {
30.                 MF();
31.             }
32.         }
33.         public void MD()
34.         {
35.             ME();
36.         }
37.         public void ME()
38.         {
39.             while (true) Thread.SpinWait(int.MaxValue / 20);
40.         }
41.         public void MF()
42.         {
43.             ML();
44.         }
45.         public void MG(object param)
46.         {
47.             MH(param);
48.         }
49.         public void MH(object param)
50.         {
51.             MI(param);
52.         }
53.         public void   MI(object  param)
54.         {
55.             if (param == "4")
56.             {
57.                 MJ();
58.             }
59.             else
60.             {
61.                 MK();
62.             }
63.         }
64.         public void MJ()
65.         {
66.             Monitor.Enter(s1); Thread.SpinWait(int.MaxValue / 20); Monitor.Enter(s2);
67.         }
68.         public void MK()
69.         {
70.             Monitor.Enter(s2);
71.             Thread.SpinWait(int.MaxValue / 10);
72.             Monitor.Enter(s1);
73.         }
74.         public void ML()
75.         {
```

```
76.            MM();
77.        }
78.        public void MM()
79.        {
80.            while (true)
81.            { Thread.SpinWait(int.MaxValue / 3);
82.              Debugger.Break();
83.            };
84.        }
85.    }
86.    class Program
87.    {
88.        static XClass obj = null;
89.        static void Main(string[] args)
90.        {
91.            obj = new XClass();
92.            Task.Factory.StartNew(obj.MA, "1");
93.            Task.Factory.StartNew(obj.MA, "2");
94.            Task.Factory.StartNew(obj.MA, "3");
95.            Task.Factory.StartNew(obj.MG, "4");
96.            Task.Factory.StartNew(obj.MG, "5");
97.            Console.ReadLine();
98.        }
99.    }
    }
```

3. Запустите отладку программы с помощью кнопки "Start Debbuging" из меню Visual Studio "Debug" или с помощью клавиши"F5":

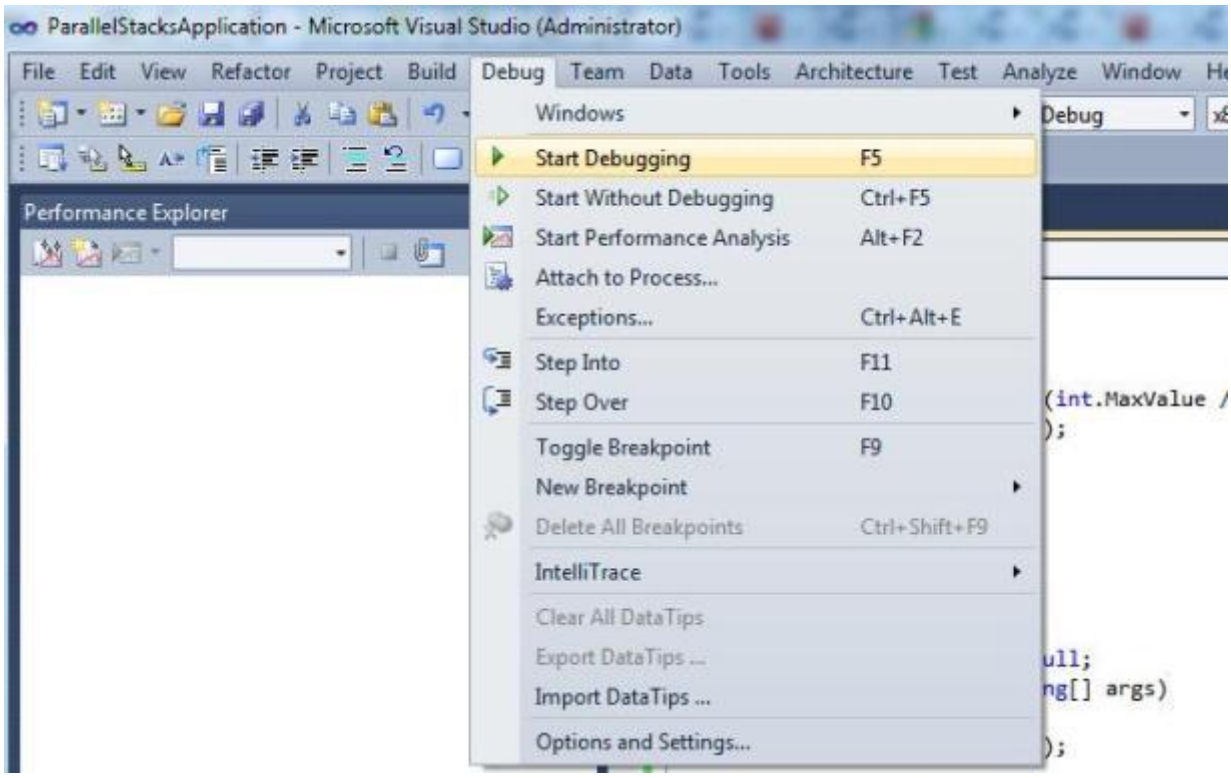

Рис. 2.

4. Через несколько секунд курсор отладчика должен остановиться на методе Debbuger.Break():

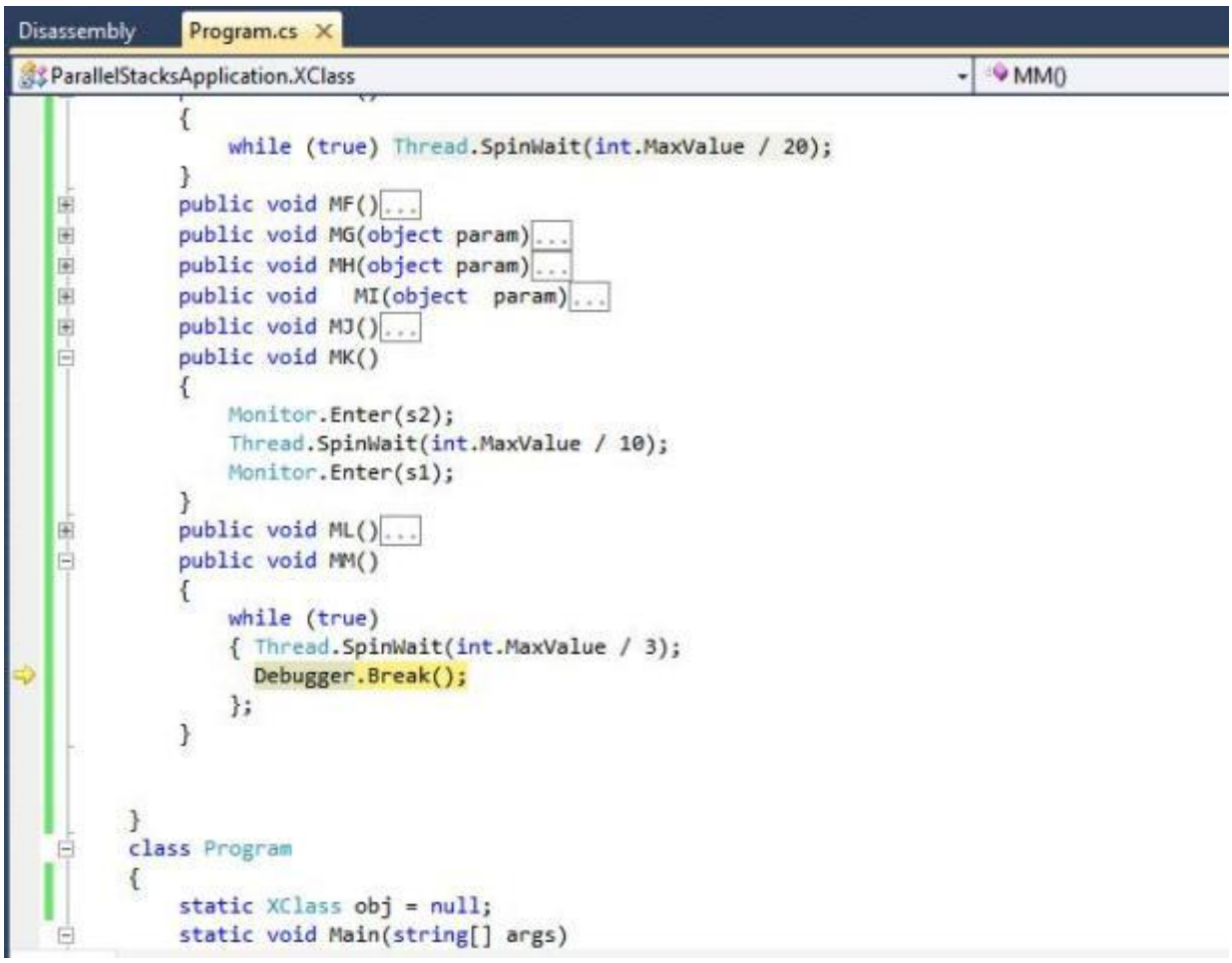

Рис. 3.

5. Запустите окно параллельных стеков (Parallel Stacks) из меню "Debug -> Windows" или с помощью сочетания клавиш"Ctrl+Shift+D,S":

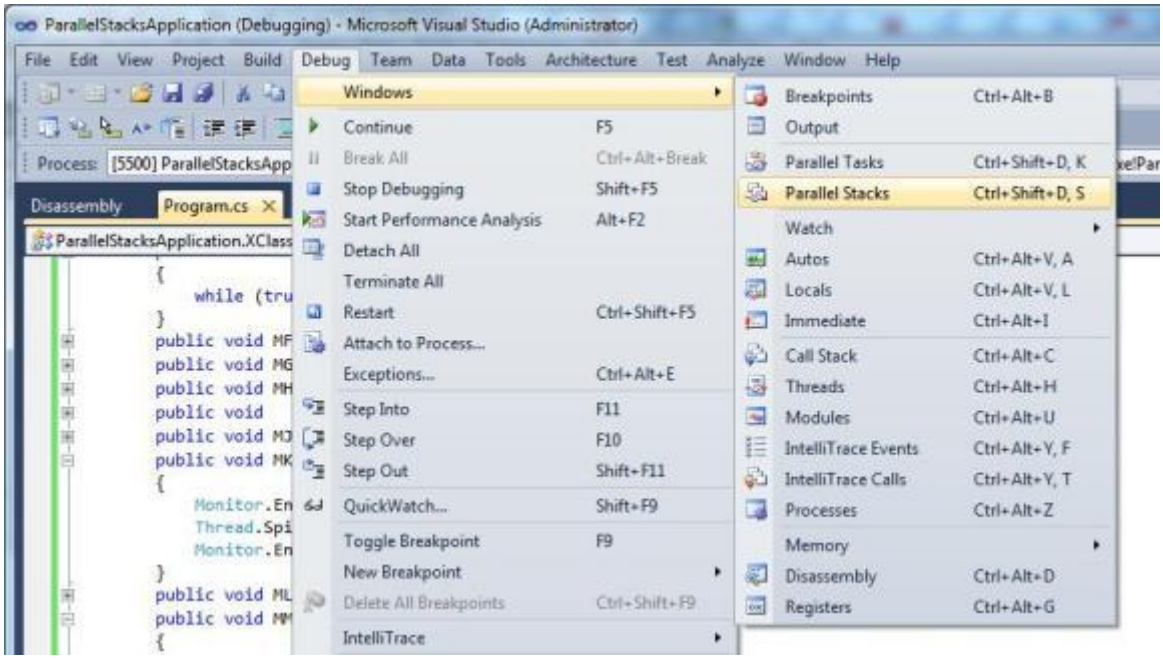

Рис. 4.

6. В результате должно отобразиться "дерево" потоков:

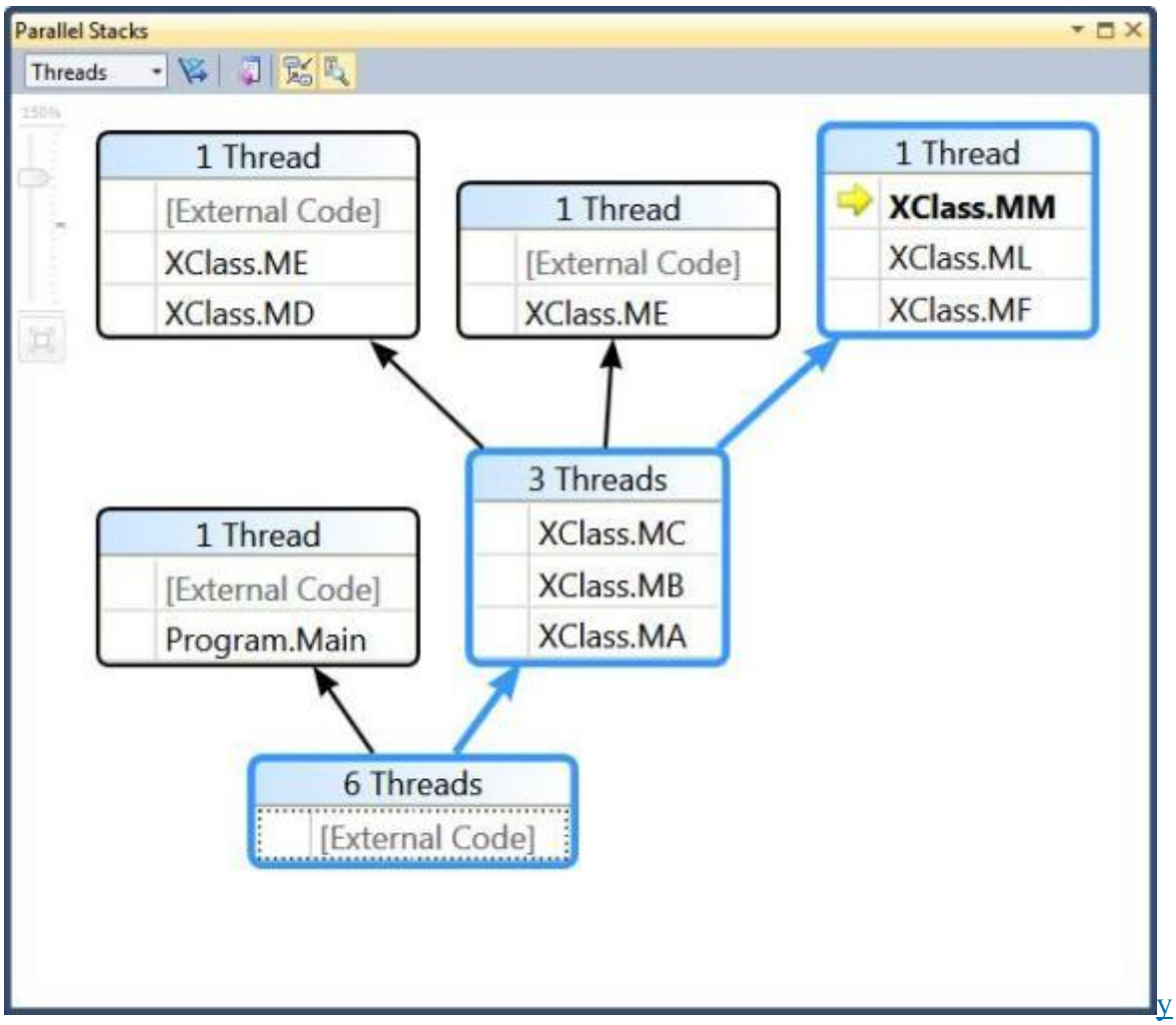

Рис. 5.

7. Переключимся на метод класса XClass - MC(). Для этого кликните правой кнопкой мыши по данному методу и выберите из списка "Switch To Frame" и установите флажок напротив значения 6568:

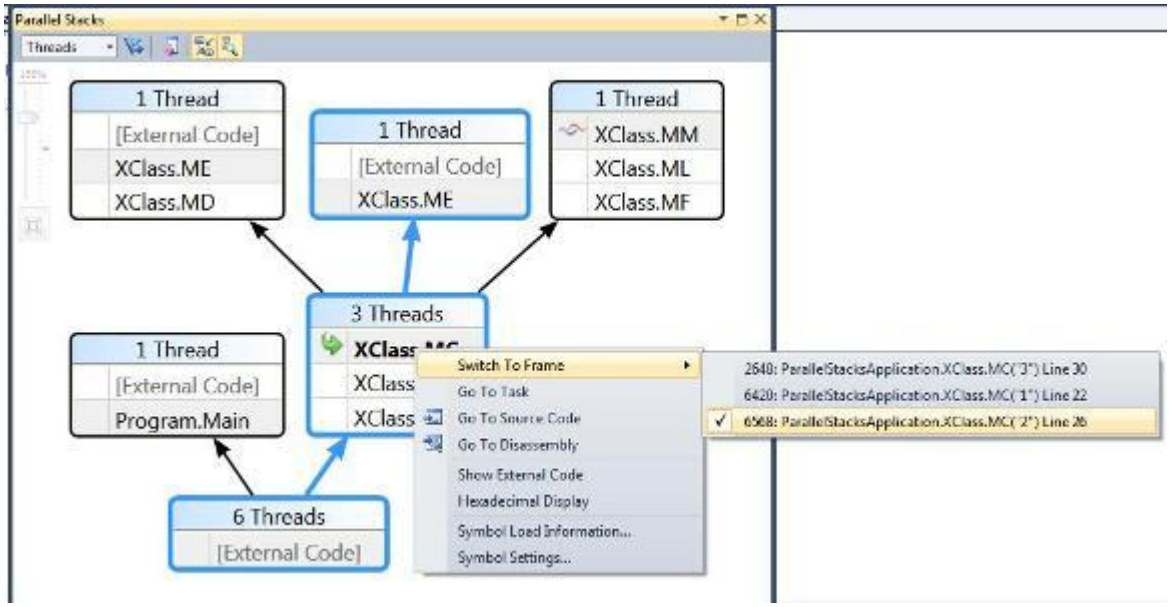

Рис. 6.

8. После чего Visual Studio перейдет на данный фрагмент кода:

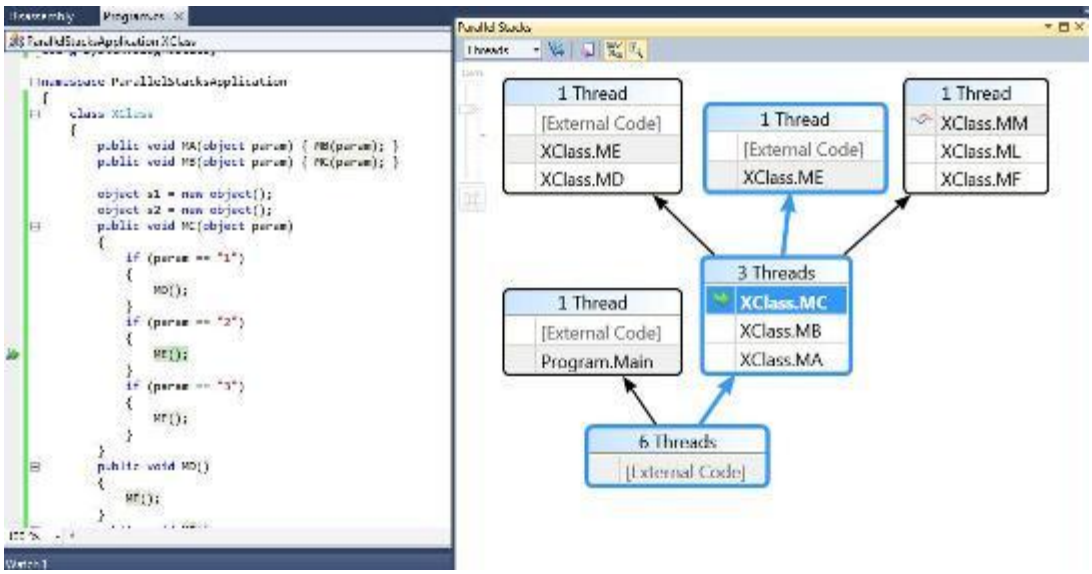

Рис. 7.

9. В примере программы используется метод Debbuger.Break(), который позволяет останавливать отладку в нужной точке. Для того, что бы отладка программы выполнялась по шагам, расставим точки останова (breakpoints) напротив нескольких методов:

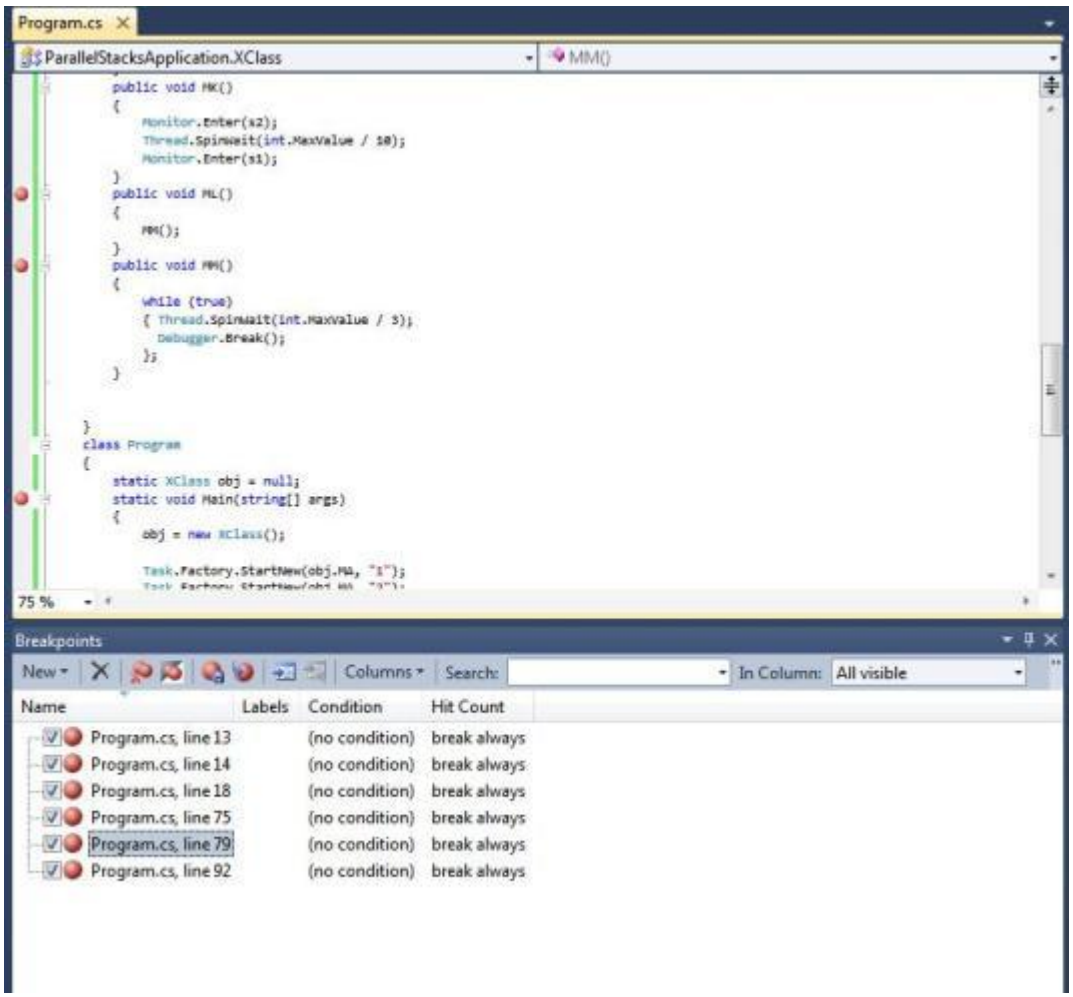

Рис. 8.

10. Запустим отладчик. В результате программа будет осуществлять отладку по шагам и соответствующие изменения (в зависимости от шага) - будут отображаться в окне вызова стеков:

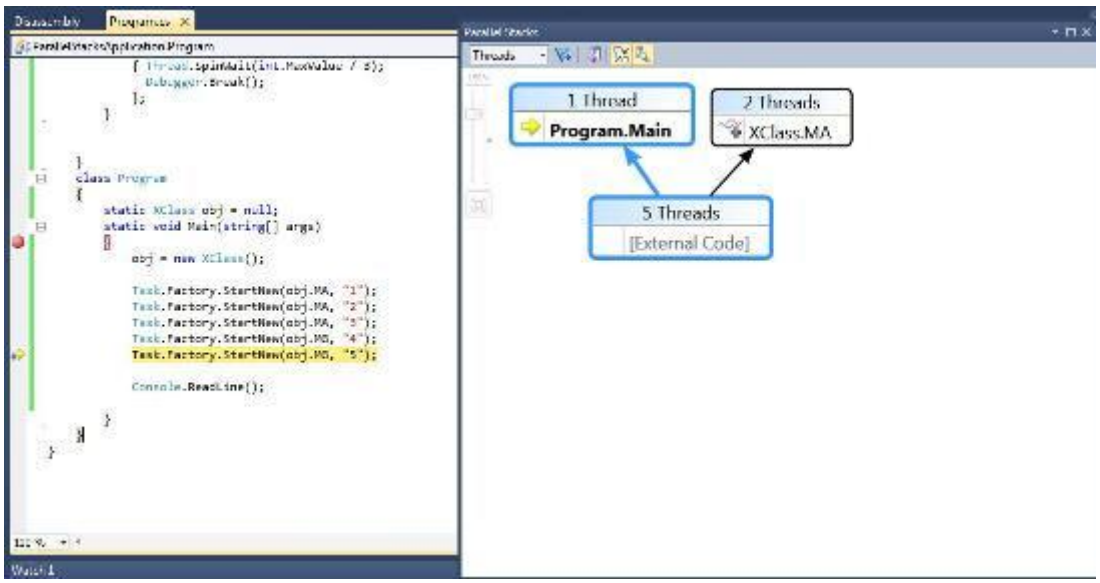

Рис. 9.

11. Для завершения процесса отладки используется пункт из меню "Debug" - "Stop Debugging" или сочетание клавиш"Shift+F5":

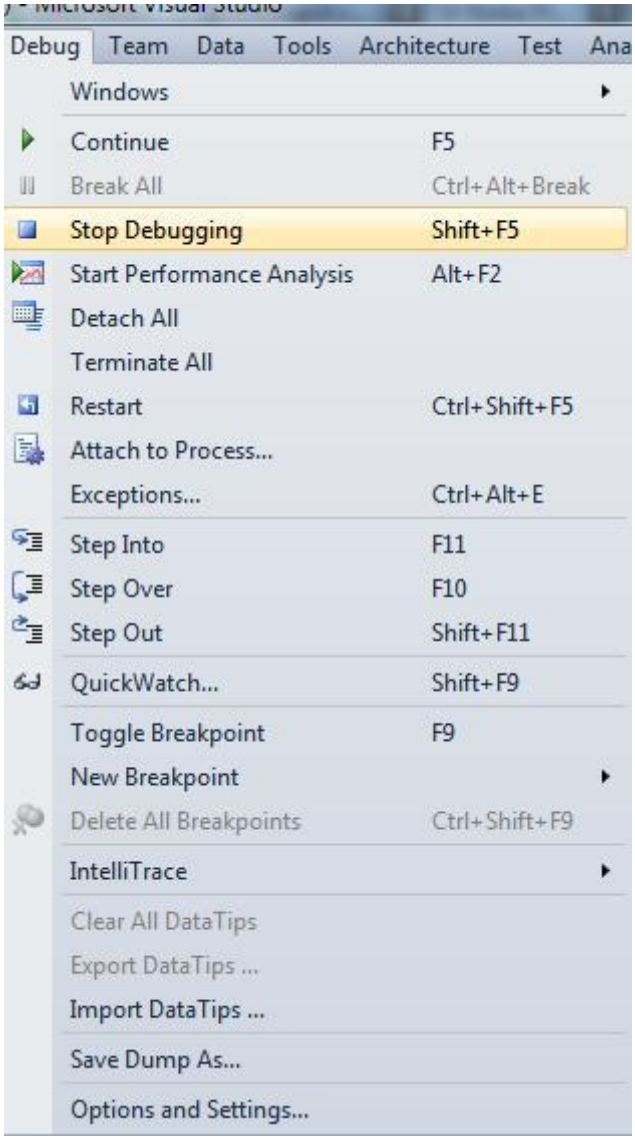

Рис. 10.