

Источник - Сергей Звездин: персональный блог (<http://blog.zvezdin.com/ru/246>)

## Использование компаса в WP7.5 Mango

---

Многие устройства на Windows Phone 7 имеют встроенный компас. В последнем обновлении инструментов разработчика для Windows Phone появился API для работы с компасом. В этой заметке я расскажу о нем.

В качестве базового объекта для работы с компасом используется [сенсор Compass](#). Как можно увидеть, этот объект унаследован от SensorBase – базового объекта для всех сенсоров в Windows Phone 7. Не все устройства имеют компас, поэтому перед началом работы следует проверить, имеет ли данное устройство компас. Общая последовательность действий для работы с компасом такова:

1. Проверка на наличие компаса в устройстве WP7. Если нет, то все остальные действия не выполняются.
2. Создание объекта Compass.
3. Подписка на событие изменения направления компаса.
4. Запуск процесса считывания данных с сенсора.
5. При необходимости можно также обрабатывать ситуации, когда требуется снять намагниченность устройства.

Для проверки доступности компаса в устройстве используется статическое свойство `IsSupported` объекта `Compass`:

```
1 | if (Compass.IsSupported == true)
```

Если компас доступен, то создается объект и запускается прослушивание сенсора. При каждом изменении значения сенсора генерируется событие `CurrentValueChanged`. В обработчике этого события можно получить доступ к объекту `CompassReading`, который содержит текущее направление компаса.

Следующий код выполняет описанные выше действия:

```
01 | private Compass _compass;  
02 |  
03 | public MainPage ()  
04 | {  
05 |     InitializeComponent ();  
06 | }
```

```
07     if (Compass.IsSupported == true)
08     {
09         _compass = new Compass();
10         _compass.CurrentValueChanged += ComapssValueChanged;
11         _compass.Start();
12
13     }
14     else
15         MessageBox.Show("Compass is not supported.");
16
17 }
18
19 private void ComapssValueChanged(object sender, SensorReadingEventArgs<compassreading> e)
20 {
21     Dispatcher.BeginInvoke(() =>
22     {
23         info.Text = e.SensorReading.TrueHeading.ToString();
24     });
25 }</compassreading>
```