```
using Microsoft.SPOT;
using System.Threading;

//Nowendig für Visual Studio 2015
namespace System.Diagnostics
{
    public enum DebuggerBrowsableState
    {
        Never,
        Collapsed,
        RootHidden
    }
}

class Program
{
    private const int PAUSE= 100;
    private BrainPad.Image imageBlau1 = new BrainPad.Image(10, 10);
    private BrainPad.Image imageBlau2 = new BrainPad.Image(10, 10);
    private BrainPad.Image imageBlau3 = new BrainPad.Image(10, 10);
    private BrainPad.Image imageBlau4 = new BrainPad.Image(10, 10);

    public void BrainPadSetup()
    {
        getImage(imageBlau1, BrainPad_Projekt_02.Properties.Resources.BinaryResources.Blau_1);
        getImage(imageBlau2, BrainPad_Projekt_02.Properties.Resources.BinaryResources.Blau_2);
        getImage(imageBlau3, BrainPad_Projekt_02.Properties.Resources.BinaryResources.Blau_3);
        getImage(imageBlau4, BrainPad_Projekt_02.Properties.Resources.BinaryResources.Blau_4);

        //Phase 0
        BrainPad.Display.DrawImage(75, 39, imageBlau1);
        BrainPad.Display.DrawImage(89, 45, imageBlau4);
        BrainPad.Display.DrawImage(95, 59, imageBlau4);
        BrainPad.Display.DrawImage(89, 73, imageBlau4);
        BrainPad.Display.DrawImage(75, 79, imageBlau4);
        BrainPad.Display.DrawImage(61, 73, imageBlau4);
        BrainPad.Display.DrawImage(55, 59, imageBlau3);
        BrainPad.Display.DrawImage(61, 45, imageBlau2);
        Thread.Sleep(PAUSE);
    }

    public void BrainPadLoop()
    {
        //Phase 1
        BrainPad.Display.DrawImage(75, 39, imageBlau2);
        BrainPad.Display.DrawImage(89, 45, imageBlau1);
        BrainPad.Display.DrawImage(55, 59, imageBlau4);
        BrainPad.Display.DrawImage(61, 45, imageBlau3);
        Thread.Sleep(PAUSE);

        //Phase 2
        BrainPad.Display.DrawImage(75, 39, imageBlau3);
```

```
            BrainPad.Display.DrawImage(89, 45, imageBlau2);
            BrainPad.Display.DrawImage(95, 59, imageBlau1);
            BrainPad.Display.DrawImage(61, 45, imageBlau4);
            Thread.Sleep(PAUSE);

            //Phase 3
            BrainPad.Display.DrawImage(75, 39, imageBlau4);
            BrainPad.Display.DrawImage(89, 45, imageBlau3);
            BrainPad.Display.DrawImage(95, 59, imageBlau2);
            BrainPad.Display.DrawImage(89, 73, imageBlau1);
            Thread.Sleep(PAUSE);

            //Phase 4
            BrainPad.Display.DrawImage(89, 45, imageBlau4);
            BrainPad.Display.DrawImage(95, 59, imageBlau3);
            BrainPad.Display.DrawImage(89, 73, imageBlau2);
            BrainPad.Display.DrawImage(75, 79, imageBlau1);
            Thread.Sleep(PAUSE);

            //Phase 5
            BrainPad.Display.DrawImage(95, 59, imageBlau4);
            BrainPad.Display.DrawImage(89, 73, imageBlau3);
            BrainPad.Display.DrawImage(75, 79, imageBlau2);
            BrainPad.Display.DrawImage(61, 73, imageBlau1);
            Thread.Sleep(PAUSE);

            //Phase 6
            BrainPad.Display.DrawImage(89, 73, imageBlau4);
            BrainPad.Display.DrawImage(75, 79, imageBlau3);
            BrainPad.Display.DrawImage(61, 73, imageBlau2);
            BrainPad.Display.DrawImage(55, 59, imageBlau1);
            Thread.Sleep(PAUSE);

            //Phase 7
            BrainPad.Display.DrawImage(75, 79, imageBlau4);
            BrainPad.Display.DrawImage(61, 73, imageBlau3);
            BrainPad.Display.DrawImage(55, 59, imageBlau2);
            BrainPad.Display.DrawImage(61, 45, imageBlau1);
            Thread.Sleep(PAUSE);

            //Phase 0
            BrainPad.Display.DrawImage(75, 39, imageBlau1);
            BrainPad.Display.DrawImage(61, 73, imageBlau4);
            BrainPad.Display.DrawImage(55, 59, imageBlau3);
            BrainPad.Display.DrawImage(61, 45, imageBlau2);
            Thread.Sleep(PAUSE);
        }

        private void getImage(BrainPad.Image image, BrainPad_Projekt_02.Properties.Resources.BinaryResources resource)
        {
            int info;
            byte[] data;
            BrainPad.Color color=new BrainPad.Color();

            data = (byte[])ResourceUtility.GetObject
```

```
          (BrainPad_Projekt_02.Properties.Resources.ResourceManager,
          resource);
        info = data[0xA];
        for (int Y = image.Height - 1; Y >= 0; Y--)
        {
            for (int X = 0; X < image.Width; X++)
            {
                byte h = data[info++];
                byte n = data[info++];
                color.B = (byte)(h & 0x1f);
                color.G = (byte)(((h & 0xe0) >> 5) + ((n & 0x07) << 3));
                color.R = (byte)((n & 0xf8) >> 3);
                image.SetPixel(Y, X, color);
            }
        }
    }
}

```