

```

// Создание виджета
function CreateDimmer (md) {
    // Обращаемся к всплывающему окну который хотим клонировать
    var popup = IR.GetItem("Dimmer" + moduleID);
    // Присваиваем переменной имя канала
    var channelName = deviceName + ":" + "channel" + md.W + moduleID;
    widget = md.subDevice.addWidget(popup.Clone("Dimmer" + md.index +
moduleID));
    // Объявления переменных необходимых для создания виджета
    var dimmer = IR.GetItem("Dimmer" + md.index + moduleID);
    var dimmerLevel = dimmer.GetItem("Level");
    var dimmerButton = dimmer.GetItem("Button");
    // Добавление каналов выбранные пользователем
    device.AddChannel(channelName, [subnetID, deviceID,
HDLCodes.singleChannelLightning, separator, parseInt(md.W, 16), 0x0, 0x0, 0x0,
0x01, 0x0, 0x0]);
    device.AddTag(channelName, [subnetID, deviceID,
HDLCodes.singleChannelLightning, separator, parseInt(md.W, 16), 0x0, 0x0]);

    // Вызов функции привязки слайдера
    var userSlider5 = new UserSlider(dimmerLevel, dimmer.GetItem("Slider"));
    // Отправка команды с уровня на HDL канал диммера
    IR.AddListener (IR.EVENT_ITEM_RELEASE, dimmerLevel, function () {
        device.Set(channelName, this.Value);
    }, dimmerLevel);
    // Кнопка для включения и отключения (0 или 100)
    IR.AddListener (IR.EVENT_ITEM_PRESS, dimmerButton, function() {
        device.Set (channelName, dimmerButton.Value * 100);
    });
}

```