PICAXE
Community

Information
PICAXE Store
Cloud Programming
Support Forum

Welcome, **Iombric**   **Notifications**   **My Profile**   **Settings**   **Log Out**

What's New?	**Forum**	**Blogs**	**PICAXE.com**	**Free Software**	**BASIC Commands**	**Manuals**	**Buy Now**	

New Posts   Private Messages   FAQ   Calendar   Community   Forum Actions   Quick Links   Advanced Search

🏠 Forum   Finished User PICAXE Projects   User Projects - Communication   2.4GHz Radio Sniffer

+ Reply to Thread   Results 1 to 1 of 1

Thread: 2.4GHz Radio Sniffer

Thread Tools   Search Thread   Rate This Thread   Display

21-07-2015, 18:29   #1

BaldwinK ○

New Member

Join Date: Sep 2013
Location: Chertsey
Posts: 17

📄 **2.4GHz Radio Sniffer**

Constructed as a diagnostic for the radio tram project, this self-contained unit borrows ideas from the hacking community.

NRF24L01 radio modules allow for automatic message checking and retransmission for up to six devices. The frequency hopping ISM spectrum splits out radio channels and you do need to be set to the same RF channel to listen in. However the sniffer must not take an active part - only listen - yet it would be a seventh device so how is it done?

The tram radio network uses the default setting of five byte addresses with the default values for the higher order bytes. Only the LSB is loaded to the chip. So if the sniffer is set to a four byte address using the (same) default higher order bytes then it can receive a message.

First we need to turn off shockburst so there will be no acknowledgements. It proved necessary to issue this instruction twice as it is often ignored and the chip tries to join in.

The fifth address byte will now be in the data field and we can match this to a tram or turnout sending or receiving. However, because we are cheating the system the actual payload data will be shifted by one bit. This is easy enough to restore so that the commands and replies can be determined.

The sniffer uses the LCM1602 two line display with nibbles strobed in via an I2C bus. Bit banging proved too slow because the display could not be ready for the next decode so hi2c instructions were used. In turn this meant that the SPI instructions to the radio module had to be done the hard way.

Most of the rest of the code involves setting up special characters to give a mimic diagram of the track turnouts and showing the position and status of each tram. The blinking cursor is used if a command is waiting a reply.

It would appear that some wireless keyboards use this chip and the key strokes can be hacked using this method.

> ✐ Attached Thumbnails
>
>
>
> ✐ Attached Files
>
> 📄 Radio Sniffer.bas (10.2 KB, 32 views)

Blog this Post | **Reply** | **Reply With Quote** |

+ Reply to Thread

Quick Navigation | **User Projects - Communication** | **Top**

« Previous Thread | Next Thread »

Tags for this Thread

None
View Tag Cloud

Add / Edit Tags

Bookmarks

🔲 Digg

■ del.icio.us

StumbleUpon

Posting Permissions   ⌃

You may post new threads   **BB code** is On
You may post replies   **Smilies** are On
You may post attachments   **[IMG]** code is On
You may edit your posts   **[VIDEO]** code is On

StumbleUpon

G Google

You may edit your posts

[VIDEO] code is Off
HTML code is Off

Forum Rules

-- Picaxe Theme ▼   -- English (US) ▼   Contact Us Main PICAXE Website Archive Top

All times are GMT +1. The time now is 07:07.

Powered by vBulletin® Version 4.2.2
Copyright © 2016 vBulletin Solutions, Inc. All rights reserved.

(c) Revolution Education Ltd 2007-15