Project “The Interceptor”:

Owning anti-drone systems with nanodrones

David Meléndez Cano
R&D Embedded Systems Engineer

@taiksoncfsxas

Taiksonprojects.blogspot.com
David Meléndez Cano

@TaiksonTexas

- R&D Embedded Software Engineer in Albalá Ingenieros, S.A. Spain
- Author of the robots: "Atropos" & "Texas Ranger"
- Author of the Book "Hacking con Drones"
- "Reincident" speaker
- Trainiac
David Meléndez Cano

@TaiksonTexas

- R&D Embedded Software Engineer in Albalá Ingenieros, S.A. Spain
- Author of the robots: "Atropos" & "Texas Ranger"
- Author of the Book  "Hacking con Drones"
- "Reincident" speaker
- Trainiac
David Meléndez Cano

@TaikosTexas

- R&D Embedded Software Engineer in Albalá Ingenieros, S.A. Spain
- Author of the robots: "Atropos" & "Texas Ranger"
- Author of the Book  "Hacking con Drones"
- "Reincident" speaker
- Trainiac
David Meléndez Cano

@TaiksonTexas

- R&D Embedded Software Engineer in Albalá Ingenieros, S.A. Spain
- Author of the robots: "Atropos" & "Texas Ranger"
- Author of the Book "Hacking con Drones"
- "Reincident" speaker
- Trainiac
A US ally shot down a $200 drone with a $3 million Patriot missile

This will be a bigger problem as more drones show up on the battlefield

by Andrew Liptak | @AndrewLiptak | Mar 16, 2017, 10:13am EDT
Previously in DEFCON...
Previously in DEFCON...

Deceiving Jammers

Defeating Jammers

Hacking Peripherals - Cellular 3G USB & GPS - Secure Command & Control

- Remote control over SSH tunnel via 3G USB cell connection. GPS & Cellular signals are illegal to jam (see FCC regulations), making it hard to defend against this type of drone.

Visit [https://transition.fcc.gov/eb/jammerenforcement/jamfaq.pdf](https://transition.fcc.gov/eb/jammerenforcement/jamfaq.pdf)

*Note: be sure to check upcoming FCC regulations about needing to keep drone within line of sight while flying.*

@taiksonetexas
Drones as a threat

- Flying computers. (*IoT over your head.*)
- Custom payloads:
  - Sniffers
  - Jammers
  - Network Analyzers
  - 3d mapping, cameras.
  - Physical attacks, explosives.
  - ...
Detection

- Thermal and standard cameras
  - A.I. to detect drone shape
  - Electronics and motor heat detection
- Characterization of drone noise
- Detected Radio Frequency and waveform
  - Radio signature
“Voluntary” measures

- No-fly zone controlled by onboard GPS and Autopilots

- Real time telemetry transmission to COPS

- Give to COPS the ability to take down your drone and “everything will be alright”
Counter-Countermeasures

- Spread-spectrum
- Frequency hopping
- Use unexpected frequencies by the jammer
- Robust protocols
First Round: “ATROPOS”

Dron ATROPOS

- WiFi Router
- PIC16F876 for PWM
- Wii Nunckuck and Motion + as onboard IMU
- HTML5 telemetry by router webserver
- WiFi comm.
- WPS Attacks with bully
First Round: “ATROPOS”

Dron ATROPOS

- WiFi
- PIC16F876 for PWM
- Wii Nunchuck and Motion + as onboard IMU
- HTML5 telemetry by router
  - HTML5 telemeter
  - WiFi communication
  - WPS Attacks with bully
First Round: “ATROPOS”

Dron ATROPOS

- WiFi
- PIC16F876 for PWM
- Wii Nunchuck and Motion + as onboard IMU
- HTML5 telemetry by router webserver
- WiFi comm.
- WPS Attacks with bully
First Round: “ATROPOS”

Dron ATROPOS

- WiFi
- PIC16F876 for PWM
- Wii Nunchuck and Motion + as onboard IMU
- HTML5 telemetry by router webserver
- WiFi comm.
- WPS Attacks with bully

@taiksonjexas
Now, what else?

“We count thirty Rebel ships, Lord Vader...

...but they're so small they're evading our turbolasers”
Project “The Interceptor”
Project “The Interceptor”

- Minimum size and weight (harder to detect)
- Low budget (no, seriously, really low)
  ~$40 + $20 with SDR
- Hacking capabilities
- “Resilient” control
Project “The Interceptor”
Vocore2
## Vocore2

### Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SIZE</strong></td>
<td>25.6mm x 25.6mm x 3.0mm</td>
</tr>
<tr>
<td><strong>CPU</strong></td>
<td>MT7628AN, 580 MHz, MIPS 24K</td>
</tr>
<tr>
<td><strong>MEMORY</strong></td>
<td>128MB, DDR2, 166MHz</td>
</tr>
<tr>
<td><strong>STORAGE</strong></td>
<td>16M NOR on board, support SDXC up to 2TB</td>
</tr>
<tr>
<td><strong>WIRELESS</strong></td>
<td>802.11n, 2T2R, speed up to 300Mbps.</td>
</tr>
<tr>
<td><strong>ANTENNA</strong></td>
<td>One U.FL slot, one on board antenna.</td>
</tr>
<tr>
<td><strong>ETHERNET</strong></td>
<td>1 port/5 ports, up to 100Mbps.</td>
</tr>
<tr>
<td><strong>USB</strong></td>
<td>Support USB 2.0, up to 480MBit/s.</td>
</tr>
<tr>
<td><strong>PCIe 1.1</strong></td>
<td>Supported</td>
</tr>
<tr>
<td><strong>GPIO</strong></td>
<td>&gt;=40 (pinmux)</td>
</tr>
<tr>
<td><strong>UART</strong></td>
<td>x3 (UART2 for debug console)</td>
</tr>
<tr>
<td><strong>PWM</strong></td>
<td>x4</td>
</tr>
<tr>
<td><strong>POWER SUPPLY</strong></td>
<td>3.6V ~ 6.0V, 500mA</td>
</tr>
<tr>
<td><strong>POWER CONSUMPTION</strong></td>
<td>74mA wifi standby, 230mA wifi full speed, 5V input.</td>
</tr>
</tbody>
</table>
Vocore2: PWM

• We need to generate x4 PWM signals to control the motors
  – Hard real time constrained. Need specific HW.

• x4 channels available, but only 2 enabled

• Last two overlap with UART2 function
  – Disable UART2 in devicetree
  – Enable PWMx4 in devicetree
Vocore2: PWM in the forum

Re: Vocore2 pwm sysfs

Emm, this is a hard way:
First, download the openwrt source from vocore.io/v2.

In `sysfs`, this is a hard way:
Vocore2: PWM in the forum

First, download the VoCore2 firmware from vocore.io/v2.

Second, find VoCore2.dts in source, try to understand the pinctrl section. You have to understand how pinctrl work, better check its source. (^_^)

Third, enable pwm driver in menuconfig.

In summary, this is a hard way:
Vocore2: PWM in the forum

and you will be a good linux hacker. 😊
Vocore2: PWM (pinmux)

```c
uart2_pins: uart2 {
    uart2 {
        ralink,group = "uart2";
        ralink,function = "uart2";
    }
};

sdxc_pins: sdxc {
    sdxc {
        ralink,group = "sdmode";
        ralink,function = "sdxc";
    }
};

pwm0_pins: pwm0 {
    pwm0 {
        ralink,group = "pwm0";
        ralink,function = "pwm0";
    }
};

pwm1_pins: pwm1 {
    pwm1 {
        ralink,group = "pwm1";
        ralink,function = "pwm1";
    }
};
```

./target/linux/ramips/dts/mt7628an.dtsi

@taiksonetexas
Vocore2: PWM (pinmux)

```c
8pinctrl1 {
    pwm_pins: pwm{
      ralink,group = "pwm0";
      ralink,function = "pwm0";
    };
    pwm1{
      ralink,group = "pwm1";
      ralink,function = "pwm1";
    };
    uart2_pwm{
      ralink,group = "uart2";
      ralink,function = "pwm";
    };
};

&i2c{
  status = "okay";
  clock-frequency = <4000000>;
  adc@0 {
    compatible = "microchip,mcp3426";
    reg = <0x68>;
  };
};

&uart2 {
  status = "disabled";
  //status = "okay";
};

&pwm {
  pinctrl-0 = <8pwm_pins>;
  status="okay";
  //status="disabled";
};
```
Vocore2: pinmux mt7628 (datasheet)

### 3.3.18 UART2 pin share scheme

Controlled by the EPHY_APGIO_AIO_EN[4:1] and UART2_MODE registers

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>4'b0000</th>
<th>4'b1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDI_TP_P2</td>
<td>MDI_TP_P2</td>
<td>UART_TXD2</td>
</tr>
<tr>
<td>MDI_TN_P2</td>
<td>MDI_TN_P2</td>
<td>UART_RXD2</td>
</tr>
</tbody>
</table>

### 3.3.19 PWM_CH0 pin share scheme

Controlled by the EPHY_APGIO_AIO_EN[4:1] and PWM0_MODE registers

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>4'b0000</th>
<th>4'b1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDI_RP_P2</td>
<td>MDI_RP_P2</td>
<td>PWM_CH0</td>
</tr>
</tbody>
</table>

### 3.3.20 PWM_CH1 pin share scheme

Controlled by the EPHY_APGIO_AIO_EN[4:1] and PWM1_MODE registers

| 4'b0000 | 4'b1111 |
Power stage

- Brushed motors (cheap as hell)
- X1 MOSFET
- X1 Capacitor
- X1 Schottky diode
Electrical motor behaviour
PID tuning
Interceptor WiFi architecture

**PILOT SIDE**
- **JOYSTICK**
- **WEB interface**
- **AES encryption**
  - 802.11 beacon injection
  - Change channel management
  - **802.11 beacon sniffing**

**AIRCRAFT SIDE**

1. 802.11 beacon sniffing
2. **AES decryption**
3. Flight control And change channel management
4. **AES encryption**
5. 802.11 beacon injection

**Control**
- from JOYSTICK to AIRCRAFT SIDE

**Telemetry**
- from WEB interface to AIRCRAFT SIDE
# Interceptor WiFi architecture

**Forged Beacon Frame injection (PILOT SIDE)**

<table>
<thead>
<tr>
<th>HEADER</th>
<th>BEACON FRAME PAYLOAD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>AES-128</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Header</th>
<th>INITIALIZATION VECTOR</th>
<th>COMMAND</th>
<th>SEQUENCE NUMBER</th>
<th>SHA256</th>
</tr>
</thead>
<tbody>
<tr>
<td>AP “INTERCEPTOR”</td>
<td>Preamble Gas Pitch Roll Yaw</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Interceptor WiFi architecture

PILOT

CH:13

My channel: 13; Target channel: 12

CH:12

My channel: 12

CH:11

My channel: 11; Target channel: 10

My channel: 11

CH:11

My channel: 11

CH:12

My channel: 12

CH:13

My channel: 13; Target channel: 12

DRONE

Etc...
Fallback FM based TX

• For a WiFi complete jamming scenario
• Transmit with an arbitrary frequency
• Demo in FM band
• Transmitting in illegal frequencies are the least problem for bad guys
• Rpi radio transmission causes harmonics. Really a problem?
Fallback FM based TX

http://asliceofraspberrypi.blogspot.com/2014/10/generating-radio-frequencies-using.html

@taiksonthexas
Fallback FM based TX

- JOYSTICK
- 4 bytes packet
- FSK modulation
- RF format conversion
- FM modulated transmission
- GPIO
- Connected to antenna When timeout
- 4 byte For flight control
- FSK demodulation
- Wav audio conversion
- SDR FM capture

@taiksonthexas
IT'S NO GOOD. I CAN'T MANEUVER!

STAY ON TOPIC!
Thank you!

Acknowledges:
José Manuel Hernández
Jesús Fernández
Javier Hernández
Vicente Polo
Daniel Iglesias
Adrian Aznar

David Meléndez Cano
R&D Embedded Systems Developer

@taikson texas

Taiksonprojects.blogspot.com