Appendix: 802.11/DSRC Detect-and-Vacate Demonstration User Guide

1 Introduction

This document offers a description of the 802.11/DSRC Detect-and-Vacate Demonsration. This demonstration is built on WARP v3, an FPGA-based programmable wireless research platform developed by Mango Communications [1], and leverages two of Mango's reference designs for WARP v3, the 802.11 Reference Design and DSRC Preamble Detector Reference Design. An overview of the demonstration is provided below. Details about the underlying reference designs are provided in Sections 2, 3, and 4. Detailed setup and usage information is provided in Section 5.

1.1 Demonstration Overview

A high-level view of the demonstration is shown in Figure 1. The demonstration consists of two logical nodes which prototype the behavior of a DSRC-aware 802.11 device. Each logical node consists of two WARP v3 boards. At each node one WARP v3 board implements an 802.11-Ethernet bridge, connecting the board's Ethernet interface to the wireless medium via a 802.11 MAC/PHY. The other WARP v3 board implements the real-time DSRC preamble detector. The detector board asserts an output signal whenever it detects a DSRC preamble. This "DSRC Detected" signal is connected to a digital input at the 802.11 board. The 802.11 MAC monitors this input and reacts whenever a DSRC detection event occurs. By default the 802.11 node reacts to a detection event by re-tuning its RF interface to a pre-defined, non-DSRC channel.

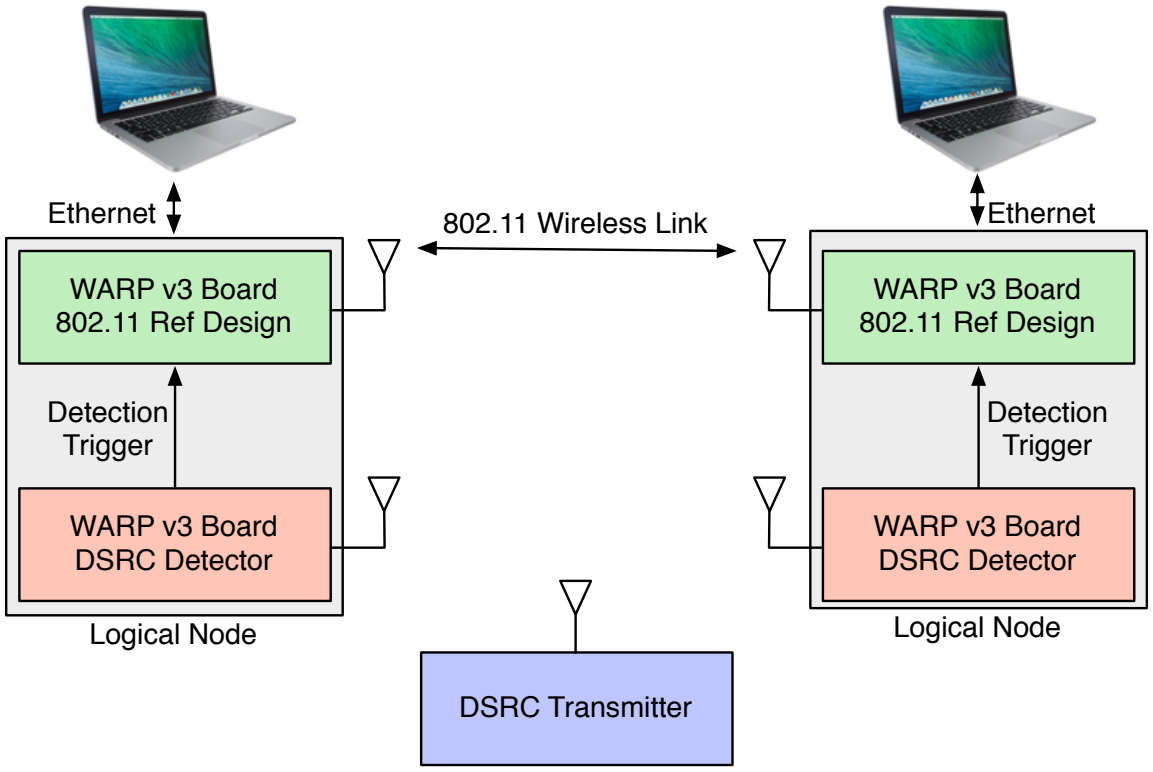


Figure 1: Overall demonstration architecture

The demonstration nodes are configured to use channel 173 (20 MHz centered at 5865 MHz) as the primary and channel 48 (20 MHz centered at 5240 MHz) as the backup. On boot or reset the 802.11 nodes will communicate on channel 173. After a DSRC detection event the 802.11 radio will be re-tuned to channel 48.

The DSRC Transmitter depicted in Figure 1 can be a commercial DSRC device or an RF vector signal generator capable of generating DSRC waveforms. A fifth WARP v3 board may also be used

as a DSRC preamble transmitter. See Section 4 for details on configuring a WARP v3 node as a DSRC preamble transmitter.

2 802.11 Reference Design

The Mango Communications 802.11 Reference Design is an FPGA implementation of the 802.11a/g/n MAC and PHY. The MAC and PHY are implemented in the FPGA of the Mango WARP v3 hardware and operate in real time. An overview of the PHY and MAC implementations is provided below in Section 2.1 and Section 2.2 respectiely. The full 802.11 Reference Design user guide is available online [2]. Extensions to the standard reference design to enable the Detect-and-Vacate behavior are described in Section 2.3. Characterization of the Detect-and-Vacate behavior is provided in Section 2.4.

2.1 PHY Overview

The 802.11 Reference Design PHY implements the IEEE 802.11-2012 Clause 18 ("802.11a/g" or "non-HT") and parts of the Clause 20 ("802.11n" or "HT") physical layers. The PHY cores support switching between non-HT and HT waveforms per packet. For transmissions the MAC selects the waveform mode per packet and configures the PHY Tx accordingly. The PHY Rx core automatically detects between non-HT/HT based on the preamble of an incoming waveform and configures its demodulation/decode pipeline accordingly. The PHY implementation interoperates with other WARP v3 nodes and with commercial 802.11 devices.

Non-HT PHY Specs:

- **Waveform**: OFDM, 64 subcarriers (52 occupied) in 20 MHz
- **Data rates**: All 8 data rates: 6, 9, 12, 18, 24, 36, 48, 54 Mbps
- **Sensitivity**: Exceeds requirements of IEEE 802.11-2012 18.3.10.2

HT PHY Specs:

- **Waveform**: OFDM, 64 subcarriers (56 occupied) in 20 MHz
- **Data rates**: MCS indexes 0 (6.5 Mbps) to 7 (65 Mbps)
- **Sensitivity**: Exceeds requirements of IEEE 802.11-2012 20.3.21.1 for supported MCS

2.2 MAC Overview

The 802.11 Reference Design MAC is implemented primarily in C code. The code runs in two Xilinx MicroBlaze CPUs implemented in the fabric of the WARP v3 FPGA. The two CPUs, labeled "CPU High" and "CPU Low", implement the high-level MAC application and the low-level DCF medium access protocol, respectively. The high-level MAC is responsible for managing association state, queueing packets for eventual transmission and interfacing to external traffic sources. The low-level MAC is responsible for managing access to the wireless medium, including virtual and physical carrier sensing, backoffs, re-transmissions, etc.

MAC Specs:

- **Applications**: AP, client (STA) and ad-hoc (IBSS) roles[1]
- **Traffic Sources**: Ethernet bridge or local arbitrary traffic generator

[1]The 802.11 Bridge described by this document will consist of 1 AP node and 1 STA node. The IBSS design is not applicable to the 802.11/DSRC Detect-and-Vacate Demonstration.

- **Management Packet Types**: Probe request/response, authentication, association request/response, beacon
- **Data Packet Types**: Data and QoS Data
- **Control Packet Types**: ACK, RTS, CTS

The low-level MAC code has full control of the Tx PHY configuration per packet. The MAC code selects the Tx waveform (non-HT or HT), data rate and Tx power for every packet it submits to the PHY. The MAC code is also responsible for selecting the current center frequency of the RF interface.

Traffic for the wireless MAC/PHY is supplied by the high-level MAC application. The high-level MAC implements either an Ethernet bridge, using the standard Ethernet 802.11 encapsulation scheme (the "Integration Function" described in IEEE 802.11-2012 Annex P), or a local traffic source/sink. The utilization level of the wireless link is implicitly controlled by the offered traffic load and selected Tx PHY params (waveform, data rate).

The high-level MAC maintains detailed statistics of all Tx/Rx events for all its partner nodes. These statistics can be used to calculate precise throughput and PER metrics.

2.3 Detect-and-Vacate Overview

For the purposes of this demonstration, the high-level MAC code is modified in two ways. First, the AP role has been modified to advertise a "future channel guidance" in transmitted beacon frames and association responses. Second, the STA role has been modified to parse this tag in received association responses and update its internal state accordingly. A STA will honor the channel specified in this new management tag as its backup channel during a DSRC Detect-and-Vacate event.

The low-level MAC code is modified to monitor the state of two additional inputs. The first, an external pin on the hardware, is used for "Physical Detection" using a separate DSRC detector board. The details of this functionality are described in Section 2.3.1.

The second input is an internal signal used for "Virtual Detection" via 802.11 protocol heuristics. This functionality is described in Section 2.3.2 and Section 2.3.3.

Whenever the low-level MAC code observes a detection event on either the physical or virtual inputs, the code immediately re-tunes the 802.11 node's RF interface to a new center frequency outside the DSRC bands. It also reports this event to the upper-level MAC for further processing. The lower-level MAC monitors the DSRC detection input at all times except when actively transmitting or receiving. This permits the 802.11 node to react to DSRC detection events as quickly as possible, even between re-transmissions of a single data packet.

2.3.1 Physical Detection

This input is connected to the "DSRC Detected" output of the separate DSRC detector board. When the MAC is notified of a DSRC detection event it executes extra code to react to the detection event.

Figure 2 shows an example timeline of events when a physical detection event occurs. The detector, described in Section 3, takes approximately 10 μsec after the start of a DSRC preamble to assert a detection output. Because this signal is frequently polled by the low-level MAC, the 802.11 design immediately reacts to the signal's presence by suspending any 802.11 transmission events and vacating the DSRC channel. Furthermore, the 802.11 design begins to re-tune its radio to a backup non-DSRC channel. The tuning process for the Mango WARP v3 board takes approximately 100 μsec, after which the 802.11 design is able to resume communication on the backup channel.

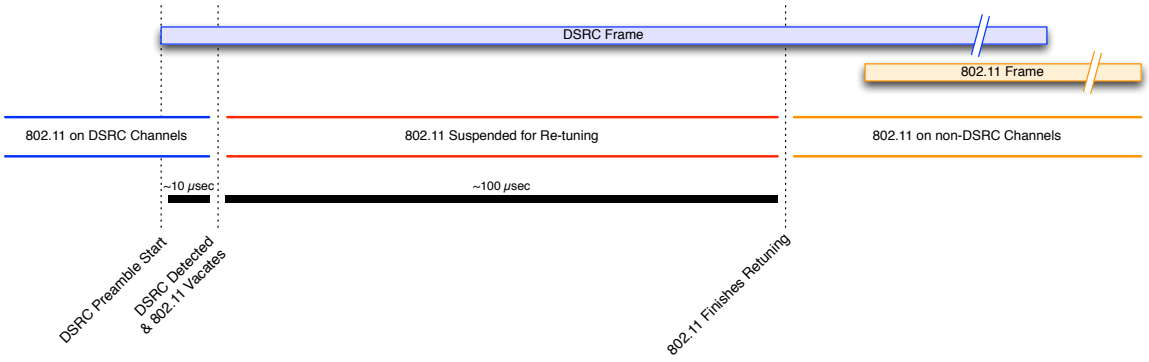


Figure 2: Timeline of Physical Detection

2.3.2 Virtual Detection: Inactivity

In addition to physical detection using a separate DSRC detector board, the 802.11 design can internally trigger a "virtual" detection if it has reason to believe that its peers in the 802.11 network have already tuned to a backup channel due to their own physical detection mechanisms.

The first such virtual detection heuristic is that of receive inactivity. If an 802.11 device has not received any frame from another 802.11 device in some amount of time, it may conclude that the other 802.11 device is no longer on the primary DSRC channel. Implicit in this strategy is that a device has some expectation that a frame should be arriving at some point in the future. This strategy is well-suited for a client STA role since, at a minimum, it expects to received periodic beacons from its associated AP.

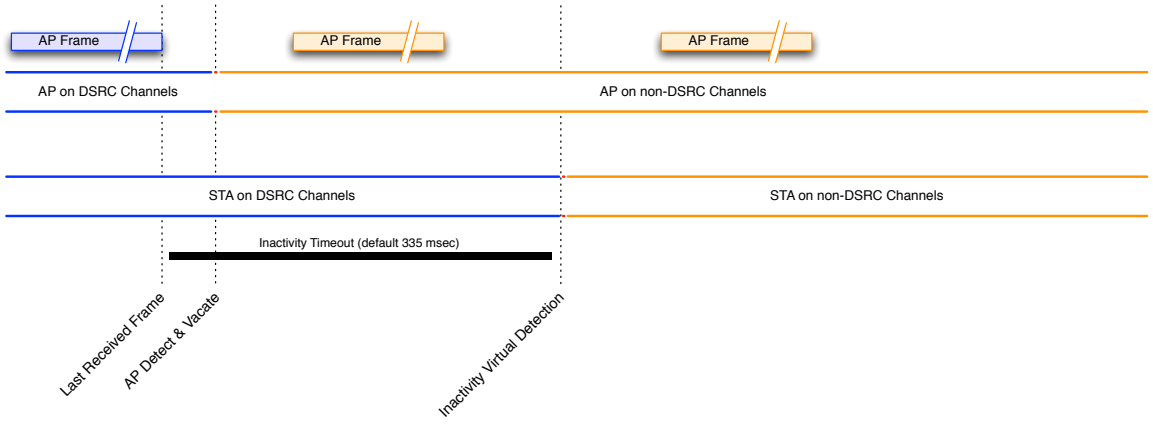


Figure 3: Timeline of Inactivity Virtual Detection

Figure 3 shows an example timeline of events for the virtual detection event based on inactivity. At some point in time, the AP performs a Detect-and-Vacate operation and leaves the DSRC channel. Whatever frame was received by the STA prior to this Detect-and-Vacate event started a timer at the STA. This timer is reset and restarted whenever the STA receives a frame from the AP. Once the AP has vacated to the backup channel, the STA stops receiving AP transmissions and the STA's timer continues to run. When the timer reaches a programmable threshold (defaults to 335 msec), the STA infers a DSRC detection and vacates to the backup non-DSRC channel to continue to communicate with the AP.

2.3.3 Virtual Detection: Transmission Failure

The second virtual detection heuristic is that of transmission failures. If an 802.11 device is reliably communicating with another 802.11 device (which is to say that unicast transmissions are being acknowledged their receiver), a virtual DSRC detection event may be asserted if the communication link is no longer reliable. If the partner device has vacated to the backup channel, it will no longer be able to send any ACKs.

The key metric for this strategy is the idea of sequential transmission failures. A vacated receiver will never be able to send an ACK on the primary DSRC channels. This is fundamentally different than other sources of transmission failures like collisions of deep channel fades since the effects of these kinds of failures are inherently probabilistic.

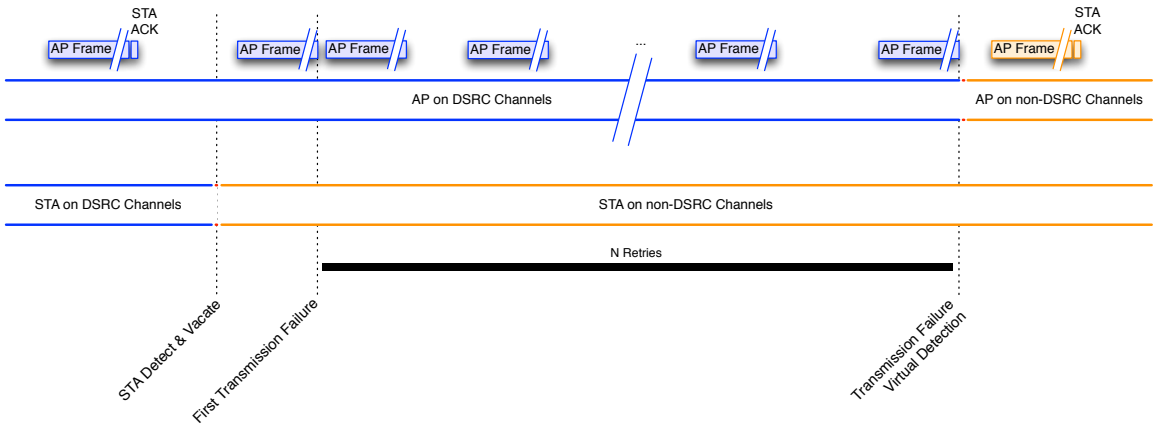


Figure 4: Timeline of Transmission Failure Virtual Detection

Figure 4 shows an example timeline of events for the virtual detection event based on transmission failures. In this scenario, an AP is sending data frames to a STA client and the STA is replying to each frame with an ACK transmission. After a Detect-and-Vacate event at the STA, it is no longer able to send ACKs to the AP. The AP naturally retries a failed transmission via the medium access behaviors of the DCF. Upon reaching a maximum number of tolerable unsuccessful retries, the AP asserts a virtual DSRC detection events and vacates to the backup non-DSRC channel. From this point forward, it is able to resume communication with the STA.

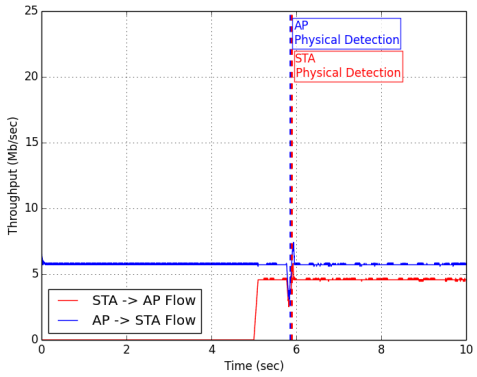
2.4 Detect-and-Vacate Characterization

On key feature of the Detect-and-Vacate feature of the 802.11 Reference Design is the ability to maintain an upper-level connection via the 802.11 wireless link, even across a re-tuning triggered by a DSRC detection event. This minimizes the impact on any traffic flow using the 802.11 link.

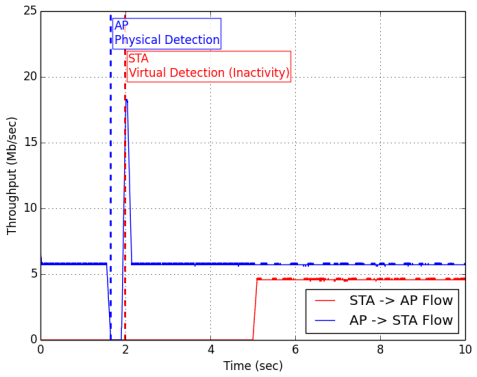
Figure 5 shows a characterization of the design undergoing different combinations of physical and virtual DSRC detection events. Data from these experiments were gathered using the 802.11 Reference Design's event log [3]. In each case, the trial consists of the following:

- **AP→STA Downlink Traffic:** ≈6 Mbps Constant Bit Rate (CBR) for 10 seconds
- **STA→AP Uplink Traffic:** No traffic for 5 seconds, followed by 5 seconds of ≈4.5 Mbps Constant Bit Rate (CBR) traffic
- All data MPDUs sent with a fixed payload size of 1400 bytes and HT MCS Index 2 (19.5 Mbps)

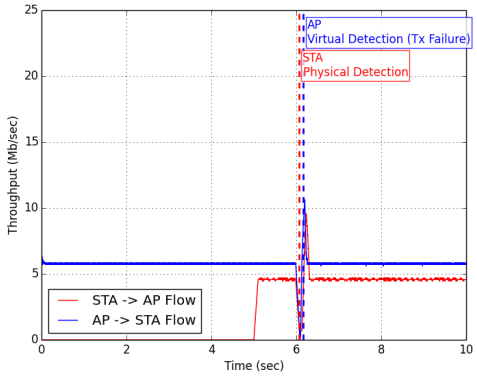
Figure 5(a) shows the AP and STA both detecting a DSRC transmission using their physical detectors. Because this detection event occurs in the latter half of the experiment when both downlink and uplink traffic flows are active, there is a brief interruption in achieved throughput in both flows.



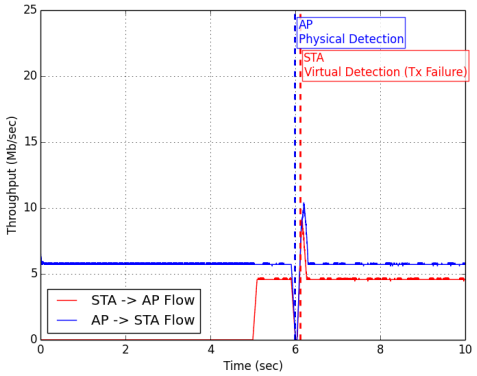
(a) Simultaneous physical detection events at AP and STA



(b) Physical detection event at AP followed by inactivity virtual detection event at STA



(c) Physical detection event at STA followed by Tx failure virtual detection event at AP



(d) Physical detection event at AP followed by Tx failure virtual detection event at STA

Figure 5: Throughput Characterization for Interruptions Caused by Detect-and-Vacate Events

The effect of the detection is minimal. At worse, a packet loss may have occurred if one device attempted to use the backup non-DSRC channel before the other device had finished re-tuning its radio.

Figure 5(b) shows the AP detecting a DSRC transmission using its physical detector. However, for the purposes of demonstrating the virtual detection modes described previously, the STA does not detect this DSRC transmission directly with its physical detector. Instead, the STA must rely on the inactivity virtual detection described in Section 2.3.2. For the inactivity virtual detector to assert, the AP and STA must be tuned to different channels for at least the duration of the inactivity timeout. During this time, no transmissions can successfully be delivered and throughput falls accordingly. After the STA tunes to the backup non-DSRC channel, the throughput of the downlink flow briefly increases above the nominal $\approx$6 Mbps rate. This throughput spike is a natural result of each node maintaining a transmit queue. New data MPDUs are being injected into the queue at a regular interval regardless of success or failure of the underlying medium access. In the interval between the two detection events (when transmissions must fail by definition), dequeueing has slowed. This is due to the fact that each data MPDU will be retried until the maximum retried limit is reach and, in between each retry, a mandatory backoff duration steadily increases. After both the AP and STA devices have tuned to the backup non-DSRC channel, all of the data MPDUs that have built up in the transmission queue are quickly emptied from the queue.

Figure 5(c) and Figure 5(d) both show the impact of the transmission failure virtual detection event on 802.11 throughput. Depending on the exact parameters of the virtual detection schemes, the throughput outage caused by the transmission failure virtual detect typically smaller than the outage caused by the inactivity virtual detect.

3 DSRC Preamble Detector Reference Design

The Mango Communications DSRC Preamble Detection Reference Design is capable of detecting the presence of the start of valid 10 MHz OFDM transmissions in four bands simultaneously. The design uses a single RF interface on the Mango WARP v3 hardware [1], capturing a 40 MHz bandwidth signal and analyzing it in real time. The 40 MHz bandwidth is nominally centered at 5875 MHz to monitor DSRC channels 172, 174, 176 and 178.

Throughout the design the four 10 MHz channels are labeled "bands" to disambiguate the detection frequencies from any particular 802.11 or DSRC channel definition. The detector baseband design views the 4 bands as 10 MHz subsets of the 40 MHz waveform, each offset by a different baseband frequency. Table 1 below summarizes the mapping of the detector "band" definition to the actual DSRC channels when the design is tuned to its nominal center frequency of 5875 MHz.

The DSRC detector can reliably detect DSRC waveforms at receive powers in -95 to -30 dBm in 10 MHz in the absence of external interferers.

Table 1: Baseband Frequency Equivalents

DSRC Channel	Channel Center Frequency	Band Baseband Frequency	Band Label
172	5860 MHz	-15 MHz	Band 1
174	5870 MHz	-5 MHz	Band 2
176	5880 MHz	+5 MHz	Band 3
178	5890 MHz	+15 MHz	Band 4

3.1 Theory of Operation

The underlying signal processing mechanism employed by the DSRC Preamble Detection Reference Design is a matched filter tuned to a single Short Training Symbol (STS) in the preamble of a 10 MHz OFDM transmission. This matched filter calculates a running cross correlation between the incoming stream of complex samples and the known 10 MHz-wide STS frequency shifted to each of the four bands. This approach realizes parallel detection of DSRC preambles in all 4 bands. All signal processing and control logic is implemented in the FPGA on the WARP v3 hardware and operates in real time.

Figure 6 presents a block diagram of the detector design. Each per-band detector consists of the following:

- A matched filter with 64 complex coefficients, calculated as a frequency-shifted, 4$\times$-interpolated version of the standard 16-sample short training symbol.
- A peak combining block which isolates the 10 peaks corresponding to the peaks in the output of the matched filter during the STF of an incoming DSRC preamble. This combiner adds processing gain to the detection system by exploiting the known period of STS repetitions in the preamble.
- The output of the peak combiner is compared to a dynamic threshold calculated as a fraction of the average amplitude of the incoming I/Q stream.

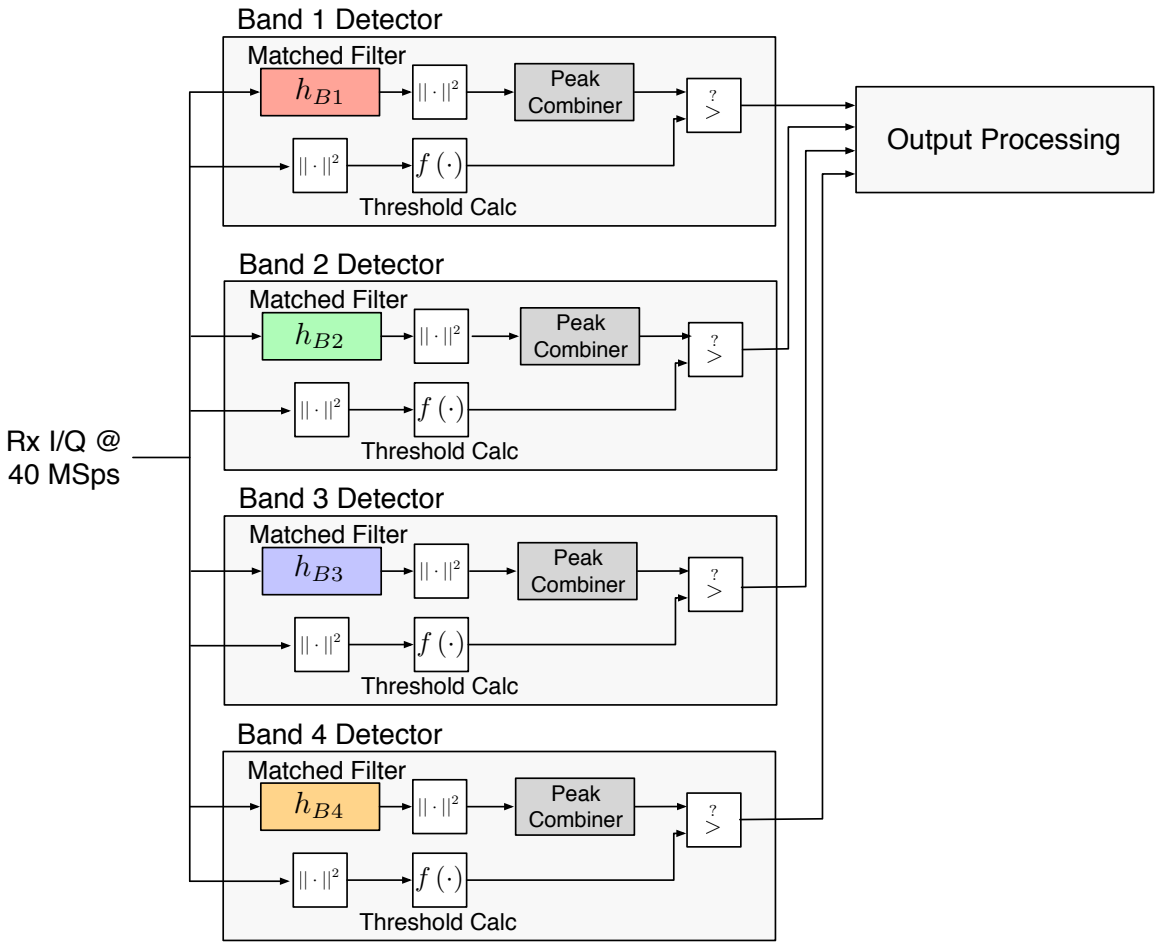


Figure 6: Block diagram of DSRC Preamble Detector architecture

- The outputs of the 4 per-band detectors are fed into an output processing block. This block implements the mapping of per-band detection events to the WARP v3 board's outputs (digital outputs, LEDs, etc).

4 DSRC Preamble Transmitter Reference Design

The Mango Communications DSRC Preamble Transmitter Reference Design is an FPGA design for WARP v3 hardware that can generate 10 MHz OFDM preambles in any of the four DSRC channels listed in Table 1. This design uses a single RF interface on WARP v3 to generate a 40 MHz bandwidth RF waveform composed of preambles in the four 10 MHz bands. Transmission in each band can be enabled or disabled at run time.

The preamble transmissions consist of only the STF portion of a valid DSRC preamble. The duty cycle of transmissions is controllable via the push buttons on the WARP v3 hardware (see Section 5 for more details).

4.1 Theory of Operation

The DSRC Preamble Transmitter Reference Design implements a simple FPGA architecture for waveform storage, sequencing, and transmission, shown in Figure 7.

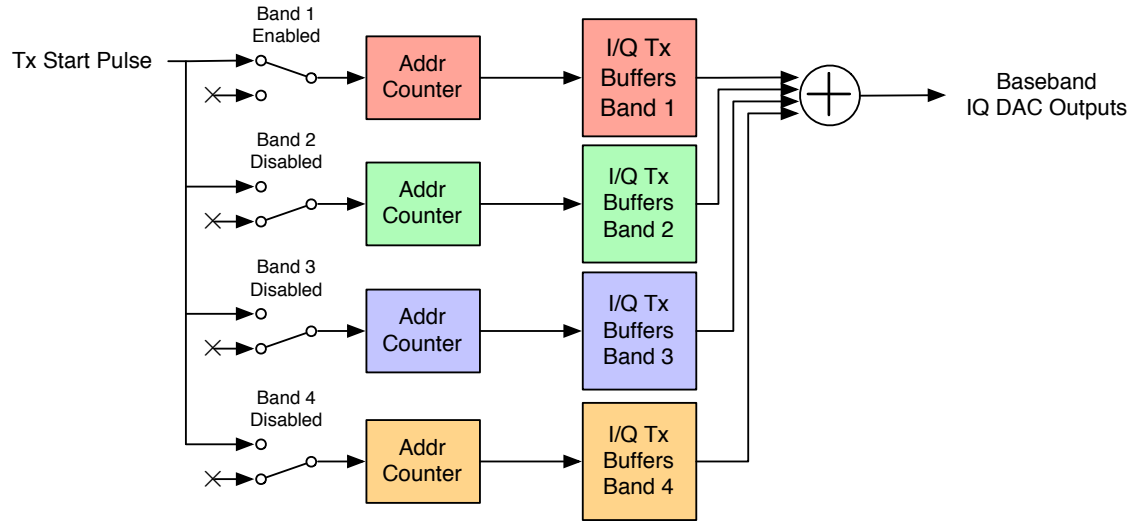


Figure 7: Block diagram of 10 MHz OFDM preamble transmitter.

Each waveform buffer stores an array of complex I/Q samples as pairs of signed fixed-point 16 bit values. By default the design populates the four waveform buffers with frequency-shifted, 4$\times$-interpolated OFDM preambles. The outputs of the per-band waveform buffers are summed (allowing transmission in multiple bands simultaneously) then driven to the WARP v3 DAC interfaces at 40 MSps. Each transmission event is triggered by a software timer.

The reference code implements a simple transmission sequence. On each Tx event the design read the DIP switch to select active bands (see Section 5.6). For each active band the software initiates a preamble transmission. For each Tx event the preamble transmissions for individual bands are separated by $\approx$2 msec. The period of Tx events is configurable and defaults to 100 msec.

5 Demonstration Setup

This section describes how to construct the 802.11/DSRC Detect-and-Vacate Demonstration, including the required connections between WARP v3 boards. Refer to Figure 8 below for the locations of the various WARP v3 hardware features described in the following design-specific sections.

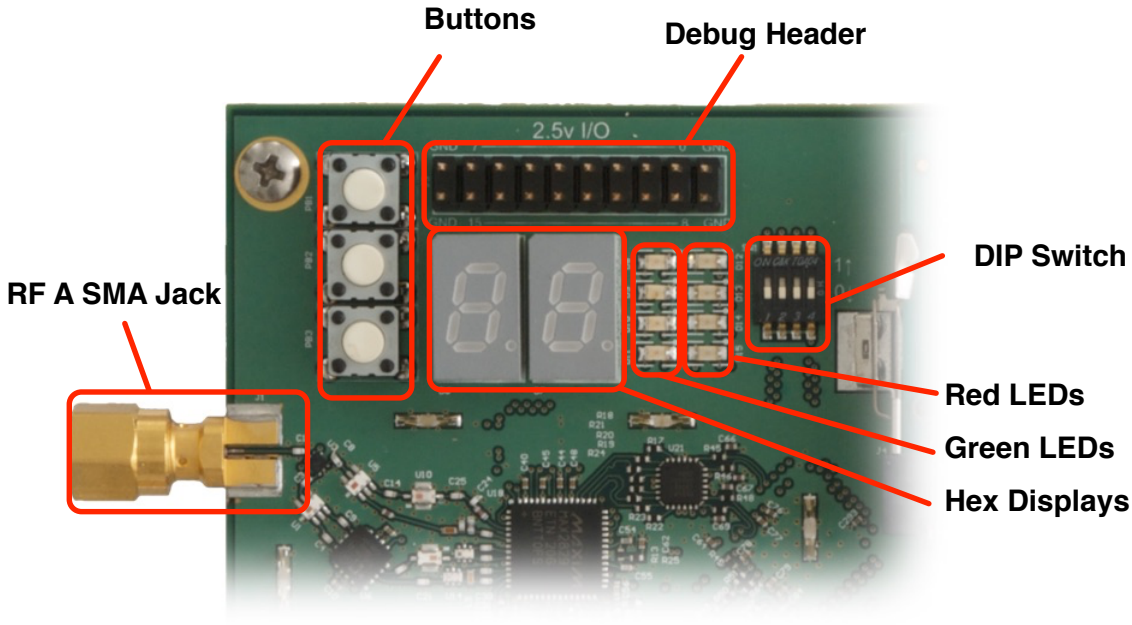


Figure 8: Detector: Hardware User I/O Usage.

5.1 DIP Switch Defaults

We suggest the following default settings for the DIP switches at each node. Refer to the following sections for details on the meaning of each node's switches.

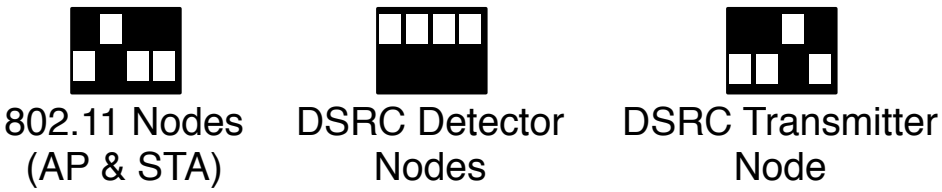


Figure 9: DIP Switch Defaults

These defaults will configure the nodes to:

- **802.11 Nodes:** create a wireless network and associate with each other via the wireless link. The wired-wireless bridge will begin operating once the nodes have associated.
- **DSRC Detectors:** monitor all 4 DSRC bands for DSRC preambles
- **DSRC Preamble Transmitter:** transmit DSRC preambles on band 3 with period 100 msec

5.2 Ethernet Connections

The default behavior of the 802.11 nodes is to implement a wireless-wired bridge between the node's wireless interface (via the 802.11 MAC/PHY) and the node's ETH A interface. In order to demonstrate the 802.11 link a PC must be connected to each 802.11 node's ETH A interface. The ETH A interfaces of the AP and STA nodes **must not** be connected together, either directly or via an Ethernet switch. A separate PC should be connected directly to each node.

The WARP v3 Ethernet connection is configured for a 1Gbps full-duplex link (1000BASE-T). The WARP v3 Ethernet interface does not implement auto-crossover (auto-MDI/MDI-X). Most PC NICs do implement auto-crossover, so a normal Ethernet patch cable is typically sufficient for connecting a PC directly to the WARP v3 board.

The 802.11 node acts as a layer-2 bridge. It does not implement any higher functions (routing, DHCP, etc). The PCs for each node should be configured with unique static IP address in the same subnet. For example, configure one PC as `10.0.1.10` and the other as `10.0.1.11`, both with subnet mask `255.255.255.0` and no default gateway.

Any application that uses the PC's Ethernet connection can be used to generate traffic for the wireless link. One option is `iperf`, a network performance measurement tool that works over local network connections.

For demonstration purposes a video stream is effective. We recommend VLC for this purpose. The VLC application can read a video file and generate a UDP stream targeted to another VLC instance on the remote PC. The receiving VLC instance receives the stream and displays the video. The traffic load can be configured by adjusting the bit rate of VLC's video transcoder.

5.3 Debug Header Connections

Each pair of 802.11 and DSRC Detector boards must be connected via a ribbon cable. The ribbon cable is connected between the 20-pin debug headers on the two WARP v3 boards. Additional pins from the 802.11 node can also be connected to external equipment for real-time monitoring.

The required cable connections are illustrated in Figure 10.

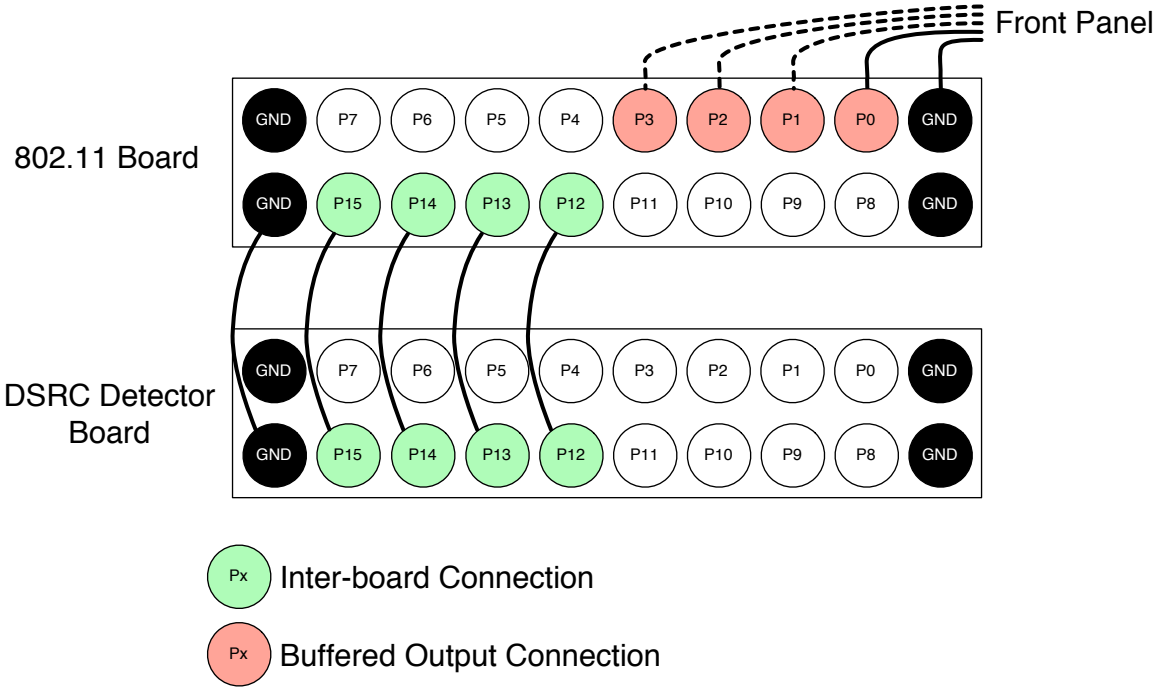


Figure 10: WARP v3 debug header cable connections

Inter-board Connections: The 802.11 and DSRC Detector nodes should be connected via pins `[12, 13, 14, 15]` on the WARP v3 debug headers. There should be a direct connection between pins of the same position on each board, as illustrated in Figure 10. The ground pins must also be connected between the boards. Any of the 4 ground pins on the debug header can be used. The pin adjacent to pin 15 is likely the most convenient.

Front-Panel Connections: The 802.11 design can drive pins `[0, 1, 2, 3]` with real-time status signals. These status signals are useful for monitoring the state of the design and measuring real-time durations of various events. In the initial demonstration design the 802.11 code drives pin `0` high when a DSRC detection event has been observed and the 802.11 MAC has re-tuned the radio to the backup channel. The design will de-assert pin `0` when the detection event is reset (see Section 5.4).

DSRC Preamble Transmitter Debug Header: The preamble transmitter design drives pins `[0, 1, 2, 3]` in real time to indicate that transmissions in bands `[1, 2, 3, 4]` are active, respectively.

5.4 802.11 Node Setup

The 802.11 node uses the WARP v3 buttons and DIP switch to configure various parameters and uses the hex displays and LEDs to display various status values.

- **RF A SMA Jack**: this is a 50 Ω standard polarity, standard gender SMA jack. An RF cable or antenna suitable to the application should be connected here. Only light weight "rubber duck" antennas should be mounted directly to the SMA jack. Bulkier antennas must be connected via a coaxial cable to avoid damage to the SMA jack's connection to the circuit board. The power input to the SMA jack **must not exceed -20 dBm** to avoid damage to the receive circuits. The SMA jack is AC-coupled to the on-board antenna switch. The max Tx power from the SMA jack is approx 20 dBm.
- **DIP Switch**: the 802.11 node uses the 2 left positions of the DIP switch to configure at-boot settings. The DIP switch values are only read at boot. The left most switch controls whether the AP and STA create a wireless network and associate with each other via the standard 802.11 association handshake. The second switch controls whether the node implements detect-and-vacate. When this switch is de-asserted the 802.11 node will ignore real and emulated DSRC detection events.


AP: Create network, normal beacon Tx
STA: Join AP's network


Ignore DSRC Detection events


AP: No network, no beacon Tx
STA: Do not join network


Monitor for DSRC Detection events

- **Green LEDs**: the green LEDs increment when wireless packets are received. The speed of the LED changes reflects the rate of wireless packet receptions.
- **Red LEDs**: the 4 red LEDs blink together whenever a DSRC detection event has occurred. The LEDs will continue blinking until the detection event is reset via the Up Button.
- **Hex Displays**: the hex displays reflect the association status of the AP and STA. Both nodes will display $\boxed{0}\boxed{1}$ when they have established a wireless connection and are members of the same wireless network. The nodes will display $\boxed{0}\boxed{0}$ when they are attempting to associate. The nodes will display $\boxed{-}\boxed{-}$ when they are configured to not attempt association via the second DIP switch.
- **Buttons**: the top and bottom buttons are used to control the 802.11 MAC's "DSRC Detected" states.
  - **Up Button**: resets the 802.11 node's "DSRC Detected" state. Resetting the 802.11 node's "DSRC Detected" state has two steps. First, push the Up Button at both nodes. The nodes' left hex displays will update to $\boxed{1}$. At this point the nodes have re-tuned to the primary channel, but all DSRC detection modes are disabled. Push the Up Button at both nodes again. The left hex displays will update to $\boxed{2}$, indicating the nodes are fully reset, monitoring for DSRC detection events. The two-step reset process ensures both nodes are able to re-tune to the primary channel and re-establish communication without trigging inferred detection events due to different re-tuning times.
  - **Down Button**: triggers an emulated DSRC detection event. Pushing this button triggers the 802.11 MAC's detect-and-vacate behaviors, emulating the response to a real DSRC detection event. The post-detection state can be reset using the Up Button, described above.
- **Debug header**: the 16 pins on the debug header are connected directly to 2.5v FPGA IO. These pins must not be driven with higher voltages.

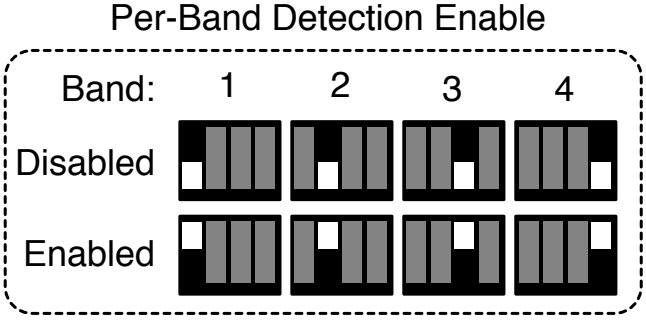
  - **Ground pins** are connected to the WARP v3 board's ground plane and should be used as the voltage reference when monitoring the values of the digital outputs
  - **Pin** `0` asserts when a DSRC detection event has occurred and remains asserted until the detection state is reset with the Up Button. This pin can be connected to an oscilloscope for real-time monitoring of the detect-and-vacate behavior.
  - **Pins** `[12, 13, 14, 15]` must be connected to the same pins on the DSRC Detector node.

5.5 DSRC Detector Node Setup

The DSRC Preamble Detection Reference Design uses a number of hardware features of the Mango WARP v3 board to configure the design's behavior and indicate its status. Figure 8 highlights these hardware features. The usage of each is described below.

When programmed with the DSRC Preamble Detector bitstream a WARP v3 kit will begin monitoring the RF waveform at the RF A SMA jack.

- **RF A SMA Jack**: this is a 50 Ω standard polarity, standard gender SMA jack. An RF cable or antenna suitable to the application should be connected here. Only light weight "rubber duck" antennas should be mounted directly to the SMA jack. Bulkier antennas must be connected via a coaxial cable to avoid damage to the SMA jack's connection to the circuit board. The power input to the SMA jack **must not exceed -20 dBm** to avoid damage to the receive circuits. The SMA jack is AC-coupled to the on-board antenna switch. The max Tx power from the SMA jack is approx 20 dBm.
- **DIP Switch**: the 4-position DIP switch configures the detector to monitor or ignore each of the 4 10 MHz bands. All 4 switches should be asserted (pushed "up") to monitor all 4 bands.

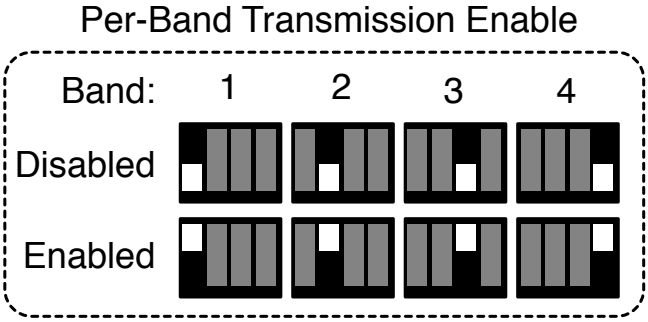


- **Green LEDs**: the 4 green LEDs are mapped to the per-band detection outputs. A green LED will illuminate for 250 msec after a detection in a single channel. The bottom LED is mapped to the lowest frequency channel.
- **Red LEDs**: the 4 red LEDs illuminate together whenever a detection event occurs in any band. The LEDs remain illuminated for 250 msec after a detection event.
- **Hex Displays**: the DSRC Detector design does not use the hex displays.
- **Buttons**: the DSRC Detector design does not use the buttons.
- **Debug header**: the 16 pins on the debug header are connected directly to 2.5v FPGA IO. These pins must not be driven with higher voltages.
  - **Ground pins** are connected to the WARP v3 board's ground plane and should be used as the voltage reference when monitoring the values of the digital outputs
  - **Pins** `[12, 13, 14, 15]` must be connected to the same pins on the 802.11 node.

5.6 DSRC Preamble Transmitter Node Setup

The DSRC Preamble Detection Reference Design uses a number of hardware features of the Mango WARP v3 board to configure the design's behavior and indicate its status. Figure 8 highlights these hardware features. The usage of each is described below.

- **RF A SMA Jack**: this is a 50 Ω standard polarity, standard gender SMA jack. An RF cable or antenna suitable to the application should be connected here. Only light weight "rubber duck" antennas should be mounted directly to the SMA jack. Bulkier antennas must be connected via a coaxial cable to avoid damage to the SMA jack's connection to the circuit board. The power input to the SMA jack **must not exceed -20 dBm** to avoid damage to the receive circuits. The SMA jack is AC-coupled to the on-board antenna switch. The max Tx power from the SMA jack is approx 10 dBm.
- **DIP Switch**: the 4-position DIP switch configures the transmitter to enable or disable each of the 4 10 MHz bands. All 4 switches should be asserted (pushed "up") to transmit in all 4 bands.



- **Green LEDs**: the four green LEDs illuminate when a transmission occurs, with each LED indicating a transmission in a single band. Band 1 maps to the lowest LED.
- **Red LEDs**: the four red LEDs are not used by the DSRC Preamble Transmitter Reference Design.
- **Hex Displays**: the hex displays show the current preamble transmission frequency as a 2-digit value in units of transmissions per second. The displays update when the transmission frequency is updated via the buttons.
- **Buttons**: the 3 push buttons configure the series detection parameters.
  - The **Up** button increments the frequency of preamble transmissions, in units of 1 transmission per second, up to a maximum of 99 transmissions per second.
  - The **Middle** button decrements frequency of preamble transmissions, in units of 1 transmission per second, down to a minimum of 1 transmission per second.
  - The **Down** button resets the frequency of preamble transmissions to 10 per second.

6 Demonstration GUI Application

The demonstration includes a graphical application which monitors and controls some parameters of the prototype devices. The GUI is **not** required to use the demonstration platform. However it provides a convenient interface for quickly configuring the nodes' transmit parameters and DSRC detection modes. When the GUI is used to control the demonstration hardarwe the other user I/O (buttons, switches) should not be used. Attempting to configure the nodes with both the GUI and switches can lead to inconsistent state.

6.1 GUI Setup

The GUI application runs on a Windows PC connected to the demonstration platform via Ethernet. The application interacts with the 802.11 Reference Design on both the AP and STA nodes. The steps below describe the necessary configuration.

1. Configure the PC's Ethernet interface with a static IP address of `10.0.0.200` and subnet mask `255.255.255.0`.
2. Connect the PC's Ethernet interface to a 1000Mbps (gigabit) Ethernet switch.
3. Connect the **ETH B** interfaces of each 802.11 node to the same switch.
4. Disconnect all other devices from the switch. The switch must **not** be connected to other PCs or to the 802.11 node's ETH A interfaces.
5. Verify the WARP v3 node serial numbers in the `config.ini` file match the 802.11 nodes.
6. Power on the demonstration hardware.
7. Run the GUI application.

6.2 GUI Operation

A screenshot of the GUI application is shown in Figure 11. Each element of the GUI is described in the following sections.

6.2.1 Current Channel

The GUI displays the channel number for the 802.11 node's current center frequency. By default the node operates the 802.11 link in a channel that overlaps DSRC frequencies. Upon DSRC detection the node will re-tune to a non-DSRC channel. The GUI will update the **Current Channel** field when the node re-tunes. The primary (DSRC) and backup (non-DSRC) channels are configured in the GUI's `config.ini` file.

6.2.2 Detection State

The GUI displays the node's current DSRC detection state:

- **Disabled**: all DSRC detection modes are disabled; the 802.11 node will ignore any DSRC detection events
- **Ready**: at least one DSRC detection mode is enabled; the 802.11 node is operating in the primary channel and will re-tune upon a DSRC detection event.
- **Vacated**: the node has re-tuned to the backup channel following a DSRC detection event. The source of the detection event is displayed in parentheses:
  - **Preamble Detected**: a DSRC waveform was detected by the DSRC Detection node
  - **Inactivity**: a detection event was inferred following an interval of no receptions from the partner node
  - **Tx Failures**: a detection event was inferred by repeated failures of transmissions to the parter node
  - **Button Press**: an emulated detection event was triggered by pushing the WARP v3 board's "Down" button

Figure 11: GUI Application Screenshot

6.2.3 Tx Data Rate

The 802.11 Reference Design supports 16 PHY data rates. Each data rate represents a PHY mode (non-HT or HT), a modulation scheme (BPSK, QPSK, 16-QAM, or 64-QAM) and a coding rate (1/2, 2/3, 3/4, or 5/6). This demonstration design uses the same PHY rate for all MPDU transmissions addressed to the partner node. Acknowledgment responses use the highest basic rate that is lower than the specified rate for outgoing MPDU transmissions. The GUI can set the PHY rate for each node. Higher PHY rates result in shorter but less robust waveforms.

6.2.4 Traffic Generator

The 802.11 Reference Design includes an arbitrary traffic generator which can inject traffic into the transmit queue for any destination address. The payload size and packet generation intervals are configurable. The demonstration design configures the traffic generator at each node to create packets addressed to the other node. Conceptually the traffic generator is similar to a UDP constant bit rate (CBR) source. The traffic flow is unidirectional and each packet has the same characteristics. When a node's local traffic generator is enabled the node ignores its ETH A Ethernet interface. Normal wired-wireless bridging will resume when the local traffic generator is disabled. The demonstration GUI provides four fields for each node's traffic generator:

- **Traffic Generator Toggle**: enables or disables the node's traffic generator.
- **New Packet Interval**: the interval at which a new packet is added to the transmit queue.

- **Packet Size**: the size in bytes of the payload included with each enqueued packet.
- **Offered Load**: the offered traffic load, computed from the payload size and packet creation rate. The Tx queue will be backlogged when the offered load exceeds the achievable throughput of the wireless link.

6.2.5 Packet Counts

Each 802.11 node tracks Tx and Rx statistics for each partner device. The GUI displays three statistics for each node:

- **Num Rx**: number of non-duplicate packets received without error from the partner node
- **Num Tx Total**: number of packets transmitted to the partner node
- **Num Tx Success**: number of packets transmitted to the partner node for which ACKs were received

6.2.6 Detection Modes

The various DSRC detection modes can be individually enabled or disabled at each 802.11 node:

- **Vacate on Detection**: controls whether the 802.11 device listens for physical DSRC detection events from the DSRC Preamble Detector device
- **Vacate on Tx Failure**: controls whether the 802.11 device implements the successive Tx failure inferred detection mode
- **Vacate on Inactivity**: controls whether the 802.11 STA node implements the Rx inactivity inferred detection mode

6.2.7 Reset Buttons

The GUI provides two reset buttons:

- **Reset Node Channels**: this button implements a post-detection reset. When pressed both nodes are re-tuned to the primary (DSRC) channel and all post-detection state is purged. Pushing this button has no effect when DSRC detection has not occurred at either node
- **Reset Node Tx/Rx Counts**: this button resets the Tx/Rx statistics at both nodes. This reset has no affect on the DSRC detection state of either node

References

[1] "WARP v3 Hardware User Guide." [Online]. Available: https://warpproject.org/trac/wiki/HardwareUsersGuides/WARPv3

[2] "802.11 Reference Design User Guide." [Online]. Available: https://warpproject.org/trac/wiki/802.11/?d1

[3] "802.11 WLAN Experiments Framework Event Log." [Online]. Available: https://warpproject.org/trac/wiki/802.11/wlan_exp/log