

Synchronized Sweep with External I and Q Output Ports

Overview

WSA5000 provides I and Q output (IQout) ports such that they can be used to mate the RF front-end with an external, high speed digitizer. The IQout ports can be used with ThinkRF's synchronized sweep engine and the GPIO port to automate the spectrum scanning process, minimizing the RF front-end control interaction.

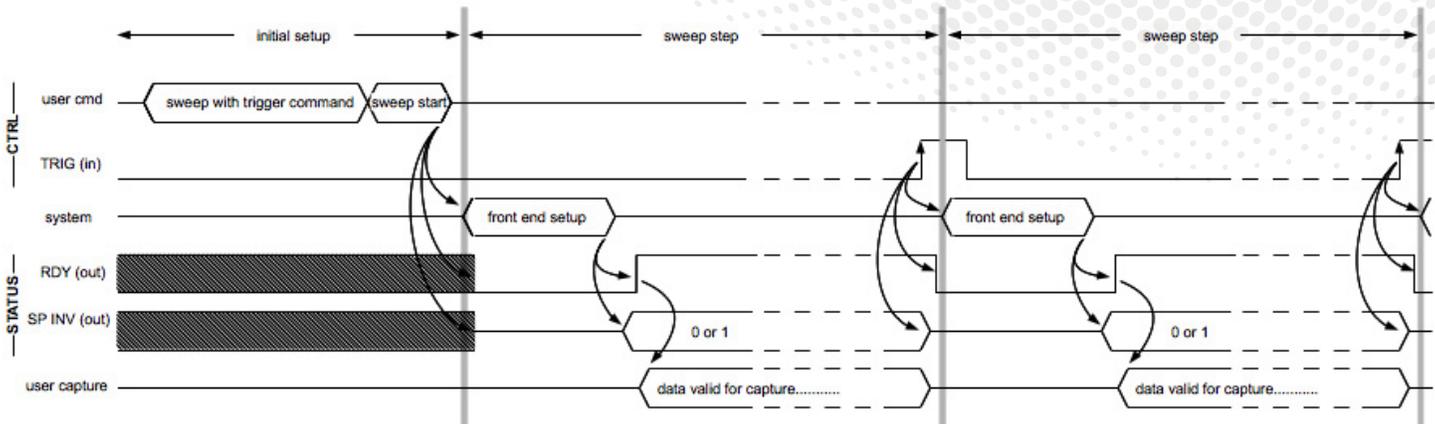
This application note provides the details necessary for setting up a synchronized sweep with the IQout and assumes that the user knows how to program a WSA. Refer to ThinkRF's Programmer's Guide and User Guide as needed, both available at <http://thinkrf.com/documentation/>

Setting Up Synchronized Sweep

The steps below provide the sequence of events for setting up the Synchronized Sweep with IQout. This document uses trigger PULSE type to simplify the explanation. See the timing diagram following the steps for the flow of the process as well as the hand-shaking sequence.

1. Set WSA5000 to IQout mode using the SCPI command:
`:OUTPUT:IQ:MODE CONNECTOR`
2. Setup Sweep with pulse or sweep trigger via SCPI commands
3. Initiate Sweep Start via SCPI command to start the first sweep step
4. Wait for the system RDY (out) signal to go high to indicate the system including front-end setup is ready/completed
5. User performs data capture. Note SP_INV (out) signal for indication of spectral inversion requirements (see in Programmer's Guide the "Trailer Word Format" section for details)
6. Pulse TRIG (in) high to inform the WSA to setup the next sweep step
7. Repeat steps 4-6 until all sweep entries performed

Synchronized Sweep for IQout



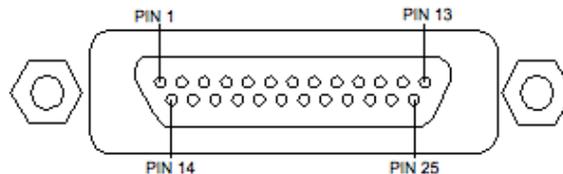
Note:

- The TRIG pulse strobe could be of any length since the device uses edge triggering.
- Once the RDY signal goes high after a setup, ensure there is enough time for data capture until the next rising TRIG (in) strobe. This can be calculated as, for example, $\text{capture time} = \text{samples size} * (1/\text{sample rate})$. Adjust this capture time accordingly as needed by your data capture and/or data process application.

GPIO Port Setup

GPIO Pinouts

The micro-DB25 as view from the WSA5000's end-plate:



The list of GPIO pins required for the synchronized sweep setup:

Pin	Direction	Voltage	Signal
4	in	3.3 V	TRIG IN
16	out	3.3 V	SP(ectral) INV(ersion)
17	out	3.3 V	RDY
23-25	power	6 V or 12 V*	VCC
9,15,21-22	ground	0 V	GND

* Please see the **Caution** note on the next page



Caution: The voltage level of these pins is WSA5000-XXX model dependent: 6V for -1XX and -2XX models and 12V for -3XX and -4XX models. Please check your model for the proper hardware interface when used to avoid potential damage.

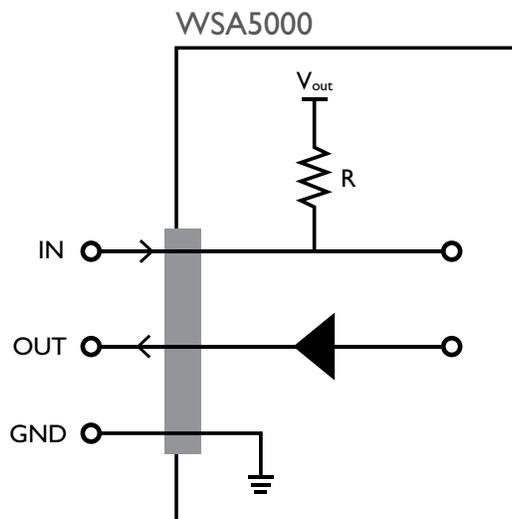
DC Characteristics

The DC characteristics for the GPIO pins is:

	Min	Max	Units
V_{IH}	2		V
V_{IL}		0.8	V
V_{OH}		3.3	V
V_{OL}	0		V
I_{supply} (From VCC)		1.5	A
$I_{sink/source}$		12	mA

Circuit Diagrams

The following circuit diagram illustrates a simplified schematic for the Input and Output pins of the GPIO.

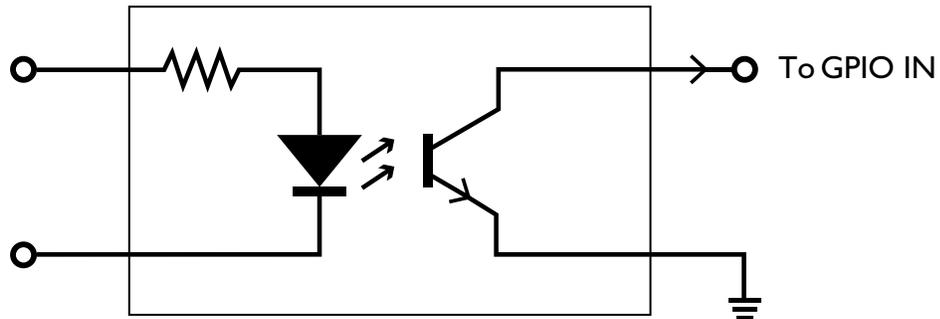


where:

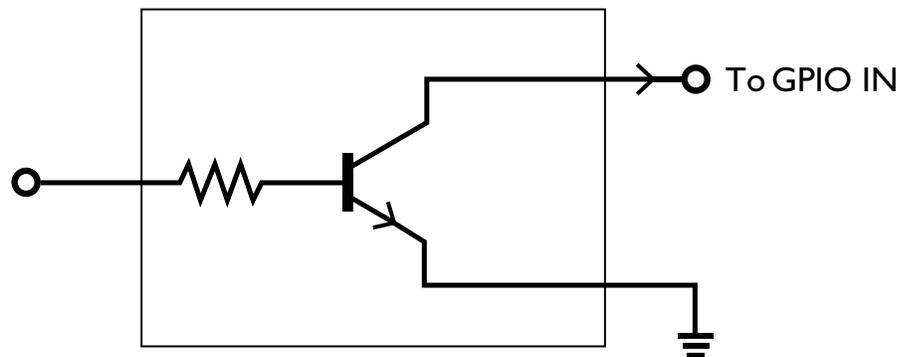
$$V_{out} = 3.3 \text{ V}$$
$$R = 10 \text{ k}\Omega$$

The following circuit diagrams illustrate, as examples, two different methods of setting up your own circuit to interface to an OUTPUT pin of the GPIO port. The ground (GND) of the interface circuit should be common to the GPIO GND.

Opto-Isolator Interface



Transistor Interface



Note: The cable in this interface should be much less than 10 m long.

Code Example

The following program is a simple SCPI setup example using pyRF, an open-source Python API developed to interface with WSA devices (see <http://thinkrf.com/pyrf/>).

```
# Import libraries
import sys
from pyrf.devices.thinkrf import WSA

# Define handle & connect to WSA
dut = WSA()
dut.connect(sys.argv[1])

# Reset WSA configuration and clean the capture buffer
dut.reset()
dut.flush()

# Set IQ path to use external connector port instead of internal digitizer
dut.scpiiset('OUTPUT:IQ:MODE CONNECTOR')

# Set the WSA's sync state to slave, default is OFF
dut.scpiiset('SYSTEM:SYNC:MASTER OFF')

#####
# Create a sweep entry
#####

# Set RFE mode
dut.scpiiset('SWEEP:ENTRY:MODE ZIF')

# Set frequency range and step size
dut.scpiiset('SWEEP:ENTRY:FREQ:CENT 1200 MHZ, 3200 MHZ')
dut.scpiiset('SWEEP:ENTRY:FREQ:STEP 10 MHZ')

# Set the WSA's trigger type in the sweep entry to use pulse type
dut.scpiiset('SWEEP:ENTRY:TRIGGER:TYPE PULSE')

# Set number of iteration to 1 time, default is 0 for infinity
dut.scpiiset('SWEEP:LIST:ITERATION 1')

# Save the sweep entry
dut.scpiiset('SWEEP:ENTRY:SAVE 0')

#####

# Start sweep mode
dut.scpiiset('SWEEP:LIST:START')

# perform data captures, etc.
....

# Stop sweep mode if want to break early or when infinite iteration is set
# dut.scpiiset('SWEEP:LIST:STOP')
```