OOT hier blocks BPSK and QPSK phase ambiguity resolver

Author: Gustavo Gonzalez
Date: 03-02-2022

INDEX

1 INTRODUCTION

1.1 OBJECTIVE

BPSK and QPSK demodulation would end up with phase ambiguity, meaning the symbol constellation could be rotated or even inverted with respect to the expected constellation because clock's and frequency's drift in the transmitter and in the receiver, this ambiguity causes data lost. There are a few methods to overpass this phase ambiguity, one of this is differential modulation, this will avoid phase ambiguity but will duplicated the bit errors. The other method is using a sync word to resolve the right phase, this method is more complex the differential modulation, but will not increase the errors and is the one used in here

Then, this document describes the design of an ad-hoc OOT hier block for BPSK and QPSK phase ambiguity resolver based on ASM detection

1.2 REFERENCE DOCUMENTS

Reference Documents hereinafter are referred to as RD.X.

[R.D.1] Phase Ambiguity Resolution for QPSK Modulation Systems - Part I: A Review - Tien Manh Nguyen, NASA May 15,1989

2 QPSK PHASE AMBIGUITY RESOLVER DESCRIPTION

.

According to [R.D.1], and depending of phase rotation and its direction, each Symbol in QPSK demodulation could be decoded following next tables states:

CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)		Output	RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		Output
	I_R=0	**Q_R=0**		**I_R=0**	**Q_R=0**	
0	I_T=0	Q_T=0	00	Q_T=0	I_T=0	00
90	-Q_T=1	I_T=0	10	I_T=0	-Q_T=1	01
180	-I_T=1	-Q_T=1	11	-Q_T=1	-I_T=1	11
270	Q_T=0	-I_T=1	01	-I_T=1	Q_T=0	10

CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)		Output	RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		Output
	I_R=0	**Q_R=1**		**I_R=0**	**Q_R=1**	
0	I_T=0	Q_T=1	01	Q_T=1	I_T=0	10
90	-Q_T=0	I_T=0	00	I_T=0	-Q_T=0	00
180	-I_T=1	-Q_T=0	10	-Q_T=0	-I_T=1	01
270	Q_T=1	-I_T=1	11	-I_T=1	Q_T=1	11

CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)			RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		
	I_R=1	**Q_R=0**	**Output**	**I_R=1**	**Q_R=0**	**Output**
0	I_T=1	Q_T=0	10	Q_T=0	I_T=1	01
90	-Q_T=1	I_T=1	11	I_T=1	-Q_T=1	11
180	-I_T=0	-Q_T=1	01	-Q_T=1	-I_T=0	10
270	Q_T=0	-I_T=0	00	-I_T=0	Q_T=0	00

CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)			RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		
	I_R=1	**Q_R=1**	**Output**	**I_R=1**	**Q_R=1**	**Output**
0	I_T=1	Q_T=1	11	Q_T=1	I_T=1	11
90	-Q_T=0	I_T=1	01	I_T=1	-Q_T=0	10
180	-I_T=0	-Q_T=0	00	-Q_T=0	-I_T=0	00
270	Q_T=1	-I_T=0	10	-I_T=0	Q_T=1	01

The resolver's proposed in this document, works at the QPSK Constellation decoder output, at this level we have a set of 2 bits per symbols and we can associated I with the MSB and Q to the LSB in the 2bits symbol.
The implementation is based on the one presented in [A.D.1], a set of correlators will look in the input stream of bits the I part of the ASM and the Q part of ASM for each bit from I and from each bit from Q detecting the matching or the reverse matching (every bit inverted)

The ASM uses in the exampled is the barker 0x1ACFFC1D and decomposed in the even bits for I and in the odd bits for Q we have:

ASM	0001 1010 1100 1111 1111 1100 0001 1101	0x1ACFFC1D
I ASM	0011 1011 1110 0010	0x3BE2
Q ASM	0100 1011 1110 0111	0x4BE7

The basic detector is below picture, where each input bit is converted to a float
0 ➔ -16 and 1 ➔ +16 to input into a Decimating FIR Filter, then, the FIR filter will outputs 16 when the Q-ASM or I-ASM matches with the input bit or -16 when matches with /Q-ASM or /I-ASM. The FIR output is compared with +16 or -16 then will generated a 1 at the threshold output, otherwise will remains in 0, allowing to detect any combination of Q-ASM or I-ASM in the input I stream or in the input Q stream

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez

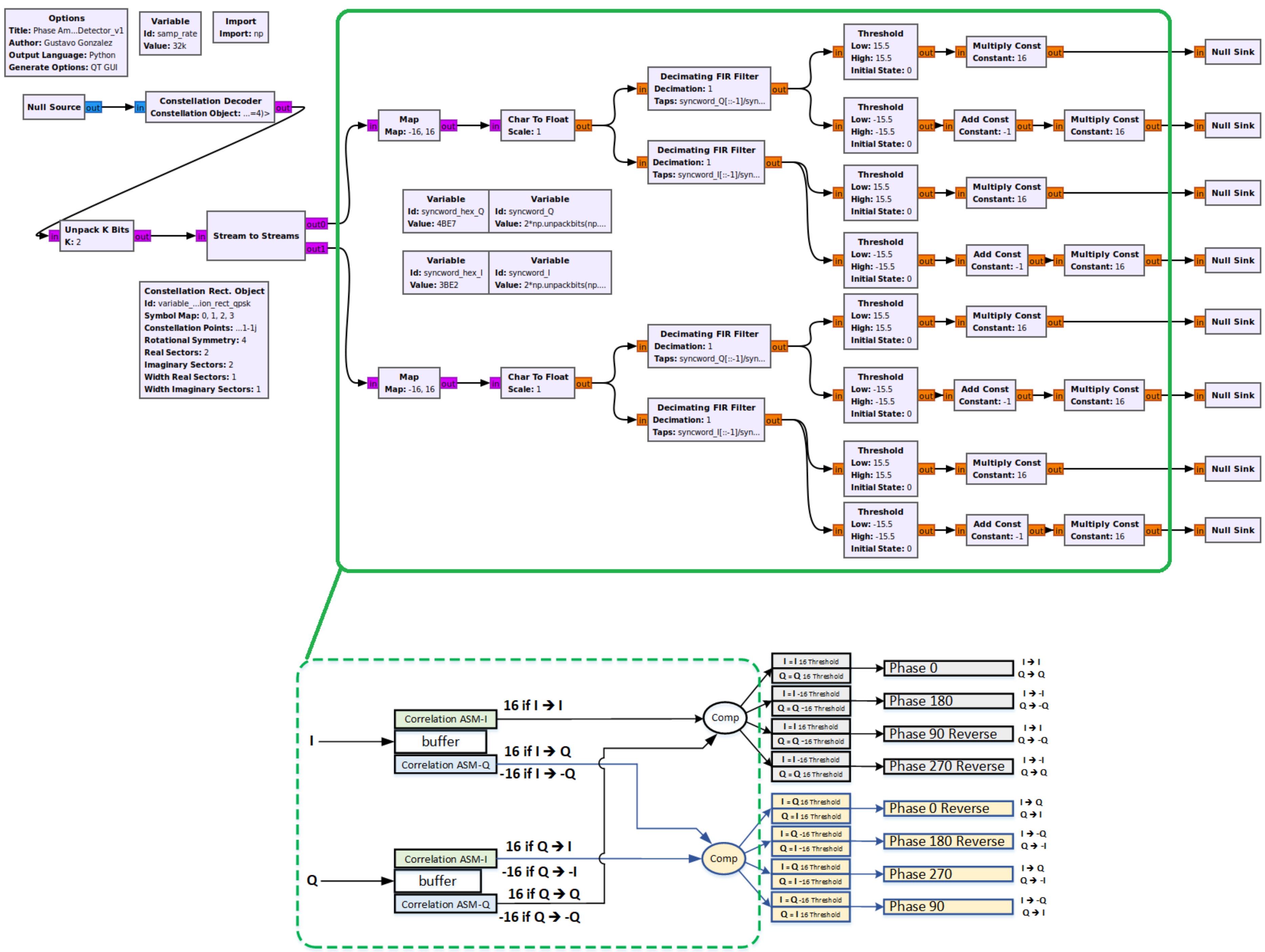


Figure 1 Implementation of a QPSK Phase Ambiguity Detector

For testing the previous implementation, an AOS frame generator is used to generated ASM+AOS Frames and stored in a file

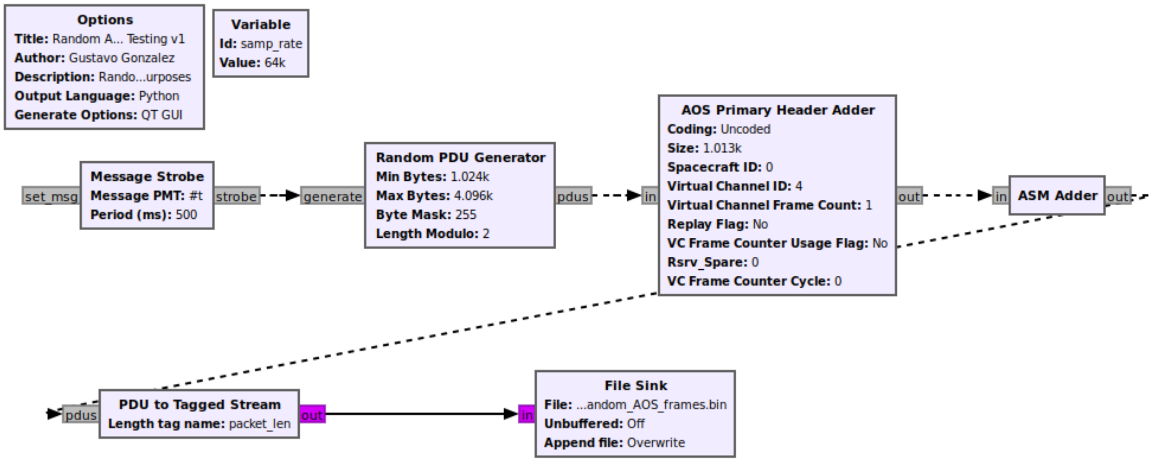


Figure 2 Random Frame Generator

The output ASM+AOS frames feed a QPSK modulator circuit, the modulated signal is then connected a set of 4 receivers, QPSK demodulators and Phase detectors, each one will received the QPSK modulated signal rotated 0º, 90º, 180º or 270º. The detector output will be displayed marking any detection.

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez

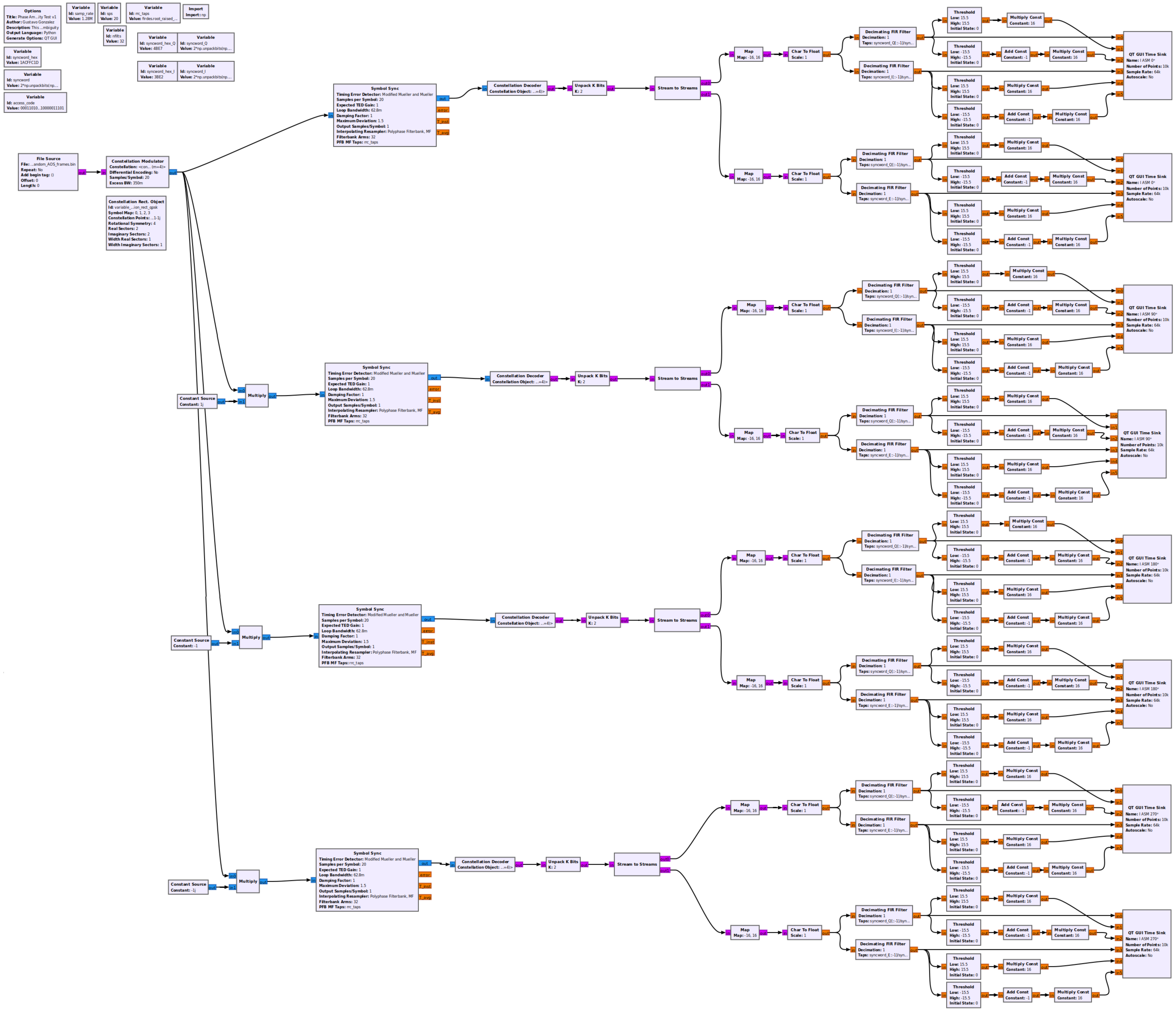


Figure 3 Phase Ambiguity Detector Test Bench for nominal Phase Rotation

Testing the implementation, its work as expected: the data file with ASM+AOS Frames (1024bytes size each block) is modulated in QPSK and transmitted, the rotated 0º, 90º, 180º or 270º degrees and the feed four QPSK demodulators and Phase ambiguity detectors. The results are the expected one as previous tables

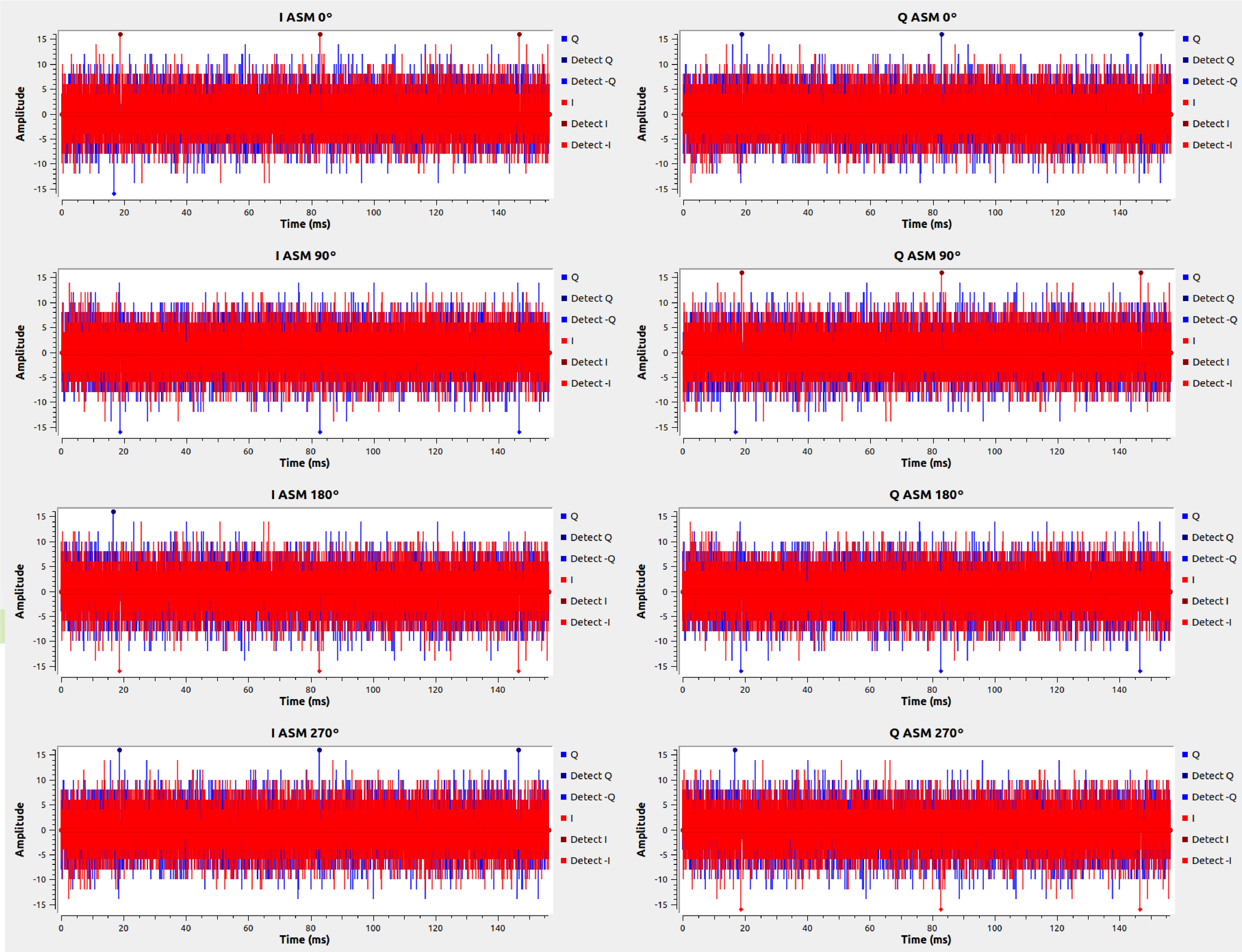


Figure 4 Detection of ASM for each Phase Nominal Rotation of a QPSK Signal

The same test as before but swapping IQ in the QPSK modulated signal lays to the expected results too

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez

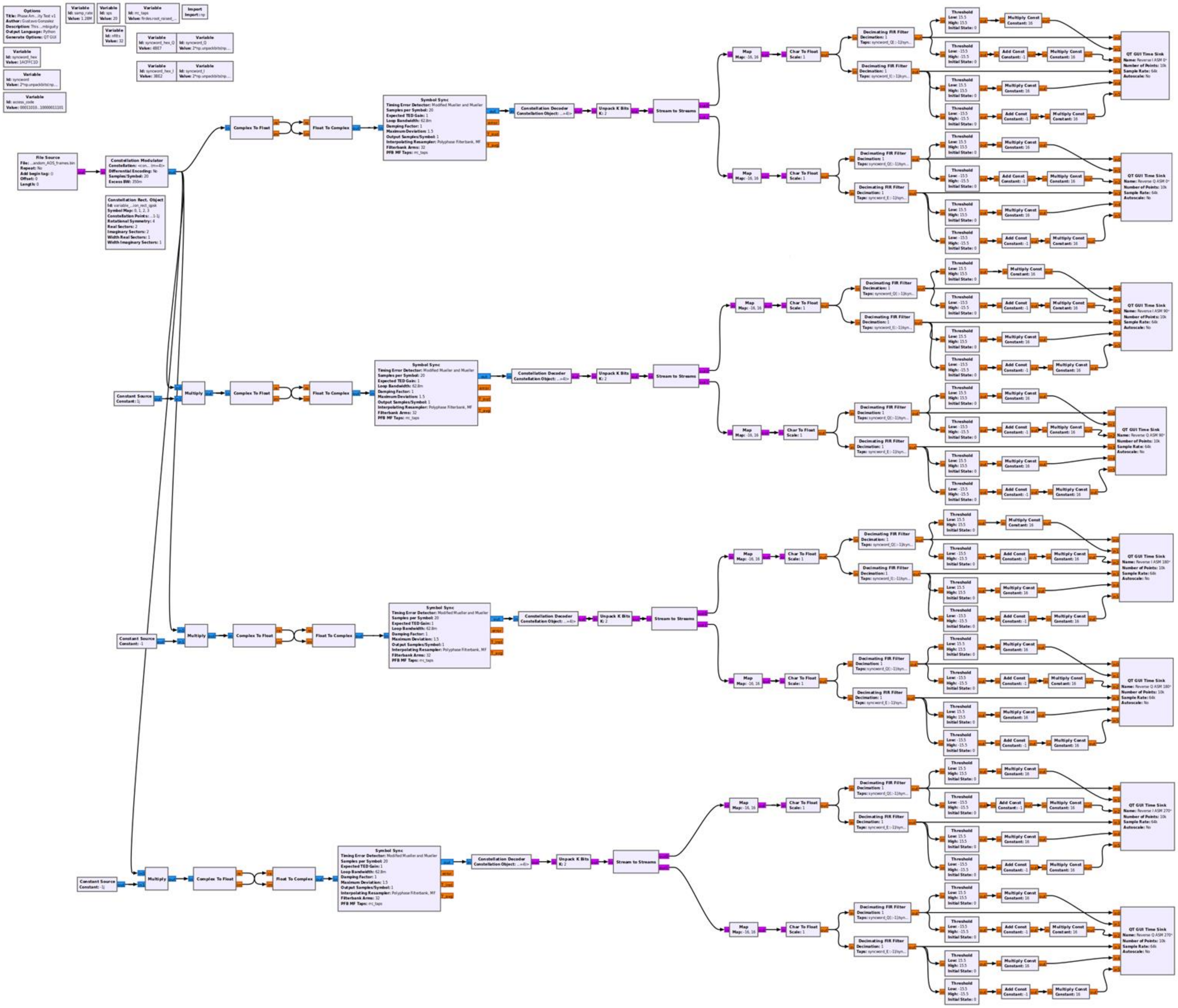


Figure 5 Phase Ambiguity Detector Test Bench for reverse Phase Rotation

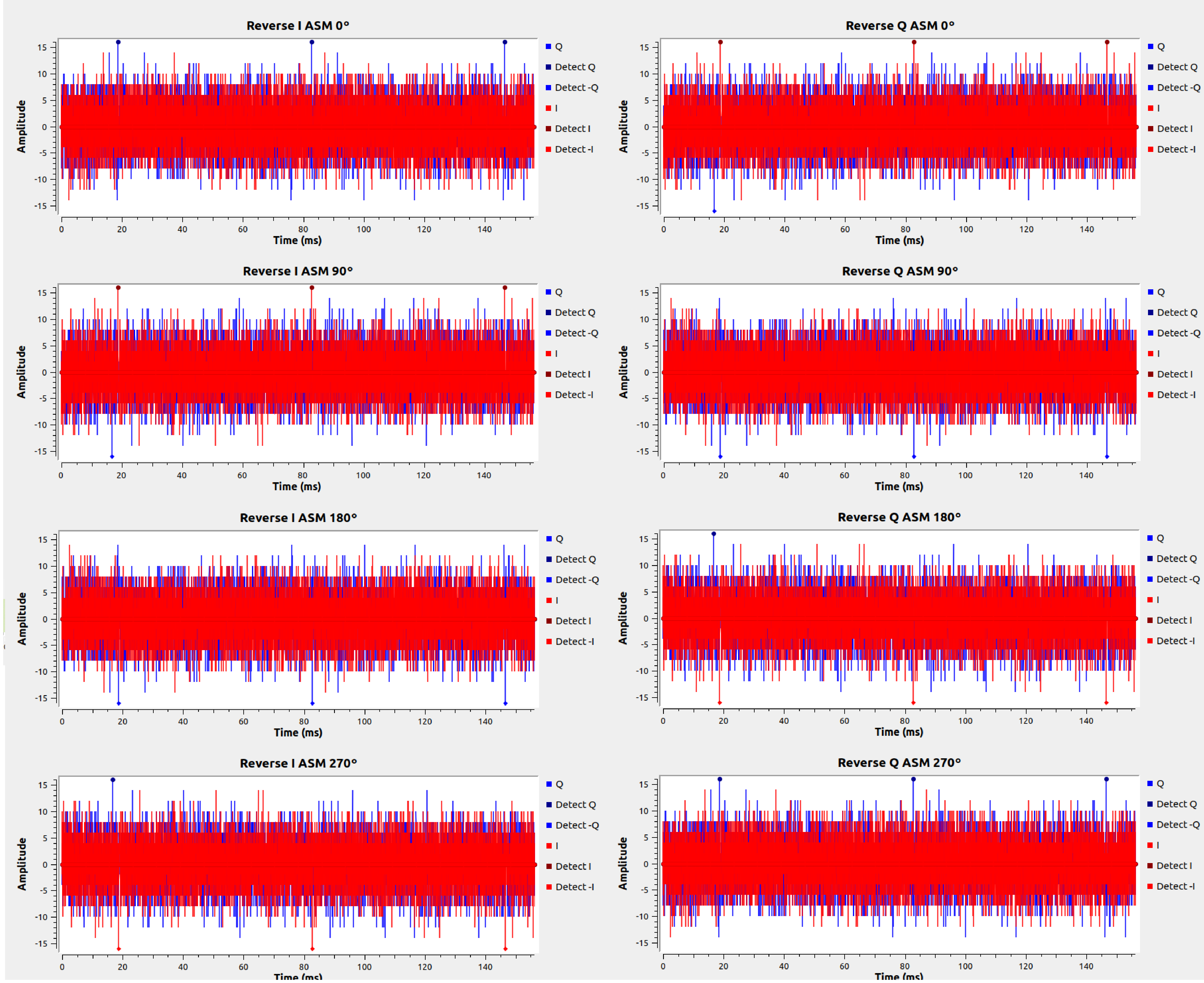


Figure 6 Detection of ASM for each Phase Reverse Rotation of a QPSK Signal

Then, it can be seen the detector work as expected outputting the next combinations:

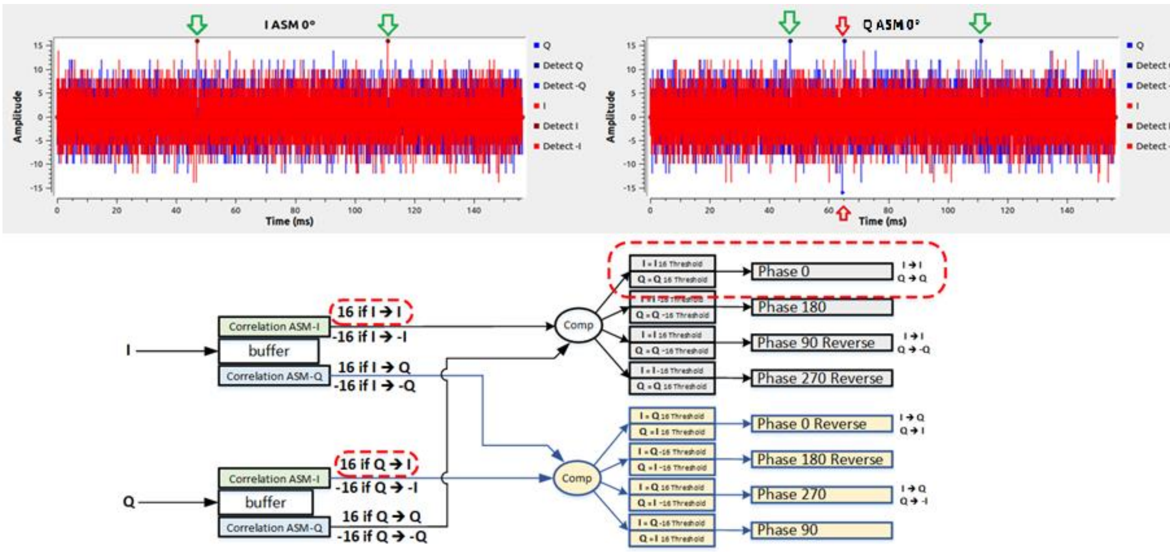


Figure 7 Detection for 0º

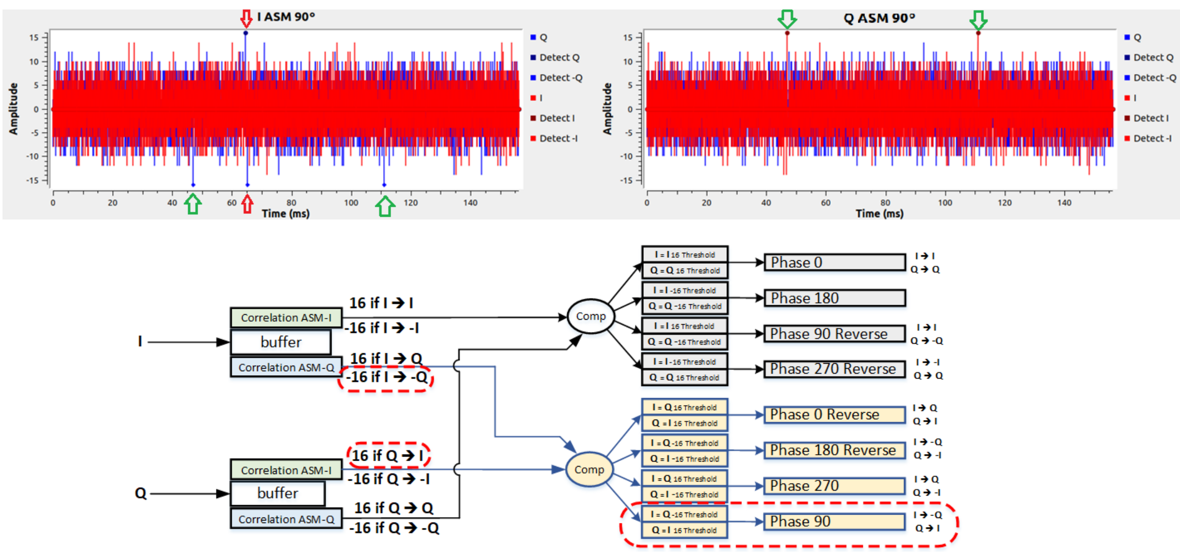


Figure 8 Detection for 90º

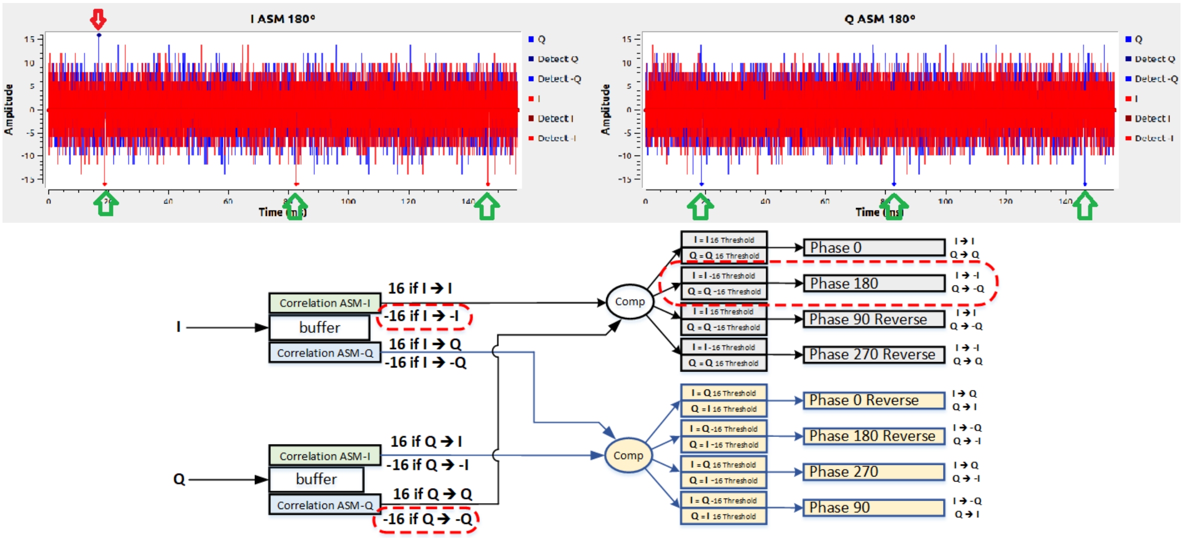


Figure 9 Detection for 180º

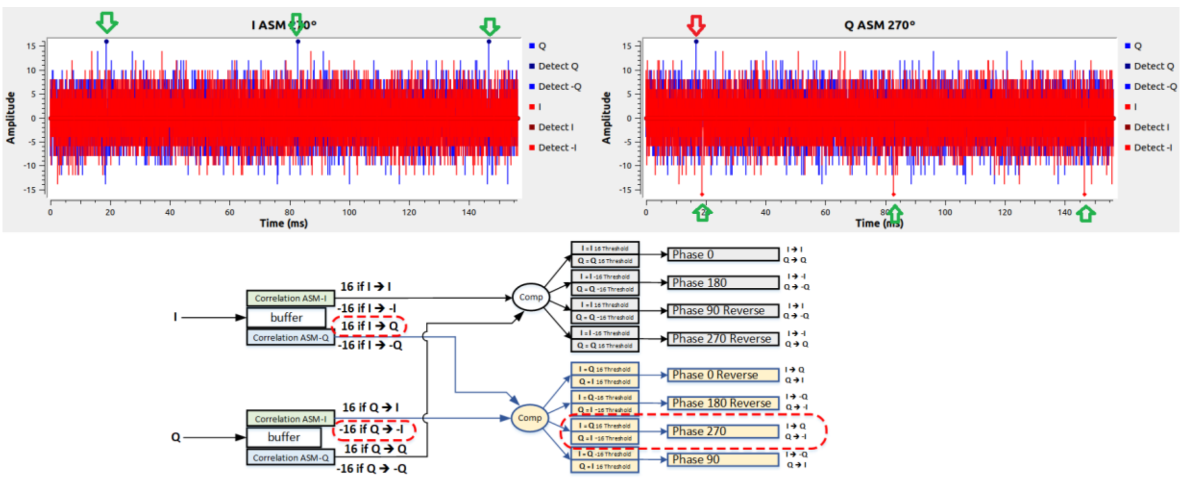


Figure 10 Detection for 270º

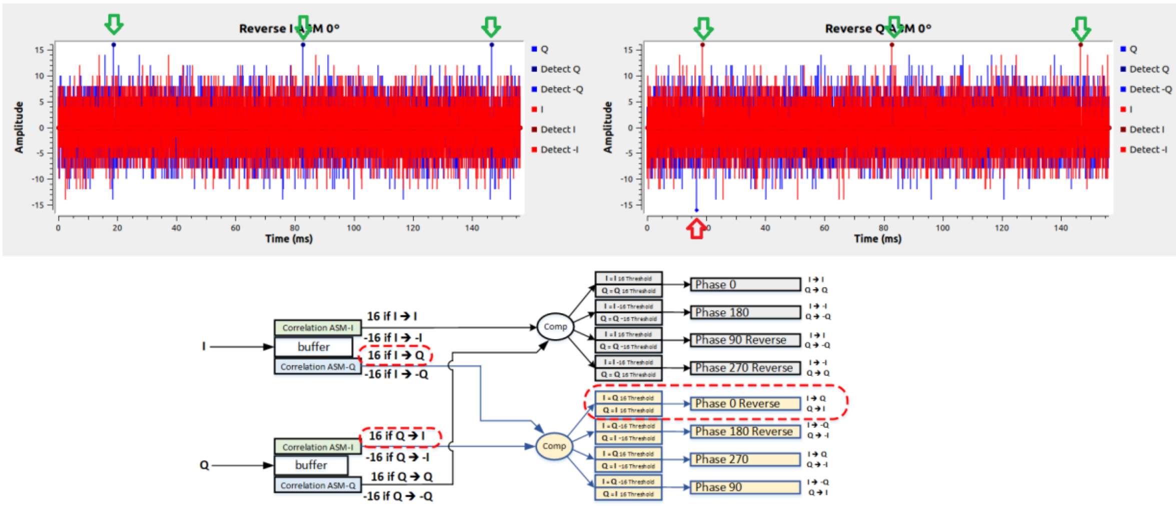


Figure 11 Detection for Reverse 0º

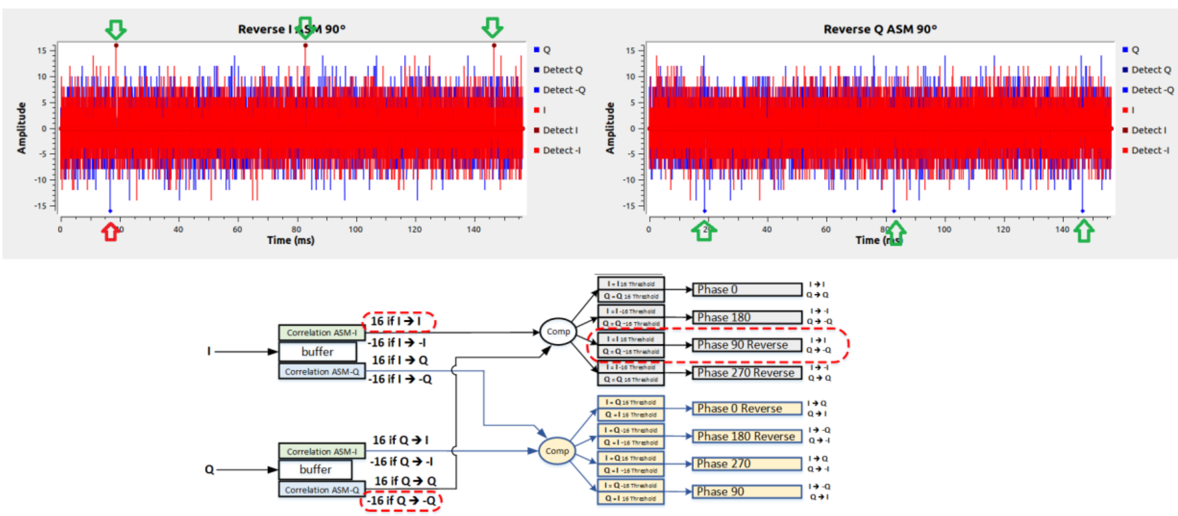


Figure 12 Detection for Reverse 90º

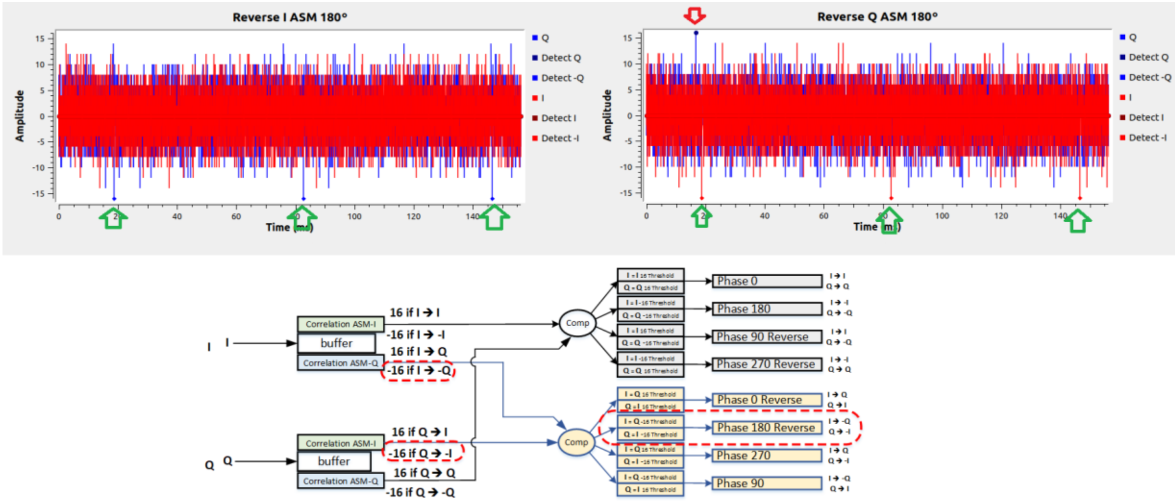


Figure 13 Detection for Reverse 180º

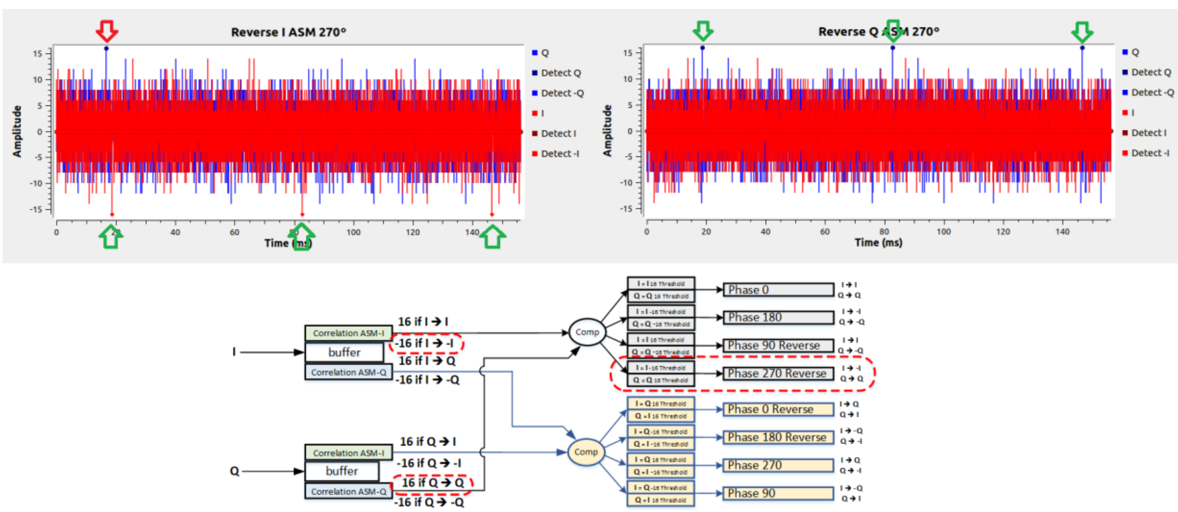
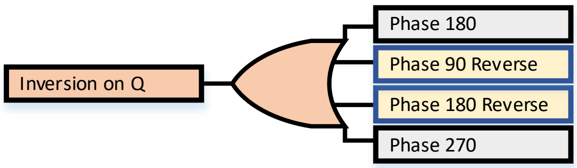


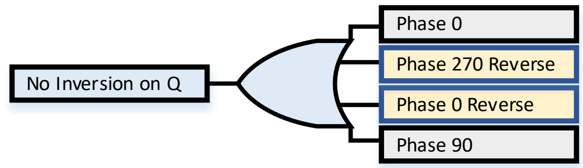
Figure 14 Detection for Reverse 270º

Using these combination we can know if Q shall be inverted or not, if I shall be inverted or not or if we need to swap I – Q using next true tables:

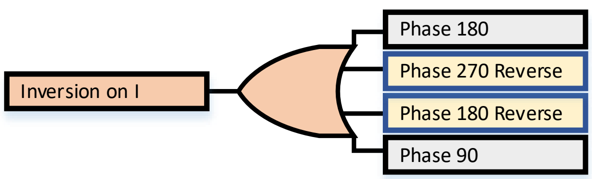
CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)			RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		
	I_R Invertion	**Swap**	**Q_R Invertion**	**I_R Invertion**	**Swap**	**Q_R Invertion**
0	No	No	No	No	Yes	No
90	Yes	Yes	No	No	No	Yes

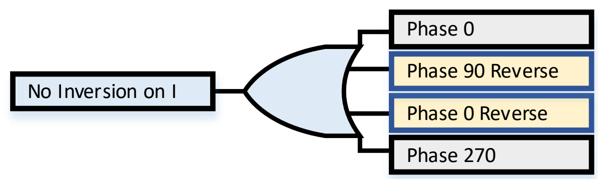
180	Yes	No	Yes	Yes	Yes	Yes
270	No	Yes	Yes	Yes	No	No



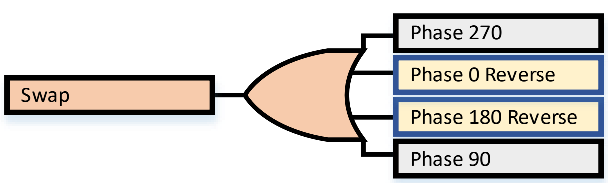


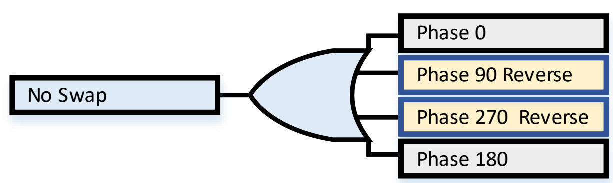
CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)			RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		
	I_R Invertion	Swap	Q_R Invertion	I_R Invertion	Swap	Q_R Invertion
0	No	No	No	No	Yes	No
90	Yes	Yes	No	No	No	Yes
180	Yes	No	Yes	Yes	Yes	Yes
270	No	Yes	Yes	Yes	No	No





CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)			RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		
	I_R Invertion	Swap	Q_R Invertion	I_R Invertion	Swap	Q_R Invertion
0	No	No	No	No	Yes	No
90	Yes	Yes	No	No	No	Yes
180	Yes	No	Yes	Yes	Yes	Yes
270	No	Yes	Yes	Yes	No	No





Testing these combination using similar test benches as before but with only one demodulator and detector and a selector to choose with QPSK modulated signal is demodulated if the 0º, 90º,180º,270º, Reverse 0º, Reverse 90º, reverse 180º or Reverse 270º

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez

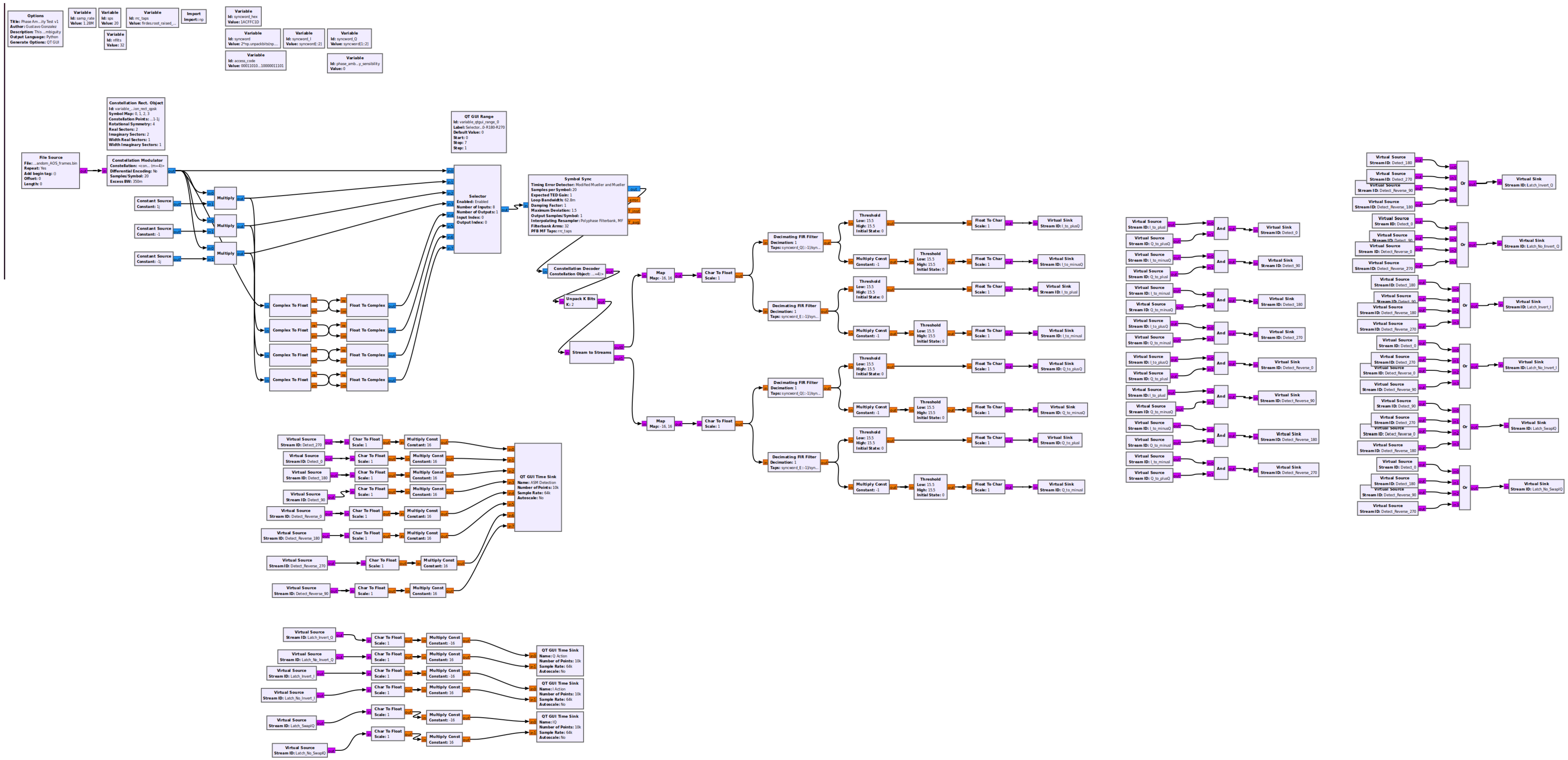


Figure 15 True Table Testing for detecting Signal Conditioning to solve Phase Ambiguity

The results are the expected one showing the right actions: detecting the ASM and requiring an inversion or not in Q, in I or an IQ swap

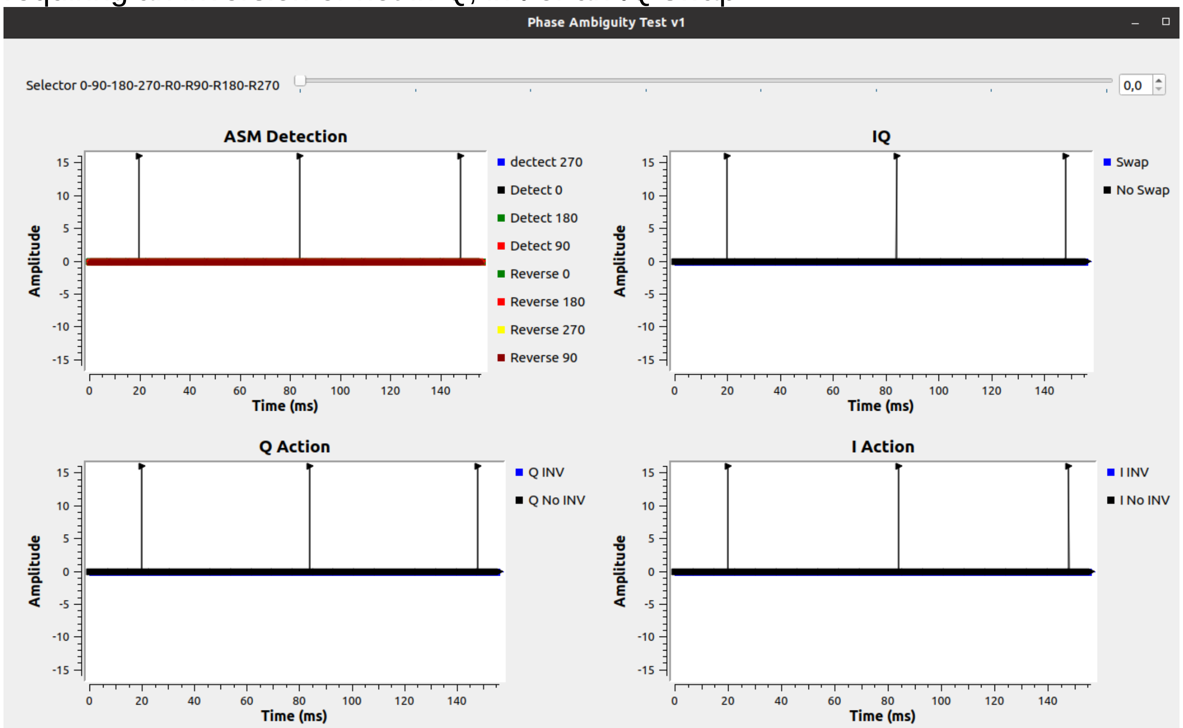


Figure 16 Detection 0º, no inversion in I, no inversion in Q, no IQ Swap required

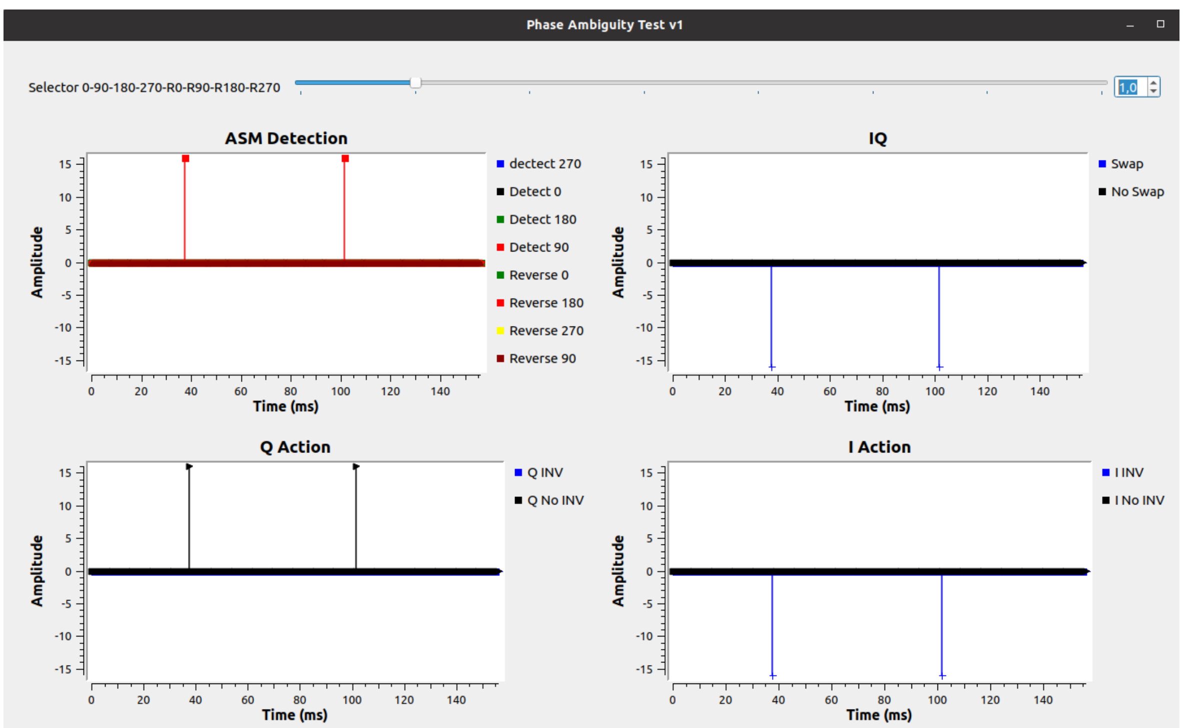


Figure 17 Detection 90º, inversion in I, no inversion in Q, IQ Swap required

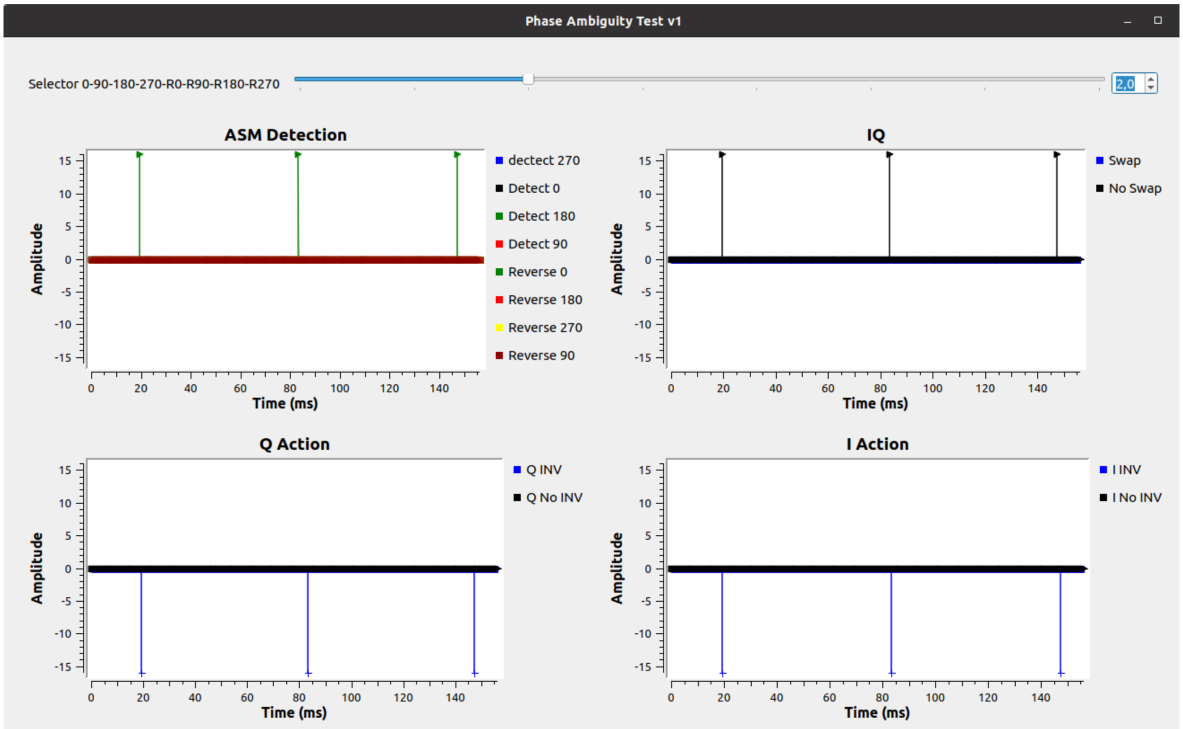


Figure 18 Detection 180º, inversion in I, inversion in Q, no IQ Swap required

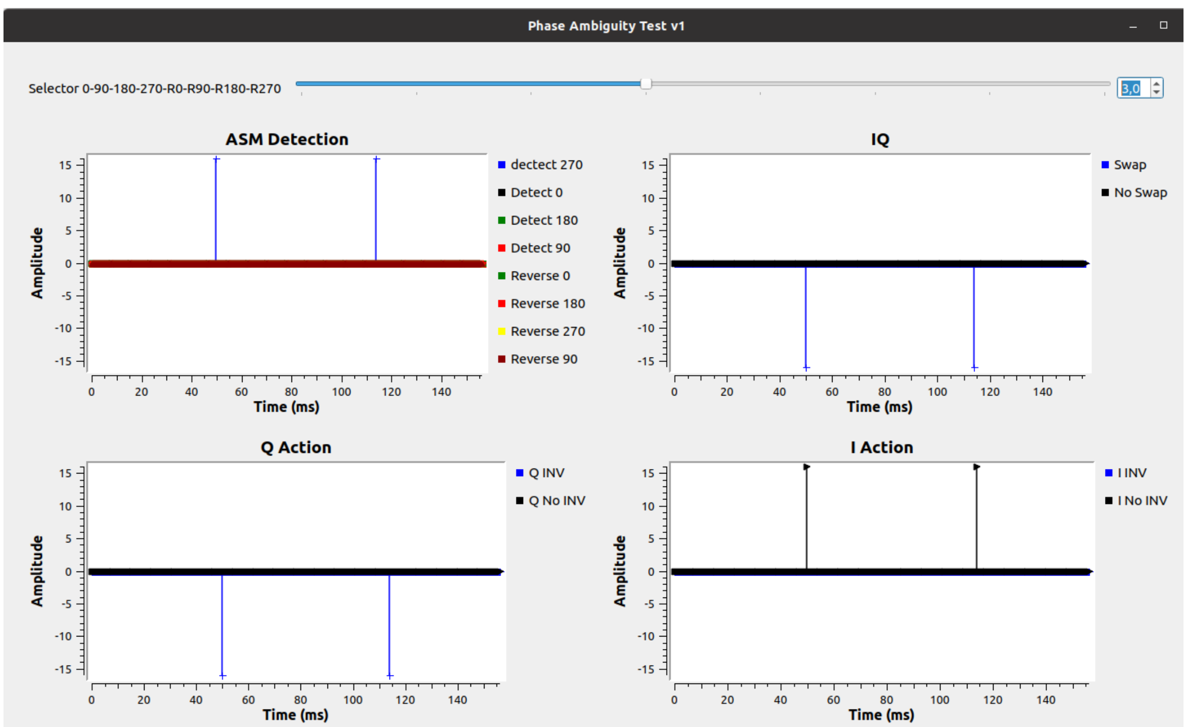


Figure 19 Detection 270º, no inversion in I, inversion in Q, IQ Swap required

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez

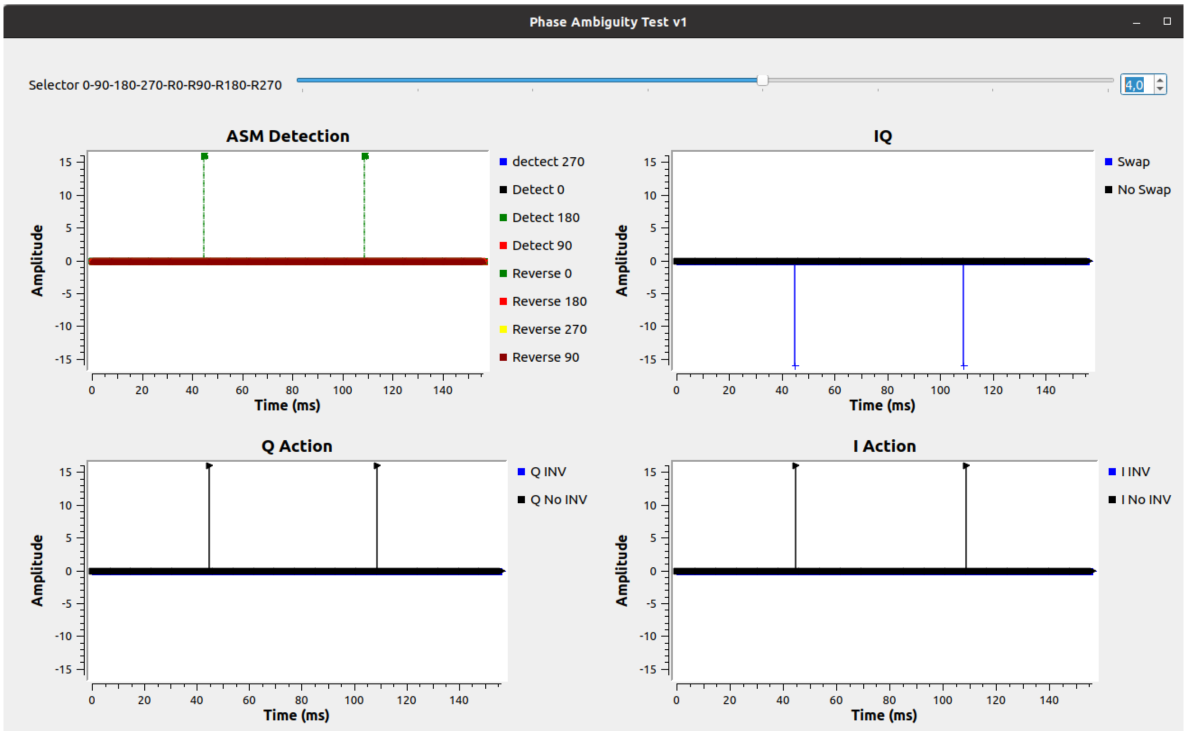


Figure 20 Detection Reverse 0º, no inversion in I, no inversion in Q, IQ Swap required

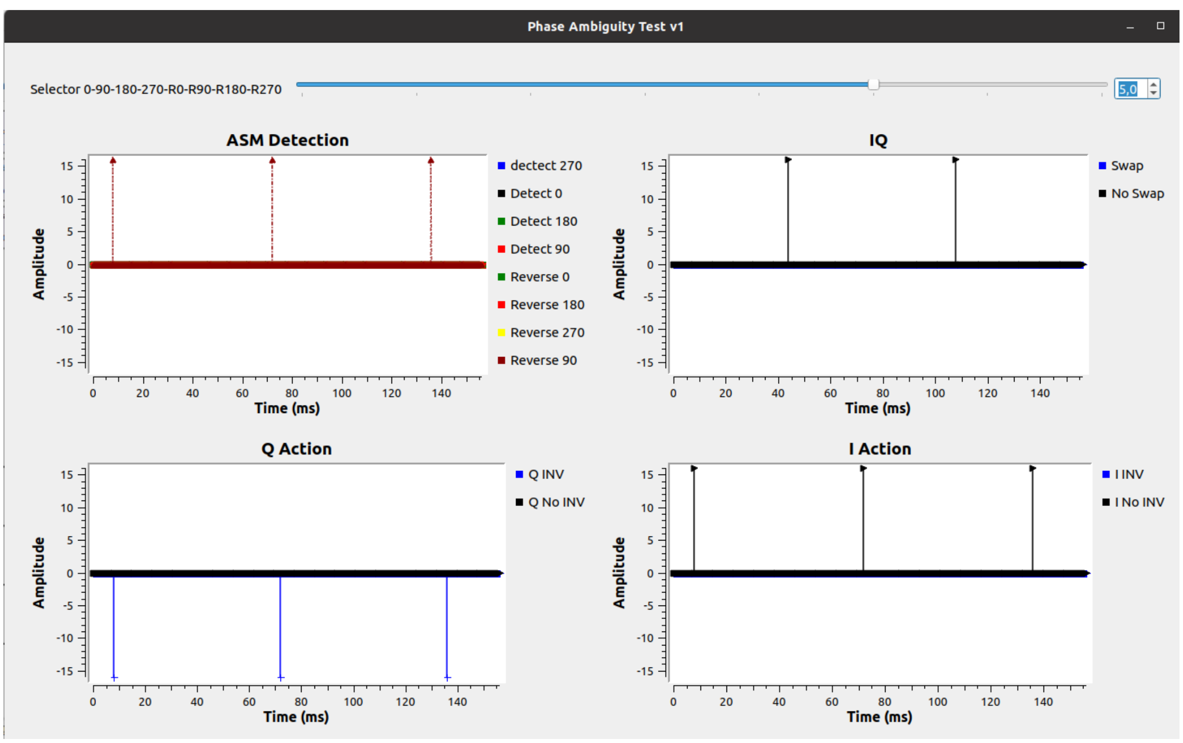


Figure 21 Detection Reverse 90º, no inversion in I, inversion in Q, no IQ Swap required

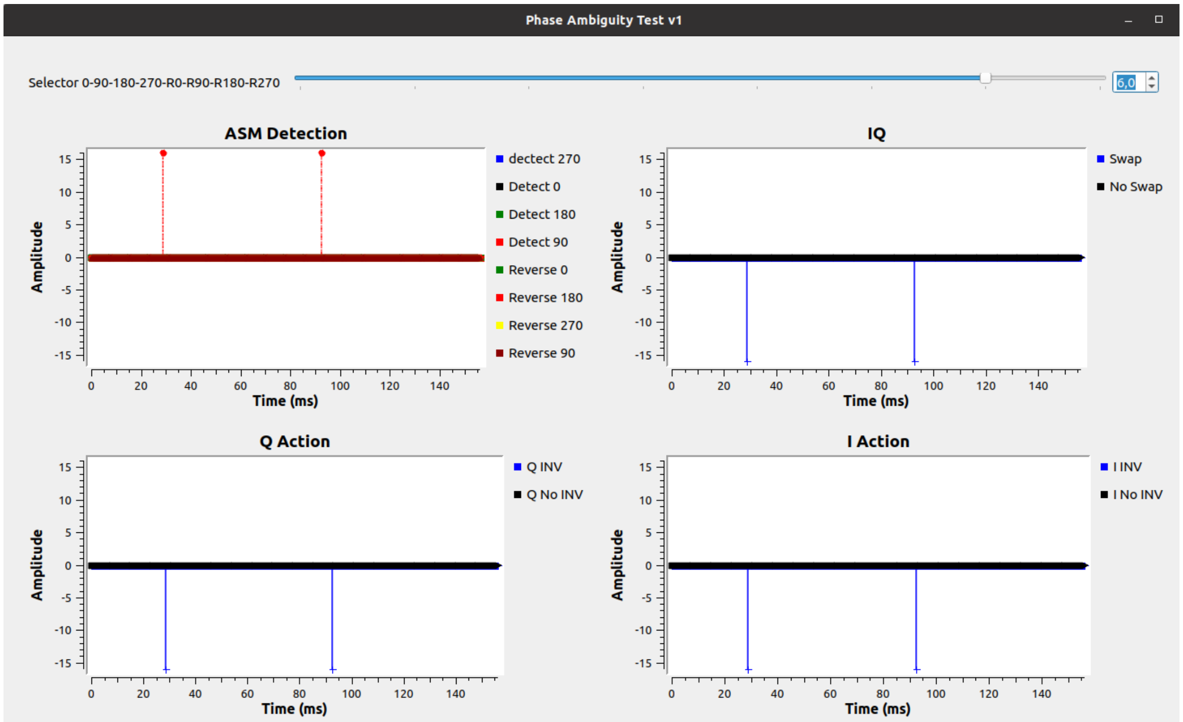


Figure 22 Detection Reverse 180º, inversion in I, inversion in Q, IQ Swap required

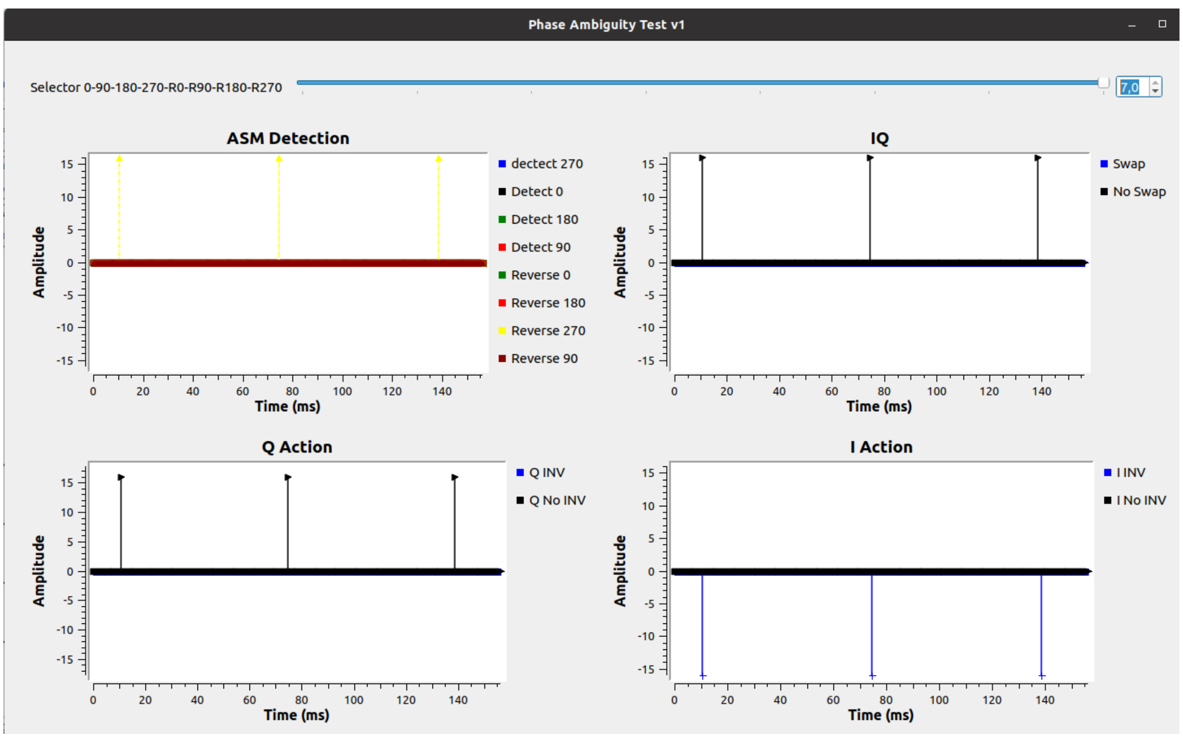


Figure 23 Detection Reverse 270º, inversion in I, no inversion in Q, no IQ Swap required

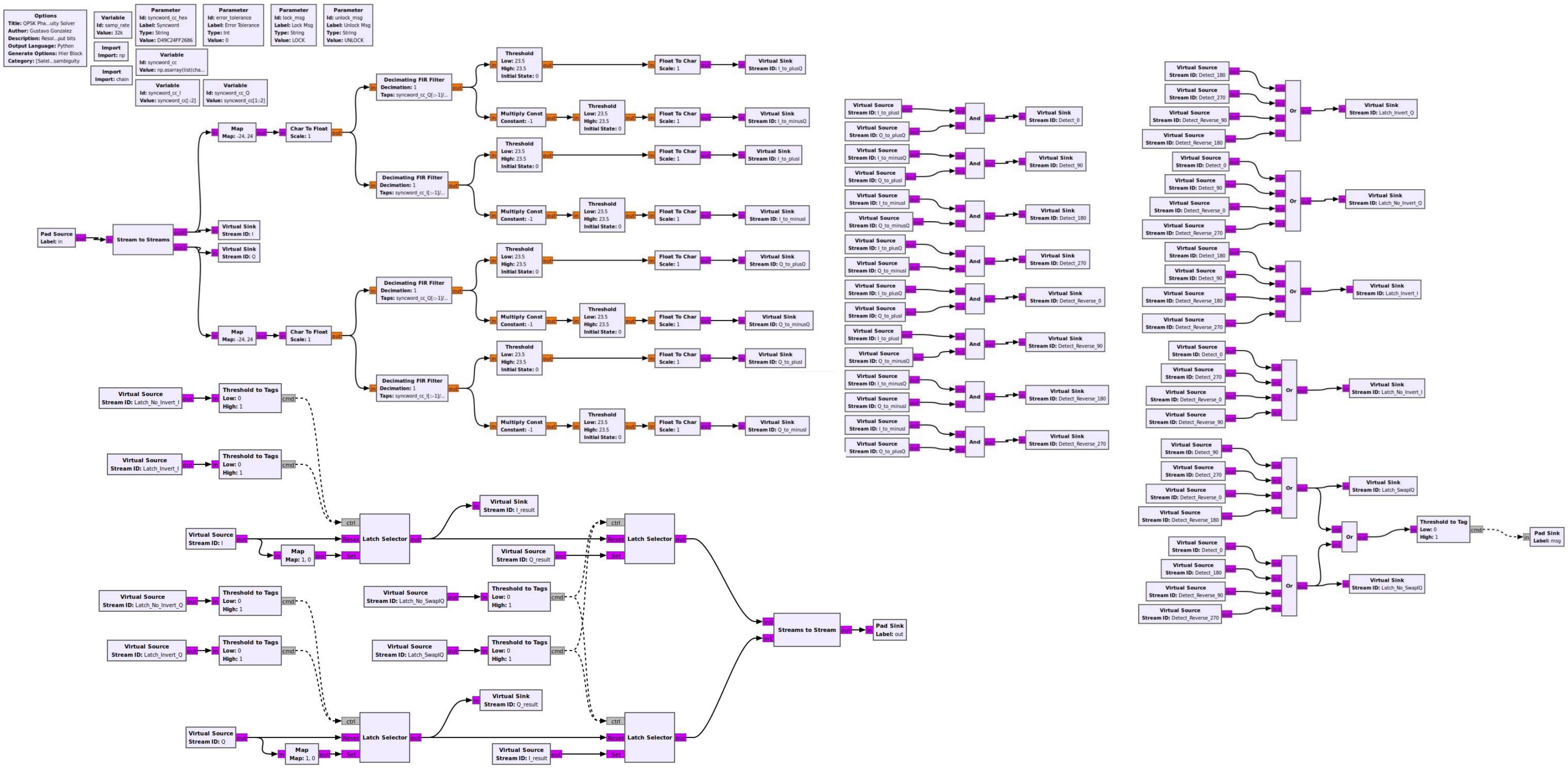
3 QPSK PHASE AMBIGUITY RESOLVER DESIGN

QPSK Phase Ambiguity Block is made with a Hier Blocks based on the previous Chapter.

The full design is in the next picture.

It will takes 4 parameters, the ASM and the Error Tolerance (indicating how many bit not matching with the ASM are allowed to be “ignored” and trigger the detection anyway) as operative parameters and the Message to send when Lock and Unlock on the ASM

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez



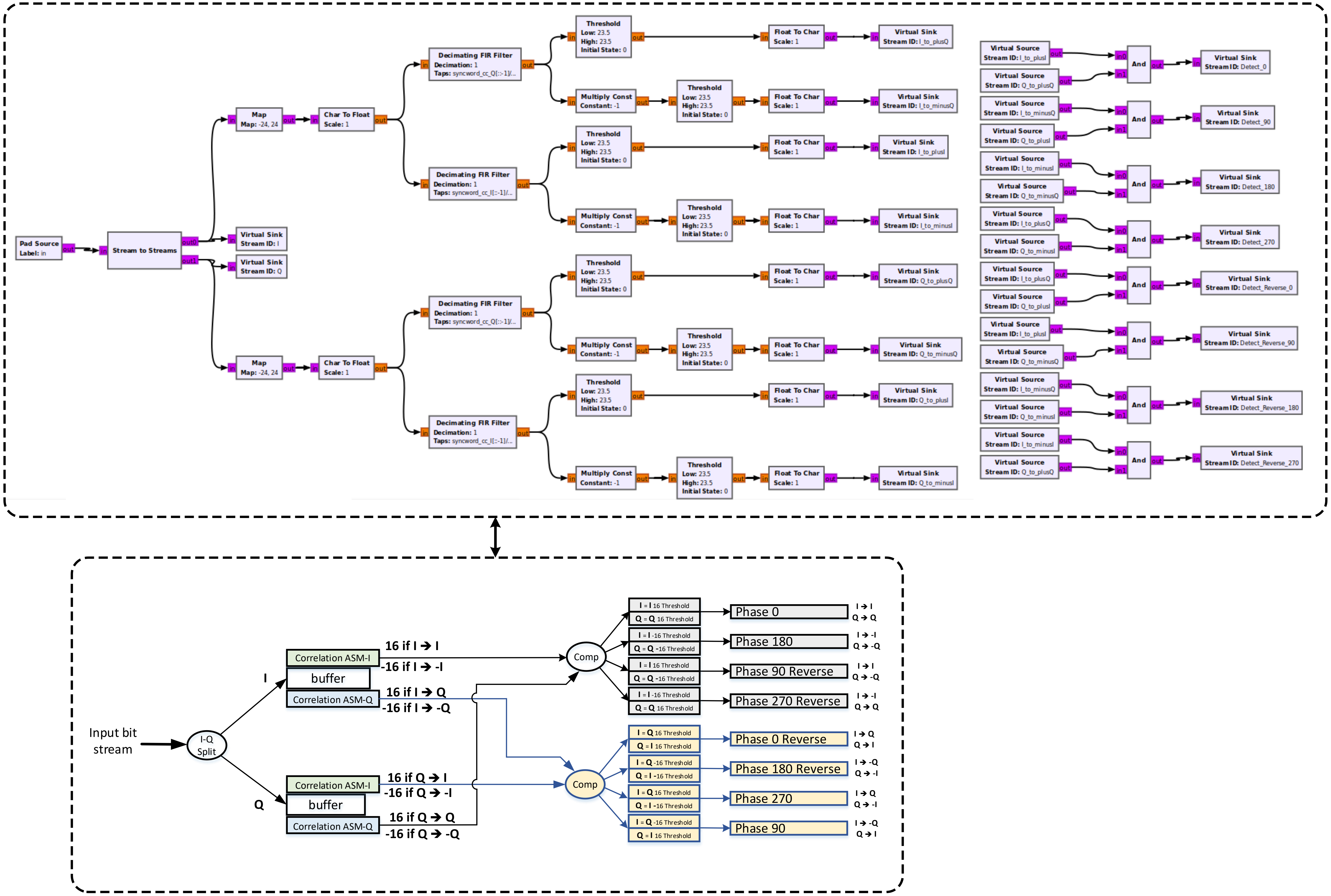
The detector section of the Phase Ambiguity Resolver has been described and explained in the previous chapter and is composed by and I-Q splitter separating the odds and even bits associated with I and Q. Each group of bit are correlated against the I or Q part of the ASM and the results of the correlation could detect if:

1) I bits match with I-ASM
2) I bits match with /I-ASM
3) I bits match with Q-ASM
4) I bits match with /Q-ASM
5) I bits not match neither with I-ASM, /I-ASM, Q-ASM nor /Q-ASM
6) Q bits match with I-ASM
7) Q bits match with /I-ASM
8) Q bits match with Q-ASM
9) Q bits match with /Q-ASM
10) Q bits not match neither with I-ASM, /I-ASM, Q-ASM nor /Q-ASM

From these possible matches, next combination produces valid Phase detection:

1) If I = I-ASM and Q = Q-ASM then ASM detection with 0º Phase
2) If I = /Q-ASM and Q = I-ASM then ASM detection with 90º Phase
3) If I = /I-ASM and Q = /Q-ASM then ASM detection with 180º Phase
4) If I = Q-ASM and Q = /I-ASM then ASM detection with 270º Phase
5) If I = Q-ASM and Q = I-ASM then ASM detection with Reverse 0º Phase
6) If I = I-ASM and Q = /Q-ASM then ASM detection with Reverse 90º Phase
7) If I = /Q-ASM and Q = /I-ASM then ASM detection with Reverse 180º Phase
8) If I = /I-ASM and Q = Q-ASM then ASM detection with Reverse 270º Phase

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez



Once the Phase is detected, then, the corrective action shall be triggered as follow:

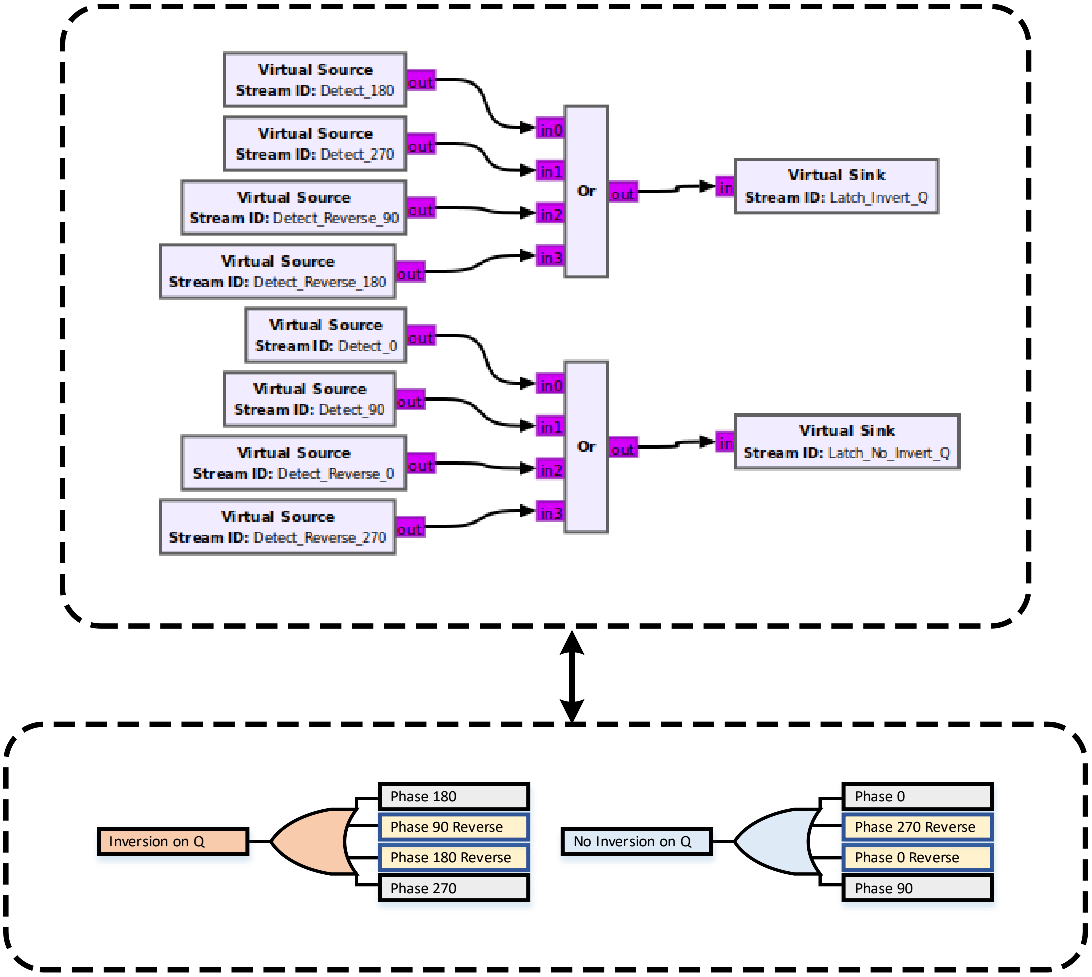
CARRIER PHASE ERROR (DEGREE)	RECEIVED DATA WITHOUT PHASE ROTATION DIRECTION AMBIGUITY (NORMAL SENSE)			RECEIVED DATA WITH PHASE ROTATION DIRECTION AMBIGUITY (REVERSE SENSE)		
	I_R Invertion	**Swap**	**Q_R Invertion**	**I_R Invertion**	**Swap**	**Q_R Invertion**
0	No	No	No	No	Yes	No
90	Yes	Yes	No	No	No	Yes
180	Yes	No	Yes	Yes	Yes	Yes
270	No	Yes	Yes	Yes	No	No

Each action combines a specific Phase detection and are implemented as follow:

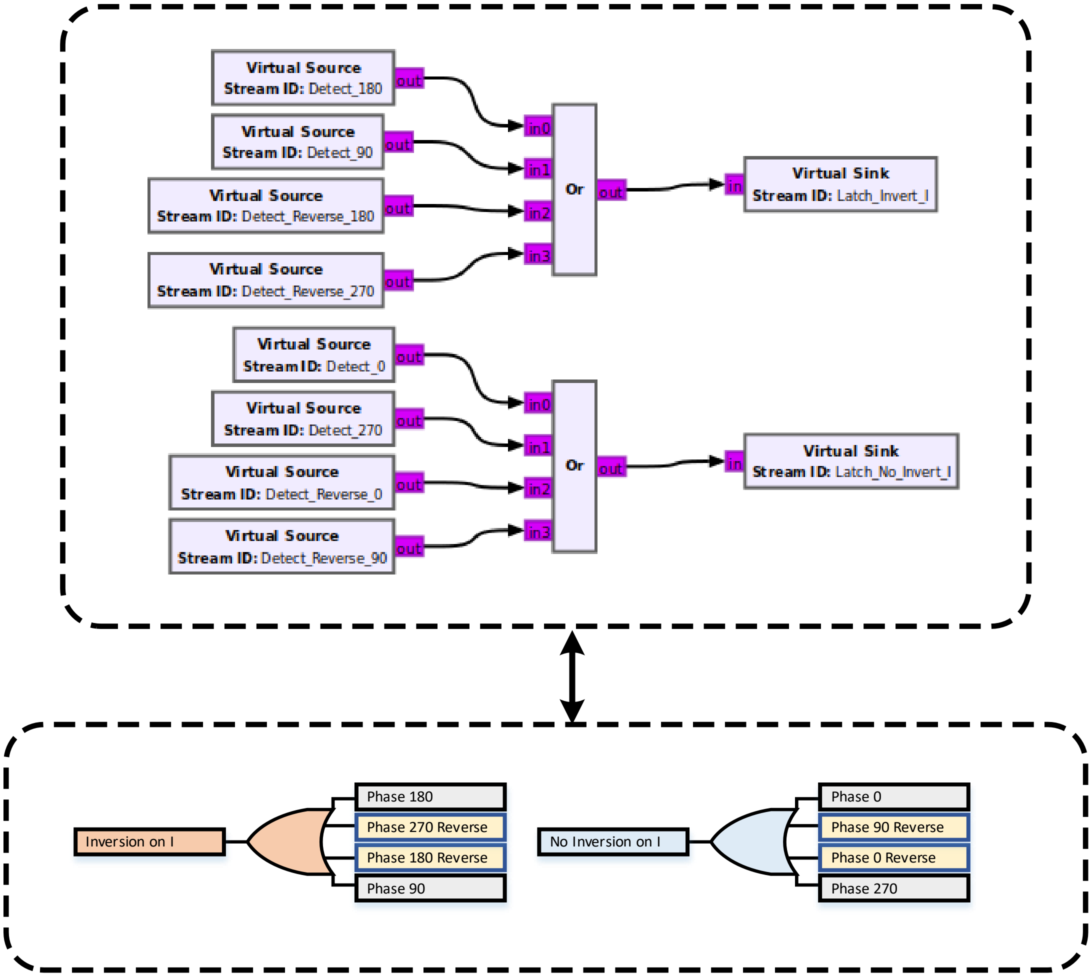
1) Q shall be NOT INVERTED when:
   a. Detect 0º Phase
   b. Detect 90º Phase
   c. Detect Reverse 0º Phase
   d. Detect Reverse 270º Phase
2) Q shall be INVERTED when:
   a. Detect 180º Phase
   b. Detect 270º Phase
   c. Detect Reverse 90º Phase
   d. Detect Reverse 180º Phase
3) I shall be NOT INVERTED when:
   a. Detect 0º Phase
   b. Detect 270º Phase
   c. Detect Reverse 0º Phase
   d. Detect Reverse 90º Phase
4) I shall be INVERTED when:
   a. Detect 90º Phase
   b. Detect 180º Phase
   c. Detect Reverse 180º Phase
   d. Detect Reverse 270º Phase
5) I-Q shall be NOT SWAPPED when:
   a. Detect 0º Phase
   b. Detect 180º Phase
   c. Detect Reverse 90º Phase
   d. Detect Reverse 270º Phase
6) I-Q shall be SWAPPED when:
   a. Detect 90º Phase
   b. Detect 270º Phase
   c. Detect Reverse 0º Phase
   d. Detect Reverse 180º Phase

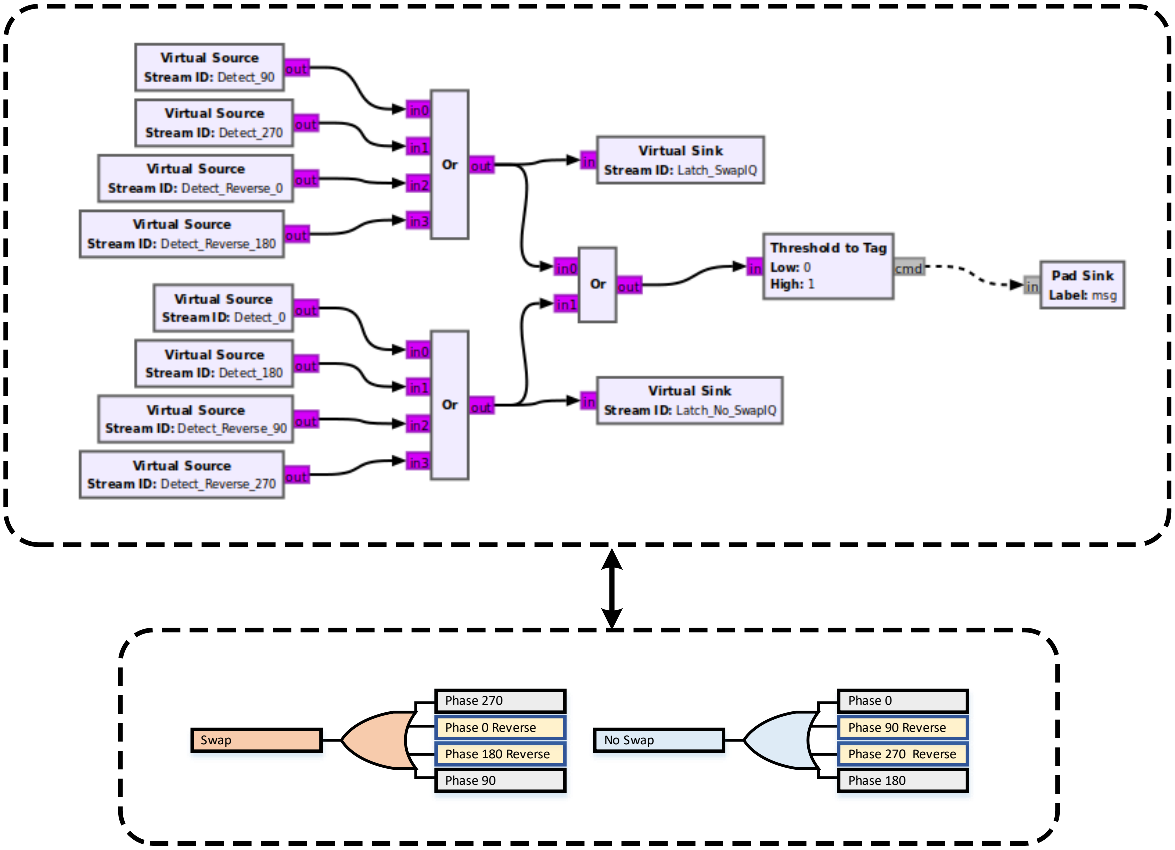
Then, the implementation is as follows:

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez

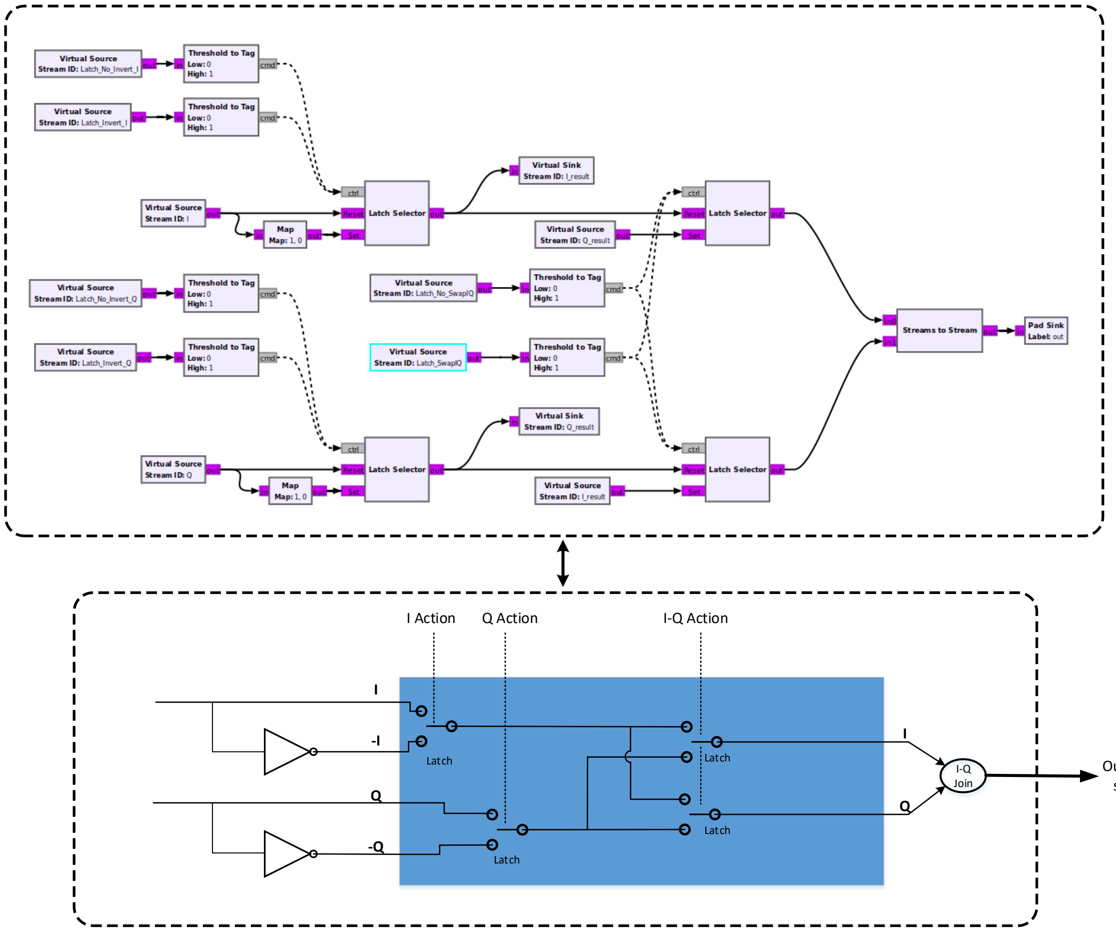


OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez





Each action shall be detected and applied to the input bits and, because each detection Action is triggered when and ASM is detected, a Latch Switch shall be used to retaint the Action until next detection. A Threshold to Tag block is used to detected the action and actuated on the Latch Selector blocks to executed each action. At the output, the I-Bit stream and the Q-Bit stream with the action applied are joined again in one output bit stream.



And additional Message output to the QPSK Phase Ambiguity Resolver block has been added: when an ASM is detected the block sends a LOCK message and when the ASM is “undetected” sends a UNLOCK message (the ASM detection duration is 1 bit, the undetected is at the next bit passing from detection to undetection, the undetected message is not sent at every bit, but once when changing from LOCK – detect - to UNLOCK – no detection)

4 BPSK PHASE AMBIGUITY RESOLVER DESIGN

BPSK Phase ambiguity Resolver is easy than the QPSK one: Only one Correlator is required, if the signal is in phase, then matching the ASM will required no inversion, and, if the signal is 180º then, the correlator will ouput the opossite polarity and the input bit stream need to be inverted to resolve the Phase Ambiguity.

As the QPSK, the BPSK has a message port outputing “LOCK” each time the ASM is detected and “UNLOCK” when lost the detection

OOT hier blocks BPSK and QPSK phase ambiguity resolver - Gustavo Gonzalez