**Εργαστήριο 1: Εισαγωγή στο SDR**

**Dennis Joosens1,\* και Maarten Weyn1**

1Ηλεκτρονική-ΤΠΕ, Σχολή Εφαρμοσμένων Μηχανικών, Πανεπιστήμιο της Αμβέρσας, Βέλγιο \*αντίστοιχα. (dennis.joosens@uantwerpen.be)

# ΑΦΗΡΗΜΈΝΟΣ

Ένα Software Defined Radio (SDR) είναι ένα ραδιόφωνο επικοινωνίας που χρησιμοποιεί λογισμικό για την υλοποίηση των αλγορίθμων που είναι απαραίτητοι για την ψηφιακή επικοινωνία. Σε αυτό το εργαστήριο, θα σχεδιάζετε και θα υλοποιείτε ένα SDR χρησιμοποιώντας υλικό Universal Software Radio Peripheral (USRP) και το λογισμικό GNU Radio.

Ο σκοπός αυτής της εισαγωγικής εργαστηριακής άσκησης είναι να εξασφαλίσει ότι οι μαθητές έχουν μια λειτουργική εγκατάσταση του GNU Radio στους υπολογιστές τους και γνωρίζουν πώς να συνδεθούν με το λογισμικό USRP defined radio.

# Εισαγωγή

Το Wireless Innovation Forum (WINNF) ορίζει το Software Defined Radio ως: *"Ραδιόφωνο στο οποίο ορισμένες ή όλες οι λειτουργίες του φυσικού επιπέδου καθορίζονται από το λογισμικό".* Το SDR αναφέρεται στην τεχνολογία όπου οι μονάδες λογισμικού που εκτελούνται σε μια γενική πλατφόρμα υλικού χρησιμοποιούνται για την υλοποίηση ραδιολειτουργιών. Συνδυάζοντας το υλικό USRP με το λογισμικό GNU Radio μπορείτε να δημιουργήσετε μια ευέλικτη και λειτουργική πλατφόρμα SDR για γρήγορη δημιουργία πρωτοτύπων ασύρματων σημάτων, συμπεριλαμβανομένου του σχεδιασμού φυσικών επιπέδων, της εγγραφής και αναπαραγωγής, της ευφυΐας σήματος, της επικύρωσης αλγορίθμων και άλλων.

## Υλικό USRP

Τα ραδιοφωνικά προϊόντα Universal Software Radio Peripheral (USRP) έχουν σχεδιαστεί για εφαρμογές ραδιοσυχνοτήτων από DC έως 6 GHz, συμπεριλαμβανομένων συστημάτων πολλαπλών κεραιών (MIMO). Παραδείγματα τομέων εφαρμογής περιλαμβάνουν λευκούς χώρους, κινητά τηλέφωνα, δημόσια ασφάλεια, παρακολούθηση φάσματος, ραδιοδικτύωση, γνωστικό ραδιόφωνο, δορυφορική πλοήγηση και ερασιτεχνικό ραδιόφωνο. Το USRP συνδέεται με έναν κεντρικό υπολογιστή δημιουργώντας ένα ραδιόφωνο καθορισμένο από λογισμικό.

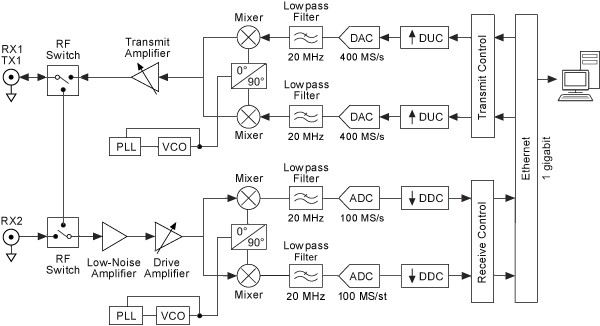
Για τη λήψη:

* Τα εισερχόμενα σήματα στις εισόδους της σύνδεσης SMA αναμιγνύονται χρησιμοποιώντας έναν δέκτη άμεσης μετατροπής σε στοιχεία εισόδου/Q βασικής ζώνης.
* Η δειγματοληψία των δεδομένων I/Q θα γίνει από έναν αναλογικό σε ψηφιακό μετατροπέα (ADC).
* Τα ψηφιοποιημένα δεδομένα I/Q ακολουθούν παράλληλες διαδρομές μέσω μιας διαδικασίας ψηφιακής μετατροπής (DDC) που αναμιγνύει, φιλτράρει και αποδεκατίζει το σήμα εισόδου σε ρυθμό καθορισμένο από το χρήστη.
* Τα δείγματα που έχουν μετατραπεί προς τα κάτω μεταβιβάζονται στον κεντρικό υπολογιστή.

Για μετάδοση:

* Τα δείγματα σήματος Baseband I/Q συντίθενται από τον κεντρικό υπολογιστή και τροφοδοτούνται στο USRP με καθορισμένο ρυθμό δειγματοληψίας μέσω USB.
* Το υλικό USRP παρεμβάλλει το εισερχόμενο σήμα σε υψηλότερο ρυθμό δειγματοληψίας χρησιμοποιώντας μια διαδικασία ψηφιακής μετατροπής (DUC).
* Στη συνέχεια, μετατρέπει το σήμα σε αναλογικό με έναν ψηφιακό σε αναλογικό μετατροπέα (DAC).
* Το προκύπτον αναλογικό σήμα στη συνέχεια αναμιγνύεται μέχρι την καθορισμένη συχνότητα φορέα και μεταδίδεται μέσω των συνδέσμων SMA.

Οι πράξεις διαβίβασης ή λήψης συνοψίζονται στο σχήμα (1).



**Εικόνα 1.** Τυπικό μπλοκ διάγραμμα ενός USRP

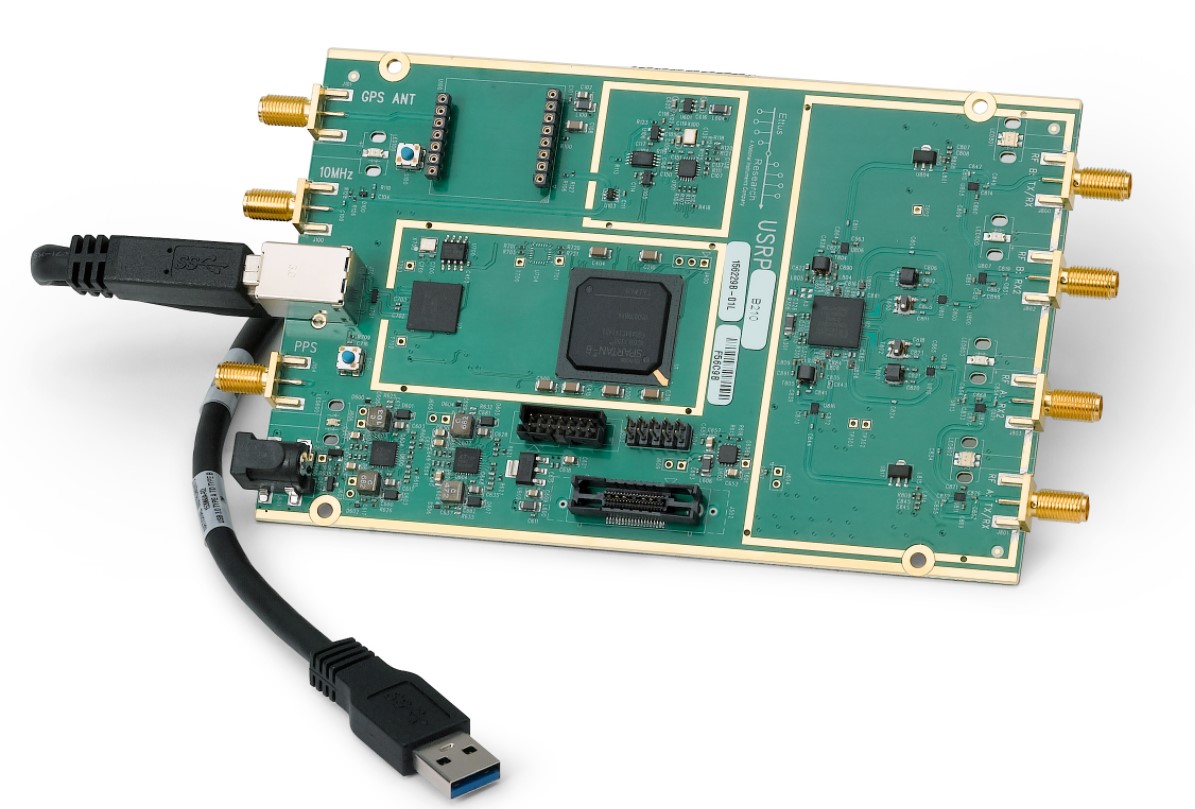
Στο εργαστήριό μας, θα χρησιμοποιήσουμε το υλικό USRP B200 και B210, δείτε τα σχήματα (2), (3) και (4). Η σειρά B(us) USRP καλύπτει συχνότητες RF από 70 MHz έως 6 GHz, διαθέτει συνδεσιμότητα Spartan-6 FPGA και USB 3.0. Αυτή η πλατφόρμα επιτρέπει τον πειραματισμό με ένα ευρύ φάσμα σημάτων, συμπεριλαμβανομένων FM και τηλεοπτικών εκπομπών, κινητής τηλεφωνίας, Wi-Fi και άλλων. Το USRP B200 διαθέτει ένα κανάλι λήψης και ένα κανάλι μετάδοσης σε σχεδιασμό που τροφοδοτείται από δίαυλο. Το USRP B210 επεκτείνει τις δυνατότητες του B200 προσφέροντας συνολικά δύο κανάλια λήψης και δύο κανάλια μετάδοσης, ενσωματώνει ένα μεγαλύτερο FPGA, GPIO και περιλαμβάνει εξωτερικό τροφοδοτικό. Και οι δύο χρησιμοποιούν ένα RFIC αναλογικών συσκευών (AD9361) για να προσφέρουν μια οικονομικά αποδοτική πλατφόρμα πειραματισμού RF και μπορούν να μεταδώσουν έως και 56 MHz στιγμιαίου εύρους ζώνης μέσω ενός διαύλου USB 3.0 υψηλού εύρους ζώνης σε επιλεγμένα chipset USB 3.0 (με συμβατότητα προς τα πίσω με USB 2.0).



**Εικόνα 2.** Μπροστινή όψη ενός USRP B210 Software Defined Radio



**Εικόνα 3.** Μπροστινή όψη ενός USRP B2x0 Software Defined Radio



**Εικόνα 4.** PCB ενός USRP B210 Software Defined Radio

## Λογισμικό ραδιοφώνου GNU

Το GNU Radio[[1]](#footnote-1)[[2]](#footnote-2) είναι μια ελεύθερη και ανοιχτού κώδικα εργαλειοθήκη ανάπτυξης λογισμικού που παρέχει μπλοκ επεξεργασίας σήματος για την υλοποίηση ραδιοφώνων λογισμικού. Μπορεί να χρησιμοποιηθεί με άμεσα διαθέσιμο εξωτερικό υλικό RF χαμηλού κόστους για τη δημιουργία ραδιοφώνων που καθορίζονται από λογισμικό ή χωρίς υλικό σε περιβάλλον προσομοίωσης. Χρησιμοποιείται ευρέως από χομπίστες, σε ακαδημαϊκά και εμπορικά περιβάλλοντα για την υποστήριξη τόσο της έρευνας ασύρματων επικοινωνιών όσο και των πραγματικών ραδιοσυστημάτων.

Το GNU Radio εκτελεί όλη την επεξεργασία σήματος. Μπορείτε να το χρησιμοποιήσετε για να γράψετε εφαρμογές για να λάβετε δεδομένα από ψηφιακές ροές ή για να προωθήσετε δεδομένα σε ψηφιακές ροές, οι οποίες στη συνέχεια μεταδίδονται χρησιμοποιώντας υλικό. Το GNU Radio έχει φίλτρα, κωδικούς καναλιών, στοιχεία συγχρονισμού, ισοσταθμιστές, αποδιαμορφωτές, αποκωδικοποιητές και πολλά άλλα στοιχεία (που ονομάζονται μπλοκ) τα οποία συνήθως βρίσκονται σε ραδιοσυστήματα. Το πιο σημαντικό, περιλαμβάνει μια μέθοδο σύνδεσης αυτών των μπλοκ και στη συνέχεια διαχειρίζεται τον τρόπο με τον οποίο τα δεδομένα μεταφέρονται από το ένα μπλοκ στο άλλο. Η επέκταση του GNU Radio είναι επίσης αρκετά εύκολη. Εάν βρείτε ένα συγκεκριμένο μπλοκ που λείπει, μπορείτε γρήγορα να το δημιουργήσετε και να το προσθέσετε. Αυτά είναι λειτουργικά στοιχεία κλήσης εκτός δέντρου και μπορούν να κατασκευαστούν γράφοντας κώδικα Python.

Δεδομένου ότι το GNU Radio είναι λογισμικό, μπορεί να χειριστεί μόνο ψηφιακά δεδομένα. Συνήθως, τα σύνθετα δείγματα βασικής ζώνης είναι ο τύπος δεδομένων εισόδου για δέκτες και ο τύπος δεδομένων εξόδου για πομπούς. Στη συνέχεια, χρησιμοποιείται αναλογικό υλικό για τη μετατόπιση του σήματος στην επιθυμητή κεντρική συχνότητα. Πέρα από αυτή την απαίτηση, οποιοσδήποτε τύπος δεδομένων μπορεί να περάσει από το ένα μπλοκ στο άλλο - είτε πρόκειται για bit, byte, διανύσματα, ριπές ή πιο σύνθετους τύπους δεδομένων.

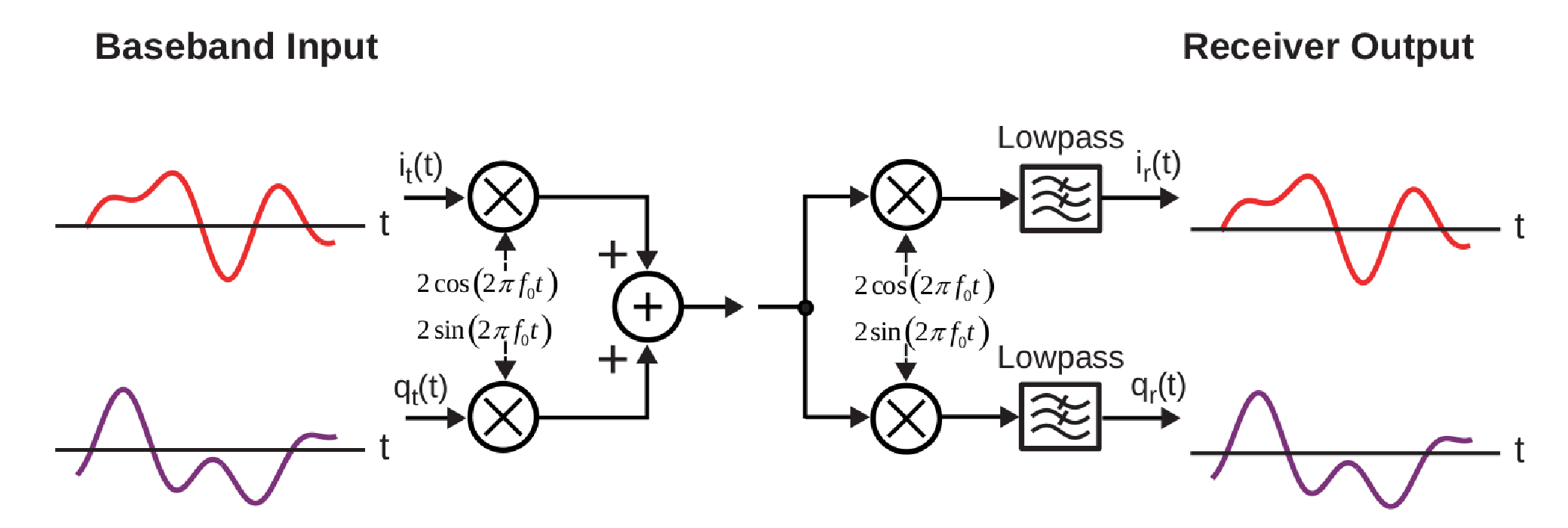
Οι εφαρμογές GNU Radio γράφονται κυρίως χρησιμοποιώντας τη γλώσσα προγραμματισμού Python, ενώ η παρεχόμενη, κρίσιμη για την απόδοση διαδρομή επεξεργασίας σήματος υλοποιείται σε C++ χρησιμοποιώντας επεκτάσεις κινητής υποδιαστολής επεξεργαστή, όπου είναι διαθέσιμες. Έτσι, ο προγραμματιστής είναι σε θέση να υλοποιήσει ραδιοσυστήματα υψηλής απόδοσης σε πραγματικό χρόνο σε ένα απλό στη χρήση, γρήγορο περιβάλλον ανάπτυξης εφαρμογών.

Το GNU Radio είναι ένα πλαίσιο που επιτρέπει στους χρήστες να σχεδιάζουν, να προσομοιώνουν και να αναπτύσσουν εξαιρετικά ικανά ραδιοσυστήματα πραγματικού κόσμου. Πρόκειται για ένα εξαιρετικά αρθρωτό πλαίσιο προσανατολισμένο στο "διάγραμμα ροής" που συνοδεύεται από μια ολοκληρωμένη βιβλιοθήκη μπλοκ επεξεργασίας που μπορούν εύκολα να συνδυαστούν για να κάνουν πολύπλοκες εφαρμογές επεξεργασίας σήματος.



# Δεδομένα IQ

Τα δεδομένα in-phase/quadrature (I/Q) είναι μια σύνθετη αναπαράσταση του μεταδιδόμενου σήματος, για το οποίο μπορούν όχι μόνο να δώσουν πληροφορίες σχετικά με το πλάτος των σημάτων αλλά και τις πληροφορίες φάσης. Τα δεδομένα IQ μπορούν να αναπαρασταθούν ως δύο κανάλια δεδομένων που είναι ορθογώνια μεταξύ τους, όπως φαίνεται στο σχήμα (5).



**Εικόνα 5.** Κανάλια In-phase και Quadrature

Τα δύο κανάλια στην πλευρά του πομπού μπορούν να γραφτούν ως εξής:

*i*(*t*) = *it*(*t*)cos(2π *fot*)

(1) *q*(*t*) = *qt*(*t*)sin(2π *fot*) που οδηγεί σε,

*y*(*t*) = *αυτό*(*t*)Μαρούλι(2π *Φωτ*)+qt(*t*)αμαρτία(2π *Φωτ*) (2)

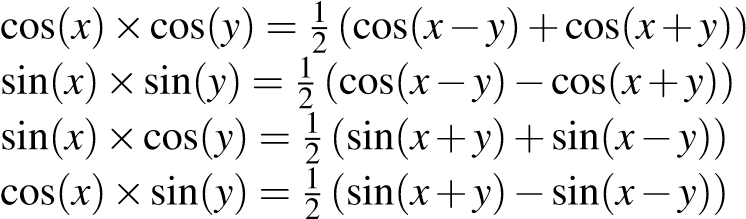
στην πλευρά του δέκτη τα δύο κανάλια I και Q μπορούν να ανακτηθούν χρησιμοποιώντας το μαθηματικό γινόμενο προς άθροισμα

ταυτότητα. [[3]](#footnote-3)

*ir*(*t*) = [*it*(*t*)cos(2π *fot*)+qt(*t*)sin(2π *fot*)]2cos(2π *fot*)

|  |  |
| --- | --- |
| = [*it*(*t*)+it(*t*)cos(4π *fot*)]+[*qt*(*t*)sin(4π *fot*)+qt(*t*)sin(0)]  = [*it*(*t*)+it(*t*)cos(4π *fot*)]+[*qt*(*t*)sin(4π *fot*)]  *qr*(*t*) = [*it*(*t*)cos(2π *fot*)+qt(*t*)sin(2π *fot*)]2sin(2π *fot*) | (3) |
| = [*it*(*t*)sin(4π *fot*)−it(*t*)sin(0)]+[*qt*(*t*)−qt(*t*)cos(4π *fot*)]  = [*it*(*t*)sin(4π *fot*)]+[*qt*(*t*)−qt(*t*)cos(4π *fot*)] | (4) |

Χρησιμοποιώντας ένα φίλτρο χαμηλής διέλευσης, τα κανάλια I και Q μπορούν να ανακτηθούν στο δέκτη.



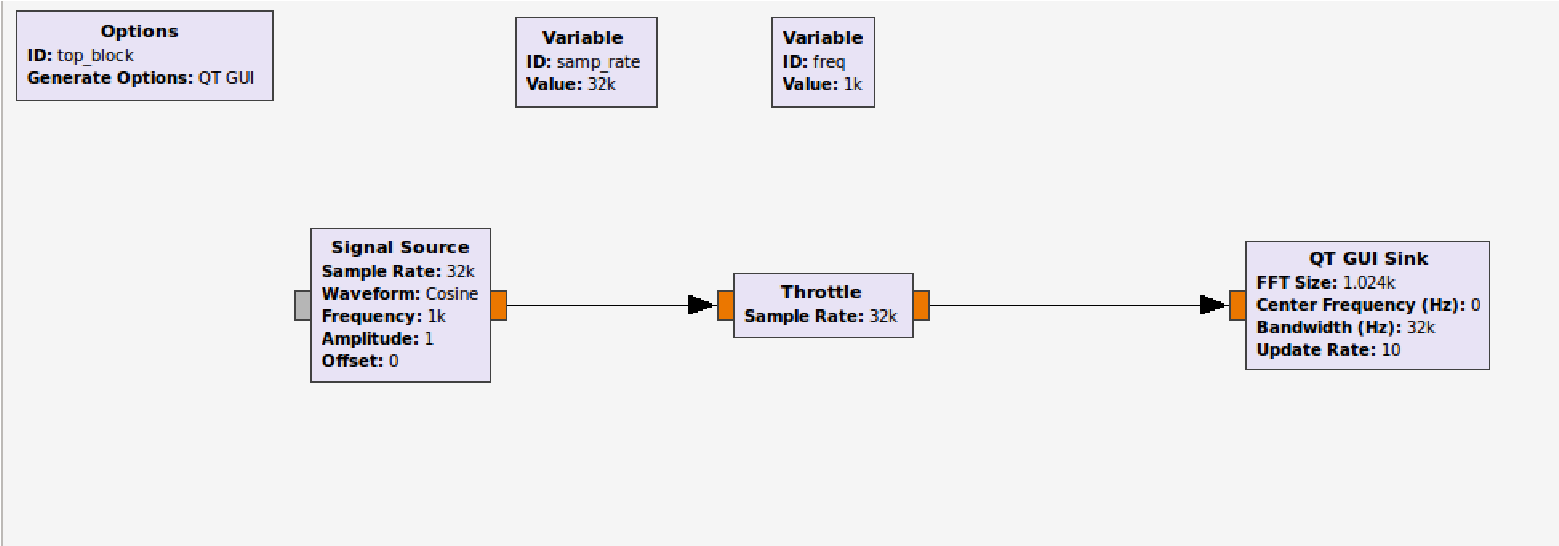
# Εργαστηριακή Διαδικασία

1. Δημιουργία διαγράμματος ροής

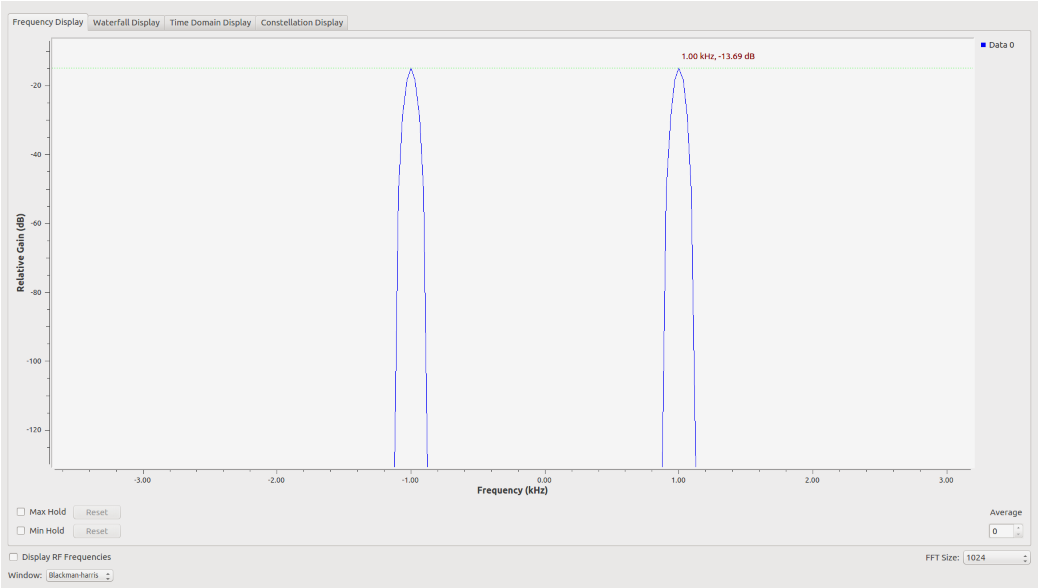
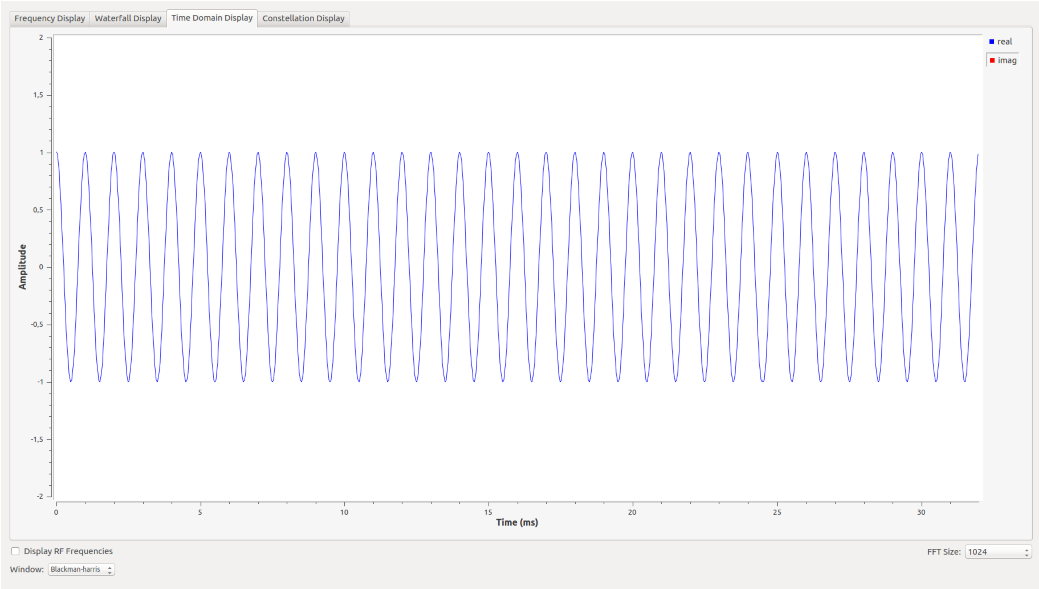
Κατά τη διάρκεια αυτής της άσκησης θα δημιουργήσετε ένα συνημιτονοειδές κύμα χρησιμοποιώντας το GNU Radio.

* + - Ανοίξτε ένα νέο διάγραμμα ροής στο GNU Radio.
    - Δύο πλαίσια θα βρίσκονται στο διάγραμμα ροής, πρώτο πλαίσιο "επιλογές", το οποίο χρησιμοποιείται για τον ορισμό καθολικών παραμέτρων για το διάγραμμα ροής. Το δεύτερο είναι "μεταβλητό", το οποίο μπορείτε να του δώσετε ένα όνομα και να το χρησιμοποιήσετε στο διάγραμμα ροής σας (προεπιλεγμένη μεταβλητή είναι ο ρυθμός δειγματοληψίας[[4]](#footnote-4)).
    - Ορίστε τις επιλογές δημιουργίας σε QT GUI.
    - Προσθέστε ένα άλλο *μπλοκ μεταβλητής* , ονομάστε το freq και ορίστε το σε 1000.
    - Προσθέστε μια *πηγή σήματος* και ορίστε τον τύπο εξόδου σε κυμαινόμενους αριθμούς, κυματομορφή σε συνημίτονο, συχνότητα σε freq, πλάτος σε 1 και τέλος μετατόπιση σε 0.
    - Προσθέστε ένα *γκάζι* και ορίστε το Type σε Float.
    - Προσθέστε έναν *νεροχύτη QT GUI* και ορίστε το "Type" σε Float και το "FFT Size" σε 1024.
    - Αλλάξτε τη συχνότητα της πηγής σήματος από 1k σε 2.5k, 5k, 16k και 100k και παρατηρήστε το πεδίο χρόνου και συχνότητας.
    - Ερώτηση 1: Ποια είναι η διαφορά και γιατί;

Τελικά, θα πρέπει να έχετε ένα διάγραμμα ροής όπως φαίνεται στο Σχήμα(6) και όταν εκτελέσετε το διάγραμμα ροής θα λάβετε ένα αποτέλεσμα όπως φαίνεται στο Σχήμα (7).



**Εικόνα 6.** πηγή σήματος GRC



A)

B)

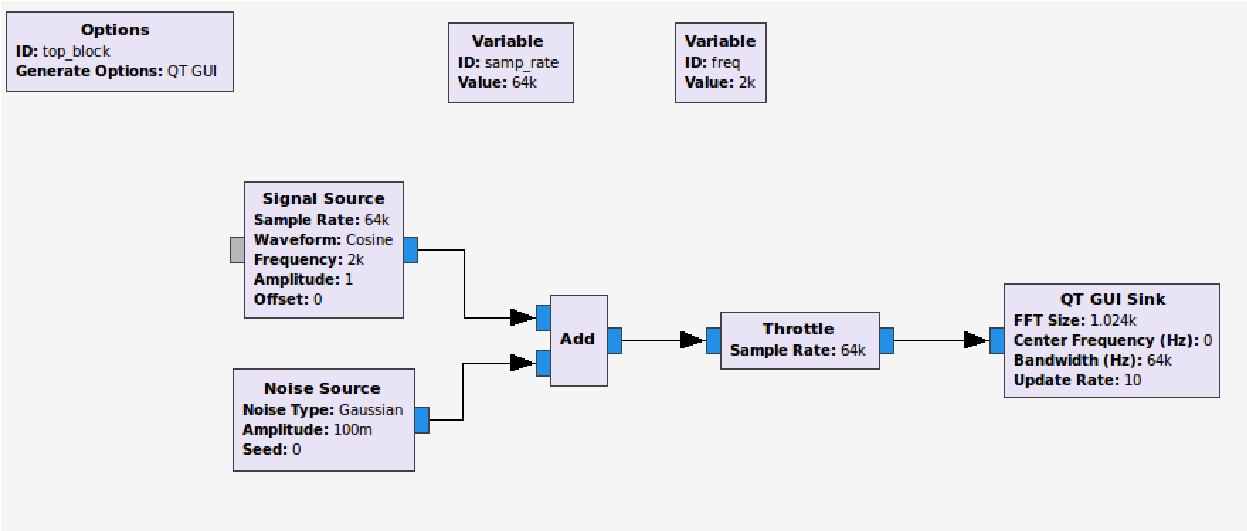
**Εικόνα 7.** Α) Το πεδίο συχνοτήτων του σήματος, Β) Το χρονικό πεδίο του σήματος

1. Σύνθετο σήμα με θόρυβο

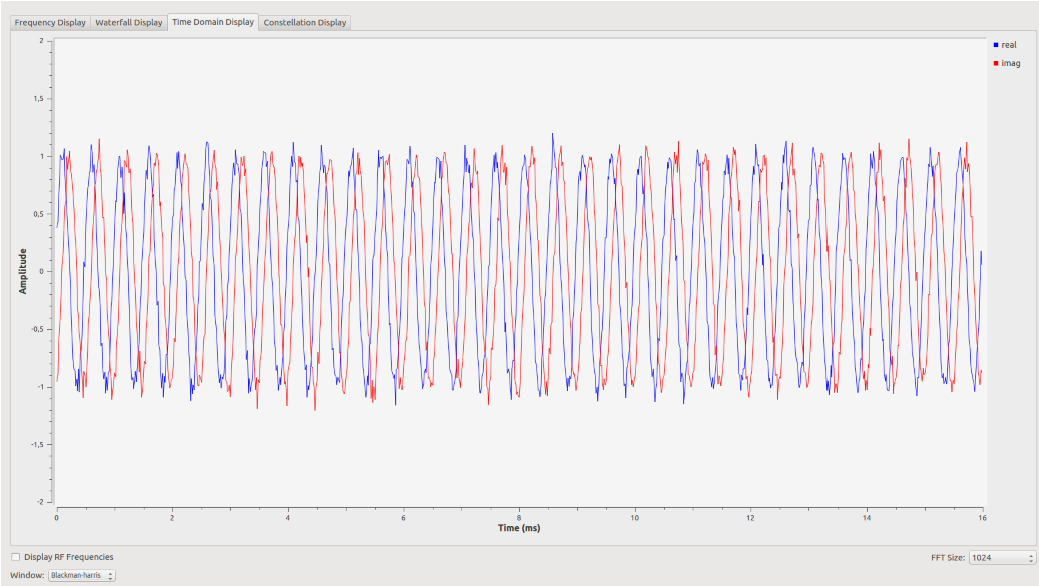
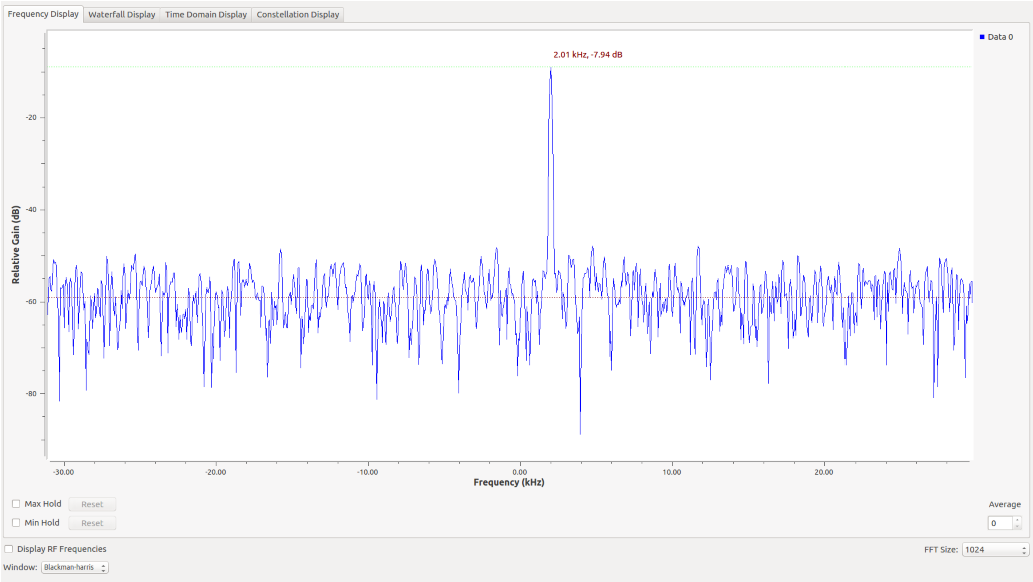
Σε αυτή την άσκηση, θα δημιουργήσετε ένα πολύπλοκο κύμα συνημιτόνου μολυσμένο με θόρυβο.

* + Ανοίξτε ένα νέο διάγραμμα ροής στο GNU Radio.
  + Ορίστε τις επιλογές δημιουργίας σε QT GUI.
  + Ορίστε τη μεταβλητή ρυθμού δειγματοληψίας σε 64000.
  + Προσθέστε ένα άλλο *μπλοκ μεταβλητής, ονομάστε το freq και ορίστε το σε 2000.*
  + Προσθέστε μια *πηγή σήματος* και ορίστε τον τύπο εξόδου σε μιγαδικούς αριθμούς, κυματομορφή σε συνημίτονο, συχνότητα σε freq, πλάτος σε 1 και τέλος μετατόπιση σε 0.
  + Προσθέστε μια *πηγή θορύβου* και ορίστε τον τύπο εξόδου σε μιγαδικούς αριθμούς, τον τύπο θορύβου σε Gaussian και το πλάτος σε 0.1.
  + Προσθέστε ένα *μπλοκ προσθήκης.*
  + Προσθέστε ένα  *μπλοκ γκαζιού* και ορίστε τον τύπο σε σύνθετο.
  + Προσθέστε έναν *νεροχύτη QT GUI* και ορίστε τον τύπο σε σύνθετο και το μέγεθος FFT σε 1024.
  + Ερώτηση 2: Πόσες κορυφές συχνότητας μπορείτε να δείτε και γιατί;
  + Ερώτηση 3: Πώς μπορείτε να μετριάσετε το φαινόμενο του θορύβου;

Στο τέλος θα πρέπει να έχετε ένα διάγραμμα ροής όπως φαίνεται στο Σχήμα(8) και όταν εκτελέσετε το διάγραμμα ροής θα λάβετε ένα αποτέλεσμα όπως φαίνεται στο Σχήμα (9).



**Εικόνα 8.** Σύνθετο σήμα με θόρυβο GRC



B)

A)

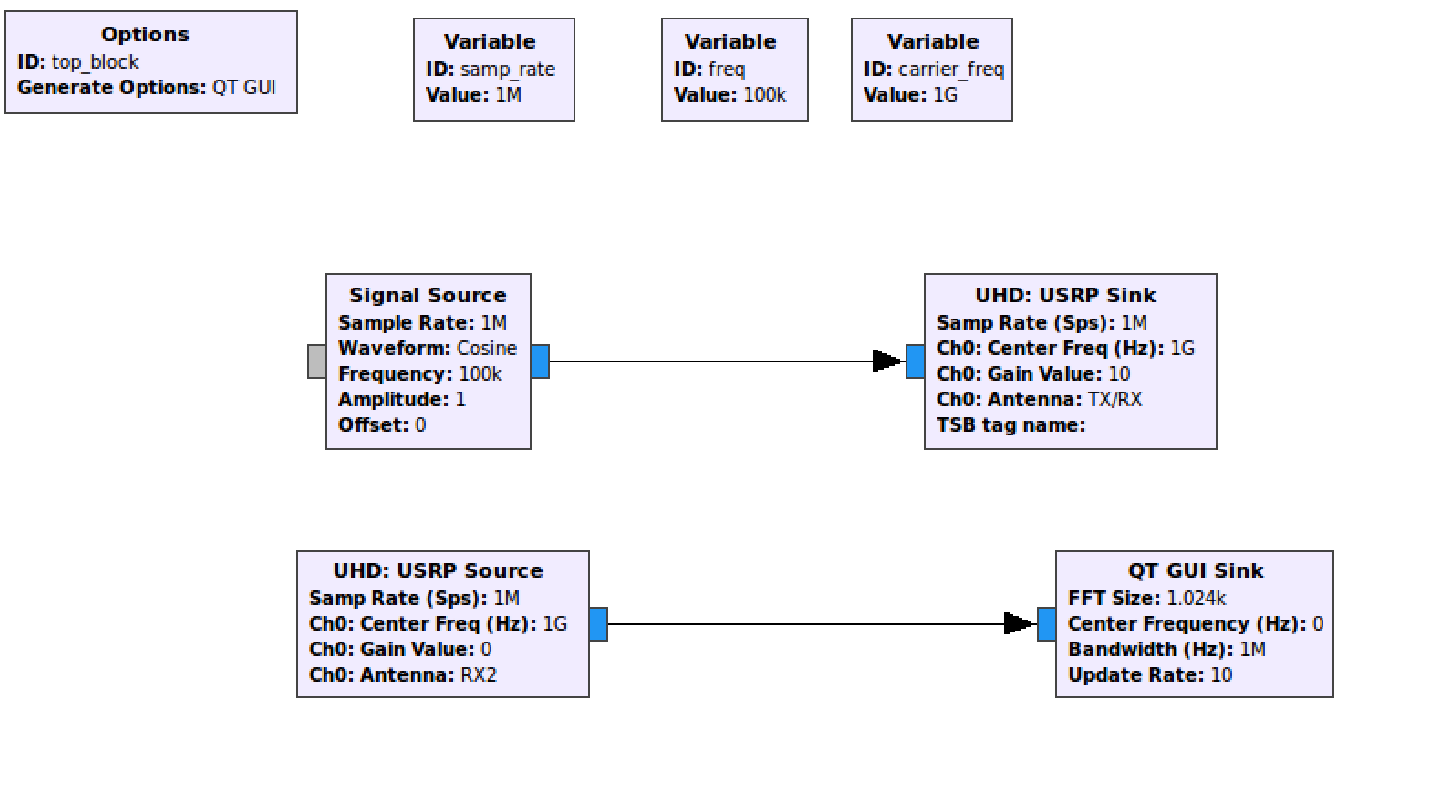
**Εικόνα 9.** Α) Το πεδίο συχνοτήτων του σήματος, Β) Το χρονικό πεδίο του σήματος

1. Σύνδεση του USRP

Αυτή η άσκηση είναι να μεταδώσει και να λάβει ένα σύνθετο κύμα συνημιτόνου χρησιμοποιώντας το USRP.

* + Ανοίξτε ένα νέο διάγραμμα ροής στο GNU Radio.
  + Ορίστε τις επιλογές δημιουργίας σε QT GUI.
  + Ορίστε τη μεταβλητή ρυθμού δειγματοληψίας σε 1e6.
  + Προσθέστε ένα άλλο *μπλοκ μεταβλητής* , ονομάστε το freq και ορίστε το σε 20e3.
  + Προσθέστε ένα άλλο *μπλοκ μεταβλητής* , ονομάστε το freq φορέα και ορίστε το σε 1e9.
  + Προσθέστε μια *πηγή σήματος* και ορίστε τον τύπο εξόδου σε μιγαδικούς αριθμούς, κυματομορφή σε συνημίτονο, συχνότητα σε freq, πλάτος σε 1 και τέλος μετατόπιση σε 0.
  + Προσθέστε έναν *νεροχύτη UHD: USRP* και ορίστε τον Τύπο εισόδου σε σύνθετο float32, κεντράρετε το Freq στο freq φορέα και το Gain Value σε 1.
  + Προσθέστε ένα *UHD: USRP Source* και ορίστε το Output Type σε complex float32, center Freq to carrier freq και Gain Value σε 0.
  + Προσθέστε *το QT GUI Sink* και ορίστε το Type σε Complex και το FFT Size σε 1024.
  + Προσθέστε ένα φίλτρο για να λαμβάνετε το σωστό σήμα με λιγότερο θόρυβο.
  + Ερώτηση 4: Ποιο φίλτρο χρησιμοποιείτε και ποιες είναι οι παράμετροι που θα επιλέξετε;

Στο τέλος θα πρέπει να έχετε ένα διάγραμμα ροής όπως φαίνεται στην εικόνα(10).



**Εικόνα 10.** Πηγή σήματος GRC

1. https://wiki.gnuradio.org [↑](#footnote-ref-1)
2. https://www.gnuradio.org/ [↑](#footnote-ref-2)
3. Χρήση των μαθηματικών ταυτοτήτων γινομένου προς άθροισμα [↑](#footnote-ref-3)
4. Σκεφτείτε ότι ο ρυθμός δειγματοληψίας χρησιμοποιείται για τον υπολογισμό του διακριτού μεγέθους βημάτων από το ένα δείγμα στο επόμενο μέσα σε μια λειτουργία DSP. Ο ρυθμός δειγματοληψίας αναφέρεται επίσης στο ρυθμό με τον οποίο τα δείγματα διέρχονται από το διάγραμμα ροής. Εάν δεν υπάρχει έλεγχος ρυθμού, ρολόι υλικού ή μηχανισμός στραγγαλισμού, τα δείγματα θα παραχθούν, θα περάσουν από το διάγραμμα ροής και θα καταναλωθούν όσο το δυνατόν γρηγορότερα (δηλαδή το διάγραμμα ροής θα είναι δεσμευμένο από την CPU). Μόνο ένα μπλοκ που αντιπροσωπεύει κάποιο υποκείμενο υλικό με το δικό του ρολόι (π.χ. USRP, κάρτα ήχου) ή το μπλοκ γκαζιού, θα χρησιμοποιήσει το "Sample Rate" για να ρυθμίσει αυτό το ρολόι υλικού και επομένως θα έχει ως αποτέλεσμα την εφαρμογή ελέγχου ρυθμού στα δείγματα στο διάγραμμα ροής. [↑](#footnote-ref-4)