

# Εργασία Τηλεπικοινωνιακά Συστήματα 2022-2023

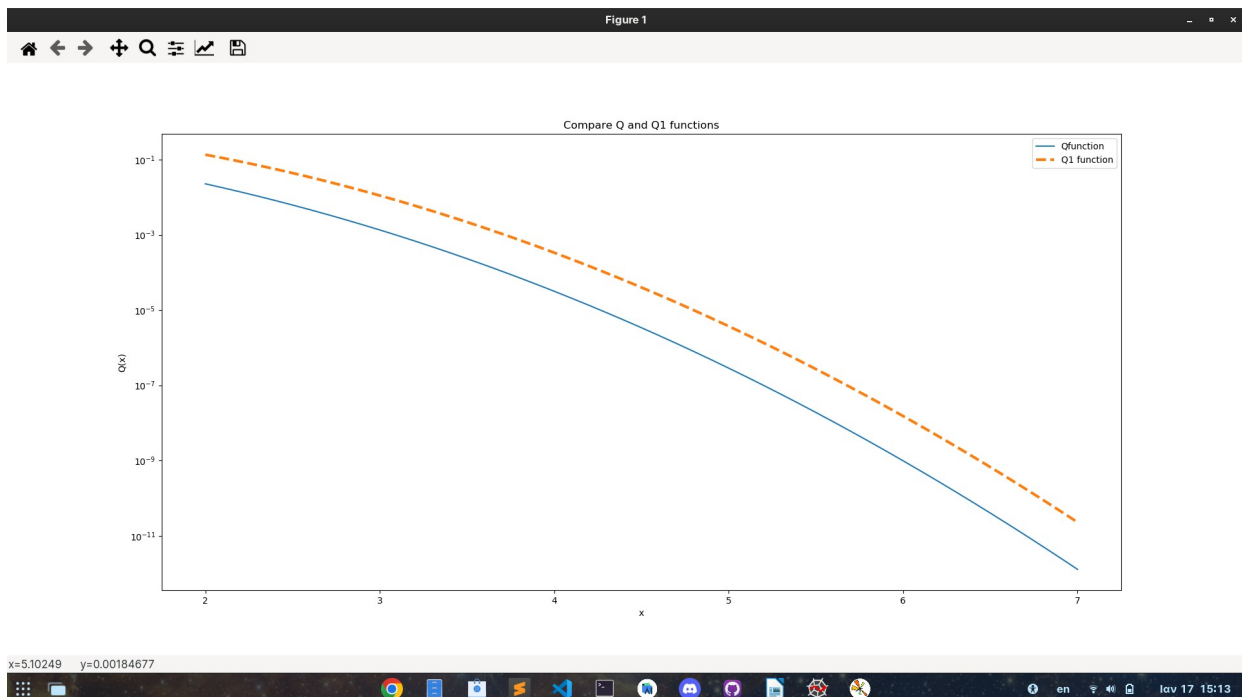
Καραπιέρης Νικόλαος Ιωάννης (ΑΜ: 22042)

## Μέρος 1ο: η συνάρτηση Q

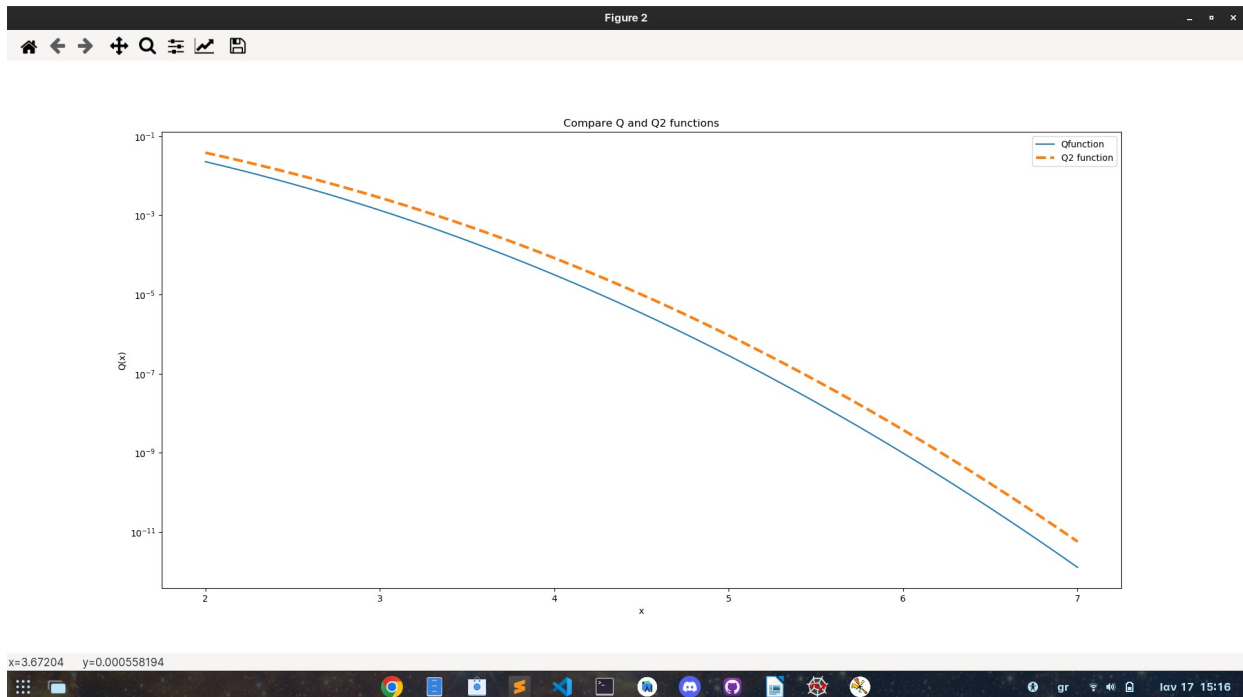
A) Αρχικά, έκανα `import` τις βιβλιοθήκες `numpy`, `matplotlib.pyplot`, `scipy` και όρισα την συνάρτηση **Qfunction**, όπως την έχουμε δει στα πλαίσια του μαθήματος. Στην συνέχεια, μέσω της συνάρτησης **linspace()**, της βιβλιοθήκης `numpy`, έθεσα τιμές στην μεταβλητή `X` από 2 έως 7, όπως και ζητάει η εκφώνηση και έθεσα 1000 σημεία. Έπειτα, δημιούργησα τις συναρτήσεις που έχουν προταθεί κατά καιρούς στην βιβλιογραφία ως προσεγγιστικές της **Qfunction**. Για την δημιουργία αυτών των συναρτήσεων, χρησιμοποίησα την συνάρτηση **exp()**, της βιβλιοθήκης `numpy`, η οποία επιστρέφει το εκθετικό.

Στην συνέχεια, μέσω της συνάρτησης **close()**, της βιβλιοθήκης `matplotlib.pyplot` και δίνοντας ως όρισμα το **'all'**, κλείνουμε πρώτα όλα τα προηγούμε **plot**. Έπειτα, με την βοήθεια της βιβλιοθήκης `matplotlib.pyplot`, μέσω της συνάρτησης **figure()**, ορίζουμε τον αριθμό για κάθε διαφορετική γραφική παράσταση, μέσω της συνάρτησης **title()**, ορίζουμε τον τίτλο κάθε γραφικής, μέσω της συνάρτησης **yscale()** και δίνοντας ως όρισμα το **"log"**, μετατρέπουμε τον άξονα του `y` στην λογαριθμική κλίμακα. Τέλος, κάνουμε **plot** τις συναρτήσεις, βάζοντας τα κατάλληλα **labels** σε κάθε μία και στις προσεγγιστικές συναρτήσεις, θέτουμε ως επιπλέον όρισμα το **linestyle='dashed'**, έτσι ώστε να εμφανίζονται με διακεκομμένες γραμμές.

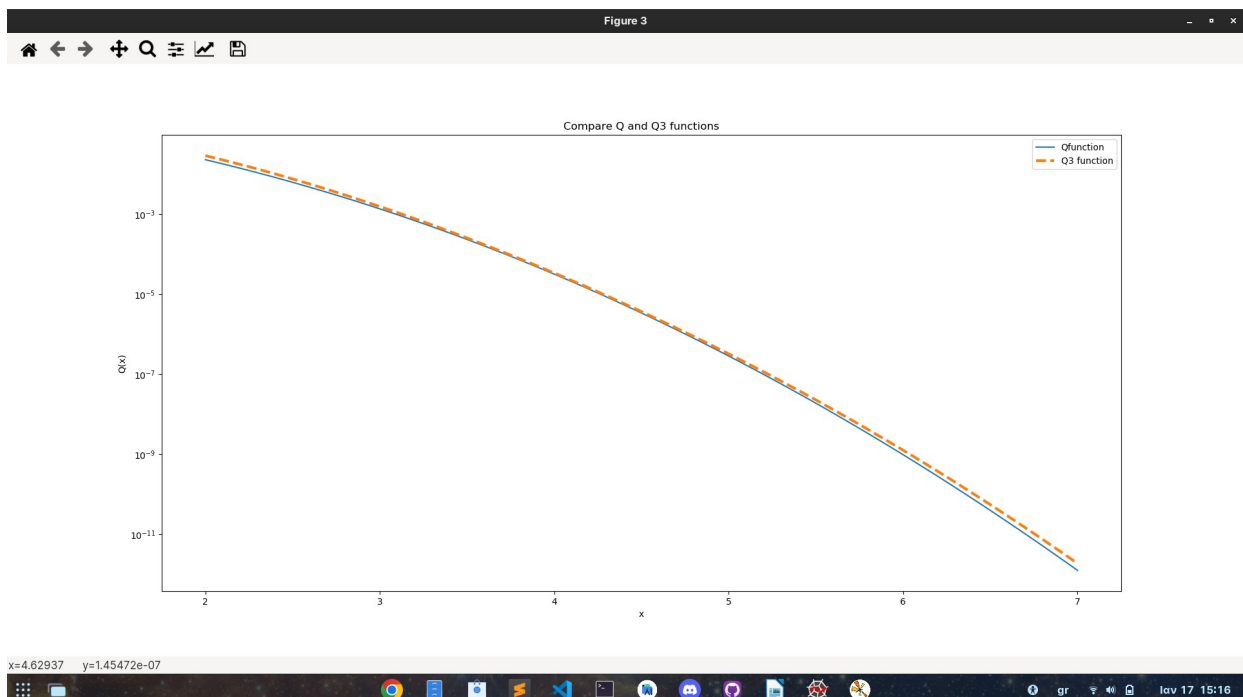
- Γραφική με τις συναρτήσεις Q και Q1:



- Γραφική με τις συναρτήσεις  $Q$  και  $Q2$ :



- Γραφική με τις συναρτήσεις  $Q$  και  $Q3$ :



**B)** Στο β ερώτημα του πρώτου μέρους έπρεπε να υπολογίσω το μέσω τετραγωνικό σφάλμα στο διάστημα  $[2,7]$  για κάθε μια προσέγγιση. Δημιούργησα 3 μεταβλητές ( $e1$ ,  $e2$ ,  $e3$ ), στις οποίες μέσω της συνάρτησης **abs()**, έβρισκα αρχικά την απόλυτη τιμή του αριθμητή και του παρονομαστή, για κάθε προσέγγιση και μέσω της συνάρτησης **trapz()**, στην συνέχεια έβρισκα το αριθμητικό ολοκλήρωμα κάθε μεταβλητής και το τύπωνα στην έξοδο.

Τα αποτελέσματα που προκύψαν είναι τα εξής:

- Μεταξύ των συναρτήσεων **Q** και **Q1**, το **e1 = 54.159243**
- Μεταξύ των συναρτήσεων **Q** και **Q2**, το **e2 = 9.886656**
- Μεταξύ των συναρτήσεων **Q** και **Q3**, το **e3 = 0.996427**

**Συμπεράσματα:** Βλέποντας, αρχικά τις γραφικές παραστάσεις, παρατηρούμε ότι την μεγαλύτερη απόκλιση από την συνάρτηση **Q**, την έχει η συνάρτηση **Q1**, ενώ αυτή που την προσεγγίζει με μεγαλύτερη ακρίβεια είναι η συνάρτηση **Q3**, όπως βλέπουμε και στα διαγράμματα. Στην συνέχεια, παρατηρούμε ότι, η συνάρτηση **Q3** έχει και το μικρότερο τετραγωνικό σφάλμα σε σχέση με τις υπόλοιπες. Οπότε, κατά την γνώμη μου η καλύτερη προσέγγιση είναι η συνάρτηση **Q3**.

## Μέρος 2ο: Κυματομορφή PAM

Αρχικά, έκανα import την βιβλιοθήκη **commLib** του μαθήματος, η οποία περιέχει τις κλάσεις **pam\_constellation**, η οποία είναι υποκλάση της κλάσης **constellation** και την κλάση **digital\_signal**, οι οποίες με βοήθησαν στην υλοποίηση του συγκεκριμένου ερωτήματος.

Στην συνέχεια, έθεσα κάποιες παραμέτρους όπως, **TS** το οποίο μας δείχνει την διάρκεια κάθε συμβόλου, **samples\_per\_symbol** το οποίο μας δείχνει τον αριθμό δειγμάτων σήματος για κάθε σύμβολο, **tinitial** και **tguard** το οποίο μας δείχνει το αρχικό και τελικό χρονικό διάστημα όπου το σήμα είναι ίσο με μηδέν εκτός της διάρκειας των συμβόλων. Τέλος, όρισα μία ακόμη μεταβλητή **Name** η οποία περιέχει το όνομα μου. Ο σκοπός της δημιουργίας της είναι, να μπορούμε να αλλάξουμε το όνομα δυναμικά, έτσι ώστε το πρόγραμμα να δουλεύει για κάθε πιθανό όνομα που μπορεί να βάλει ο χρήστης ως είσοδο.

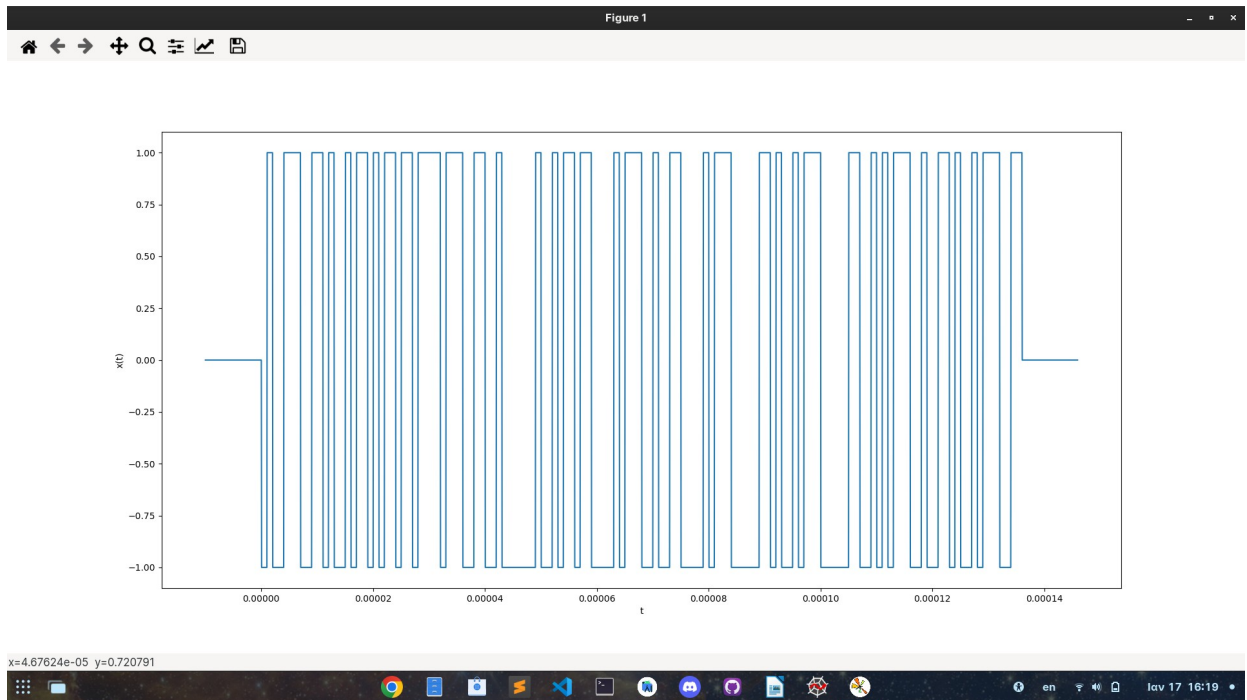
Έπειτα, μέσω των συναρτήσεων **join()**, **ord()** και **format()**, μετέτρεψα το όνομα μου από string σε **binary code**, με βάση τον κώδικα ASCII (Κάθε γράμμα αντιστοιχεί σε μια ομάδα των 8 μπίτς, συμπεριλαμβανομένου και του χαρακτήρα **space**). Την ιδέα της συγκεκριμένης υλοποίησης, την πήρα από την παρακάτω πηγή [geekForGeeks](http://www.geekforgeeks.com).

Στην συνέχεια, καλώ την κλάση **pam\_constellation**, κάθε φορά με διαφορετικό όρισμα(2,4,8,16), ώστε να υλοποιήσουμε τις κυματομορφές **2-PAM**, **4-PAM**, **8-PAM**, **16-PAM**. Έπειτα, καλώ την κλάση **digital\_signal**, δίνοντάς της ως ορίσματα τα εξής, **TS**, **samples\_per\_symbol**, **tinitial**, **tguard**, και **constellation** ο οποίος ισούται κάθε φορά με την ανάλογη κυματομορφή **PAM**.

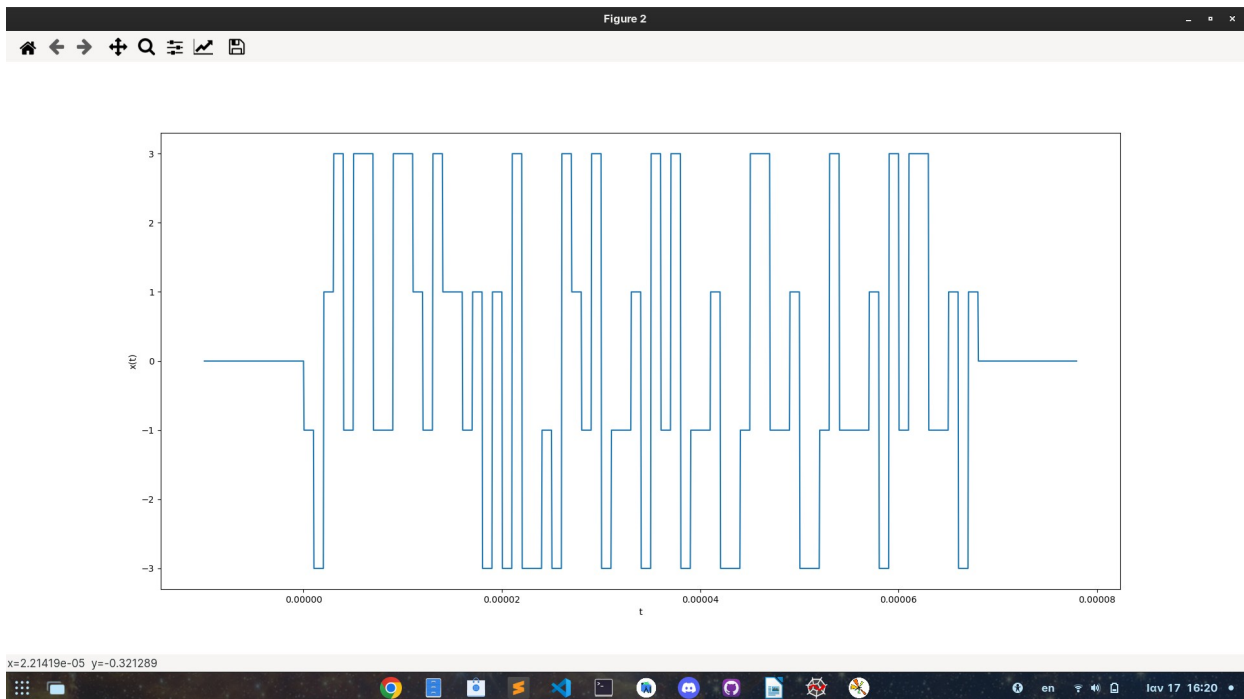
Στην απαίτηση “Σε περίπτωση όπου στην τελευταία ομάδα bits που μεταδίδουμε απομένουν λιγότερα από logM bits συμπληρώνουμε μηδενικά bits ώστε να έχουμε logM bits.”, έπραξα ως εξής: Για παράδειγμα στην κυματομορφή **8-PAM**, τα μπίτς μεταδίδονται σε ομάδες των τριών μπίτς, οπότε μέσω της συνάρτησης **len()** έβρισκα τον αριθμό των μπίτς που μεταδίδονται ανάλογα με το όνομα κάθε φορά και έλεγχα αν το υπόλοιπο της διαιρέσης αυτού με τον αριθμό 3 είναι διάφορο του 0. Αν ναι τότε πρέπει να συμπληρώσουμε μηδενικά μπίτς, μέχρις ότου το υπόλοιπο της διαίρεσης τους να ισούται με 0. Με το ίδιο σκεπτικό έπραξα και στις υπόλοιπες κυματομορφές, για παράδειγμα στην κυματομορφή **16-PAM**, το μόνο που αλλάζει είναι ότι αντί για τον αριθμό 3 τώρα έχουμε τον αριθμό 4, αφού στην συγκεκριμένη περίπτωση τα μπίτς μεταδίδονται σε ομάδες των τεσσάρων μπίτς.

Τέλος, μέσω της συνάρτησης **modulate\_from\_bits()**, της κλάσης **digital\_signal**, δημιουργούμε την κυματομορφή δίνοντας κάθε φορά ως είσοδο τα μπίτς που μεταδίδουμε(**bits**), και τον αντίστοιχο αστερισμό (**constellation = c1 ή c2 ή c3**) και μέσω της συνάρτησης **plot()** τα κάνουμε plot, την πρώτη φορά που καλούμε την συγκεκριμένη συνάρτηση της κλάσης **digital\_signal**, δίνουμε ως όρισμα πέρα από τον αριθμό του κάθε διαφορετικού **figure** και την τιμή **True**, με σκοπό πριν κάνουμε οποιοδήποτε plot, πρώτα να κλείσουμε όλα τα προηγούμενα.

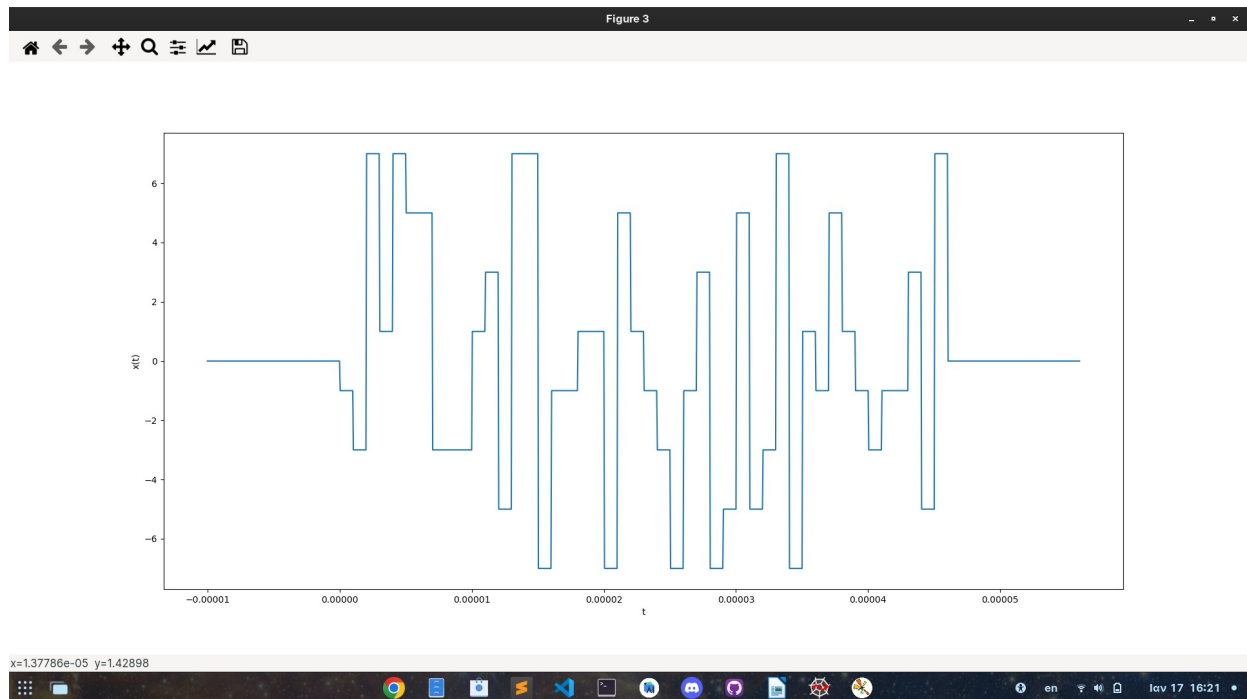
- Κυματομορφή 2-PAM:



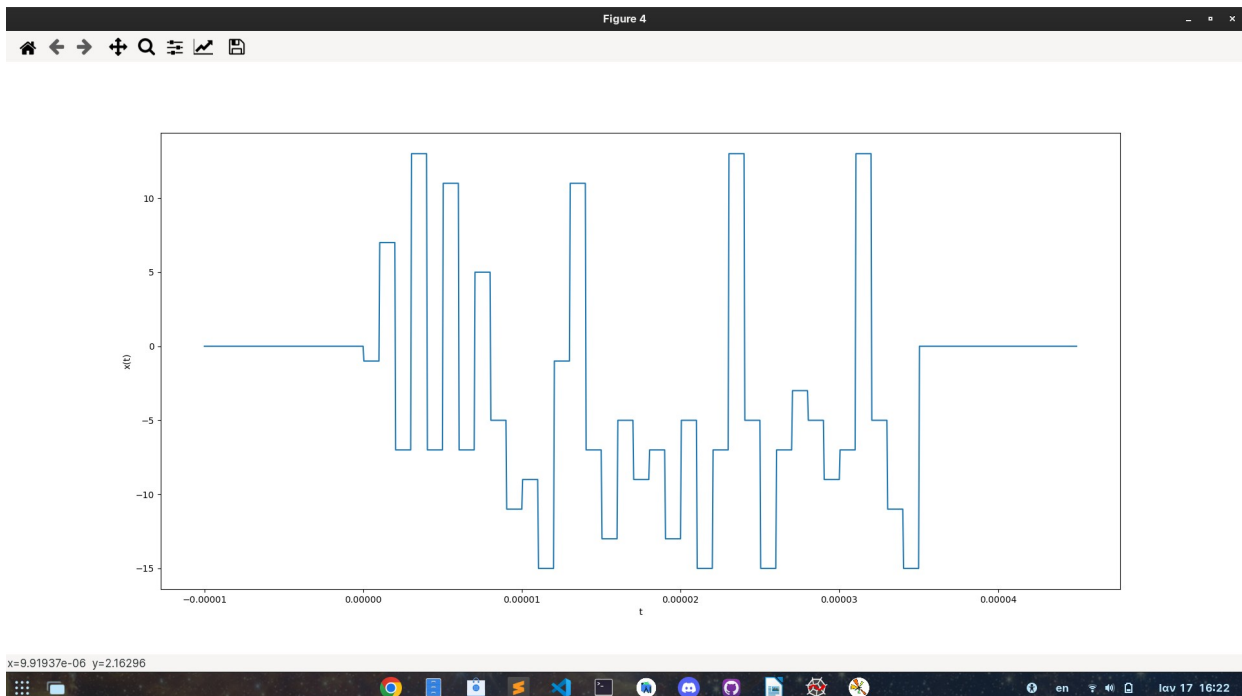
- Κυματομορφή 4-PAM:



- **Κυματομορφή 8-PAM:**



- **Κυματομορφή 16-PAM:**



**Σχολιασμός:** Γενικότερα το PAM που χρησιμοποιεί τα λιγότερα σύμβολα, έχει και την μικρότερη πιθανότητα σφάλματος, σε σχέση με τα υπόλοιπα δηλαδή το **2-PAM**, αυτό συμβαίνει καθώς, αν επιλέξουμε να μεταδώσουμε **16-PAM** ουσιαστικά μεταδίδουμε με την ίδια ενέργεια οχταπλάσιο αριθμό μπίτς σε σχέση με το **2-PAM**, οπότε είναι λογικό να έχουμε μεγαλύτερη πιθανότητα σφάλματος στο **16-PAM**. Οπότε, αν μας ενδιαφέρει ξεκάθαρα και μόνο το να έχουμε μικρή πιθανότητα σφάλματος, θα επιλέξουμε την κυματομορφή **2-PAM**, ενώ αν δεν έχουμε πολύ μεγάλο διαθέσιμο **εύρος ζώνης**, θα επιλέξουμε την κυματομορφή **16-PAM**.