

Εργασία στην ενότητα Πολυεπεξεργαστικά Υπολογιστικά Συστήματα: Ανάπτυξη εφαρμογής στον συνεπεξεργαστή ARM NEON της Xilinx Zynq-7000

16 Δεκεμβρίου, 2024

Ομάδα Εργασίας

αα	Όνοματεπώνυμο	Αριθμός Μητρώου	Email ή τηλέφωνο
1.			
2.			

1. Εισαγωγή

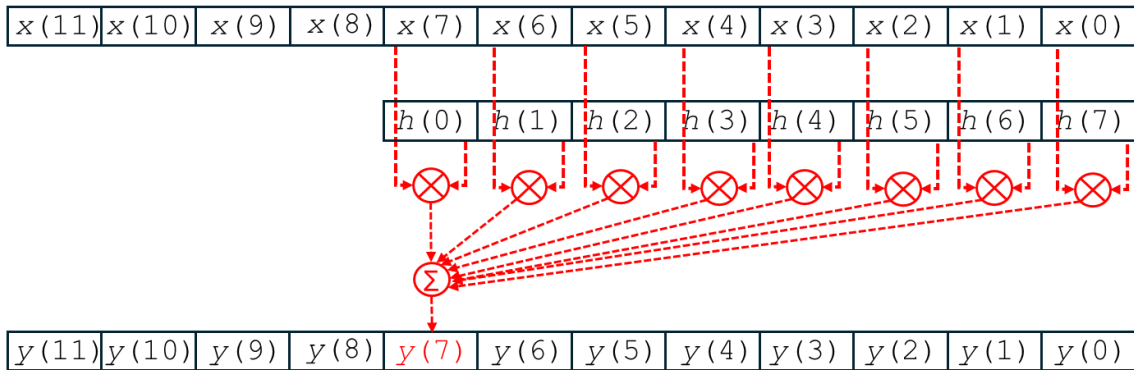
Η εργασία αφορά την υλοποίηση ενός απλού προγράμματος C που θα εκτελείται στον συνεπεξεργαστή NEON του ARM του τσιπ Xilinx Zynq-7000 που βρίσκεται στην πλακέτα Zybo. Ο σκοπός της εργασίας είναι να ασχοληθείτε με την ανάπτυξη μίας απλής εφαρμογής σε ένα επεξεργαστικό σύστημα που προσφέρει δυνατότητα παράλληλης επεξεργασίας της κατηγορίας SIMD (Single Instruction Multiple Data).

2. Σύνομη περιγραφή του συστήματος

Το σύστημα θα πρέπει να πραγματοποιεί ένα ψηφιακό φίλτρο πεπερασμένης κρουστικής απόκρισης (FIR filter – Finite Impulse Response filter). Ένα FIR φίλτρο δέχεται σαν είσοδο μία ακολουθία δειγμάτων $x(n)$ (σήμα διακριτού χρόνου) και αποδίδει στην έξοδο μία άλλη ακολουθία δειγμάτων $y(n)$, με $n \in \mathbb{Z}$. Η σχέση εισόδου-εξόδου δίνεται από την εξίσωση της συνέλιξης

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k]$$

Η ακολουθία $h(k)$, $k = 0 \dots N - 1$ λέγεται κρουστική απόκριση του φίλτρου και ο ακέραιος N τάξη του φίλτρου. Η συμπεριφορά του φίλτρου καθορίζεται αποκλειστικά από την κρουστική απόκριση. Ο υπολογισμός του δείγματος εξόδου $y(7)$ για ένα φίλτρο τάξης $N=8$ απεικονίζεται στο παρακάτω σχήμα:



Πραγματοποιήστε δύο εφαρμογές, μία για σειραϊκό υπολογισμό της συνέλιξης και μία για παράλληλο υπολογισμό με χρήση των καταχωρητών Q του NEON. Και για τις δύο εφαρμογές οι υπολογισμοί να γίνονται με αριθμητική κινητής υποδιαστολής απλής ακρίβειας (τύποι της C float και float32x4_t, αντίστοιχα). Για κρουστική απόκριση $h(k)$ του φίλτρου χρησιμοποιήστε τις ακόλουθες τιμές (τάξη φίλτρου $N=8$):

$h(0)$	$h(1)$	$h(2)$	$h(3)$	$h(4)$	$h(5)$	$h(6)$	$h(7)$
-0.0010	0.1219	0.1862	0.2429	0.2429	0.1862	0.1219	-0.0010

Σαν είσοδο χρησιμοποιήστε συνημιτονοειδές σήμα της μορφής $x(n)=\cos(\omega n)$, $n=0 \dots 31$, δηλαδή πλάτος 1, αρχική φάση 0, και κυκλική (ή γωνιακή) συχνότητα ω . Για κυκλική συχνότητα ω πειραματιστείτε με δύο τιμές, $\omega=\omega_1=2\pi/16$ και $\omega=\omega_2=2\pi/8$. (Περίοδος $T_1=16$ δείγματα και $T_2=8$ δείγματα, αντίστοιχα).

Το κάθε πρόγραμμα θα πρέπει να υπολογίζει την έξοδο του φίλτρου $y(n)$ για $n=7 \dots 31$.

Υπολογίστε την επιτάχυνση (speedup) και την απόδοση (efficiency) του παράλληλου αλγόριθμου σε σχέση με τον αντίστοιχο σειραϊκό. Στις μετρήσεις χρόνου θα λάβετε υπ' όψη αποκλειστικά τον υπολογισμό της συνέλιξης και όχι τους χρόνους προετοιμασίας της μνήμης με το σήμα εισόδου $x(n)$ και τους χρόνους εμφάνισης (σε ASCII μορφή) των σημάτων εισόδου $x(n)$ και εξόδου $y(n)$ στο τερματικό.

3. Παρατηρήσεις

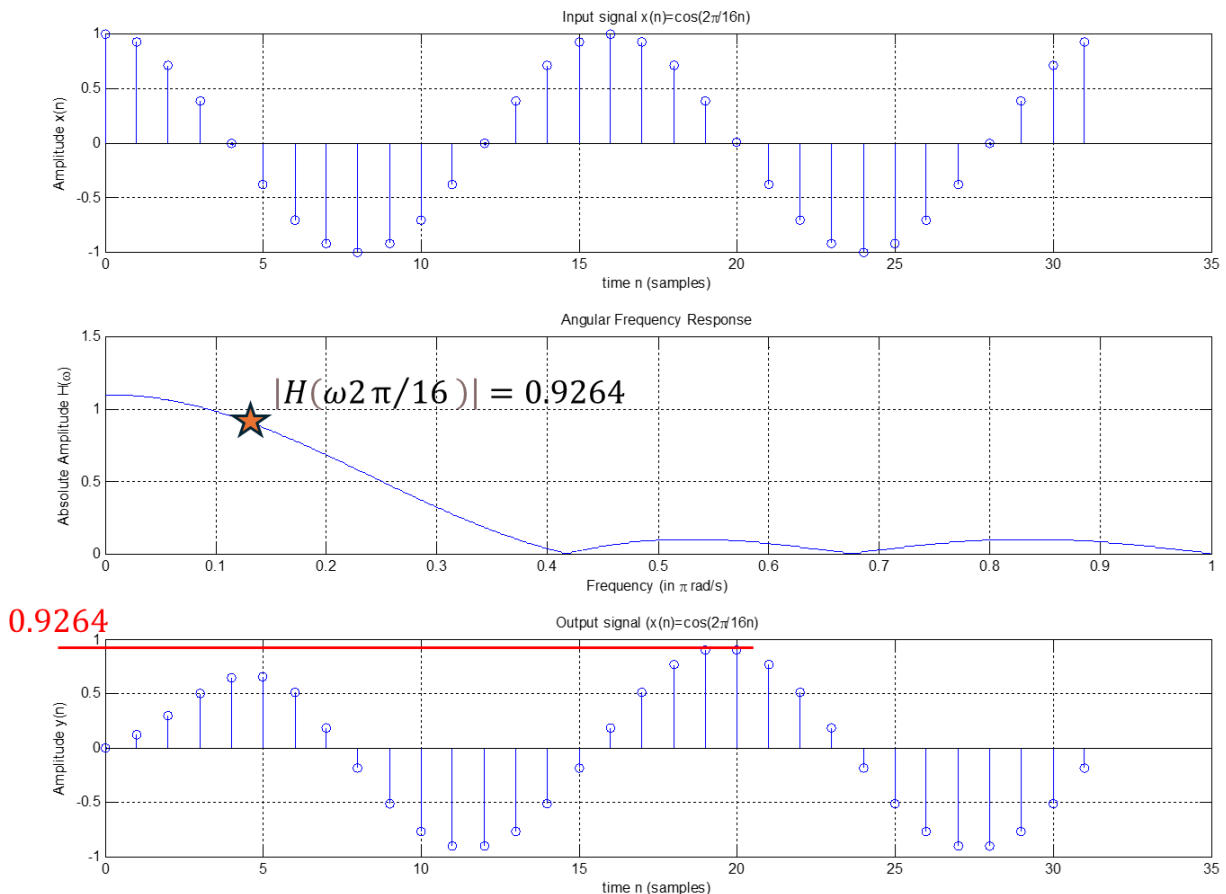
Μία βασική ιδιότητα ενός FIR φίλτρου είναι ότι η διέγερσή του με ένα συνημίτονο (ή ημίτονο) δίνει απόκριση στην έξοδο συνημίτονο (ή ημίτονο) ακριβώς ίδιας κυκλικής συχνότητας ω , αλλά με διαφορετικό πλάτος και φάση. Δηλαδή,

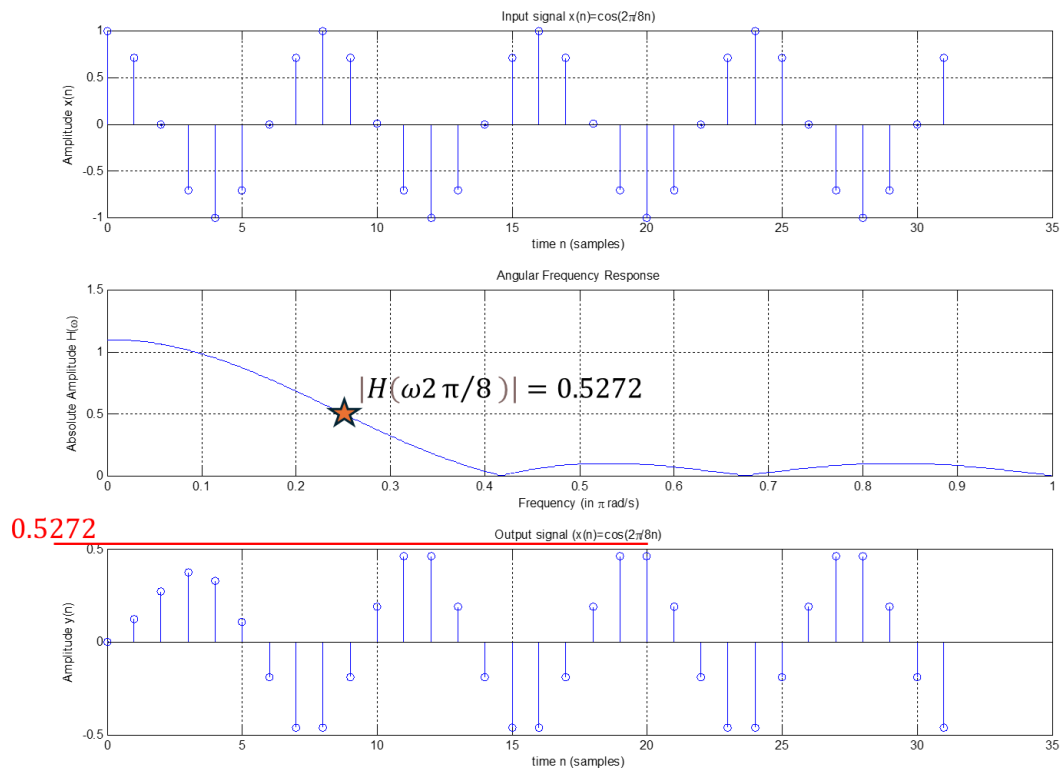
$$x(n) = \cos(\omega n) \xrightarrow{FIR} y(n) = |H(\omega)| \cos(\omega n - \varphi)$$

όπου $|H(\omega)|$ είναι η απόλυτη τιμή της απόκρισης συχνότητας του φίλτρου που εξαρτάται από την κρουστική του απόκριση $h(n)$. Για την συγκεκριμένη κρουστική απόκριση που δίνεται και για τις κυκλικές συχνότητες $\omega_1=2\pi/16$ και $\omega_2=2\pi/8$, ισχύει ότι $|H(2\pi/16)| = 0.9264$ και $|H(2\pi/8)| = 0.5272$.

Τα δύο επόμενα σχήματα δείχνουν την είσοδο, απόλυτη τιμή κρουστικής απόκρισης και έξοδο του φίλτρου για τις περιπτώσεις συχνότητας $\omega_1=2\pi/16$ και $\omega_2=2\pi/8$. Παρατηρήστε ότι στην πρώτη περίπτωση ($\omega_1=2\pi/16$) το πλάτος του σήματος εξόδου $y(n)$ είναι περίπου 0.9264, ενώ δεύτερη περίπτωση ($\omega_2=2\pi/8$) το πλάτος του σήματος εξόδου $y(n)$ πέφτει σε τιμή περίπου 0.5272. Από την απόκριση συχνότητας προκύπτει ότι το φίλτρο είναι χαμηλοπερατό (lowpass filter) με συχνότητα αποκοπής (cut-off frequency) $\pi/4$.

Οι παραπάνω παρατηρήσεις θα σας βοηθήσουν να διαπιστώσετε την ορθή λειτουργία των προγραμμάτων σας.





Επί πλέον, για τη χρήση του NEON απαιτείται:

- Η χρήση της οδηγίας `#include <arm_neon.h>` στον πηγαίο κώδικα.
- Η χρήση της οδηγίας `-mfpu=neon` στον μεταγλωττιστή (C/C++ Building Settings → ARM v7 gcc compiler → Miscellaneous → Other flags).

Για τη χρήση της συνάρτησης `cos()` και της σταθεράς `π` (`M_PI`) απαιτείται:

- Η χρήση της οδηγίας `#include <math.h>` στον πηγαίο κώδικα.
- Η χρήση της οδηγίας `-lm` στον linker (C/C++ Building Settings → ARM v7 gcc linker → Libraries → Other flags).

4. Σύσταση Ομάδων – Παράδοση Εργασίας

- ☑ Κάθε ομάδα εργασίας μπορεί να αποτελείται μέχρι 2 φοιτητές. Συμπληρώστε τα στοιχεία των μελών της ομάδας στον πίνακα της πρώτης σελίδας αυτής της εκφώνησης (όνομα, αριθμός μητρώου και τουλάχιστον ένα email επικοινωνίας ή τηλέφωνο για κάθε ομάδα).
- ☑ Θα πρέπει να παραδώσετε: (α) τους πηγαίους κώδικες, (β) εμφάνιση των αποτελεσμάτων που παίρνετε για κάθε περίπτωση σήματος εισόδου, και (γ) τις μετρήσεις απόλυτων χρόνων για τα δύο προγράμματα (σειραϊκό και παράλληλο) καθώς και τις μετρήσεις επιτάχυνσης και απόδοσης. Τα δύο τελευταία (β και γ) σε ένα ξεχωριστό αρχείο Word.
- ☑ Ημερομηνία παράδοσης της εργασίας: μέρα εξέτασης του μαθήματος, δηλ **31 Ιανουαρίου 2025**.
- ☑ Θα οριστεί ημερομηνία για την εξέταση της εργασίας.
- ☑ Η παράδοση της εργασίας θα γίνει σε ηλεκτρονική μορφή στην ιστοσελίδα του μαθήματος.