École de technologie supérieure

Université du Québec

Laboratoire 3

Présenté à M. Marcel GABREA

Dans le cadre du cours

*SYS835 – Processeur numérique du signal et ses applications*

PAR

Eric LACERTE LACE23038502

Philippe LAVOIE LAVP05067200

MONTRÉAL, LE 6 DECEMBRE 2017

# 3.3 Méthodes d’optimisation

## Création du projet

Se référer au rapport de laboratoire 2 section 2.3.1

## Création de target de simulation

Se référer au rapport de laboratoire 2 section 2.3.2

## Écriture du fichier main :

* Définition globale de 2 tableaux x et y.
* Initialisation des tableaux.
* Déclaration des fonctions.

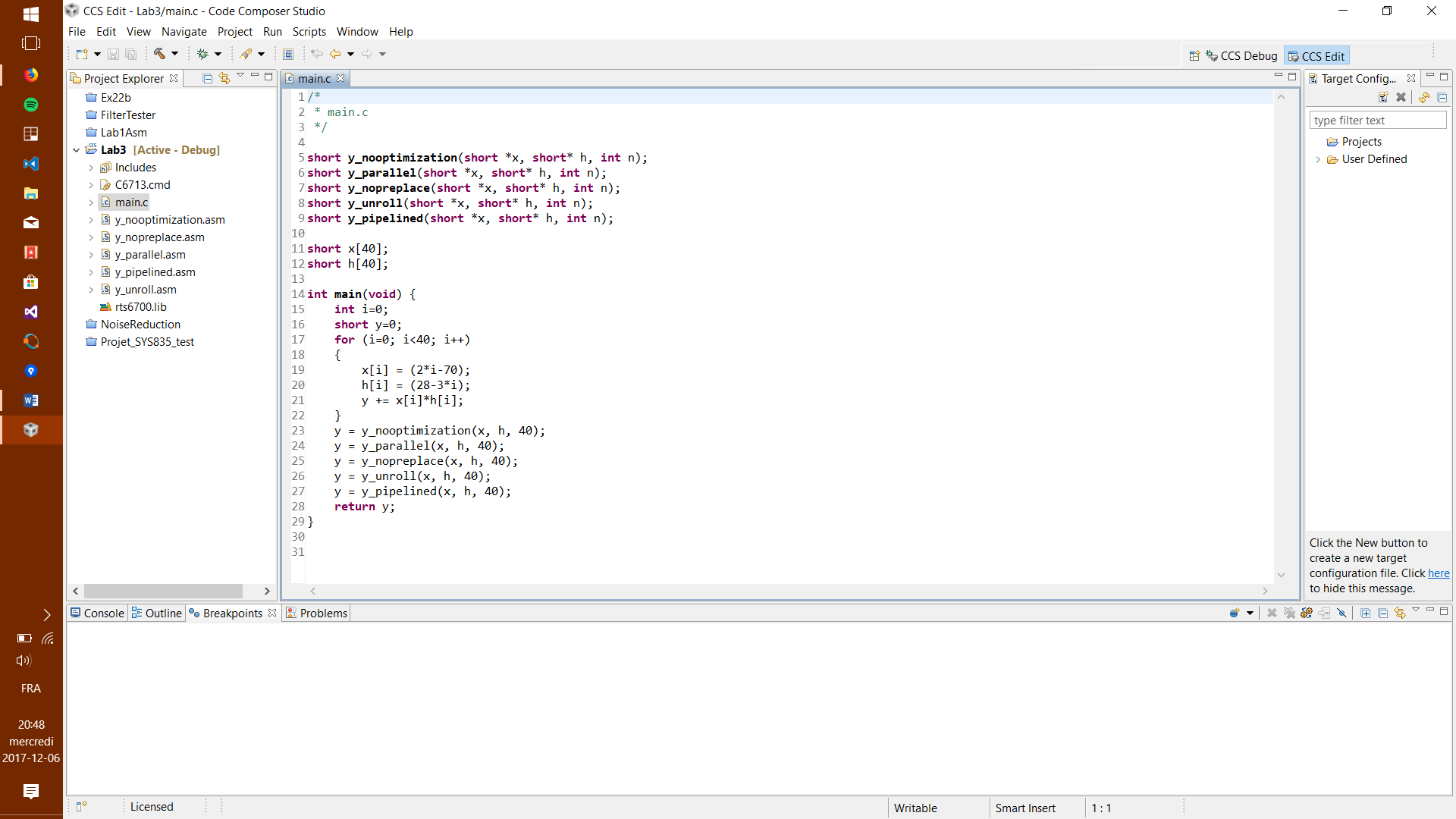


Figure main.c

# Création de la fonction assembleur sans optimisation

La fonction y\_nooptimization ne comporte aucune optimisation.

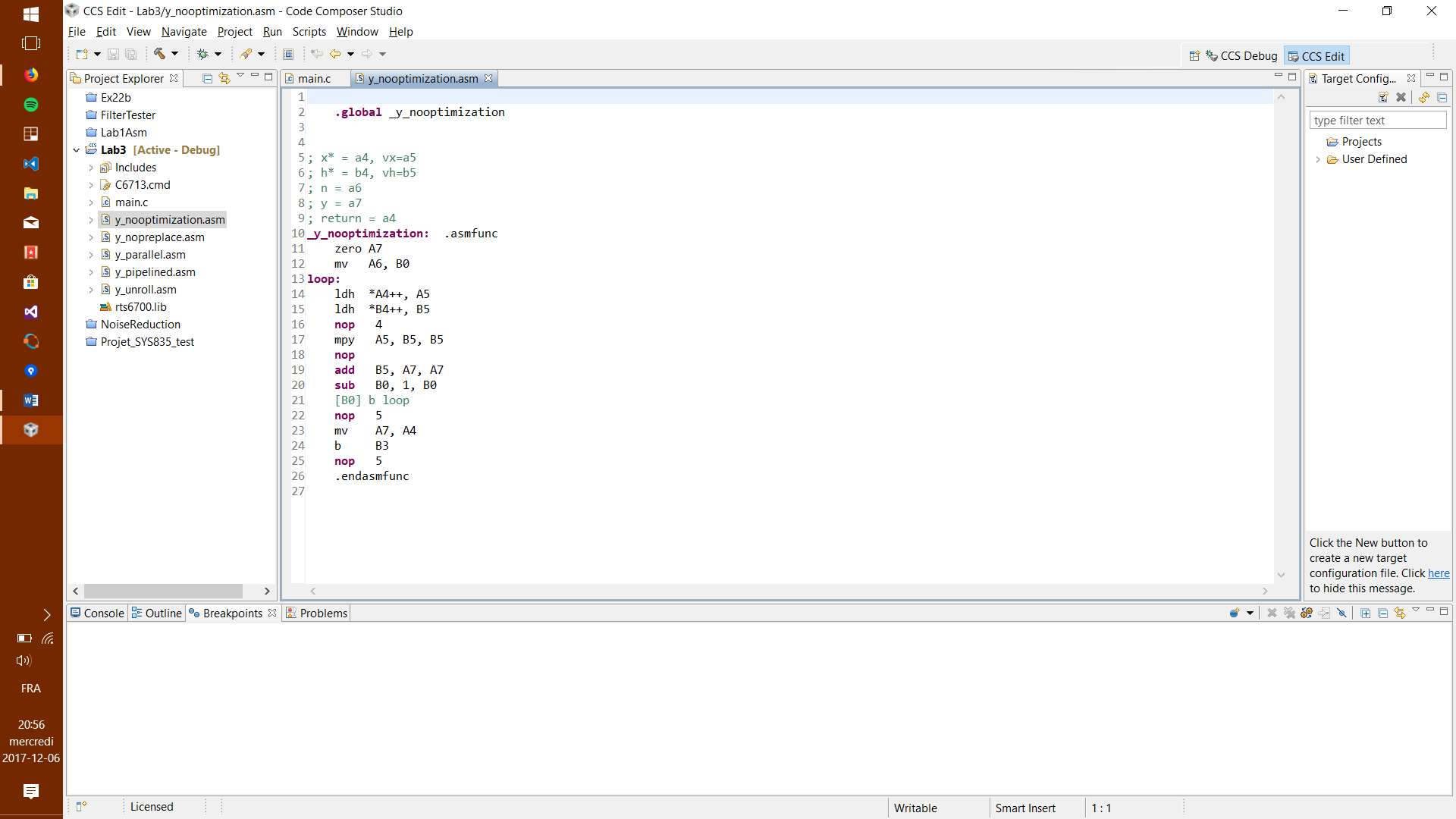


Figure 2 y\_nooptimization.asm

# Création de la fonction assembleur optimisation parallèle

La fonction y\_parallel comporte une optimisation en utilisant les instructions admissibles en parallèle.

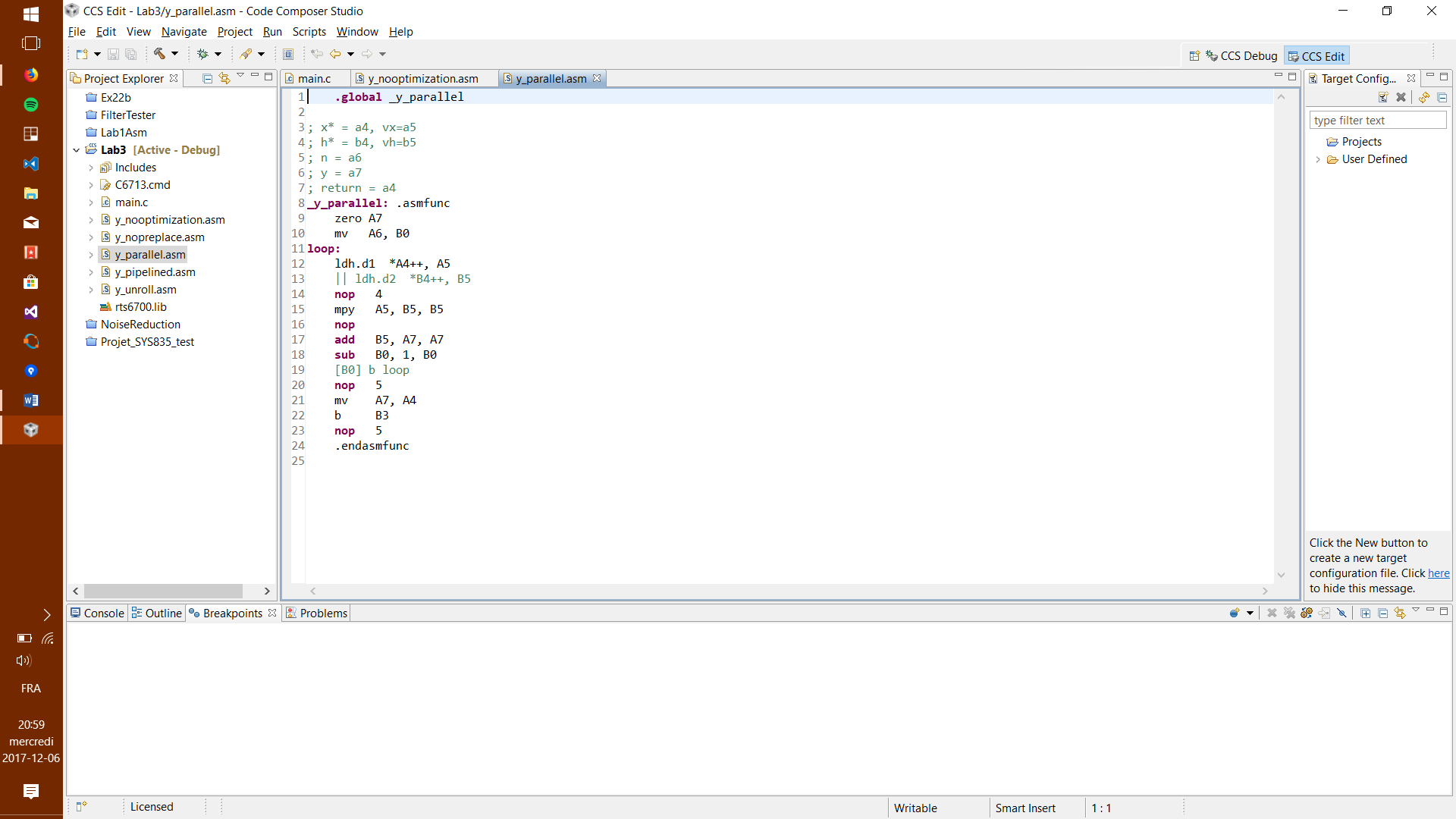


Figure 3 y\_parallel.asm

# Création de la fonction assembleur optimisation des NOP

La fonction y\_nopreplace comporte une optimisation en restructurant l’ordre des instructions afin de minimiser les instructions NOP

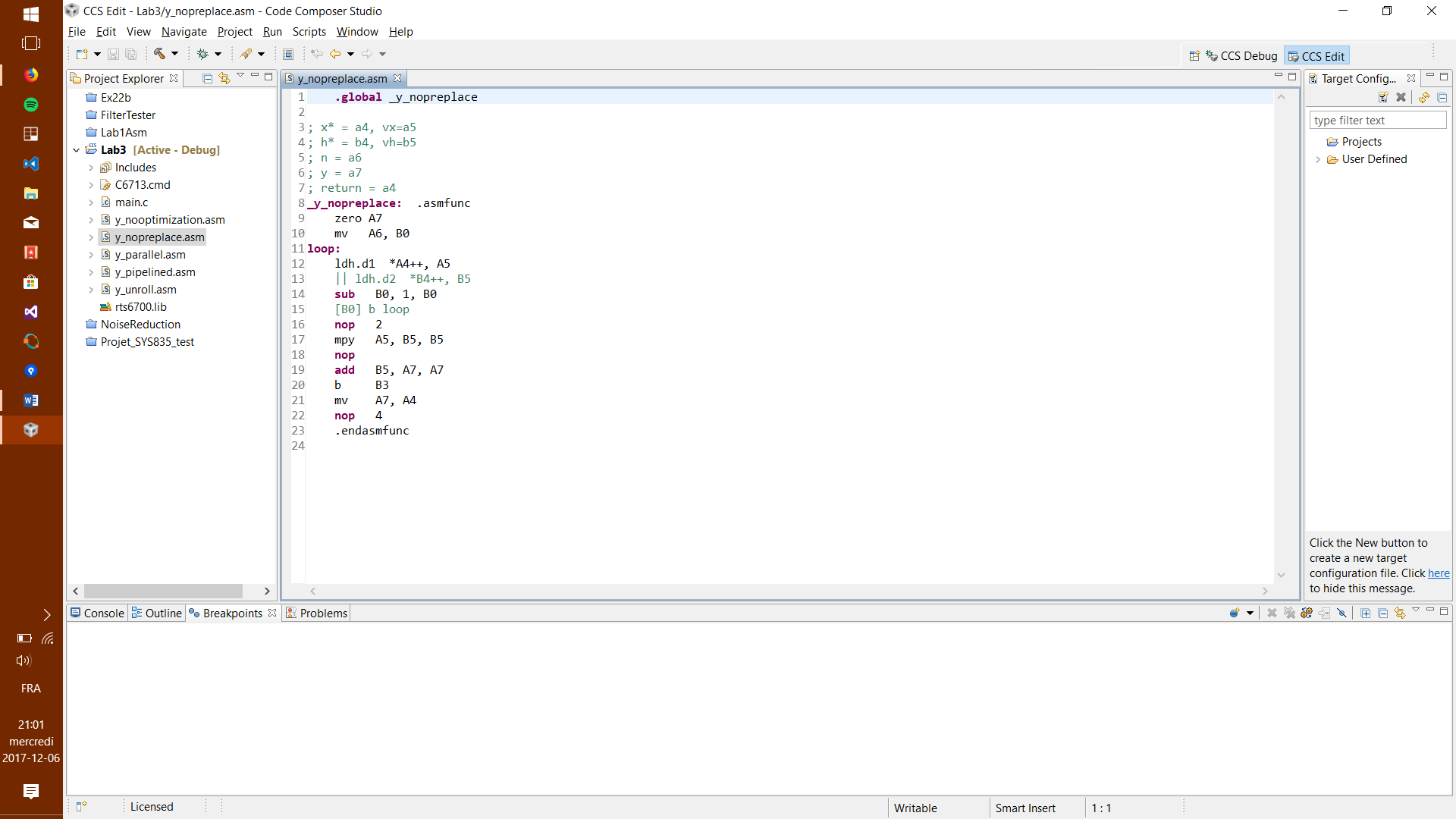


Figure 4 y\_nopreplace.asm

# Création de la fonction assembleur optimisation déroulement et « LoadWord »

La fonction y\_unroll comporte une optimisation en remplacant les lh (*load half word*) par des instructions lw (*load Word*). Cela permet de charger 2 valeurs à la fois (partie haute, partie basse du *word*).

On « déroule » la boucle afin de minimiser le nombre de vérification de « branch » utiliser pour retourner au début de la boucle.

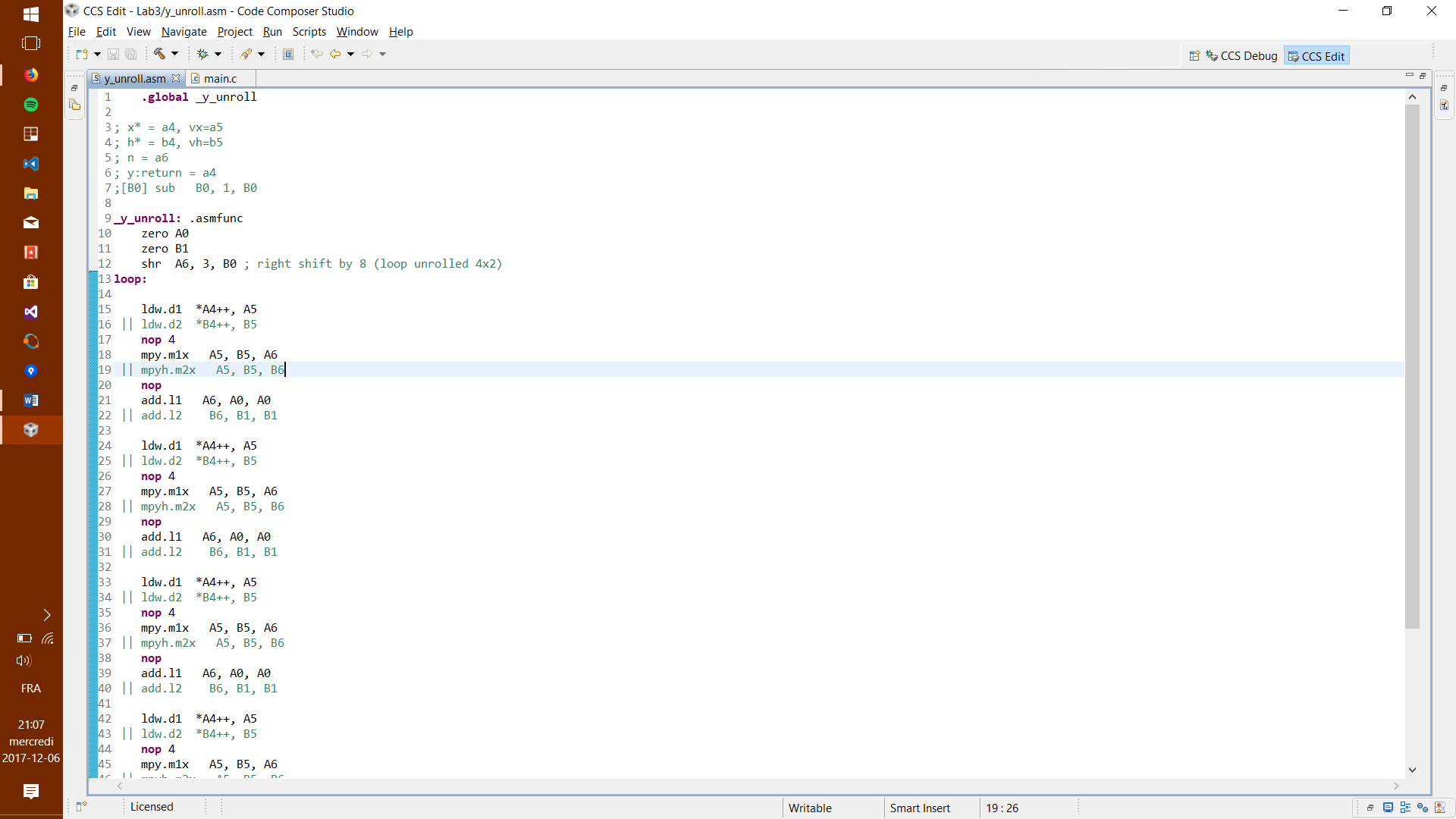
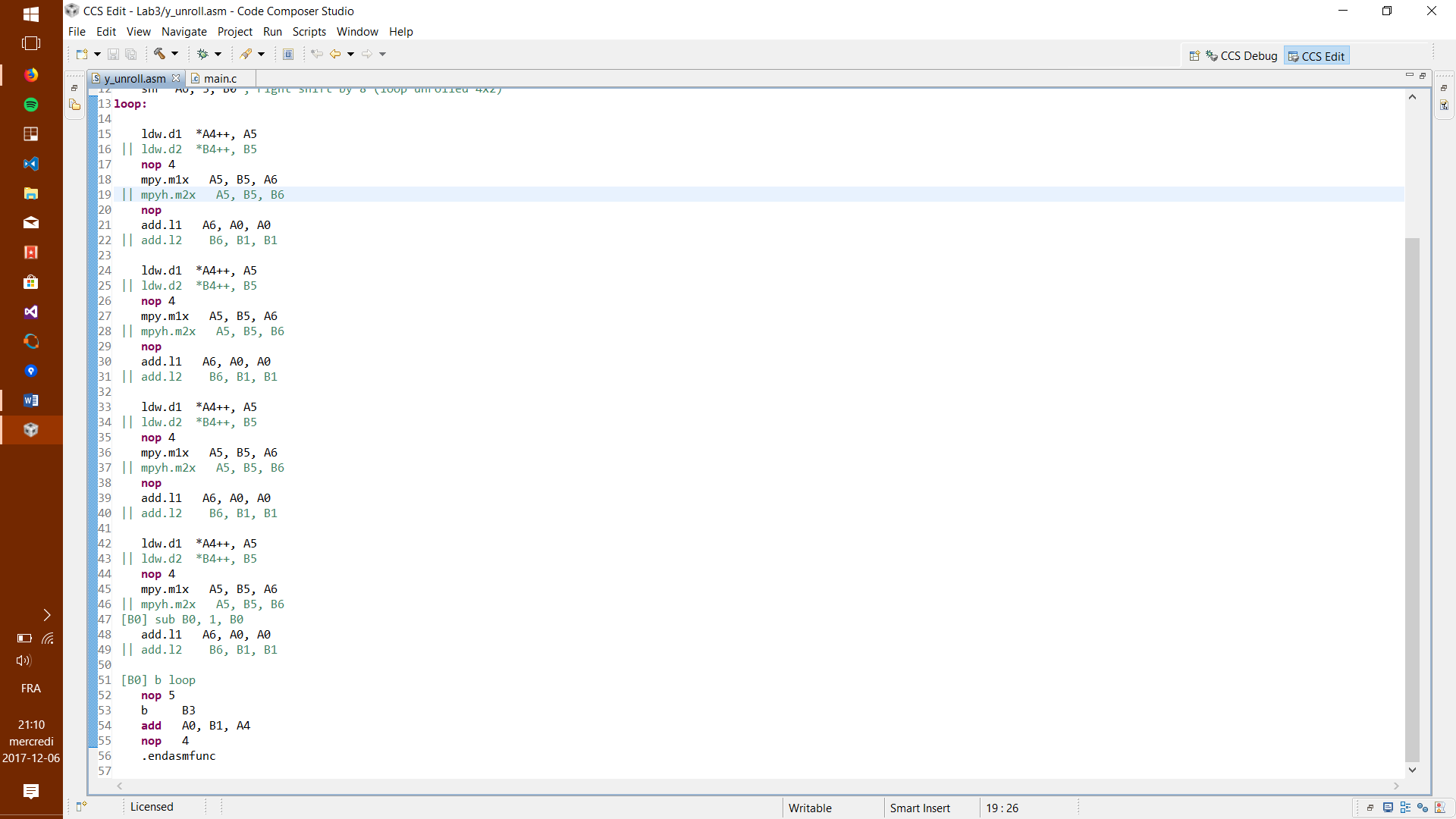
 

Figure 5 y\_unroll.asm

# Création de la fonction assembleur optimisation « pipeline »

La fonction y\_pipelined contient une optimisation par le pipeline. Nous amorçons le pipeline avec l’épilogue c0 à c6. Ensuite nous performons la boucle.

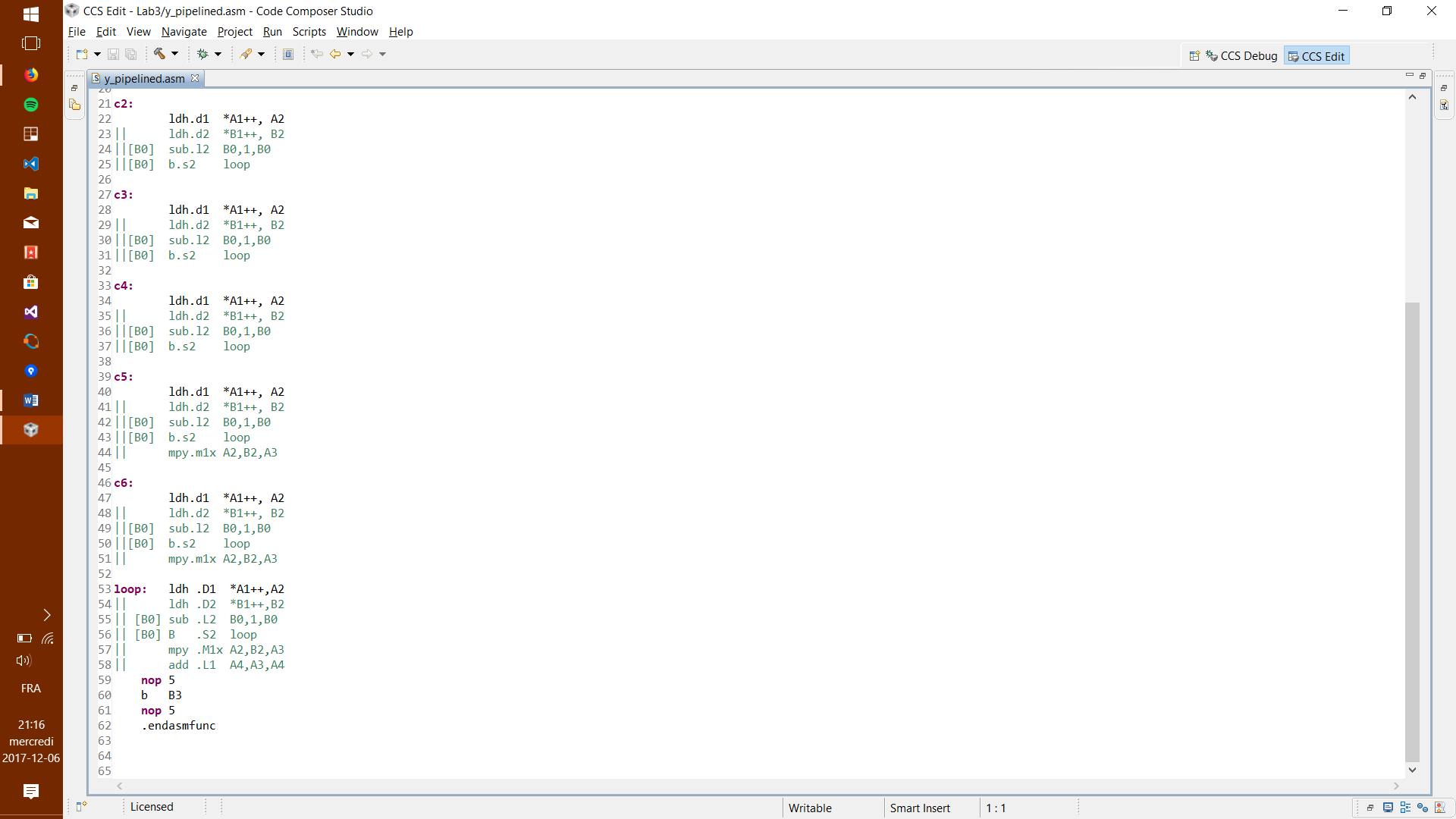
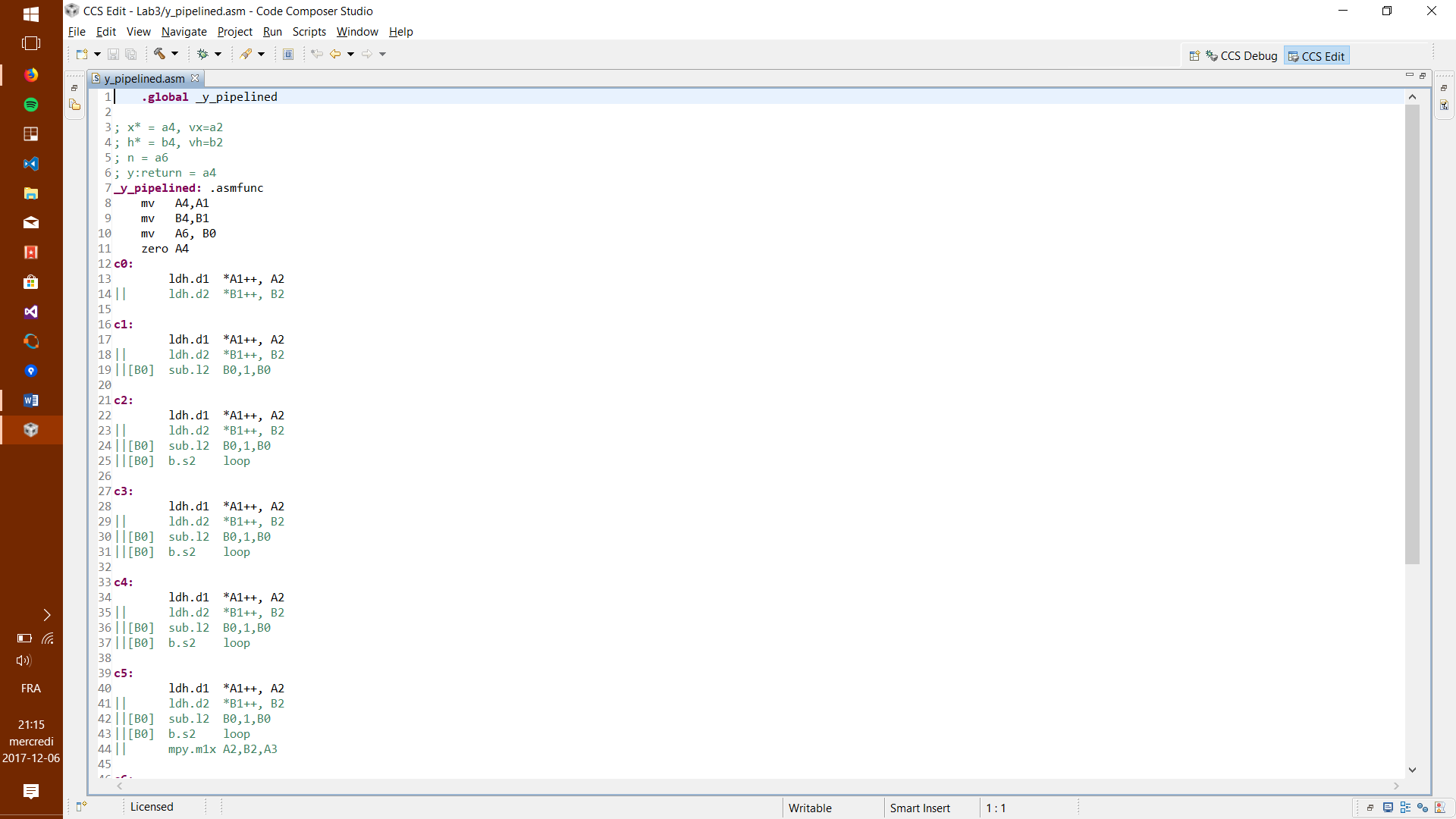
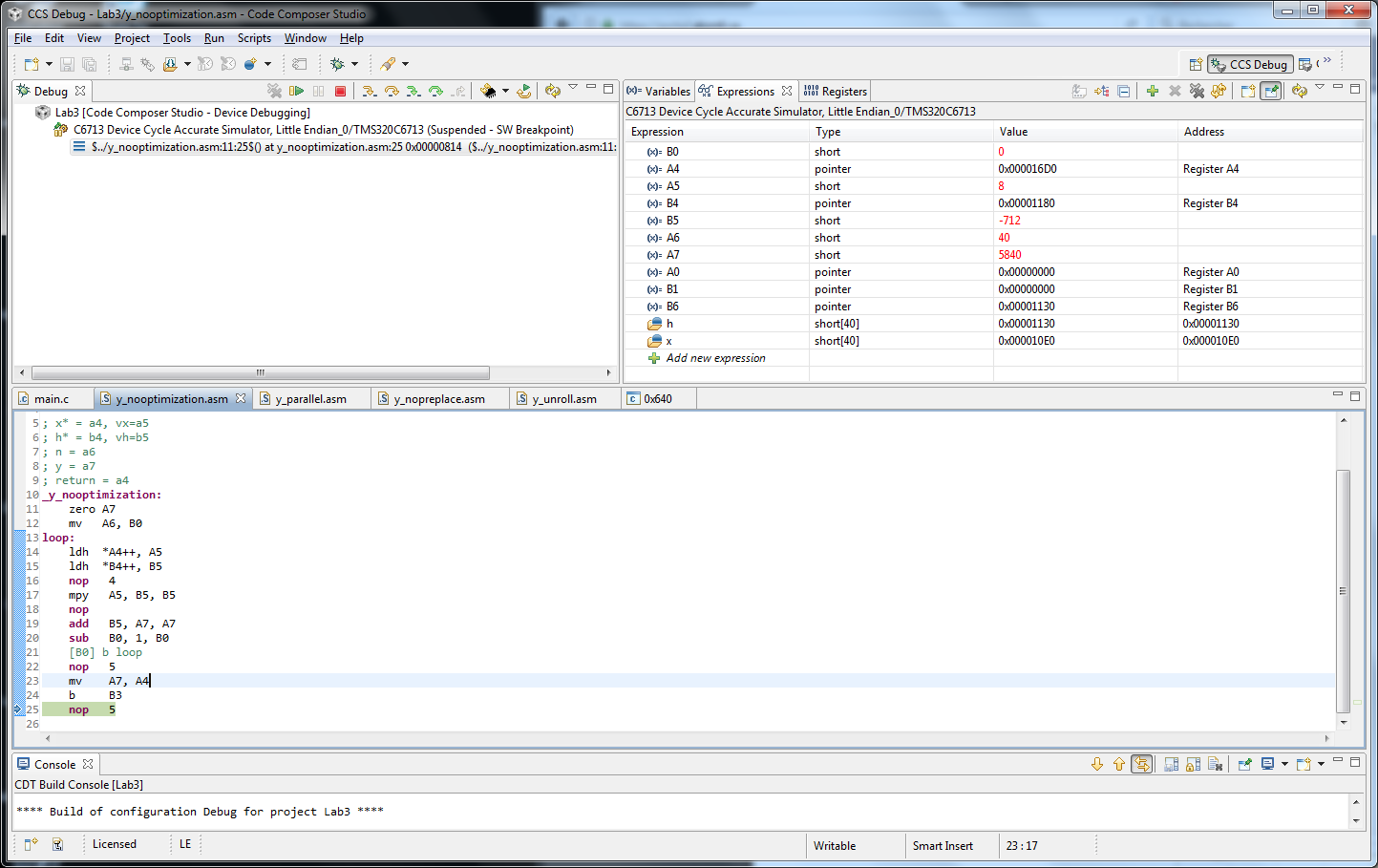


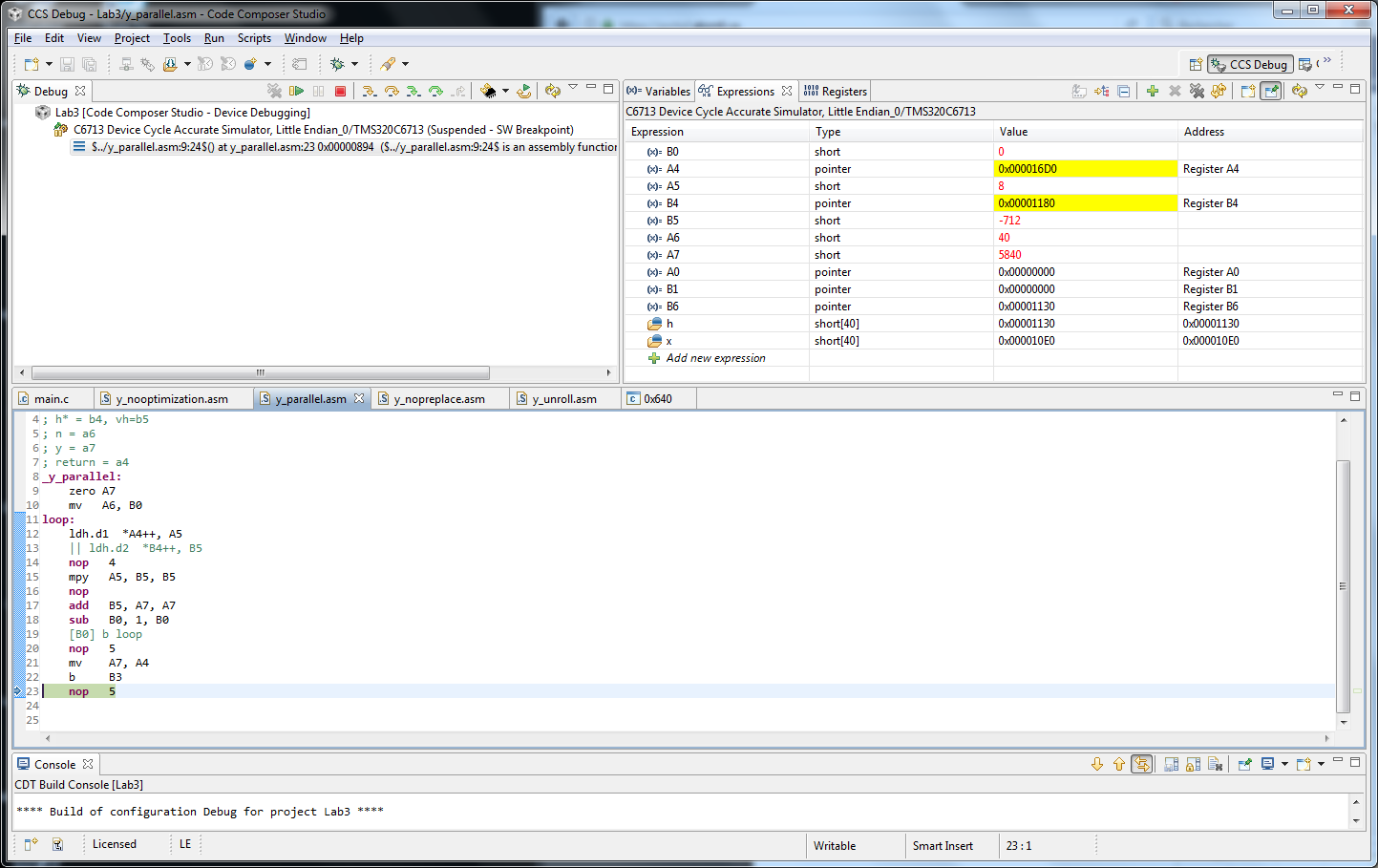
Figure y\_pipelined.asm

TODO

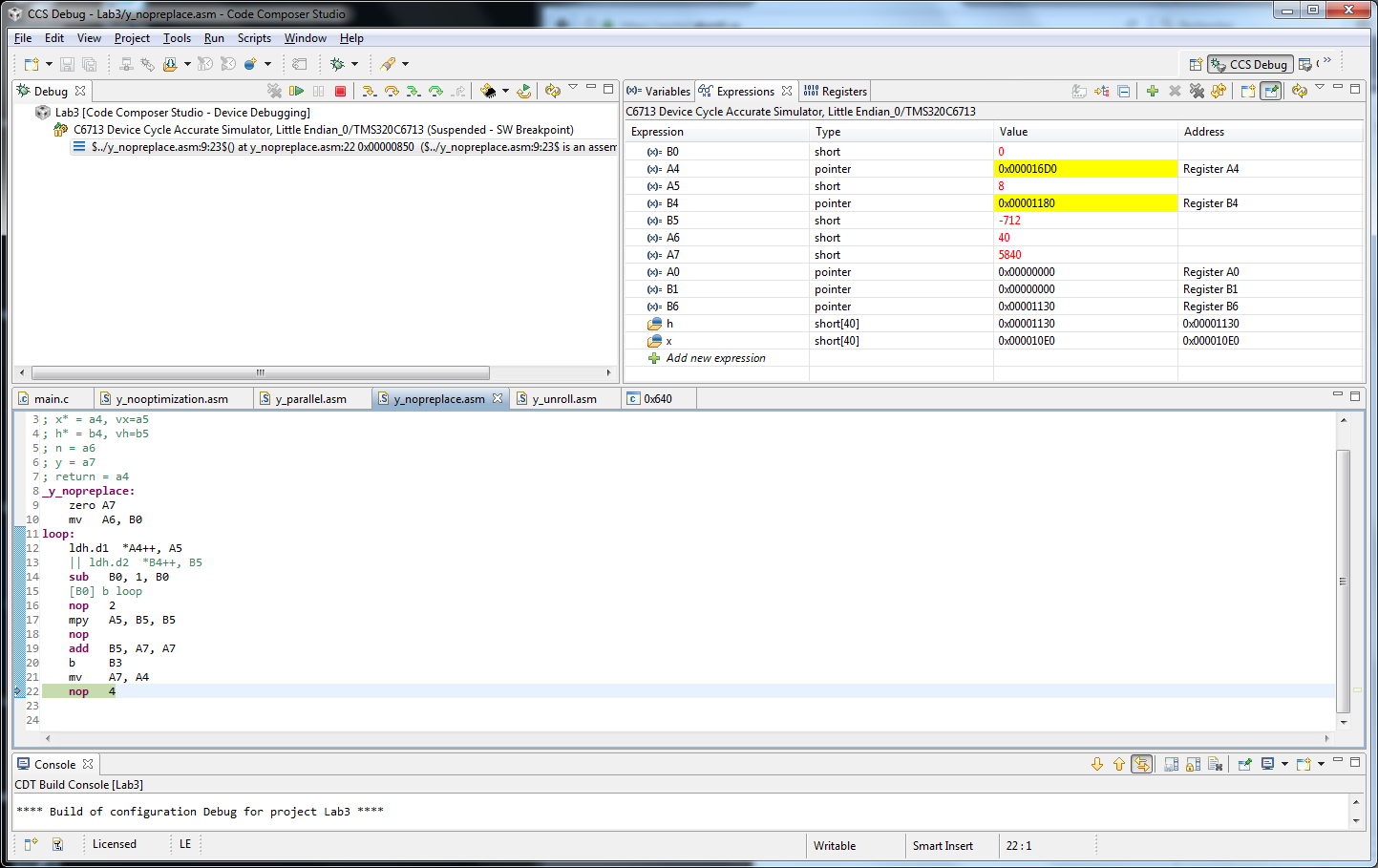
Exécution de y\_nooptimization et vérification du résultat (dans A7, retourné dans A4).



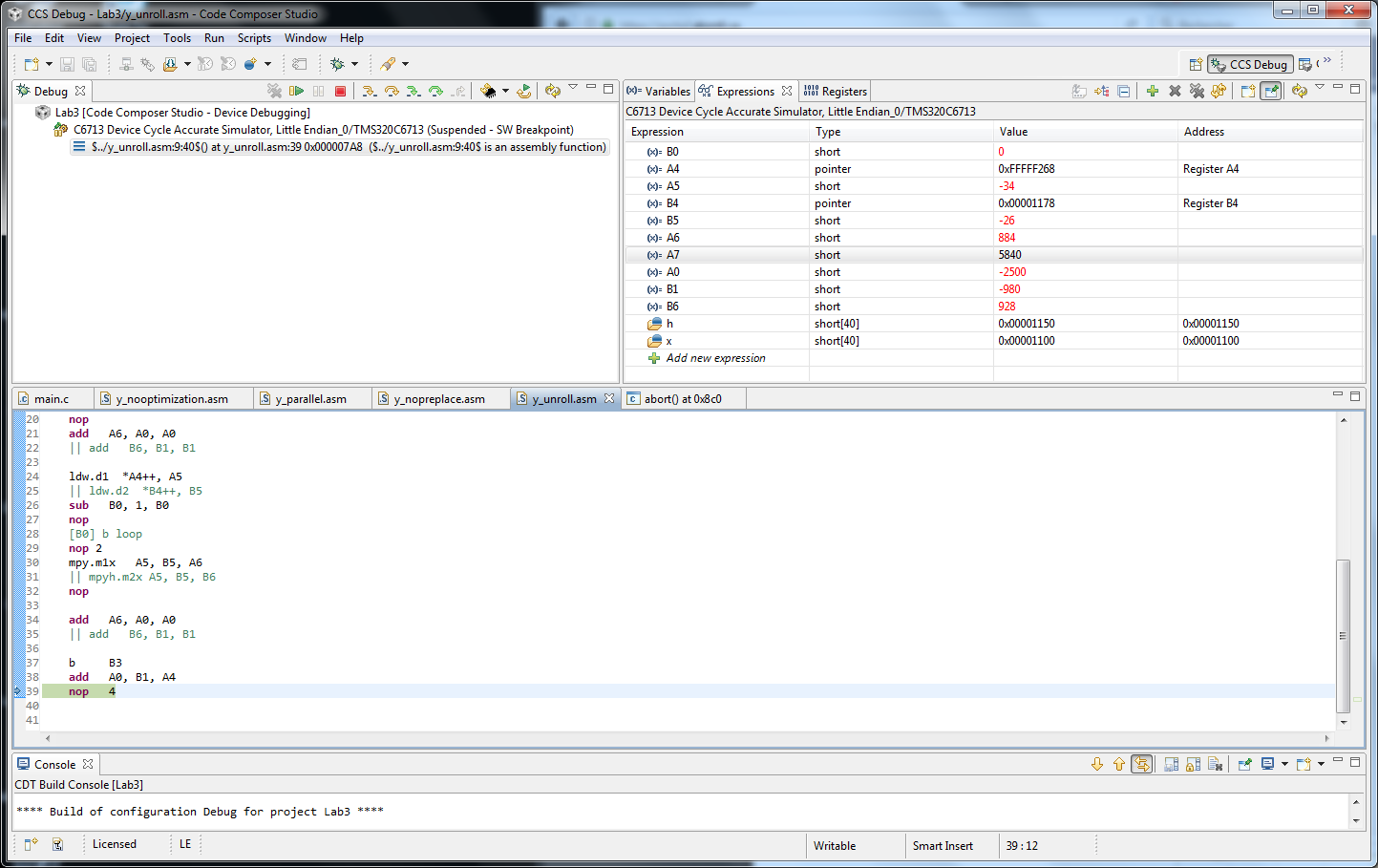
Exécution de y\_parallel et vérification du résultat dans A7.



Exécution de y\_nopreplace



Execution y\_unroll :



Profiling:

