Honeywell-Resideo

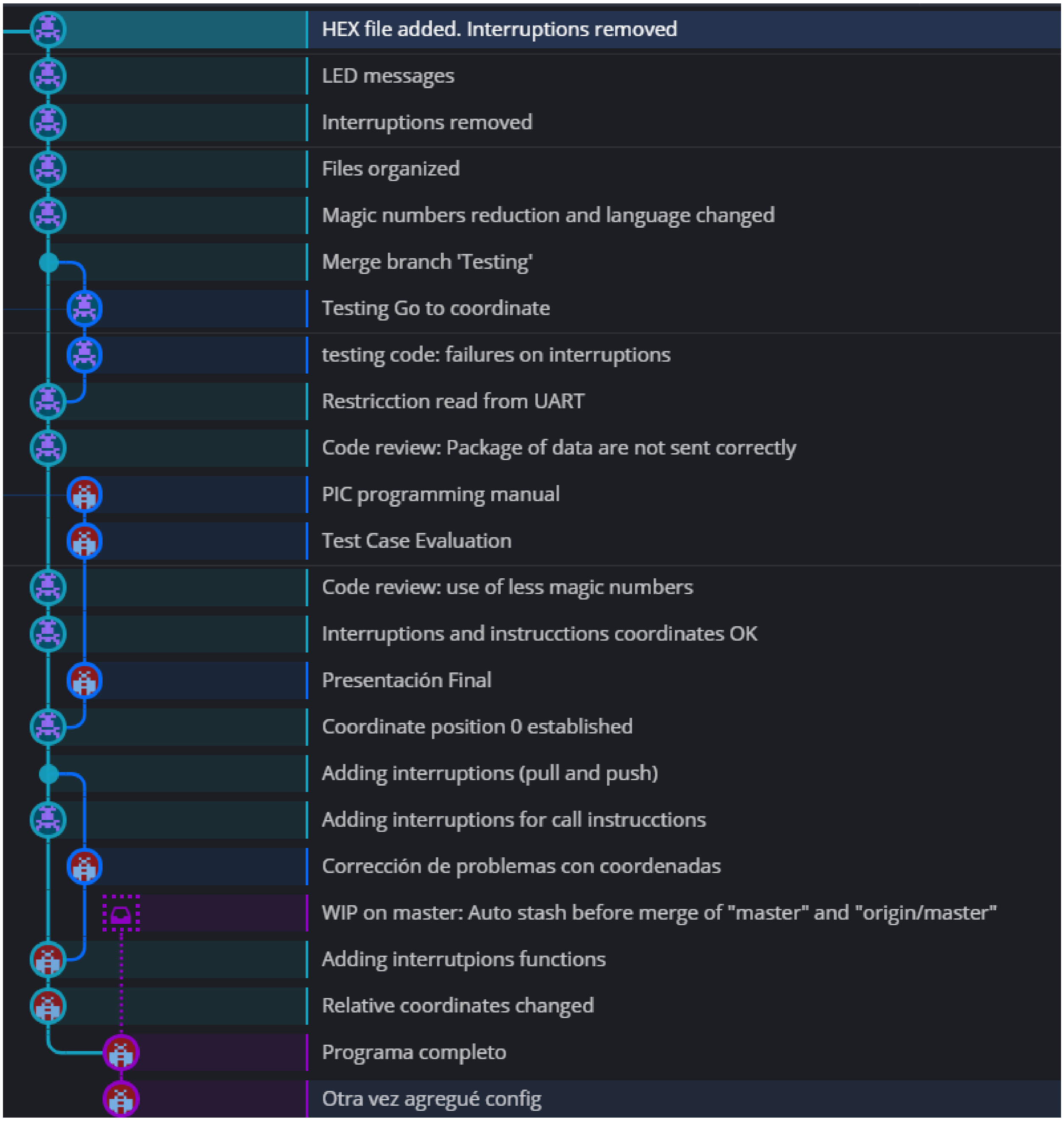
Haydé Zamudio	A01363956
Angélica Luna	A01187815
Felipe Rojas	A01566041

resideo

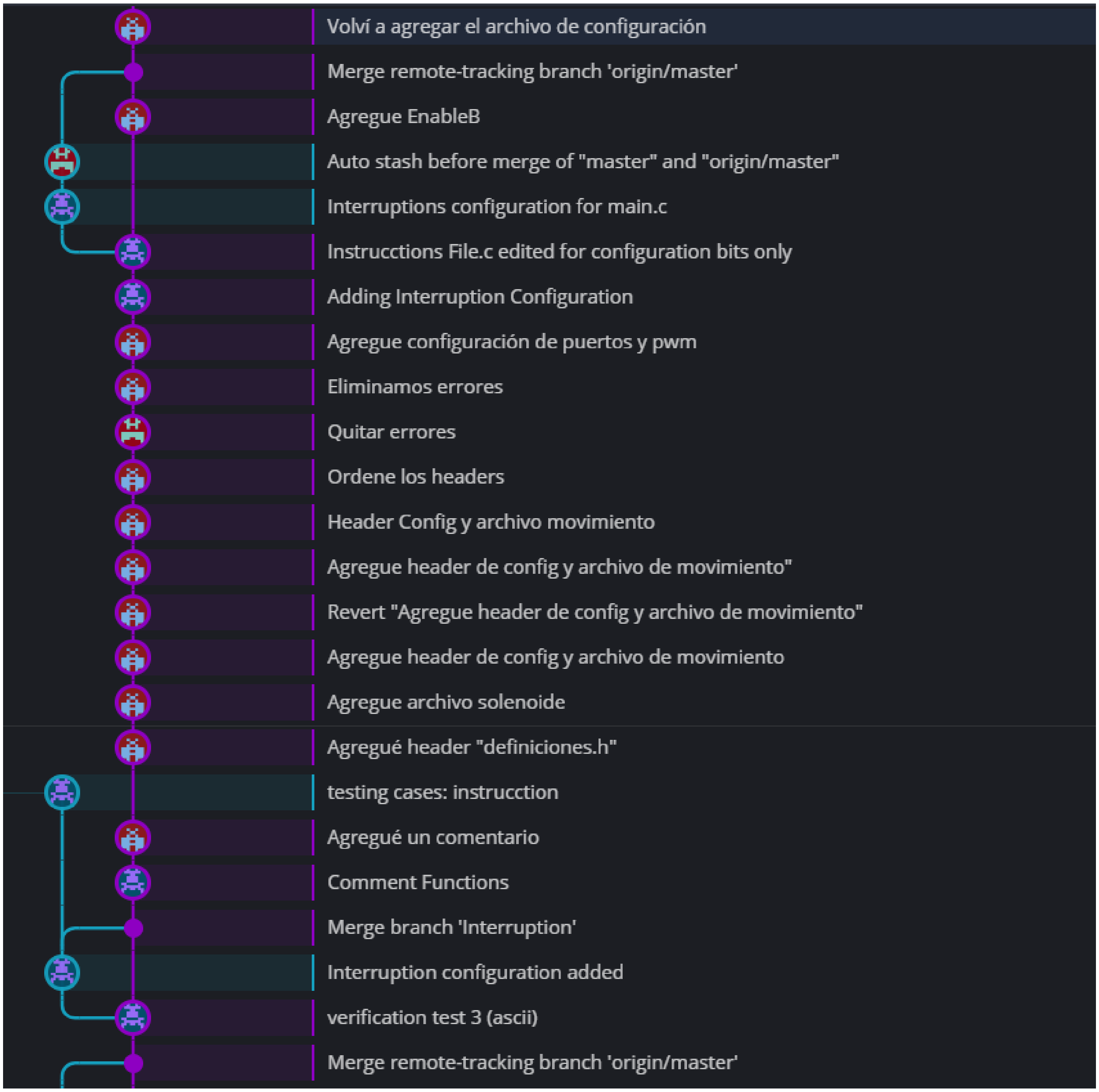
Tecnológico
de Monterrey

Code Review evidence

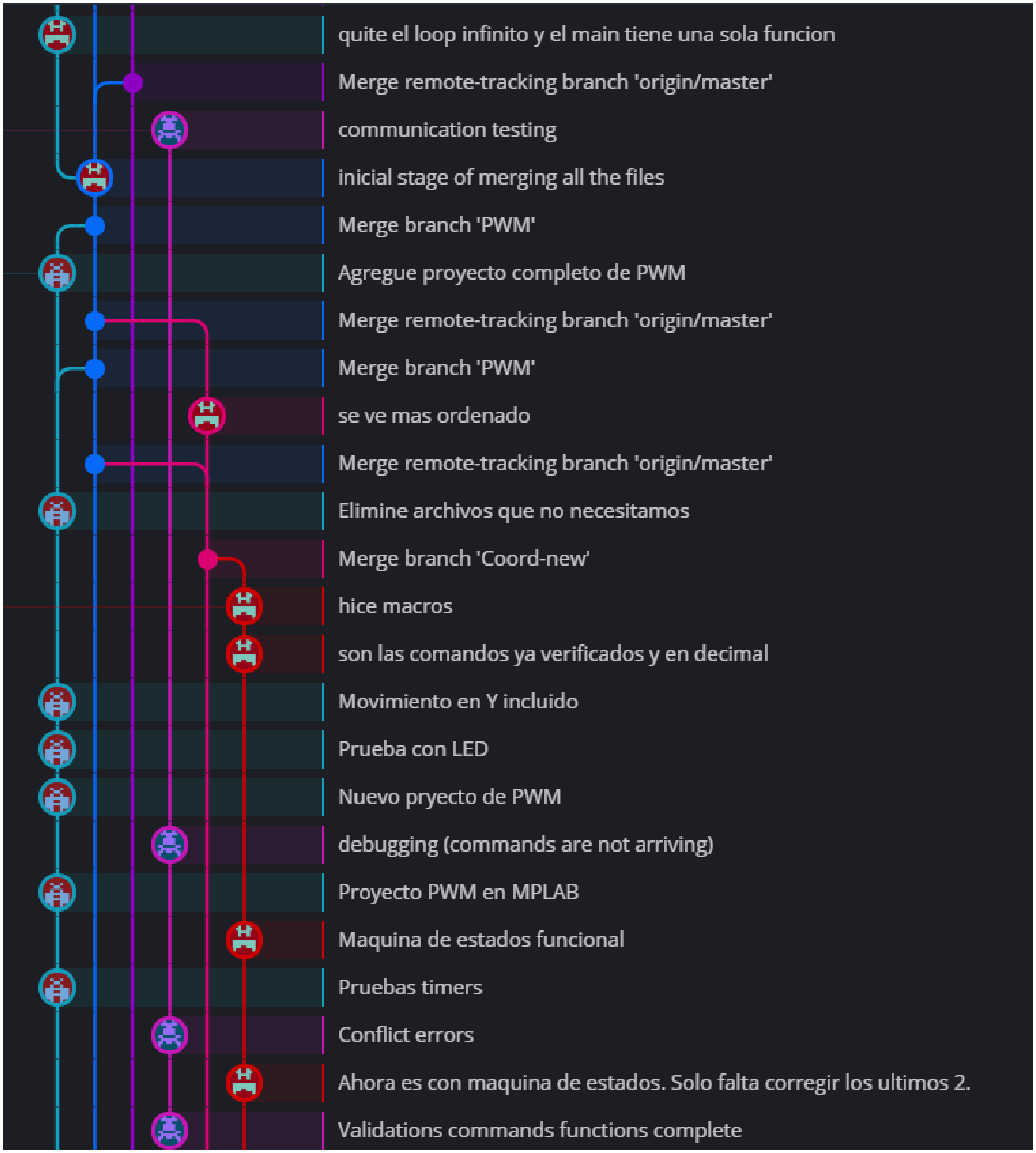
Using GitKraken as a Git GIU, the next images shows the commits and branches the project had.



HEX file added. Interruptions removed
LED messages
Interruptions removed
Files organized
Magic numbers reduction and language changed
Merge branch 'Testing'
Testing Go to coordinate
testing code: failures on interruptions
Restricction read from UART
Code review: Package of data are not sent correctly
PIC programming manual
Test Case Evaluation
Code review: use of less magic numbers
Interruptions and instrucctions coordinates OK
Presentación Final
Coordinate position 0 established
Adding interruptions (pull and push)
Adding interruptions for call instrucctions
Corrección de problemas con coordenadas
WIP on master: Auto stash before merge of "master" and "origin/master"
Adding interrutpions functions
Relative coordinates changed
Programa completo
Otra vez agregué config



Volví a agregar el archivo de configuración
Merge remote-tracking branch 'origin/master'
Agregue EnableB
Auto stash before merge of "master" and "origin/master"
Interruptions configuration for main.c
Instrucctions File.c edited for configuration bits only
Adding Interruption Configuration
Agregue configuración de puertos y pwm
Eliminamos errores
Quitar errores
Ordene los headers
Header Config y archivo movimiento
Agregue header de config y archivo de movimiento"
Revert "Agregue header de config y archivo de movimiento"
Agregue header de config y archivo de movimiento
Agregue archivo solenoide
Agregué header "definiciones.h"
testing cases: instrucction
Agregué un comentario
Comment Functions
Merge branch 'Interruption'
Interruption configuration added
verification test 3 (ascii)
Merge remote-tracking branch 'origin/master'



quite el loop infinito y el main tiene una sola funcion

Merge remote-tracking branch 'origin/master'

communication testing

inicial stage of merging all the files

Merge branch 'PWM'

Agregue proyecto completo de PWM

Merge remote-tracking branch 'origin/master'

Merge branch 'PWM'

se ve mas ordenado

Merge remote-tracking branch 'origin/master'

Elimine archivos que no necesitamos

Merge branch 'Coord-new'

hice macros

son las comandos ya verificados y en decimal

Movimiento en Y incluido

Prueba con LED

Nuevo pryecto de PWM

debugging (commands are not arriving)

Proyecto PWM en MPLAB

Maquina de estados funcional

Pruebas timers

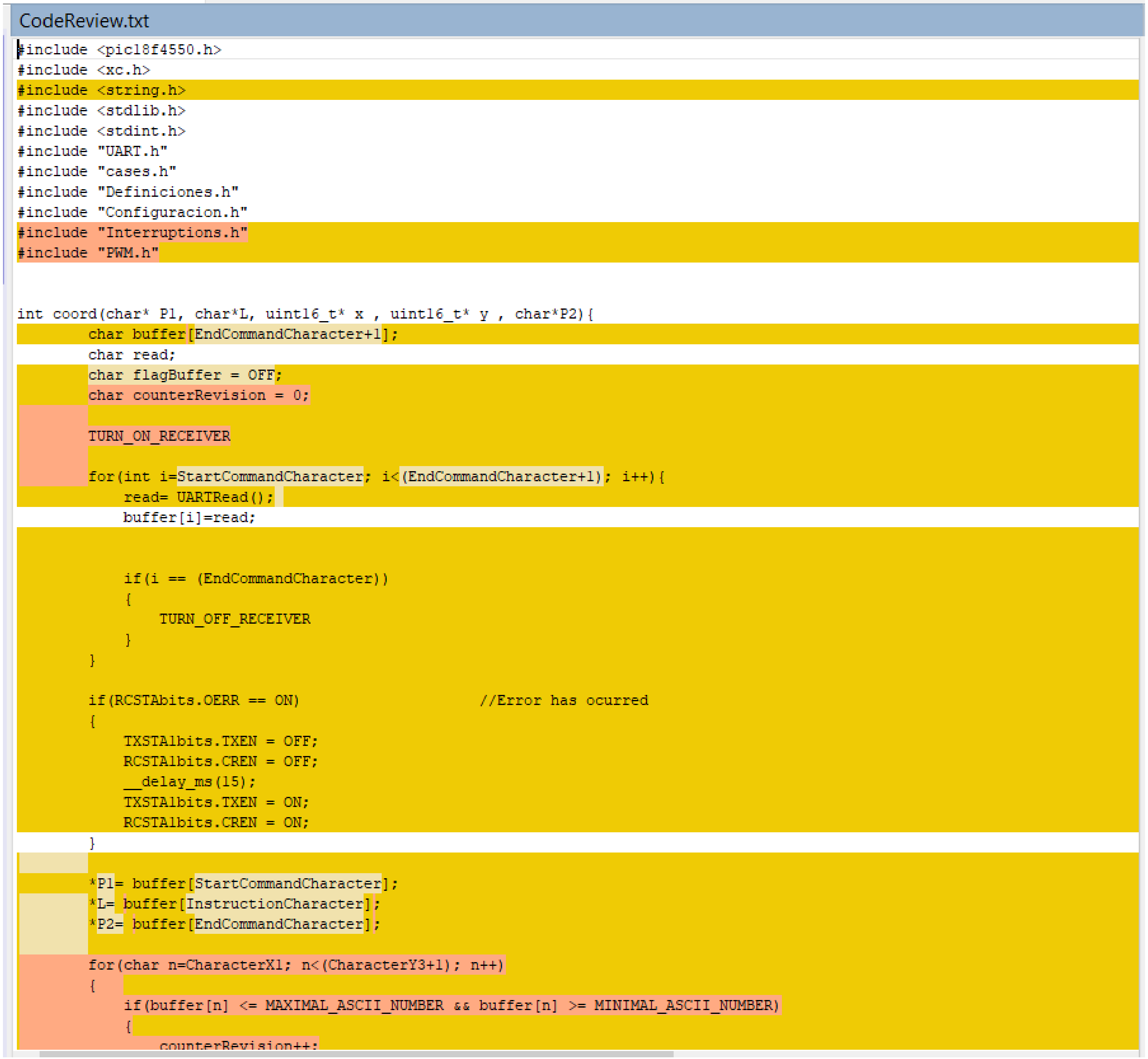
Conflict errors

Ahora es con maquina de estados. Solo falta corregir los ultimos 2.

Validations commands functions complete

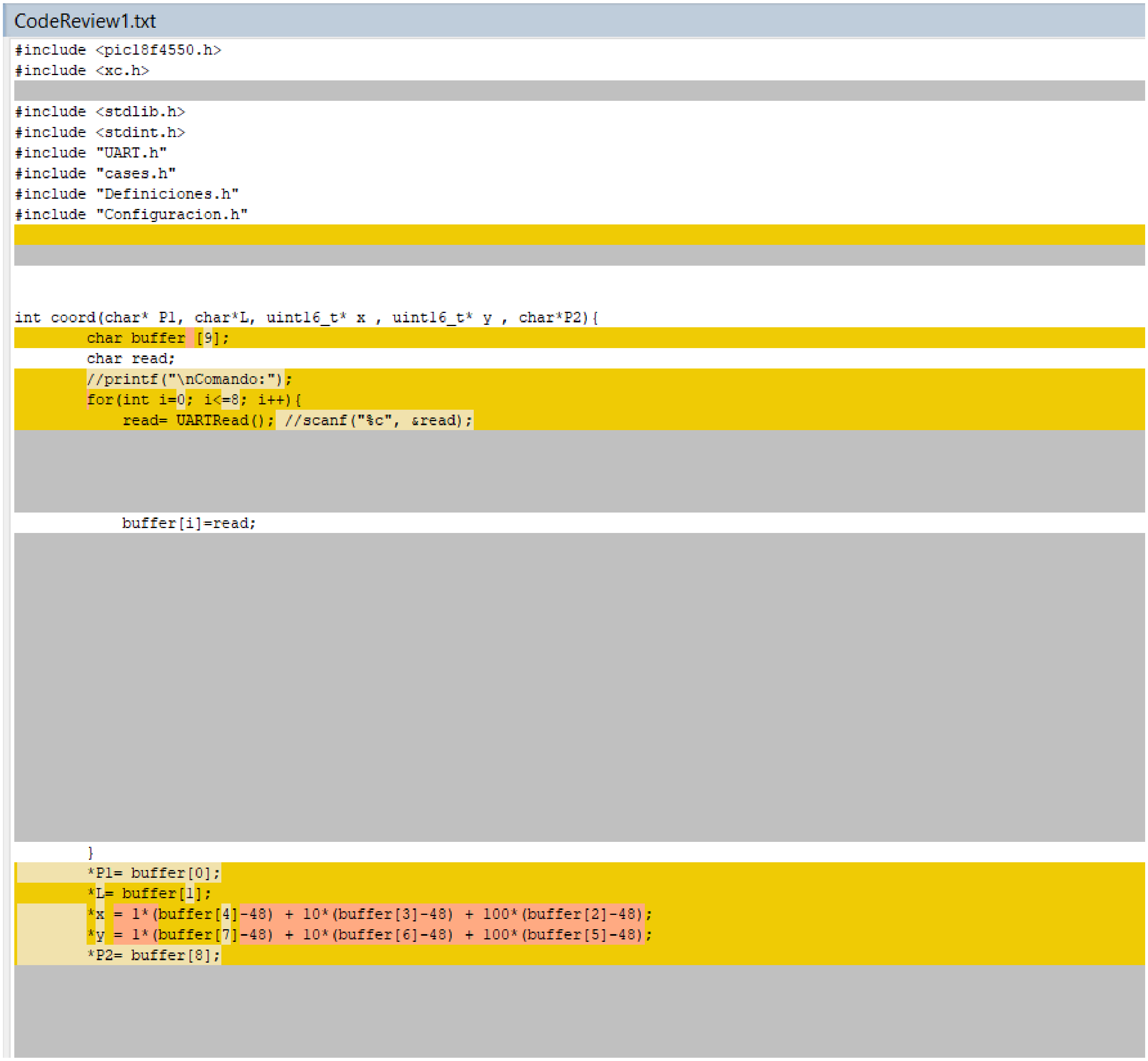
Code review and improvement: Code was improved not only in better algorithm but also on syntax, readability and easier to maintain. Other tools, such TortoiseSVN, helped to compare code.

Using *diff with* tool, it was easier to compare files. An example is this file from the project, where magic numbers were used and some conditions/instrucctions hadn't been implemented.



CodeReview.txt

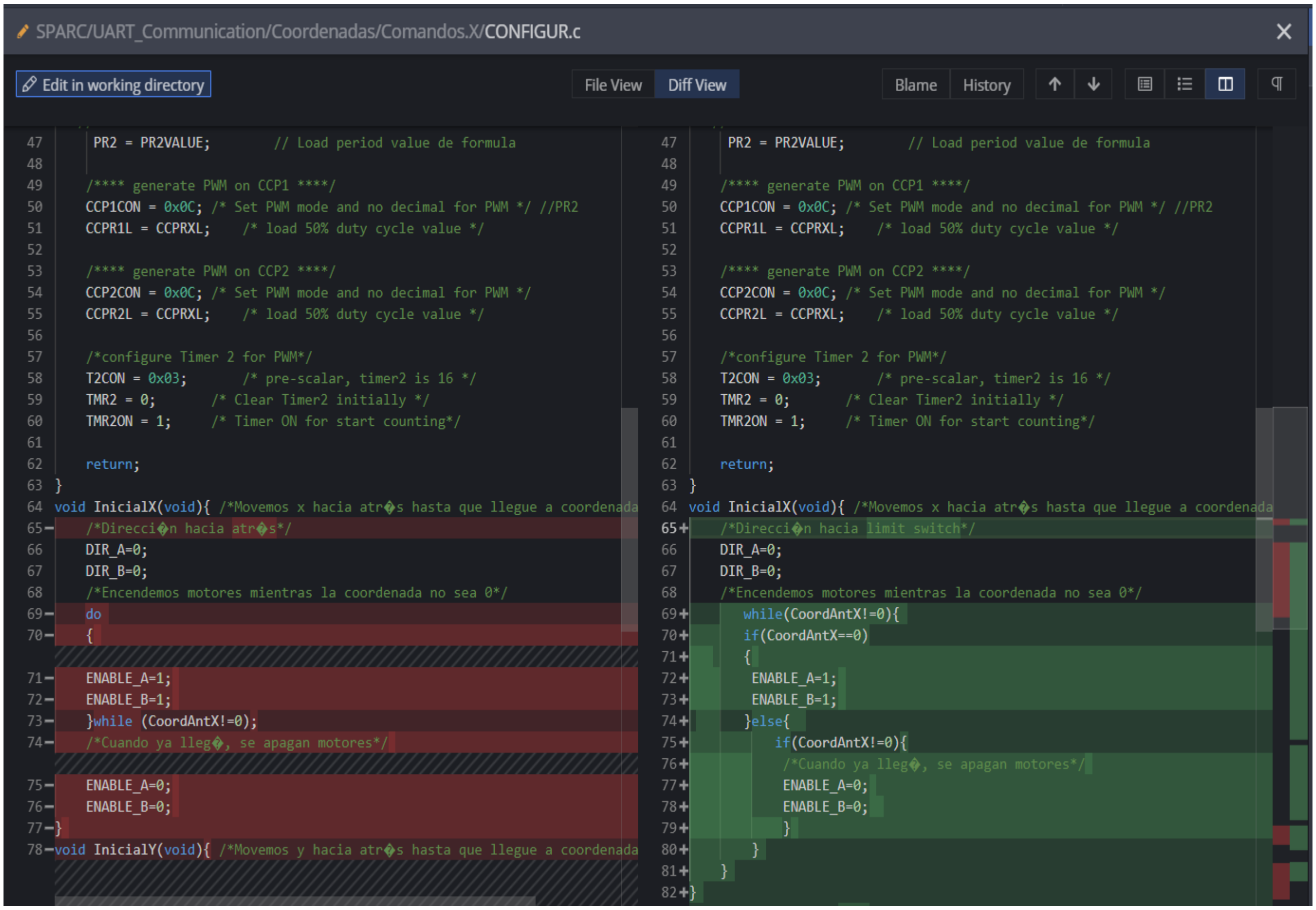
```
#include <pic18f4550.h>
#include <xc.h>
#include <string.h>
#include <stdlib.h>
#include <stdint.h>
#include "UART.h"
#include "cases.h"
#include "Definiciones.h"
#include "Configuracion.h"
#include "Interruptions.h"
#include "PWM.h"


int coord(char* P1, char*L, uint16_t* x , uint16_t* y , char*P2){
        char buffer[EndCommandCharacter+1];
        char read;
        char flagBuffer = OFF;
        char counterRevision = 0;

        TURN_ON_RECEIVER

        for(int i=StartCommandCharacter; i<(EndCommandCharacter+1); i++){
            read= UARTRead();
            buffer[i]=read;


            if(i == (EndCommandCharacter))
            {
                TURN_OFF_RECEIVER
            }
        }

        if(RCSTAbits.OERR == ON)                        //Error has ocurred
        {
            TXSTA1bits.TXEN = OFF;
            RCSTA1bits.CREN = OFF;
            __delay_ms(15);
            TXSTA1bits.TXEN = ON;
            RCSTA1bits.CREN = ON;
        }

        *P1= buffer[StartCommandCharacter];
        *L= buffer[InstructionCharacter];
        *P2= buffer[EndCommandCharacter];

        for(char n=CharacterX1; n<(CharacterY3+1); n++)
        {
            if(buffer[n] <= MAXIMAL_ASCII_NUMBER && buffer[n] >= MINIMAL_ASCII_NUMBER)
            {
                counterRevision++;
```



CodeReview1.txt

```
#include <pic18f4550.h>
#include <xc.h>

#include <stdlib.h>
#include <stdint.h>
#include "UART.h"
#include "cases.h"
#include "Definiciones.h"
#include "Configuracion.h"




int coord(char* P1, char*L, uint16_t* x , uint16_t* y , char*P2){
        char buffer [9];
        char read;
        //printf("\nComando:");
        for(int i=0; i<=8; i++){
            read= UARTRead(); //scanf("%c", &read);

            buffer[i]=read;

        }
        *P1= buffer[0];
        *L= buffer[1];
        *x = 1*(buffer[4]-48) + 10*(buffer[3]-48) + 100*(buffer[2]-48);
        *y = 1*(buffer[7]-48) + 10*(buffer[6]-48) + 100*(buffer[5]-48);
        *P2= buffer[8];
```

Also, GitKraken can compare commits


SPARC/UART_Communication/Coordenadas/Comandos.X/CONFIGUR.c
Edit in working directory
File View
Diff View
Blame
History

```c
     PR2 = PR2VALUE;        // Load period value de formula

    /**** generate PWM on CCP1 ****/
    CCP1CON = 0x0C; /* Set PWM mode and no decimal for PWM */ //PR2
    CCPR1L = CCPRXL;    /* load 50% duty cycle value */

    /**** generate PWM on CCP2 ****/
    CCP2CON = 0x0C; /* Set PWM mode and no decimal for PWM */
    CCPR2L = CCPRXL;    /* load 50% duty cycle value */

    /*configure Timer 2 for PWM*/
    T2CON = 0x03;       /* pre-scalar, timer2 is 16 */
    TMR2 = 0;       /* Clear Timer2 initially */
    TMR2ON = 1;     /* Timer ON for start counting*/

    return;
}
void InicialX(void){ /*Movemos x hacia atr�s hasta que llegue a coordenada
-    /*Direcci�n hacia atr�s*/
    DIR_A=0;
    DIR_B=0;
    /*Encendemos motores mientras la coordenada no sea 0*/
-    do
-    {
-    ENABLE_A=1;
-    ENABLE_B=1;
-    }while (CoordAntX!=0);
-    /*Cuando ya lleg�, se apagan motores*/
-    ENABLE_A=0;
-    ENABLE_B=0;
-}
-void InicialY(void){ /*Movemos y hacia atr�s hasta que llegue a coordenada
```

```c
     PR2 = PR2VALUE;        // Load period value de formula

    /**** generate PWM on CCP1 ****/
    CCP1CON = 0x0C; /* Set PWM mode and no decimal for PWM */ //PR2
    CCPR1L = CCPRXL;    /* load 50% duty cycle value */

    /**** generate PWM on CCP2 ****/
    CCP2CON = 0x0C; /* Set PWM mode and no decimal for PWM */
    CCPR2L = CCPRXL;    /* load 50% duty cycle value */

    /*configure Timer 2 for PWM*/
    T2CON = 0x03;       /* pre-scalar, timer2 is 16 */
    TMR2 = 0;       /* Clear Timer2 initially */
    TMR2ON = 1;     /* Timer ON for start counting*/

    return;
}
void InicialX(void){ /*Movemos x hacia atr�s hasta que llegue a coordenada
+    /*Direcci�n hacia limit switch*/
    DIR_A=0;
    DIR_B=0;
    /*Encendemos motores mientras la coordenada no sea 0*/
+       while(CoordAntX!=0){
+       if(CoordAntX==0)
+       {
+        ENABLE_A=1;
+        ENABLE_B=1;
+       }else{
+           if(CoordAntX!=0){
+            /*Cuando ya lleg�, se apagan motores*/
+            ENABLE_A=0;
+            ENABLE_B=0;
+            }
+        }
+    }
+}
```