THIS ISSUE CAN BE FIXED AT THE COST OF MAKING THE CODE LESS RESPONSIVE AND ADDING MORE **LAG** IN CLOSED LOOP SYSTEM

BUT THIS IS THE BEST POSSIBLE SOLUTION. IT IS PLACED IN THE LAST. CURRENTLY WE DECIDED TO TAKE IT LIKE THAT ONLY.

ISR\_init();

// LED2=1;

for(;;)

{

uart0\_tx\_string(" :::::::::::::: I'm now in main Loop !! :::::::::::: ");

//This function compensates the sen\_data array and puts the result into com\_data array

compensate\_value(sen\_data,com\_data);

……..

…

..

}

// -ve correction means -ve error, ie turn right

// +ve correction means +ve error ie turn left.

servo\_output = 1500-(unsigned int)correction;

servo\_set (servo\_output)

}

ATD0CTL5\_SCAN=0; //This will start a new conversion ....

}

:::::::::::::: I'm now in main Loop !! ::::::::::::

:::::::::::::: I'm

:::: Interrupt ADC complete haha !! :::**m now** in main Loop !! :::::::::::: :w i

:::::::::::::: I'm now in main Loop !! ::::::::::::te haha !! :::

::::::

:::::::::::::: I'm

:::: Interrupt ADC complete haha !! ::: now in ma:::

:::::::::::::: I'm now in main Loop !! ::::::::::::Loop !! :::::::::::: ::::::::

:::::::::::::: I'm

:::: Interrupt ADC complete haha !! :::

p !! ::::::::::::

:::::::::::::: I'm now in main Loop !! :::

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::'m now in main Loop !! :::::::::::: :::

:::: Interrupt ADC complete haha !! :::::::::::::::::::::

in

::

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::m now in main

:::: Interrupt ADC complete haha !! ::::::::::::::oop !! :::::::::::: ::::::::

:::::::::::::: I'm now in main Loop !! ::::::::::::haha !! :::haha !! :::

:::: Interrupt ADC complete haha !! :::::

oop !! :::::::::::: :::

:::::::::::::: I'm now in main Loop !! ::::::::::::::

:::::::::::::: I'

:::: Interrupt ADC complete haha !! :::: now in main Loop !! :::::::::::: ::::

:::::::::::::: I'm now in main Loop !! :::::::::::::

:::::::::::::: I'

:::: Interrupt ADC complete haha !! ::::now in main Loop !! :::::::::::: :::::

:::::::::::::: I'm now in main Loop !! :::::::::::::

L

::

:::::::::::::: I'

:::: Interrupt ADC complete haha !! ::::ow in main Loo

:::::::::::::: I'm now in main Loop !! :::::::::::::

p !! :::::::::::: ::::::::::

:::::::::::::: I'

:::: Interrupt ADC complete haha !! ::::

:::::::::::::: I'm now in main Loop !! ::::

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::::::: :::::::::::::: I'm now

:::::::

:::: Interrupt ADC complete haha !! :::::::::::::::::::::

'

p

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! ::::::: ::::::::

:::::::

:::: Interrupt ADC complete haha !! :::::::::::::

:::::: I'm now in main Loop !

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::haha !! :::

:::::::

:::: Interrupt ADC complete haha !! ::::

:::::::::::::: I'm

:::: Interrupt ADC complete haha !! :::: :::::::::::::: I'm now in main Loop !

:::::::::::::: I'm now in main Loop !! :::::::::::::

! ::

:::::::::::::: I'

:::: Interrupt ADC complete haha !! :::: :::::::::::w

:::::::::::::: I'm now in main Loop !! ::::::::::::::: I'm now in main Loop !! :::

:::::::::::::: I'

:::: Interrupt ADC complete haha !! ::::

:::::::::::::: I'm now in main Loop !! ::::

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::::::::::::::: I'm now in main Loop !! :o

:::: Interrupt ADC complete haha !! :::::::::::::::::::::

::::

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::::::::::::::::

:::: Interrupt ADC complete haha !! :::::::::::::I'm now in main Loop !! ::::::

:::::::::::::: I'm now in main Loop !! ::::::::::::aha !! :::haha !! :::

:::: Interrupt ADC complete haha !! ::::

FIXED DATA

uart0\_9600\_init(); //To be removed in the final version of the software.

// LED2=1;

for(;;)

{

uart0\_tx\_string(" :::::::::::::: I'm now in main Loop !! :::::::::::: ");

//This function compensates the sen\_data array and puts the result into com\_data array

compensate\_value(sen\_data,com\_data);

// Placing it here, will increase the lag between value read and action taken.. hence

// making the system unresponsive ...

ATD0CTL5\_SCAN=0; //This will start a new conversion ....

if(com\_data[0] <=5 && com\_data[1] <=5 && com\_data[2] <=5 && com\_data[3] <=5 && com\_data[4] <=5 && com\_data[5] <=5 && com\_data[6] <=5 && com\_data[7] <=5)

{

//The flag (ATDSTAT0\_SCF=0;) clears automatically when we access status register and then the result register..

//dc\_motor\_stop();