CS 431-1 Embedded Systems

HW 1

Opened: Wednesday, 29 September 2021, 12:00 AM
Due: Friday, 8 October 2021, 4:59 PM
To do: Make a submission

This is a fairly simple lab. The main goal is to get you acquinted with edsim51 and Keil uVision 5 while introducing you to the extreme limitations of your development environment. Instructions are purposefully open-ended because for embedded systems, you'll see that there are hundreds of thousands of boards and development environments. And such, becoming an expert in "1" is absolutely useless. Learning to become an expert is priceless.

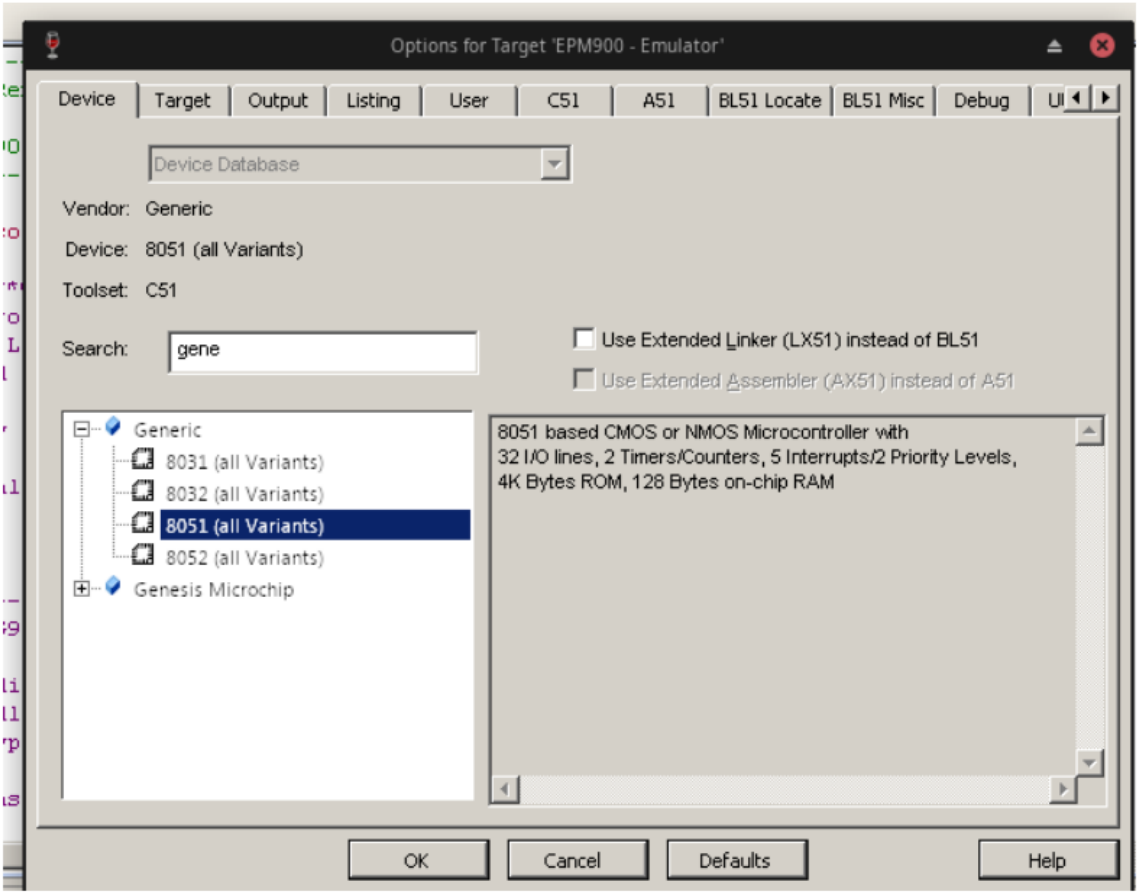
This works perfectly well in Windows, slightly glitchy but errorless in Linux with PlayOnLinux (both verified). If you have macOS, you're on your own*

Grab the modified edsim51 from here, you'll start to need it instead of the original: Modded edsim51 . The original is here: https://www.edsim51.com.

Grab Keil uVision 5 (C51 version) from here: https://www.keil.com/demo/eval/c51.htm

I would advise you to use one of the sample projects to start off from (blinky or event simpler; something without printfs).

Your target (Project->Options for Target) device in Keil uVision should be Generic->8051 (all variants)



And then make sure "Create Hex File" is selected under "Output" tab. Because you will then use the Hex output to feed into edsim51 to simulate your program as explained here: https://www.edsim51.com/intelHex.html

Now the actual lab requirements:

Implement a fixed-size (array-based) multi-function list in C (stack & queue). This fixed size should be a precompiler macro (i.e. #define). Demonstrate that you can:

1. read values from left/right (and reject reading when empty), put current size in a global variable after success (i.e return), else return -1

2. write values to left/right (reject writing if full), put current size in a global after success (i.e. return), else return -1

3. keep current size in a global variable.

Hint1: use a global struct to hold the array, indexes and current size.

Hint2: you can return multiple values via use of pointers

Hint3: a total of 7 functions is expected (append_left, append_right, pop_left, pop_right, seek_left, seek_right, length).

Upload your C files and your project files (.uvproj) in an archive (zip, gzip etc.) for ease of checking for your TA.Late submissions will lose points (could be none, could be all, please dont take the risk).

Notes:

1. Assume there is no dynamic memory allocation, you'll be working with a slightly modified C99 variant. It's a very strict environment. The details (if you're curious) are here: http://www.keil.com/c51/c51.asp. Pretty sure malloc wouldnt work, and be VERY careful when using pointers.

2. You will realise immediately that printf is NOT available :'). However edsim51 allows you to look into RAM contents. Demonstrations should be done on this basis.

3. structs and unions are your frenemies, be VERY careful if and how you use them.

I'm opening a discussion thread (HW 1 Discussion) for potential questions. You should use here for discussions not private emails.

Grading:

Each item is 30 points, and remaining 10 is for in-demo questions.

*edsim51 is java so it'll work -very poorly- because java on macOs.... Keil may work with wine under macOS but you may as well buy a lottery ticket and hope you'll win it. I have set up an ubuntu/manjaro VM and work with that. Just my two cents but; next time you're buying a computer, buy a PC and you may just save yourselves a few rents.

If you're feeling more adventureous: see if you can output the "return values" to LEDs
If you're feeling more adventureous: See here how you can -sorta- have dynamic memory and play around.
https://www.keil.com/support/man/docs/c51/c51_ap_memalloc.htm

Submission status

Submission status	No attempt
Grading status	Not graded
Time remaining	7 days
Last modified	-
Submission comments	Comments (0)

Add submission

You have not made a submission yet.

◄ Summer 2019 Lecture Notes

Jump to...

HW 1 Discussion ►