Building an Allegro Library Game: Lander-2.5

Table of Contents
Locate and Download Lander-2.5 Source.................................................................2
Locate, Download, and Unzip DUMB Library......................................................3
Build and install DUMB Library........................................................................4
  Open a command window inside the DUMB folder .........................................4
  Build DUMB Library........................................................................................5
  Install DUMB libraries inside MinGW...............................................................6
  DUMB Special Requirements...........................................................................7
  Location of installed libraries..........................................................................8
Extract lander-2.5 sources..................................................................................9
Use CodeBlocks to Create and Build a Lander Project.......................................10
  Project: Create Lander Project......................................................................10
  Project: Add all sources to the active project................................................11
  Project: Add required Linker settings for allegro build process......................12
  Project: Build Lander......................................................................................13
  Project: Run Lander.........................................................................................13
  Project: Running on another computer.........................................................14
  Project: Turn in screen capture of running lander........................................14
  Project: Unzipped lander folder......................................................................15
Locate and Download Lander-2.5 Source

Go to allegro web site (www.allegro.cc).

Locate and download Lander 2.5 version source code. Search for lander program on allegro home page. Search option is in the list of options in the list of on the upper left of the page.

Illustration 1: Allegro.cc home page

Illustration 2: Lander-2.5 page on Allegro.cc Depot
Locate, Download, and Unzip DUMB Library.

Lander also requires the DUMB library for audio output even though the documentation does not indicate its need. When trying to compile without DUMB, an error message about missing `aldumb.h` is printed. This error message indicates that the dumb library is missing.


Follow the link to the home page for DUMB. Download a copy of the library source code, version 0.9.3. Unzip DUMB to its own folder.
Build and install DUMB Library

These instructions specifically use the MinGW compiler included with the CodeBlocks distribution (see Exercise 1).

Use the Windows explorer to navigate inside the DUMB folder where the following view (Illustration 4) appears. Select (highlight) the full path and Edit->Copy to clipboard to paste as shown below.

Illustration 4: Dumb library folder (with path to copy highlighted)

Open a command window inside the DUMB folder.

Do this by Start->Run which opens as follows:

Illustration 5: Run Cmd with options and paste path

Type `cmd /k cd` as shown (Illustration 5) and paste the path to inside the DUMB folder. Make sure to add a quotation mark to the path beginning and end (Illustration 6). A console window will open inside the DUMB folder to compile the library.
**Build DUMB Library**

Build the DUMB library inside the command window. Enter: `mingw32-make` to build the library.

Answer the questions as shown on the screen shown in Illustration 7.
Illustration 8 shows a successful build result.

Install DUMB libraries inside MinGW

Type `mingw32-make install` in the command window (Illustration 9) to insert the libraries inside the `MinGW\lib` and `MinGW\include` folders, completing the install. (Illustration 11)
DUMB Special Requirements

Note: Some libraries have special requirements in order to be used. The following information is included in the DUMB HOWTO Documentation. The Lander program will not link properly without following this advice exactly. Correct usage of the library is shown later in this document. (Illustration 17)

Illustration 10: DUMB requirements for linker in CodeBlocks
Location of installed libraries

Illustration 11 shows a view of the MinGW/lib folder where the mingw32-make install command placed the libraries inside the library folder.
Extract lander-2.5 sources

Extract the lander-2.5 sources to a folder. This is illustrated below with the extract wizard. (Illustration 12)

Once the folder is created the contents are extracted. Open the uncompressed folder with Explorer and note the path to the inner folder named `lander`. (Illustration 13)
Use CodeBlocks to Create and Build a Lander Project

Project: Create Lander Project

Create a CodeBlocks project forcing CodeBlocks to use the inner lander folder. Open CodeBlocks and select File->New->Project. Create a new C Language Console project which will open the wizard pictured in Illustration 14. The shown sequence forces CodeBlocks to create its configuration file inside the lander folder (Illustration 22) instead of creating a new folder. To do this, it is necessary to use the names exactly as shown in Illustration 14.

Note that creating the project creates a main.c file (shown in the Management window - Illustration 15). This file could be a problem with another game using the same name as one of its files, but not with lander-2.5 which uses main.cpp. For another project, a blank project could be created instead of console project which would not create a main.c file.

Illustration 14: Project Wizard dialogs
As shown, set the Project title to the same name as the folder (lander). Activate the folder browse dialog by selecting the button at Folder to create project in: Select lander-2.5 as shown with red box in Illustration 14.
Project: Add all sources to the active project

Within CodeBlocks, go to Project->Add files recursively. Allow the automatic selection process to check all detected sources, then click the OK button.

Illustration 15: Project-> Add Files Recursively

The automatic selection process adds some extra files that will not build properly. These need to be eliminated from the active project.

To eliminate the extra sources from the project, right click on the file name and then pick Remove file from project on the pop-up dialog. Three files need to be eliminated listed below and shown in Illustration 16 in the file management portion of the CodeBlocks workspace.

Eliminate the following extra files, one at a time (Illustration 16):

- coreconv.cpp
- coreset.cpp
- main.c
Project: Add required Linker settings for allegro build process

CodeBlocks project Build options need to specifically add allegro and DUMB libraries to the Linker settings as shown in Illustration 17. The dialog can be selected by Project->Build Options->Linker Settings. The libraries have to be set in this specific order (see Illustration 10). Click the Add button to enter each of the following libraries, one at a time:

- libaldmb.a
- libdumb.a
- liballeg.a
Project: Build Lander

Config and data files need to be in the same directory to run lander. Move the following files to the Release directory from the original unzipped lander folder (see Illustration 22).

Illustration 18: Successful Build of Lander

Project: Run Lander

Config and data files need to be in the same directory to run lander. Move the following files to the Release directory from the original unzipped lander folder (see Illustration 22).

Illustration 19: Release Directory contents
Project: Running on another computer

If the lander-2.5 is desired to run on another computer, a copy of the allegro dynamic load library needs to be included in the same directory as lander.exe and its configuration and data files. The alleg42.dll library can be obtained from the `all42\allegro\lib\mingw32` library directory.

The collective files in the Release folder can then be copied to another computer and run as a complete program.

Project: Turn in screen capture of running lander and `\bin\Release` folder

Make a copy of the running version of lander-2.5 as shown in Illustration 21. Use screen print procedure to make copy. Do the same thing with a view of the `\bin\Release` folder (as shown in Illustration 19). This will show the configuration and data files along with the executable. Turn in this copy as evidence of completion of Exercise 2.

Illustration 21: Capture of Lander running in window mode
Project: Unzipped lander folder

This view also shows the lander.cbp project file used with CodeBlocks.

Illustration 22: Lander folder contents