



Synerduino STM Fork of INAV and the
Synerduino Target

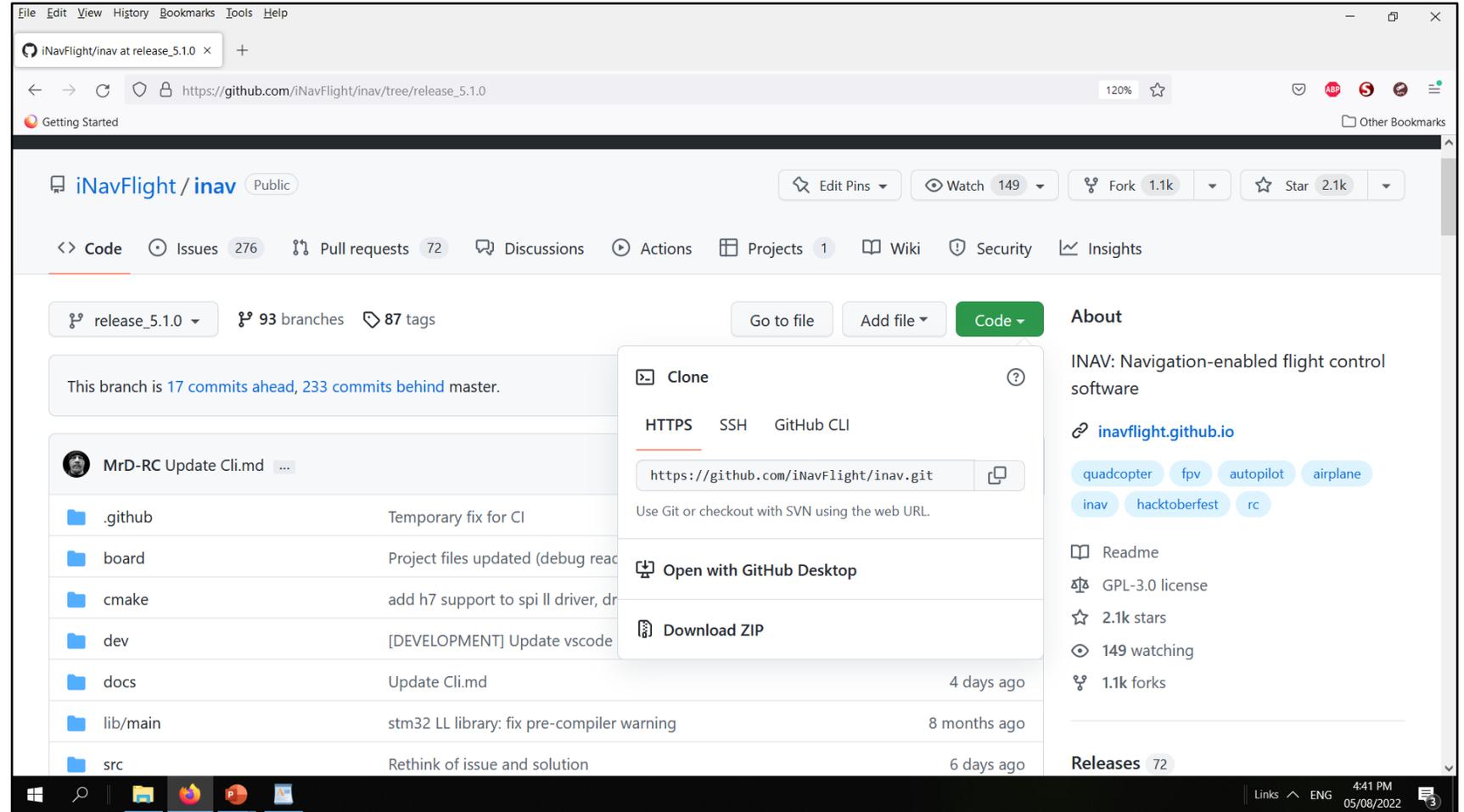
Synerduino STM Target
Compilation into Hexfile

Download the INAV
Release 5.1.0
Release 6.0.0-FP1

First you need to download
the INAV release version

These are available in the
official github page or
Synerflight SynerduinoSTM in
download tab

Prepare to Build the Hex File



The screenshot shows the GitHub repository page for iNavFlight/inav at the release_5.1.0 tag. The repository is public and has 149 watchers and 1.1k forks. The repository structure is displayed, including folders like .github, board, cmake, dev, docs, lib/main, and src. A 'Code' dropdown menu is open, showing options to clone the repository using HTTPS, SSH, or GitHub CLI, or to download the ZIP file. The repository is public and has 149 watchers and 1.1k forks.

https://github.com/iNavFlight/inav/tree/release_5.1.0

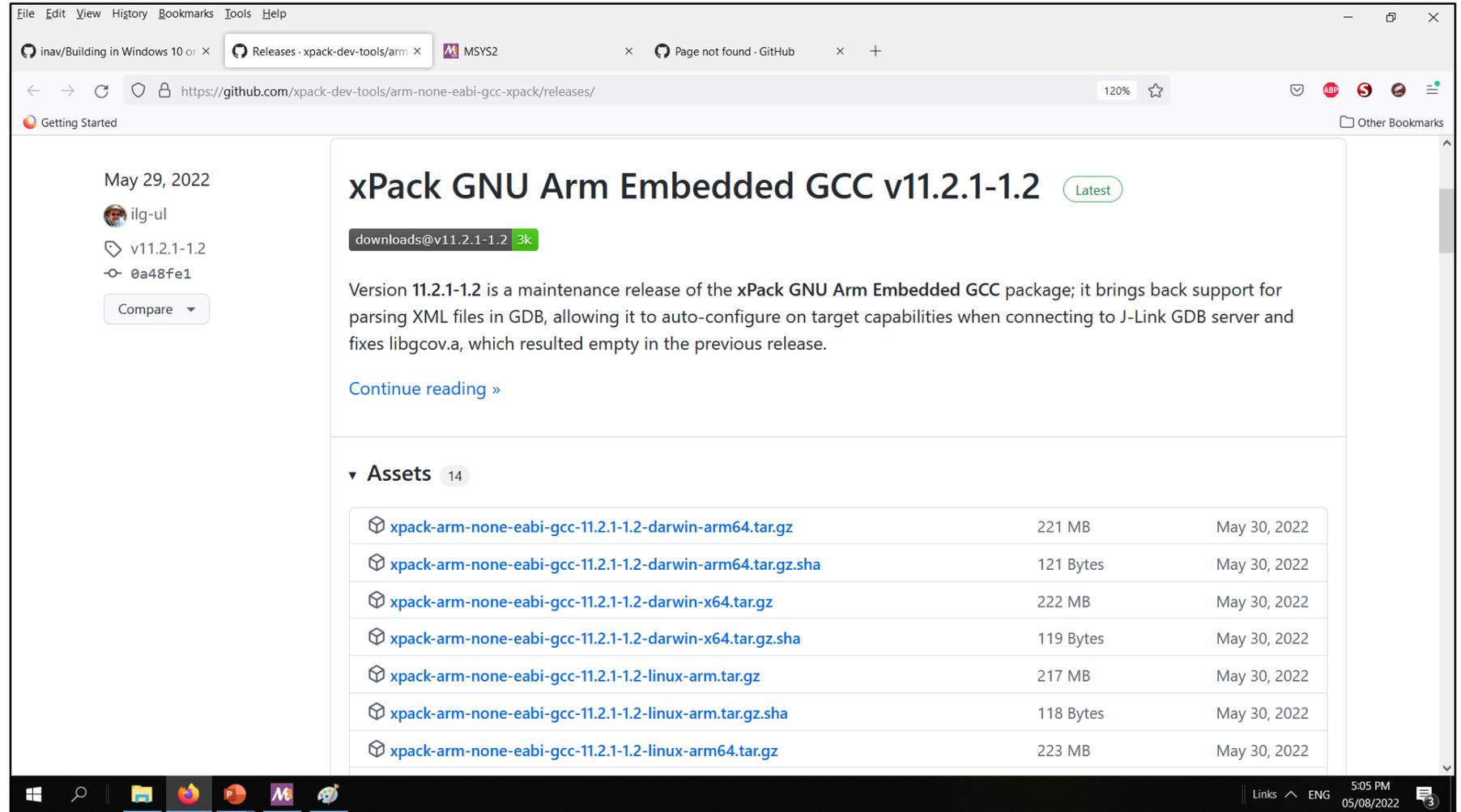
<https://github.com/iNavFlight/inav/tree/6.0.0-FP1>

Download the Xpack
Release 9.2.1-1.1-Win32-
x64 or higher

Im using
INAV-5.1.0
INAV-6.0.0-FP1

xpack-arm-none-eabi-gcc -10.2.1-1.1

These are available in the
official github page or
Synerflight SynerduinoSTM in
download tab



May 29, 2022
ilg-ul
v11.2.1-1.2
0a48fe1
Compare

xPack GNU Arm Embedded GCC v11.2.1-1.2 Latest

downloads@v11.2.1-1.2 3k

Version 11.2.1-1.2 is a maintenance release of the xPack GNU Arm Embedded GCC package; it brings back support for parsing XML files in GDB, allowing it to auto-configure on target capabilities when connecting to J-Link GDB server and fixes libgcov.a, which resulted empty in the previous release.

[Continue reading »](#)

Assets 14

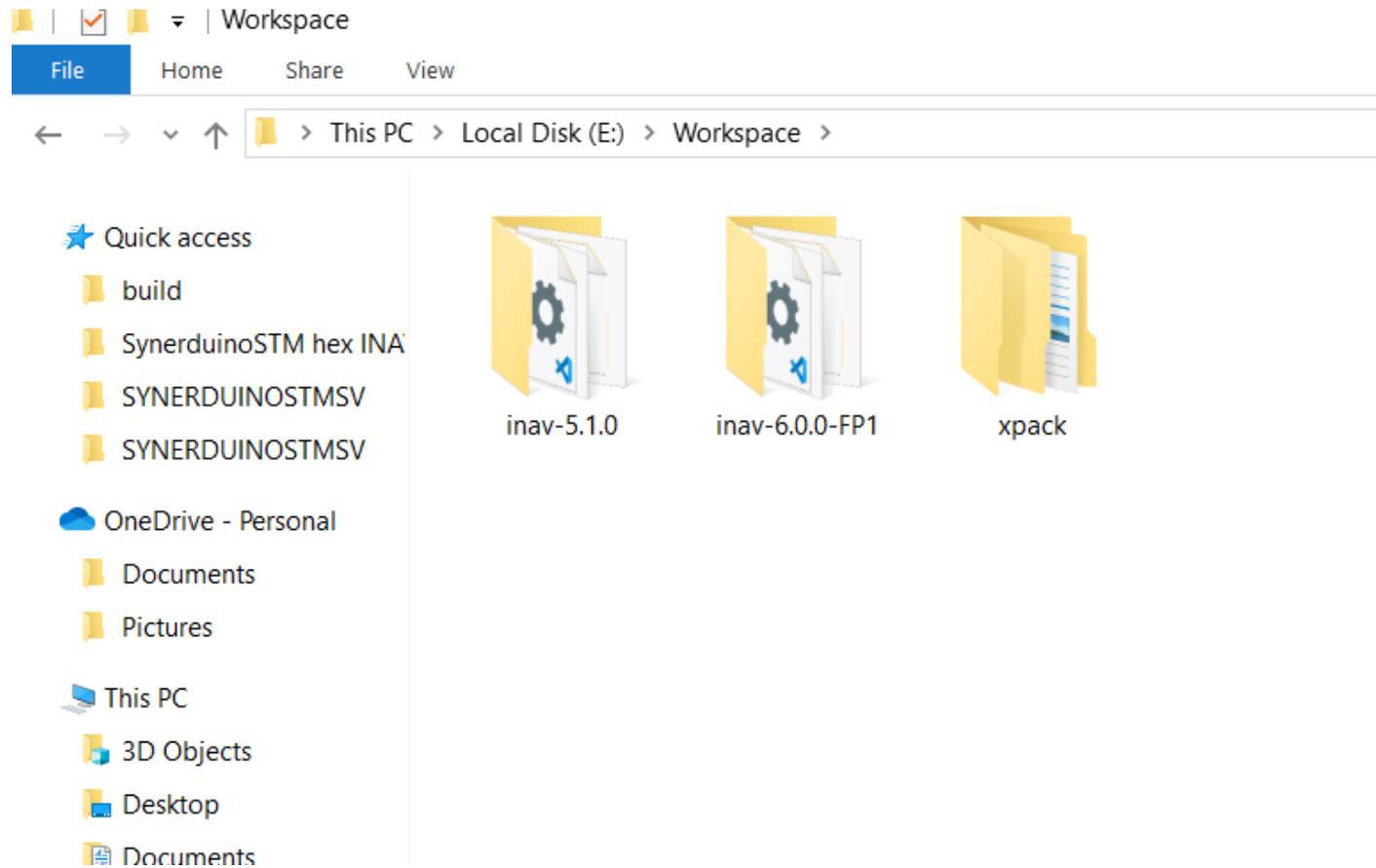
xpack-arm-none-eabi-gcc-11.2.1-1.2-darwin-arm64.tar.gz	221 MB	May 30, 2022
xpack-arm-none-eabi-gcc-11.2.1-1.2-darwin-arm64.tar.gz.sha	121 Bytes	May 30, 2022
xpack-arm-none-eabi-gcc-11.2.1-1.2-darwin-x64.tar.gz	222 MB	May 30, 2022
xpack-arm-none-eabi-gcc-11.2.1-1.2-darwin-x64.tar.gz.sha	119 Bytes	May 30, 2022
xpack-arm-none-eabi-gcc-11.2.1-1.2-linux-arm.tar.gz	217 MB	May 30, 2022
xpack-arm-none-eabi-gcc-11.2.1-1.2-linux-arm.tar.gz.sha	118 Bytes	May 30, 2022
xpack-arm-none-eabi-gcc-11.2.1-1.2-linux-arm64.tar.gz	223 MB	May 30, 2022

<https://github.com/xpack-dev-tools/arm-none-eabi-gcc-xpack/releases/>

<https://github.com/xpack-dev-tools/arm-none-eabi-gcc-xpack/releases/download/v9.2.1-1.1/xpack-arm-none-eabi-gcc-9.2.1-1.1-win32-x64.zip>

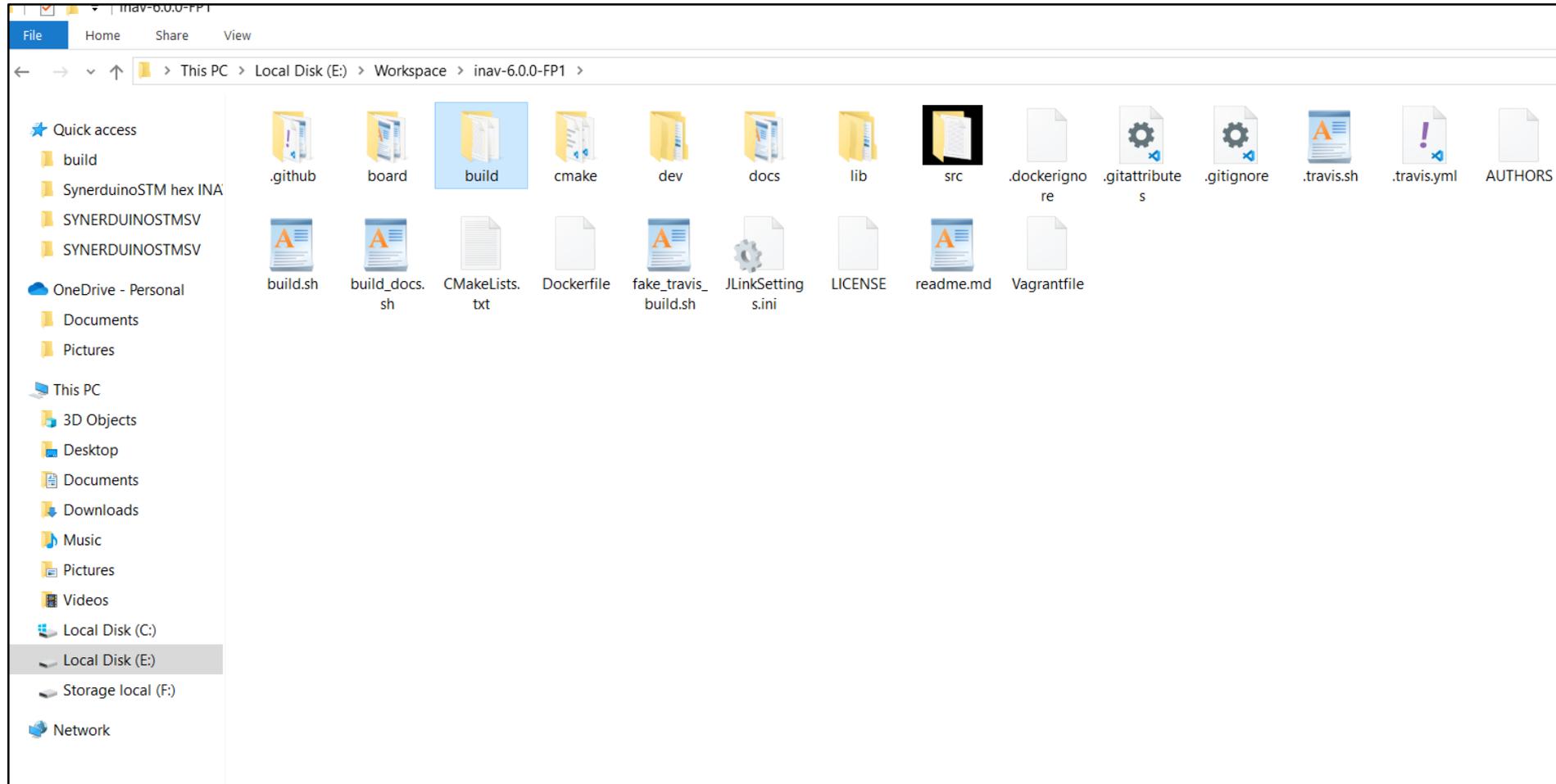
<https://github.com/xpack-dev-tools/arm-none-eabi-gcc-xpack/releases/download/v10.2.1-1.1/xpack-arm-none-eabi-gcc-10.2.1-1.1-win32-x64.zip>

After installation Go Create a Workspace folder in your local Drive in my case Drive E , and extract and place the following directories [INAV-5.1.0 or INAV-6.0.0-FP1 Folder](#) , [Xpack folder](#)

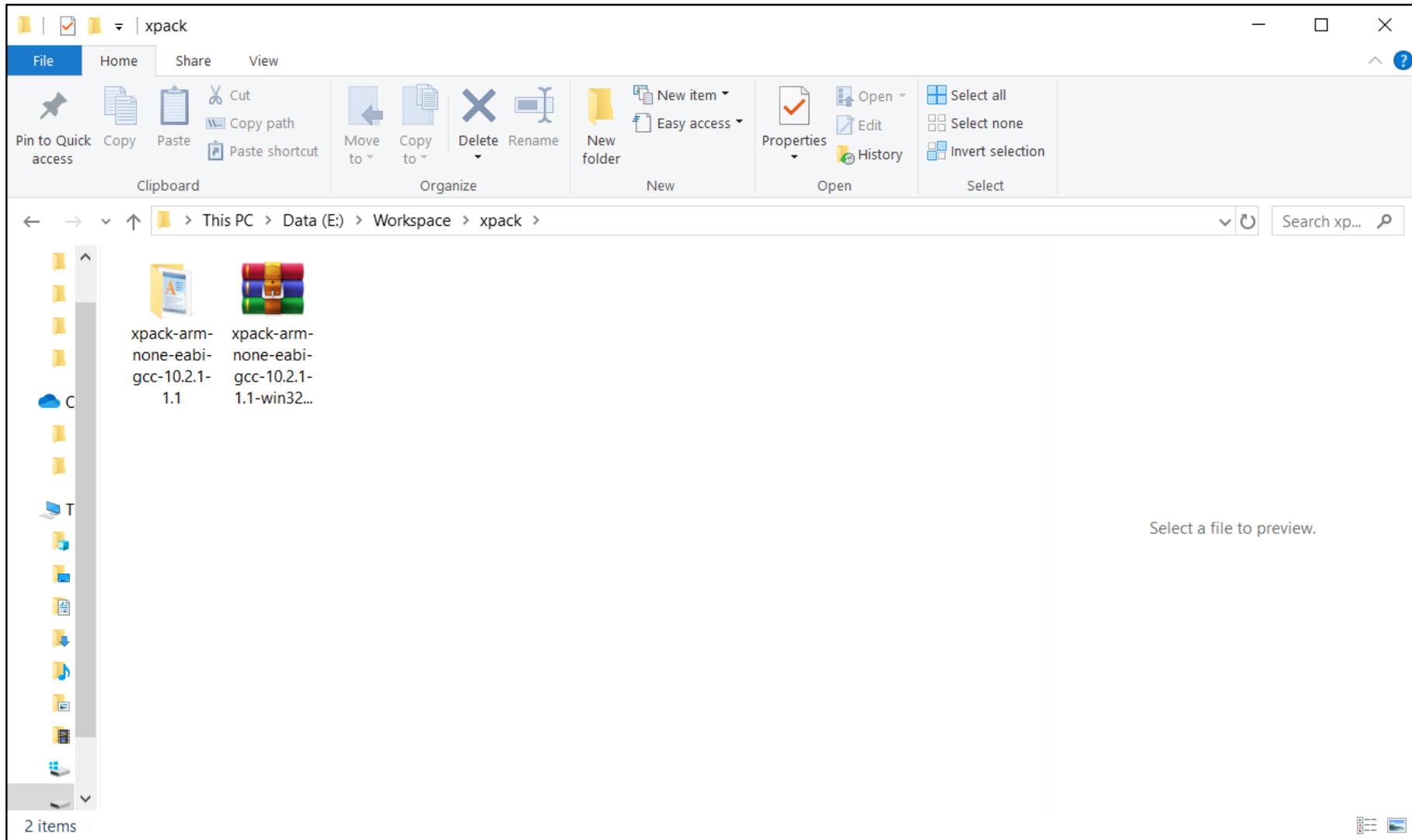


Note: I place my Workspace Folder on Drive E: as I was using SSD on my Drive C so not to fill it up

Create new folder inside INAV 5.1.0 or INAV 6.0.0-FP1 Directory and Name it build



Inside the Xpack Folder Extract the xpack-arm-none-eabi-gcc -10.2.1-1.1



Im using xpack-arm-none-eabi-gcc -
10.2.1-1.1

xpack-arm-none-eabi-gcc-10.2.1-1.1 is extracted to the Xpack folder

Download the Synerduino Target Folder and Extract to Local drive and place it in the following (E:/Workspace/inav-5.1.0/scr/main/target) or (E:/Workspace/inav-6.0.0-FP1/scr/main/target)
Readme file is also available in the download

The screenshot shows a Windows File Explorer window titled 'target'. The address bar indicates the path: > This PC > Local Disk (E:) > Workspace > inav-6.0.0-FP1 > src > main > target >. The main pane displays a list of files and folders with columns for Name, Date modified, Type, and Size. The following table represents the data shown in the file list:

Name	Date modified	Type	Size
SPEEDYBEEF4	07/11/2022 9:15 PM	File folder	
SPEEDYBEEF7	07/11/2022 9:15 PM	File folder	
SPEEDYBEEF7MINI	07/11/2022 9:15 PM	File folder	
SPEEDYBEEF7V2	07/11/2022 9:15 PM	File folder	
SPEEDYBEEF7V3	07/11/2022 9:15 PM	File folder	
SPEEDYBEEF405V3	07/11/2022 9:15 PM	File folder	
SPEEDYBEEF745AIO	07/11/2022 9:15 PM	File folder	
SPRACINGF4EVO	07/11/2022 9:15 PM	File folder	
SPRACINGF7DUAL	07/11/2022 9:15 PM	File folder	
SYNERDUINO	08/11/2022 9:01 A...	File folder	
SYNERDUINO2	08/11/2022 9:01 A...	File folder	
SYNERDUINOSTM	07/12/2022 6:43 PM	File folder	
SYNERDUINOSTMSV	14/12/2022 1:45 PM	File folder	
TMOTORF7	07/11/2022 9:15 PM	File folder	
TMOTORF7V2	07/11/2022 9:15 PM	File folder	
YUPIF4	07/11/2022 9:15 PM	File folder	
YUPIF7	07/11/2022 9:15 PM	File folder	
ZEEZF7	07/11/2022 9:15 PM	File folder	
CMakeLists.txt	22/10/2022 10:38 ...	Text Document	1 KB
common.h	22/10/2022 10:38 ...	C Header Source F...	5 KB
common_hardware.c	22/10/2022 10:38 ...	C Source File	18 KB
common_post.h	22/10/2022 10:38 ...	C Header Source F...	3 KB
sanity_check.h	22/10/2022 10:38 ...	C Header Source F...	3 KB
stm32f7xx_hal_conf.h	22/10/2022 10:38 ...	C Header Source F...	17 KB
stm32h7xx_hal_conf.h	22/10/2022 10:38 ...	C Header Source F...	15 KB
system.h	22/10/2022 10:38 ...	C Header Source F...	1 KB
system_stm32f4xx.c	08/11/2022 9:25 A...	C Source File	48 KB
system_stm32f4xx.h	22/10/2022 10:38 ...	C Header Source F...	2 KB
system_stm32f7xx.c	22/10/2022 10:38 ...	C Source File	14 KB
system_stm32f7xx.h	22/10/2022 10:38 ...	C Header Source F...	2 KB
system_stm32h7xx.c	22/10/2022 10:38 ...	C Source File	32 KB
system_stm32h7xx.h	22/10/2022 10:38 ...	C Header Source F...	4 KB

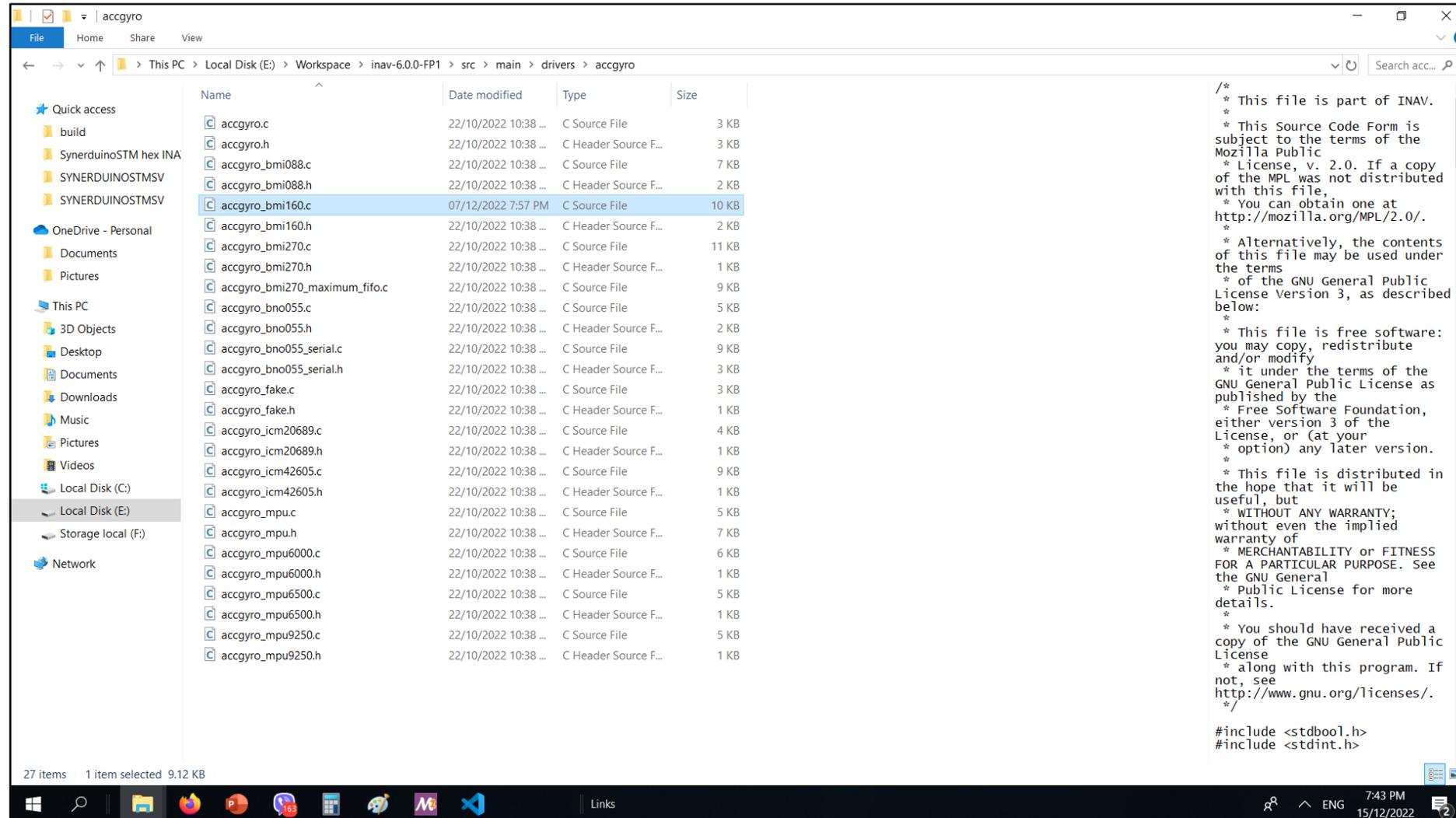
The right-hand pane shows the contents of a selected file, which is a license agreement for MCD Application. The text is as follows:

```
/**
*****
*****
*****
* @file    system_stm32f4xx.h
* @author  MCD Application
*****
Team
* @version V1.6.1
* @date    21-October-2015
* @brief   CMSIS Cortex-M4
Device System Source File for
STM32F4xx devices.
*****
*****
*****
* @attention
*
* <h2><center>&copy;
COPYRIGHT 2015
STMicroelectronics</center>
</h2>
*
* Licensed under MCD-ST
Liberty SW License Agreement
V2, (the "License");
* You may not use this file
except in compliance with the
License.
* You may obtain a copy of
the License at:
*
*
* http://www.st.com/software_lic
nse_agreement_liberty_v2
*
* Unless required by
applicable law or agreed to in
writing, software
* distributed under the
License is distributed on an
"AS IS" BASIS,
* WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either
express or implied.
* See the License for the
specific language governing
permissions and
* limitations under the
License.
**
```

Other Download of the Synerduino Target Folder is the sensor driver this is also Extract to Local drive and place it in the following

(E:/Workspace/inav-5.1.0/src/main/drivers/accgyro) or (E:/Workspace/inav-6.0.0-FP1/src/main/drivers/accgyro)

Readme file is also available in the download (accgyro_BMI160) is also modify to use with the SynerduinoSTM shield



General Info

This is a guide on how to use Windows MSYS2 distribution and building platform to build iNav firmware. This environment is very simple to manage and does not require installing docker for Windows which may get in the way of VMWare or any other virtualization software you already have running for other reasons. Another benefit of this approach is that the compiler runs natively on Windows, so performance is much better than compiling in a virtual environment or a container.

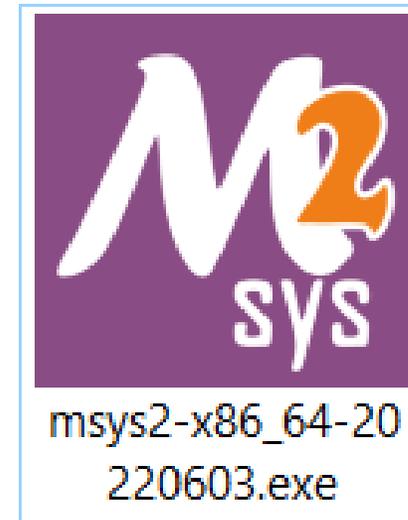
Download Arm Embedded GCC toolkit from The xPack Project

<https://xpack.github.io/arm-none-eabi-gcc/>

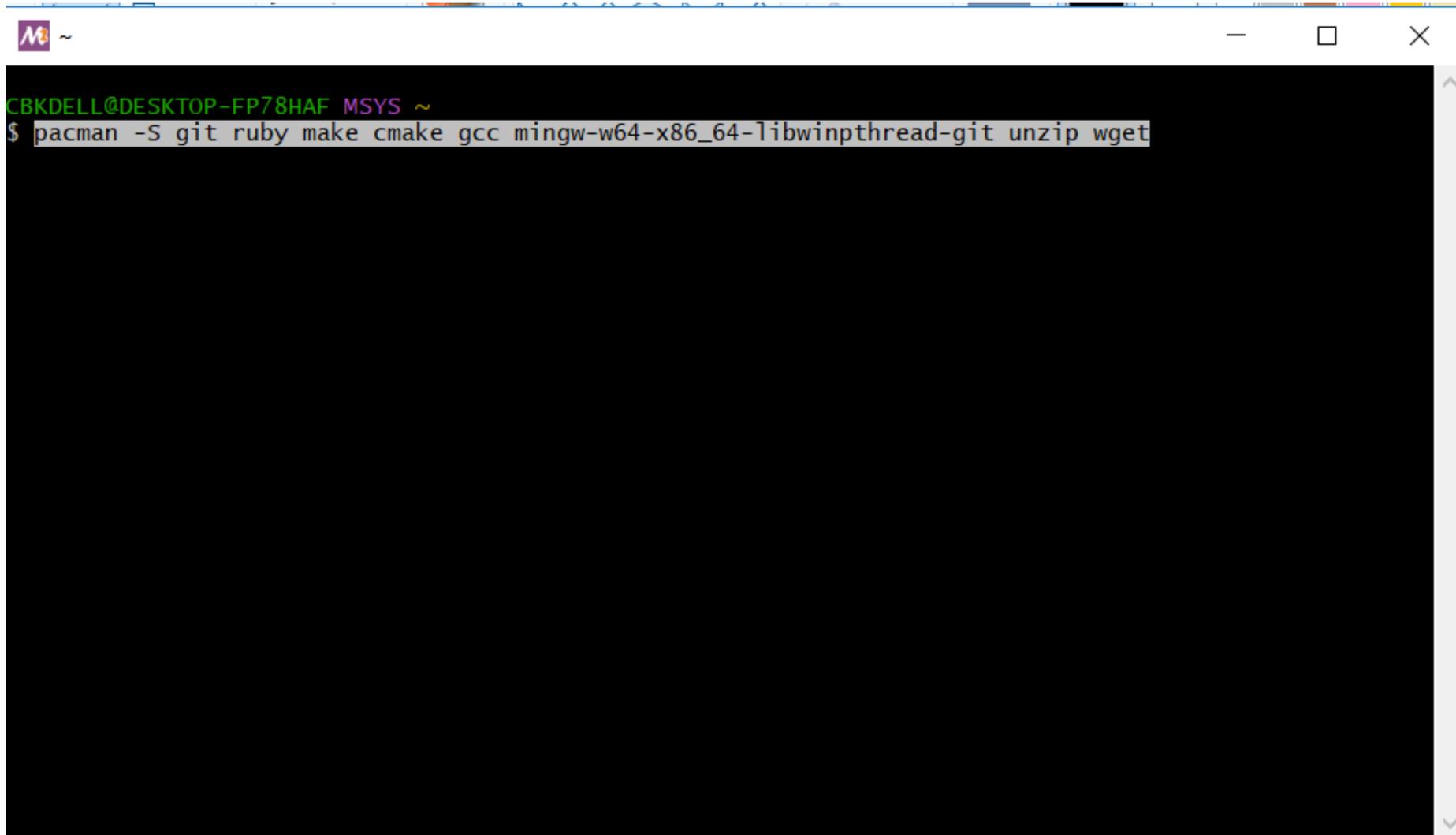
Download MSYS2 for your architecture (most likely 64-bit)

<https://www.msys2.org/wiki/MSYS2-installation/>

Install and Run msys2-x86



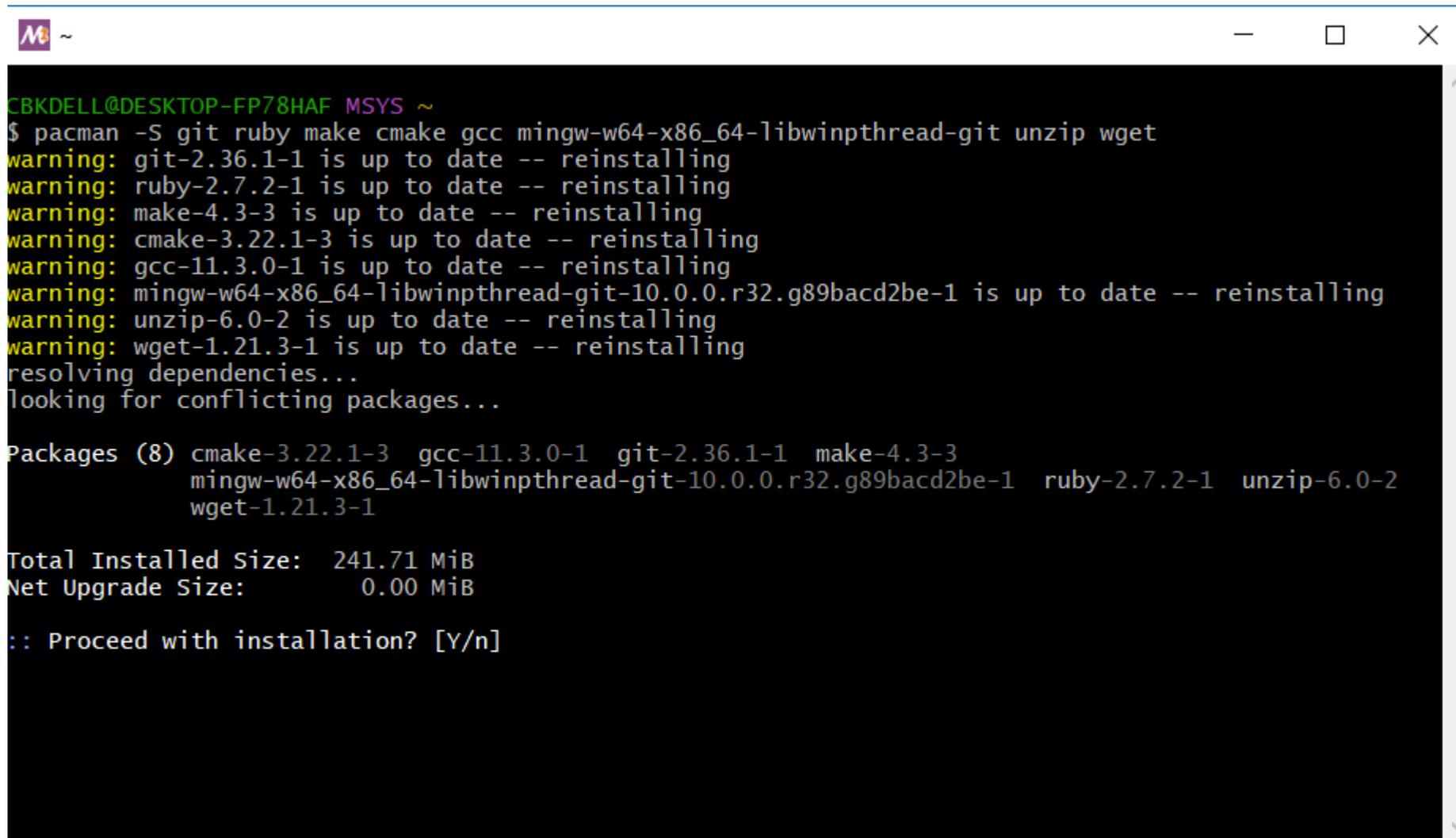
Copy and paste this script to install Ruby ,Cmake , Make and Gcc

A terminal window with a black background and white text. The window title bar shows a purple icon and a tilde symbol. The prompt is 'CBKDELL@DESKTOP-FP78HAF MSYS ~'. The command 'pacman -S git ruby make cmake gcc mingw-w64-x86_64-libwinpthread-git unzip wget' is entered and highlighted with a light gray background. The rest of the terminal is empty.

```
CBKDELL@DESKTOP-FP78HAF MSYS ~  
$ pacman -S git ruby make cmake gcc mingw-w64-x86_64-libwinpthread-git unzip wget
```

```
pacman -S git ruby make cmake gcc mingw-w64-x86_64-libwinpthread-git unzip wget
```

Y to Proceed installation

A terminal window with a black background and white text. The window title bar shows a purple icon, a tilde symbol, and standard window controls (minimize, maximize, close). The terminal output shows a pacman command being executed, followed by several warning messages about packages being up to date and being reinstalled. It then lists the packages to be installed, their total size, and the net upgrade size. Finally, it prompts the user to proceed with installation, with 'Y/n' as the expected input.

```
CBKDELL@DESKTOP-FP78HAF MSYS ~  
$ pacman -S git ruby make cmake gcc mingw-w64-x86_64-libwinpthread-git unzip wget  
warning: git-2.36.1-1 is up to date -- reinstalling  
warning: ruby-2.7.2-1 is up to date -- reinstalling  
warning: make-4.3-3 is up to date -- reinstalling  
warning: cmake-3.22.1-3 is up to date -- reinstalling  
warning: gcc-11.3.0-1 is up to date -- reinstalling  
warning: mingw-w64-x86_64-libwinpthread-git-10.0.0.r32.g89bacd2be-1 is up to date -- reinstalling  
warning: unzip-6.0-2 is up to date -- reinstalling  
warning: wget-1.21.3-1 is up to date -- reinstalling  
resolving dependencies...  
looking for conflicting packages...  
  
Packages (8) cmake-3.22.1-3 gcc-11.3.0-1 git-2.36.1-1 make-4.3-3  
mingw-w64-x86_64-libwinpthread-git-10.0.0.r32.g89bacd2be-1 ruby-2.7.2-1 unzip-6.0-2  
wget-1.21.3-1  
  
Total Installed Size: 241.71 MiB  
Net Upgrade Size: 0.00 MiB  
  
:: Proceed with installation? [Y/n]
```

```
pacman -S git ruby make cmake gcc mingw-w64-x86_64-libwinpthread-git unzip wget
```

After installation

```
resolving dependencies...
looking for conflicting packages...

Packages (8) cmake-3.22.1-3 gcc-11.3.0-1 git-2.36.1-1 make-4.3-3
             mingw-w64-x86_64-libwinpthread-git-10.0.0.r32.g89bacd2be-1 ruby-2.7.2-1 unzip-6.0-2
             wget-1.21.3-1

Total Installed Size: 241.71 MiB
Net Upgrade Size:      0.00 MiB

:: Proceed with installation? [Y/n]
(8/8) checking keys in keyring [#####] 100%
(8/8) checking package integrity [#####] 100%
(8/8) loading package files [#####] 100%
(8/8) checking for file conflicts [#####] 100%
(8/8) checking available disk space [#####] 100%
:: Processing package changes...
(1/8) reinstalling git [#####] 100%
(2/8) reinstalling ruby [#####] 100%
(3/8) reinstalling make [#####] 100%
(4/8) reinstalling cmake [#####] 100%
(5/8) reinstalling gcc [#####] 100%
(6/8) reinstalling mingw-w64-x86_64-libwinpthread-git [#####] 100%
(7/8) reinstalling unzip [#####] 100%
(8/8) reinstalling wget [#####] 100%
:: Running post-transaction hooks...
(1/1) Updating the info directory file...

CBKDELL@DESKTOP-FP78HAF MSYS ~
```

```
pacman -S git ruby make cmake gcc mingw-w64-x86_64-libwinpthread-git unzip wget
```

Set a Path for the Exported GCC to the Environment Pls make sure the directory is same name as the folder

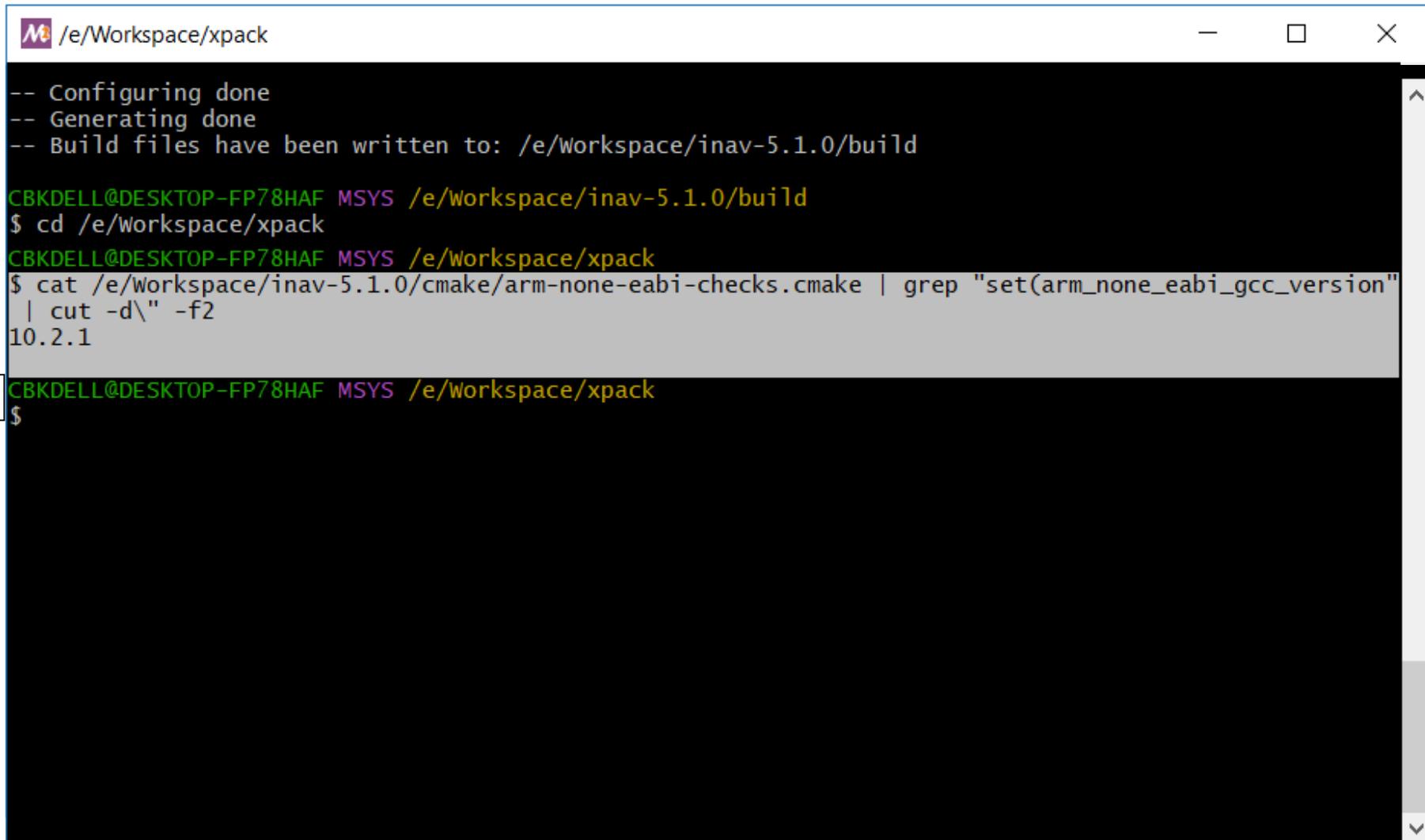
```
Me ~
:: Proceed with installation? [Y/n]
(8/8) checking keys in keyring [#####] 100%
(8/8) checking package integrity [#####] 100%
(8/8) loading package files [#####] 100%
(8/8) checking for file conflicts [#####] 100%
(8/8) checking available disk space [#####] 100%
:: Processing package changes...
(1/8) reinstalling git [#####] 100%
(2/8) reinstalling ruby [#####] 100%
(3/8) reinstalling make [#####] 100%
(4/8) reinstalling cmake [#####] 100%
(5/8) reinstalling gcc [#####] 100%
(6/8) reinstalling mingw-w64-x86_64-libwinpthread-git [#####] 100%
(7/8) reinstalling unzip [#####] 100%
(8/8) reinstalling wget [#####] 100%
:: Running post-transaction hooks...
(1/1) Updating the info directory file...

CBKDELL@DESKTOP-FP78HAF MSYS ~
$ export PATH=/e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin:$PATH
```

export PATH=/e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin:\$PATH

get the toolkit version you need for your INAV version

for INAV version 5.0.0, 5.1.0 , 6.0.0 toolchain version needed is 10.2.1



```
Me /e/Workspace/xpack
-- Configuring done
-- Generating done
-- Build files have been written to: /e/Workspace/inav-5.1.0/build

CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/inav-5.1.0/build
$ cd /e/Workspace/xpack
CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/xpack
$ cat /e/Workspace/inav-5.1.0/cmake/arm-none-eabi-checks.cmake | grep "set(arm_none_eabi_gcc_version"
| cut -d\" -f2
10.2.1
CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/xpack
$
```

Go to Xpack directory
by typing

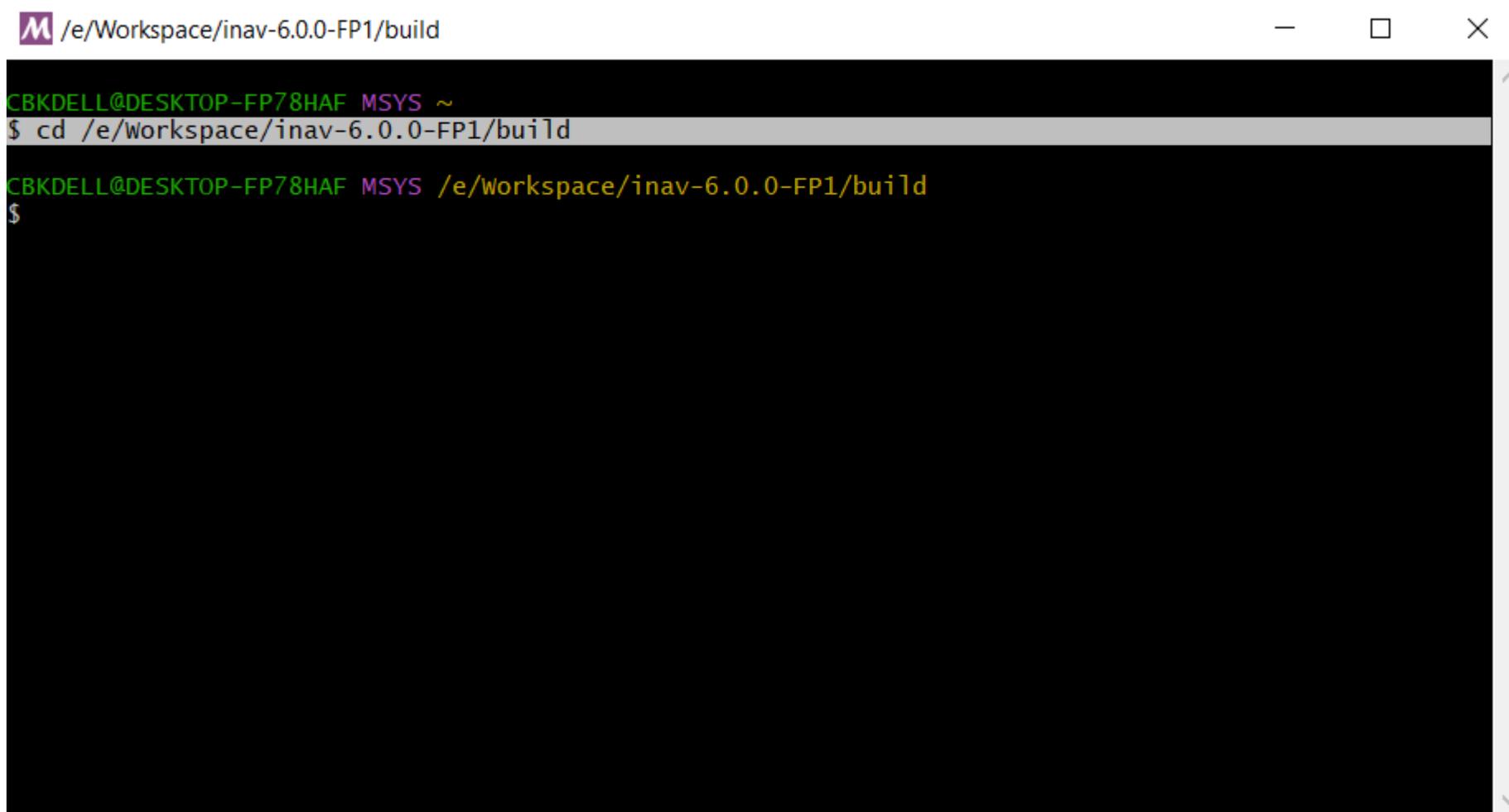
```
cd /e/Workspace/xpack
```

Source the version number that needs to be downloaded

```
cat /e/Workspace/inav-5.1.0/cmake/arm-none-eabi-checks.cmake | grep "set(arm_none_eabi_gcc_version" | cut -d\" -f2
```

```
cat /e/Workspace/inav-6.0.0-FP1/cmake/arm-none-eabi-checks.cmake | grep "set(arm_none_eabi_gcc_version" | cut -d\" -f2
```

Go to the build directory by entering this command



A terminal window with a title bar showing the path `/e/Workspace/inav-6.0.0-FP1/build`. The terminal content shows the user `CBKDELL@DESKTOP-FP78HAF` in the `MSYS` environment. The command `cd /e/Workspace/inav-6.0.0-FP1/build` is entered and executed, resulting in the prompt changing to `/e/Workspace/inav-6.0.0-FP1/build`.

`cd /e/Workspace/inav-5.1.0/build`

`cd /e/Workspace/inav-6.0.0-FP1/build`

This will extract the Cmake into the build folder creating the Environment

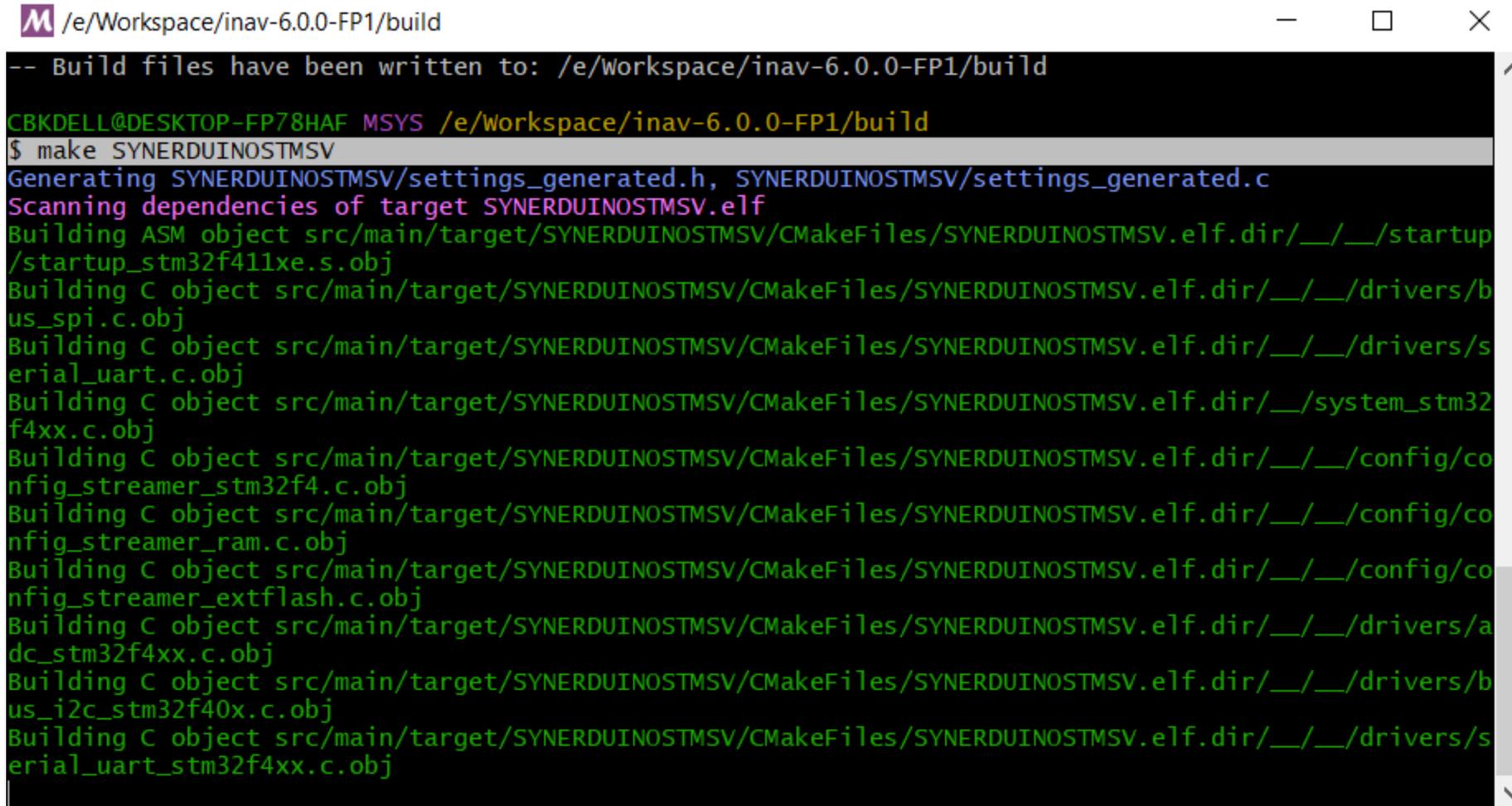
```
M /e/Workspace/inav-6.0.0-FP1/build
CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/inav-6.0.0-FP1/build
$ export PATH=/e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin:$PATH

CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/inav-6.0.0-FP1/build
$ Cmake ..
-- found arm-none-eabi-gcc 10.2.1 at /e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin/arm-n
one-eabi-gcc.exe
-- The C compiler identification is GNU 10.2.1
-- The CXX compiler identification is GNU 10.2.1
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working C compiler: /e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin/arm-none-
eabi-gcc.exe - skipped
-- Detecting C compile features
-- Detecting C compile features - done
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin/arm-non
e-eabi-g++.exe - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- The ASM compiler identification is GNU
-- Found assembler: /e/Workspace/xpack/xpack-arm-none-eabi-gcc-10.2.1-1.1/bin/arm-none-eabi-gcc.exe
-- toolchain: arm-none-eabi, WARNINGS_AS_ERRORS: OFF
-- Could not find openocd, debugging won't be available
-- DEBUG_HARDFULTS: OFF, SEMIHOSTING: OFF
```

Cmake ..

You may need to run `rm -rf *` in build directory if you had any failed previous runs now run cmake

This will build the Hexfile Targeted the Synerduino board and its settings to the Build Folder the Hexfile is use as Firmware when loading the configurator



```
M /e/Workspace/inav-6.0.0-FP1/build
-- Build files have been written to: /e/Workspace/inav-6.0.0-FP1/build
CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/inav-6.0.0-FP1/build
$ make SYNERDUINOSTMSV
Generating SYNERDUINOSTMSV/settings_generated.h, SYNERDUINOSTMSV/settings_generated.c
Scanning dependencies of target SYNERDUINOSTMSV.elf
Building ASM object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/startup/startup_stm32f411xe.s.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/drivers/bus_spi.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/drivers/serial_uart.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/system_stm32f4xx.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/config/config_streamer_stm32f4.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/config/config_streamer_ram.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/config/config_streamer_extflash.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/drivers/adc_stm32f4xx.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/drivers/bus_i2c_stm32f40x.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/drivers/serial_uart_stm32f4xx.c.obj
```

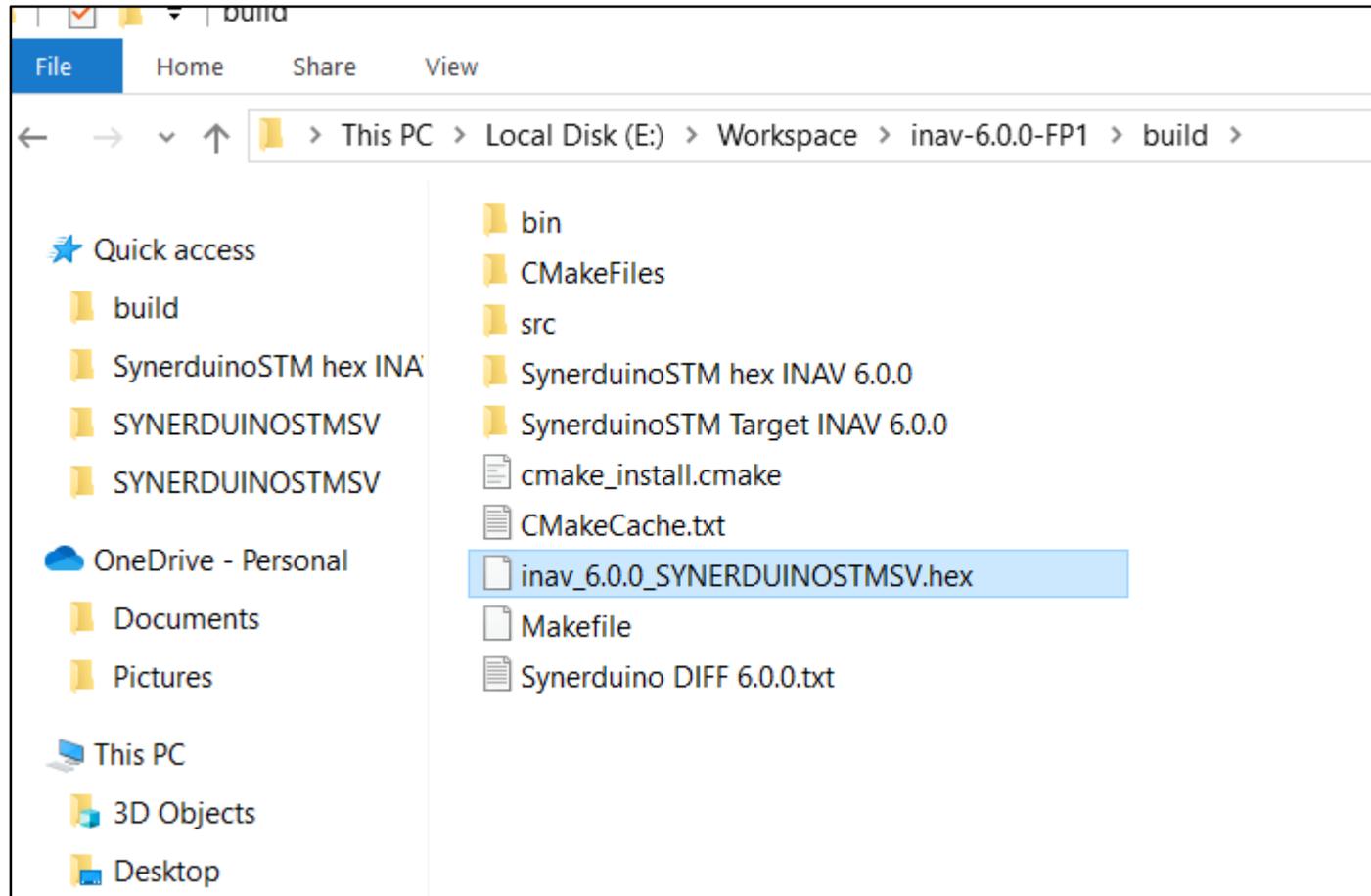
make SYNERDUINOSTM

make SYNERDUINOSTMSV

Once completed the Hexfile can be found on the Build folder

```
M /e/Workspace/inav-6.0.0-FP1/build
/ibus_shared.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/ibus.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/ltm.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/mavlink.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/msp_shared.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/smartport.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/sim.c.obj
Building C object src/main/target/SYNERDUINOSTMSV/CMakeFiles/SYNERDUINOSTMSV.elf.dir/__/__/telemetry
/telemetry.c.obj
Linking C executable ../../../../bin/SYNERDUINOSTMSV.elf
Memory region      Used Size  Region Size  %age Used
      FLASH:         1108 B      16 KB      6.76%
    FLASH_CONFIG:         0 GB      16 KB      0.00%
      FLASH1:    458570 B      480 KB     93.30%
      RAM:         92056 B      128 KB     70.23%
    MEMORY_B1:         0 GB      0 GB
Built target SYNERDUINOSTMSV.elf
Built target SYNERDUINOSTMSV
CBKDELL@DESKTOP-FP78HAF MSYS /e/Workspace/inav-6.0.0-FP1/build
$
```

And... its here . The INAV_5.1.0 SYNERDUINOSTM.hex or INAV_6.0.0_SYNERDUINOSTM.hex
INAV_5.1.0 SYNERDUINOSTMSV.hex INAV_6.0.0_SYNERDUINOSTMSV.hex



Open this file in the Configurator as
this is the firmware
Find it in the Build folder of the INAV
directory

```
cd /e/Workspace/inav-5.1.0/build
```

```
cd /e/Workspace/inav-6.0.0-FP1/build
```