Hawkeye2 - Homemade Telecine Project

Intro

The purpose of this doc is to provide the instruction on how to setup and use the Hawkeye2 homemade telecine. The appendix covers the actual construction details that could be useful for people interested in building their own unit.

Background Info

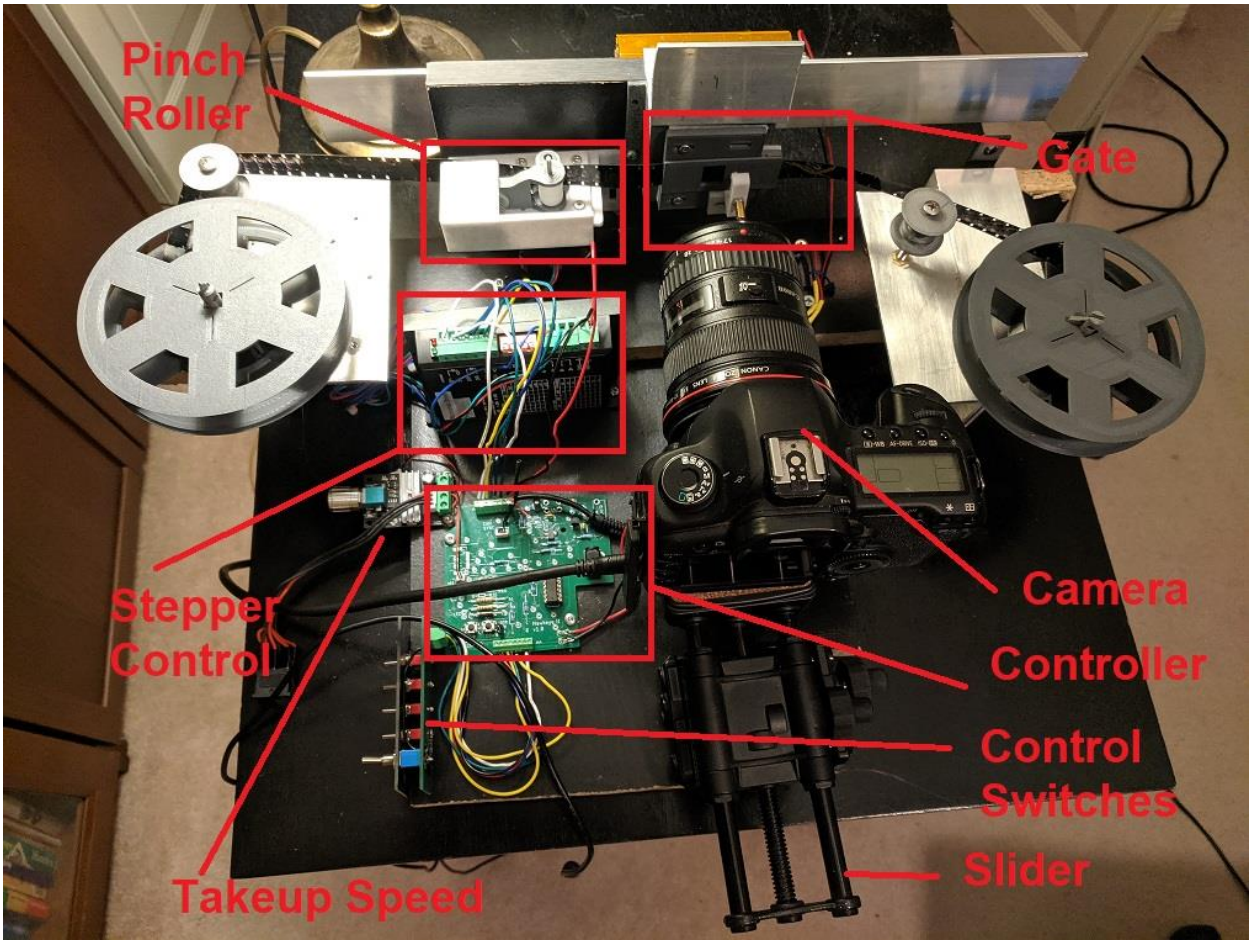
Here is a bit on the background of this project. A while back I developed a custom controller for the Wolverine scanner and called it the Hawkeye. The idea was to be able to use a better camera but still reusing the Wolverine mechanics and the housing. That worked out pretty nicely and if interested check the following link for more info on the project.

https://8mmforum.film-tech.com/vbb/forum/film-to-digital-conversion/27570-wolverine-hawkeye-for-newbies

But one of the issues with the Hawkeye is that it can do 8mm and super 8 film only. I had requests 16mm and even 35mm negatives. So, I decided to do the homemade machine that is capable of different formats. One of the 8mm forum users (David Brown) helped with the 3D prints and also some of the logistics on how to make the swappable components for different film formats.

It is a very simple machine the can accommodate different types of cameras and different lenses providing a user with very high flexibility. One of the key components of the machine is the pinch roller driven by the stepper motor. A tuning system was implemented where the step size can be stored in the memory for different film formats. Unlike some other similar machines, Hawkeye2 does not register the film perforation holes. The tuned step size is used instead. This results in a small drift during the scan caused by film shrinkage, but the drift can be easily removed by using a special deshaker software. This allows for the system to be very simple and relatively low cost.

Machine Components



Pinch Roller:

The pinch rollers consist of the active roller driven by the stepper motor an d an idler roller that is spring loaded and pushes against the active roller. The film gets advanced by being squeezed between the two rollers. The frame sync is maintained by programming the stepper during the tuning procedure.

Controller:

The controller is based on the Hawkeye1 controller. It consists of the MSP430 microprocessor and some support circuitry. The main purpose of the controller is to drive the stepper by sending pulses to the stepper controller. Additionally the controller contains the firmware that provides the camera trigger signals, the takeup reel motor control, and the polling of the switches. The tuning procedure is also implemented in the MSP430 firmware. Currently the firmware supports up to 4 different film formats.

Stepper Controller:

The stepper controller is an off the shelf module that translates the pulses from the main controller into the stepper control voltages. Check the list of materials at the end of the document.

Control Switches:

The control switches control the operation of the unit.

RUN Switch – Pauses the operation

ALIGN – Slowly advances the film until the frame is centered in the camera. Also turns tuning on if activated in run mode.

REW – Turns takeup continuously on if run switch is off. Selects the film format if run is off.

F1/F2 – In conjunction with the REW switch selects one of the four film formats. The format selection is read by the firmware upon board power up and cannot be changed after that during normal operation.

UP and DOWN switches– Used during tuning mode to run change the length of stepper advance.

Up and DOWN buttons – Located on the controller board, used for fine tuning of the stepper

REV jumper – Located on the board, reverses the motor rotation. It can also be done by the takeup motor reverse control switch.

Stepper Control:

It acts as an interface between the controller and the stepper motor.

Takeup Speed:

It is an off the shelf board. It controls the takeup speed and also reverses the motor rotation for rewind.

Camera:

Preferably a DSLR camera with a macro lens or reverse mount good quality lens.

Slider:

The slider provides for camera mounting and is used to adjust the camera focus.

Gate:

Custom designed film gate with the pressure adjusting screw. The gate and the gate mount are designed such that different film formats can be accommodated just by sliding in a different size gate.

LED Light:

The LED light uses a 12V LED chip mounted behind the gate. The gate also has a diffuser incorporated as a part of the gate mounting bracket which ensures even light distribution over the entire frame.

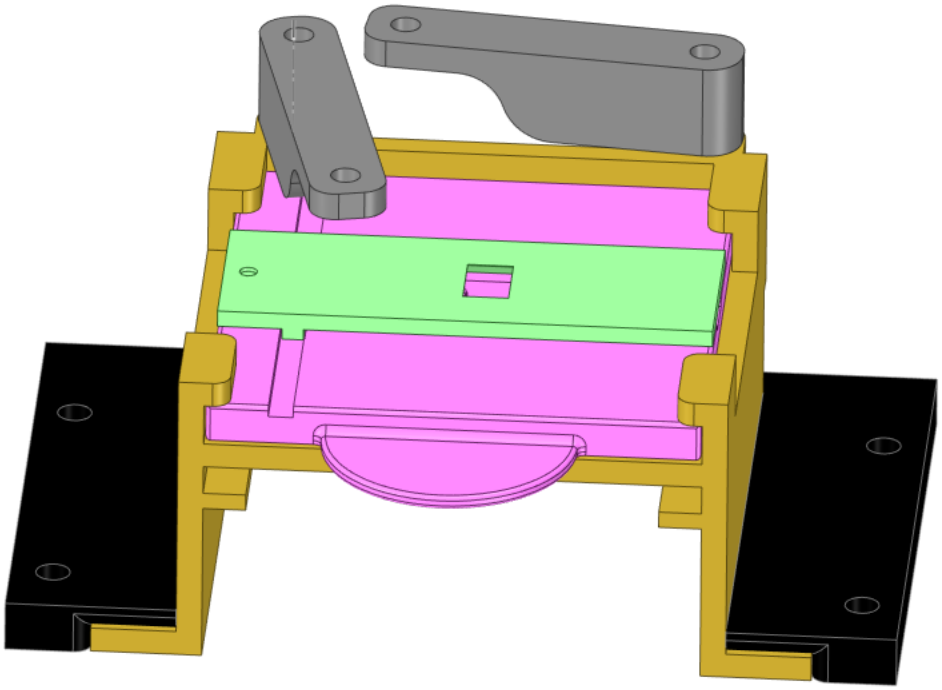
Power Supply:

It is a 12 VDC adapter that is capable of providing at least 2 Amps of current.

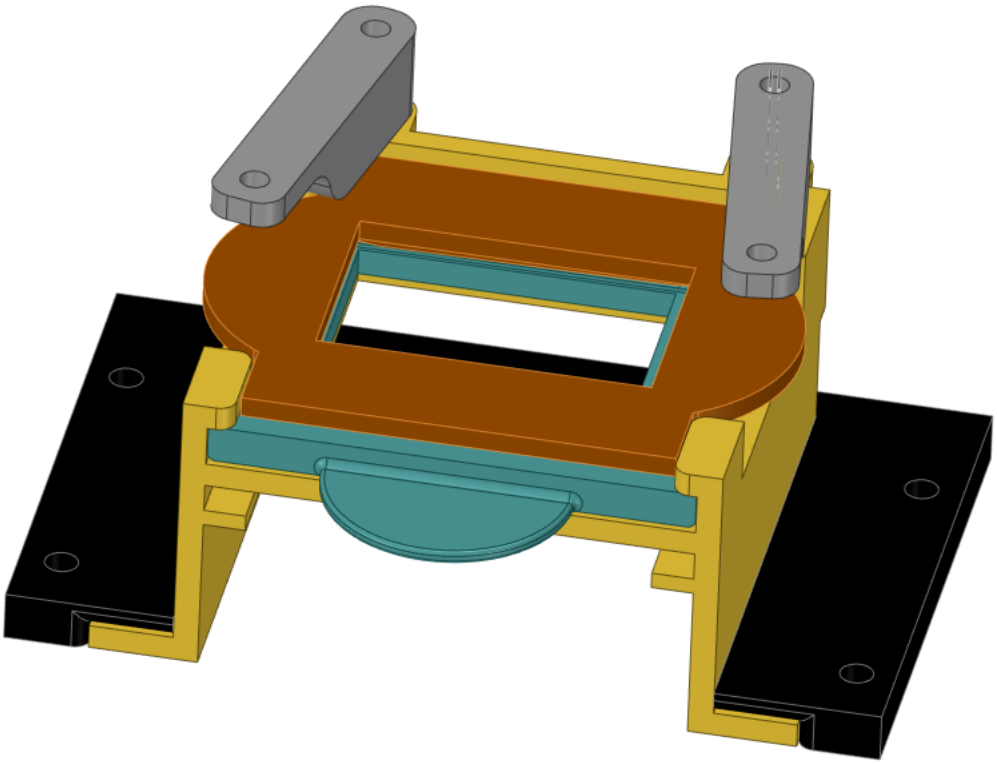
Gate Design

The gate design is made flexible to accommodate different film formats. The whole gate assembly can essentially be slid out from the mounting bracket and the new film format assembly can be inserted in place of the old one.

The picture below shows the assembly for the 8mm film.



And shown below is the assembly for the 16mm film.



Please note that the pictures above are concept pictures only and the final design is still in progress. But the pictures are there to give you a general idea on how this should work.

Pinch Roller Design

It is a pretty simple design that utilizes the stepper and idle rollers that pinch the film. As the stepper roller rotates it drives the idle roller and the film with it. The advantage of this design is in its simplicity. Additionally, unlike the claw type design, this design will have no issues with the damaged film perforations.

The disadvantage of this type of a design is that there will be some drift present in the output video caused by film shrinkage. But the drift can easily fixed by the custom deshaker software described later in this document.

Setting up the Machine

Gate Installation

Install the film gate by sliding it into the slots on the mounting bracket.

Add Image here…

Mount the camera with the macro lens on the camera slider.

Insert a test film clip into the gate.

Adjust the camera focus by using the forward/backward knob on the sider.

If the macro lens has a zoom feature, then the zoom may have to be adjusted as well. It may be necessary to go back and forth between the slider knob and the zoom to get the image in sharp focus and for it to be framed properly.

Slide the gate up and down if necessary so that the image is centered vertically. Note that it may be necessary to loosen the gate mounting screws slightly to allow the gate to move. Make sure to retighten the screws back once done.

Adjust the slider left/right knob to get the image centered.

That completes the gate installation.

Pinch Rollers

The pinch roller bracket may need to be adjusted so that the film going through the rollers is aligned with the gate.

Important Note: Make sure that the pinch rollers are properly aligned with the gate. Bad alignment will result in inconsistent film feed and large drifts from frame to frame.

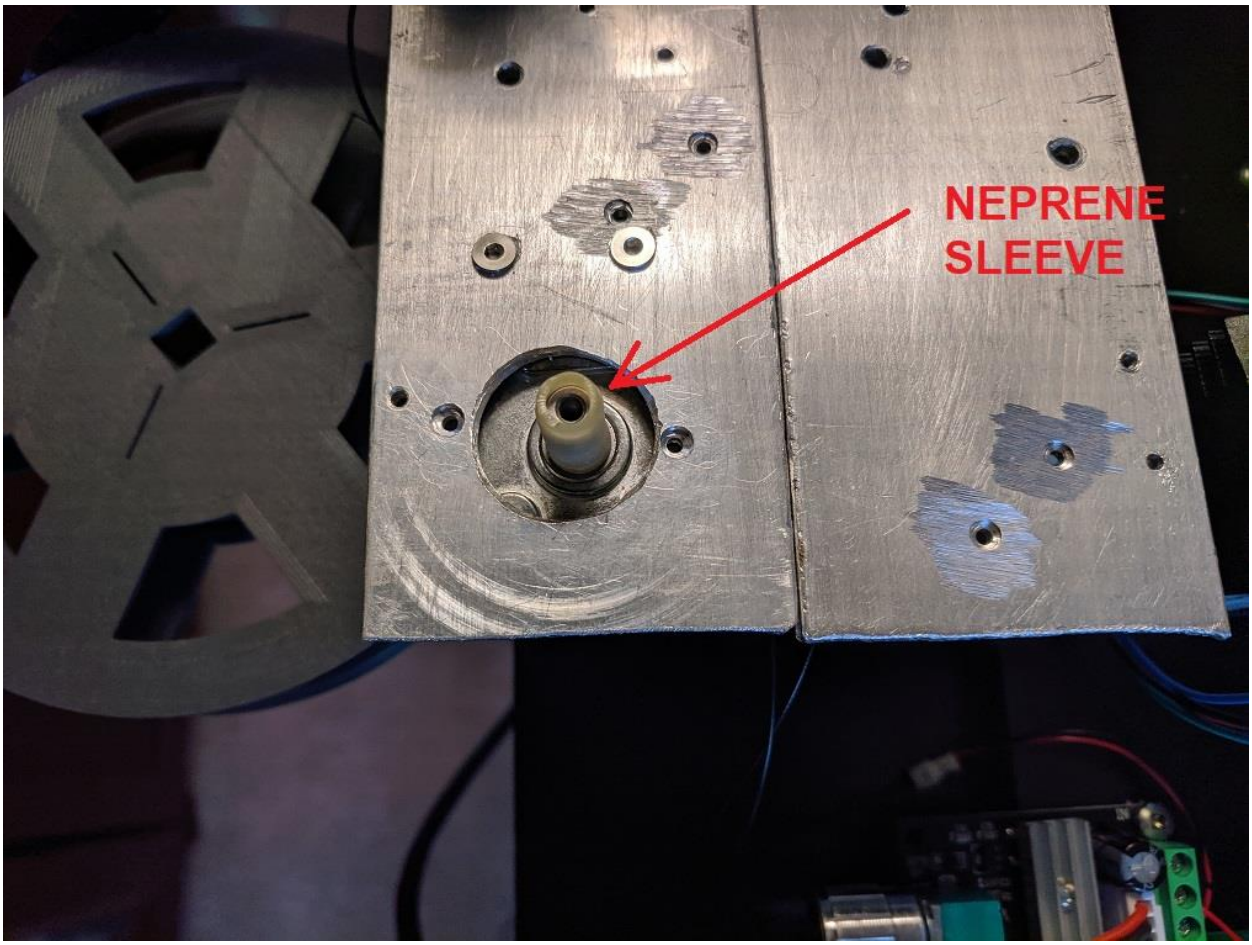
Reel Adapters

Each film format has its set of reel adapters.

The set consists of two adapters, one for the supply reel and the other one for the takeup reel.

The picture above shows the takeup adapters that get supplied with the kit (once the kit becomes available).

Note that the supply and takeup reel adapter are the same shape. The takeup stepper shaft has a neoprene friction sleeve.



Make sure that the neoprene sleeve is on the stepper shaft. Sometimes it could come off and stay within the adapter during the adapter exchange. The sleeve provides enough friction for the takeup but not too excessive in order not to tear the film.

Film Guides

Two types of Film Guides are used.

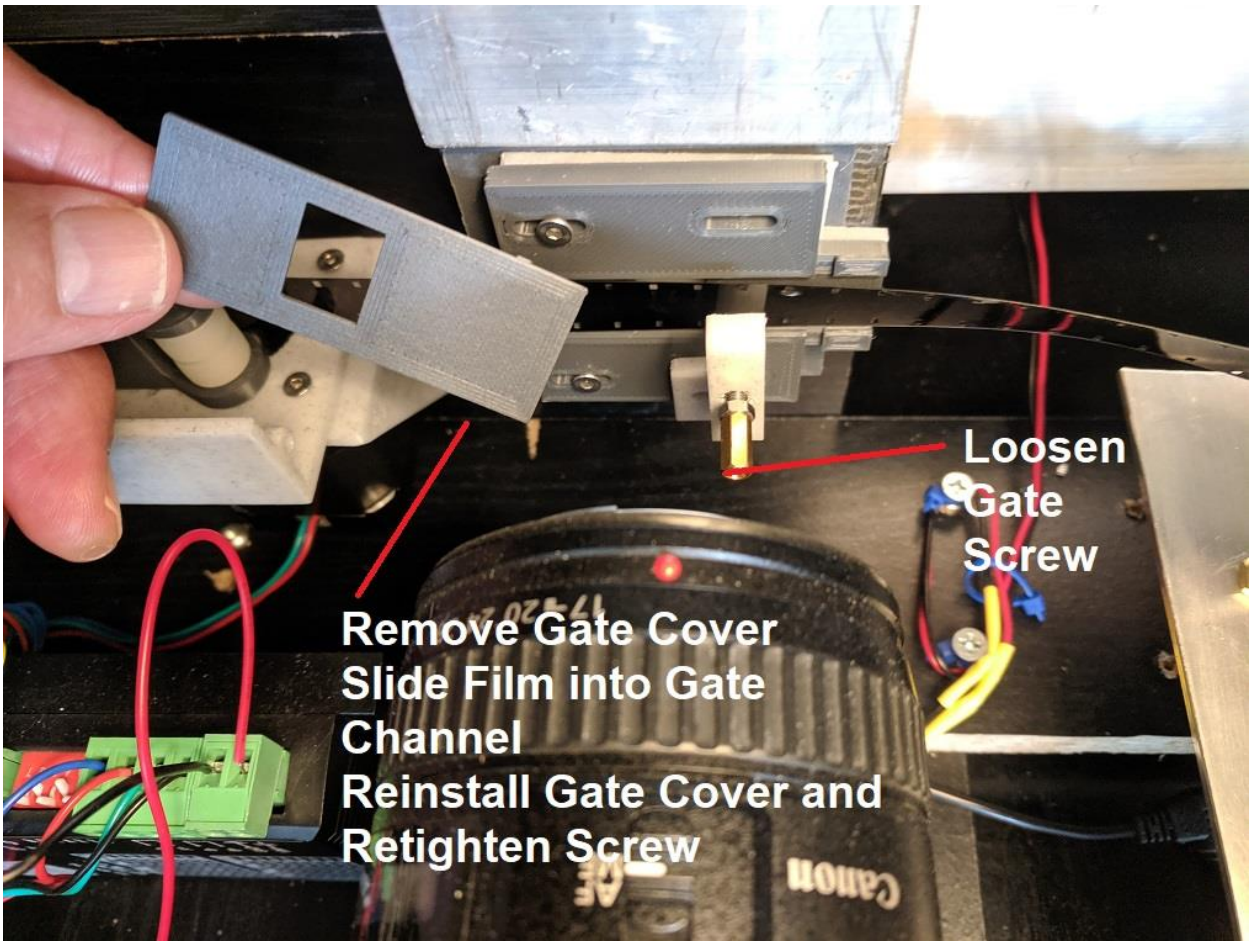
One style of the guides can be used for both 8mm and 16mm film formats and the other style for the 35mm only.

The guides are attached to the mounting panel by the mounting nuts and can be easily removed or installed by removing or installing the mounting nuts.

Threading the Film

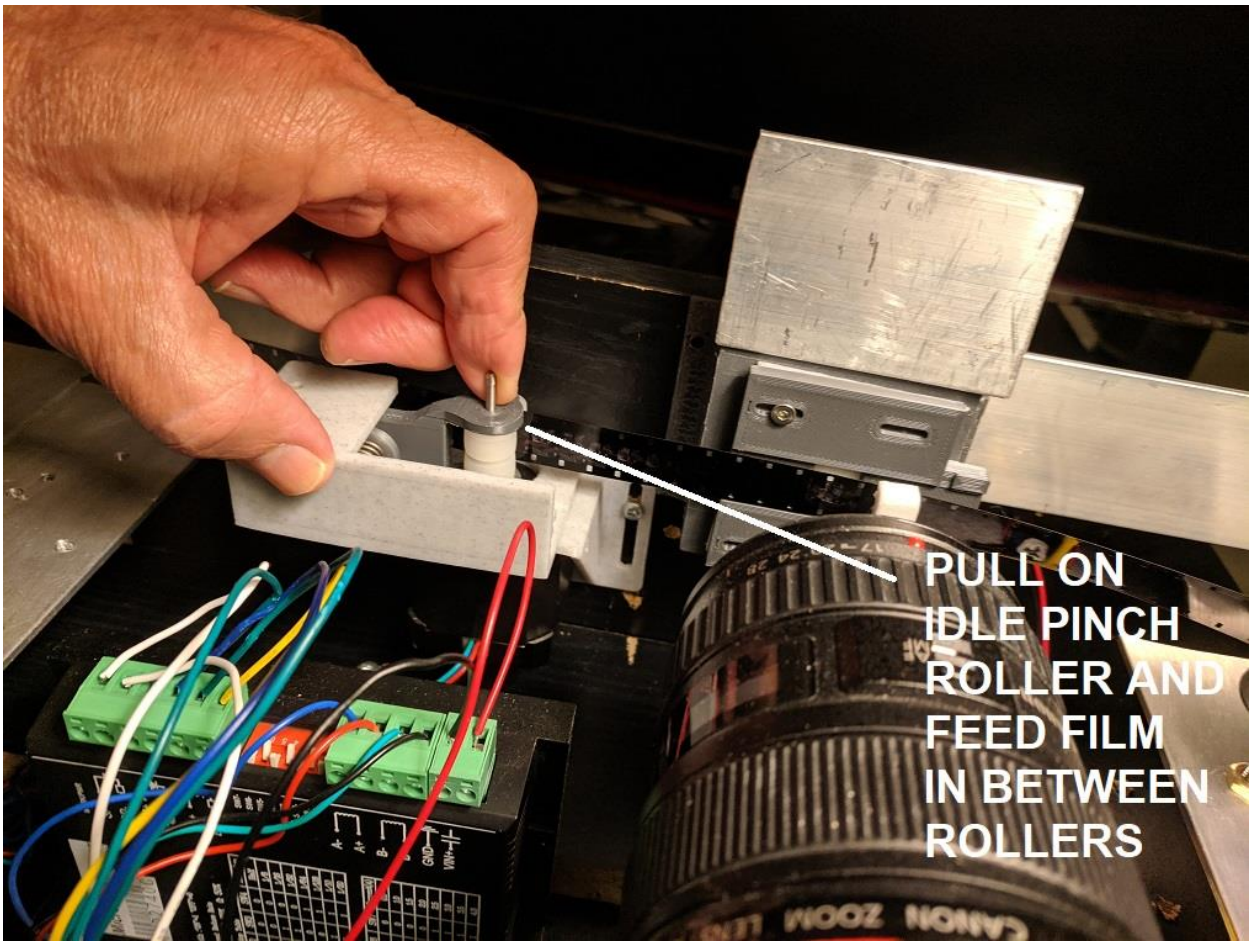
Mount the Supply reel onto the supply adapter.

Loosen the gate friction screw and remove the gate cover.



Place the film into the gate channel.

Pull on the idler roller shaft and feed the film between the idler roller and the stepper roller. Also make sure that the film fits into the gate channel loosely.



Let go of the roller shaft. The film will get squeezed between the rollers. Make sure that is is nice and square and that it aligns with the film gate.

Put the gate cover back on and tighten the screw.

Make sure that the film is threaded properly and not binding.

Loosen the screw if the film is in too tight. The screw should not be too tight, just enough to hold the film in place but low friction.

Selecting Film Format

The controller has four non-volatile memory banks where it keeps the information on the film type used. The non-volatile memory retains the information even after the complete power down.

The values in the memory banks are set by using the tuning procedure. At the end of the tuning procedure the tuned value gets saved to the selected memory bank and it will be available the next time the board gets powered up in that mode. It is up to the user to decide what film format belongs to which memory bank but it is good idea to mark that somewhere.

Here is how the memory banks are selected.

Before powering up the board, use the F1/F1 and REW switches to select one of the memory banks.

REW DOWN, F1	Memory1
REW UP, F1	Memory2

REW DOWN, F2 Memory3

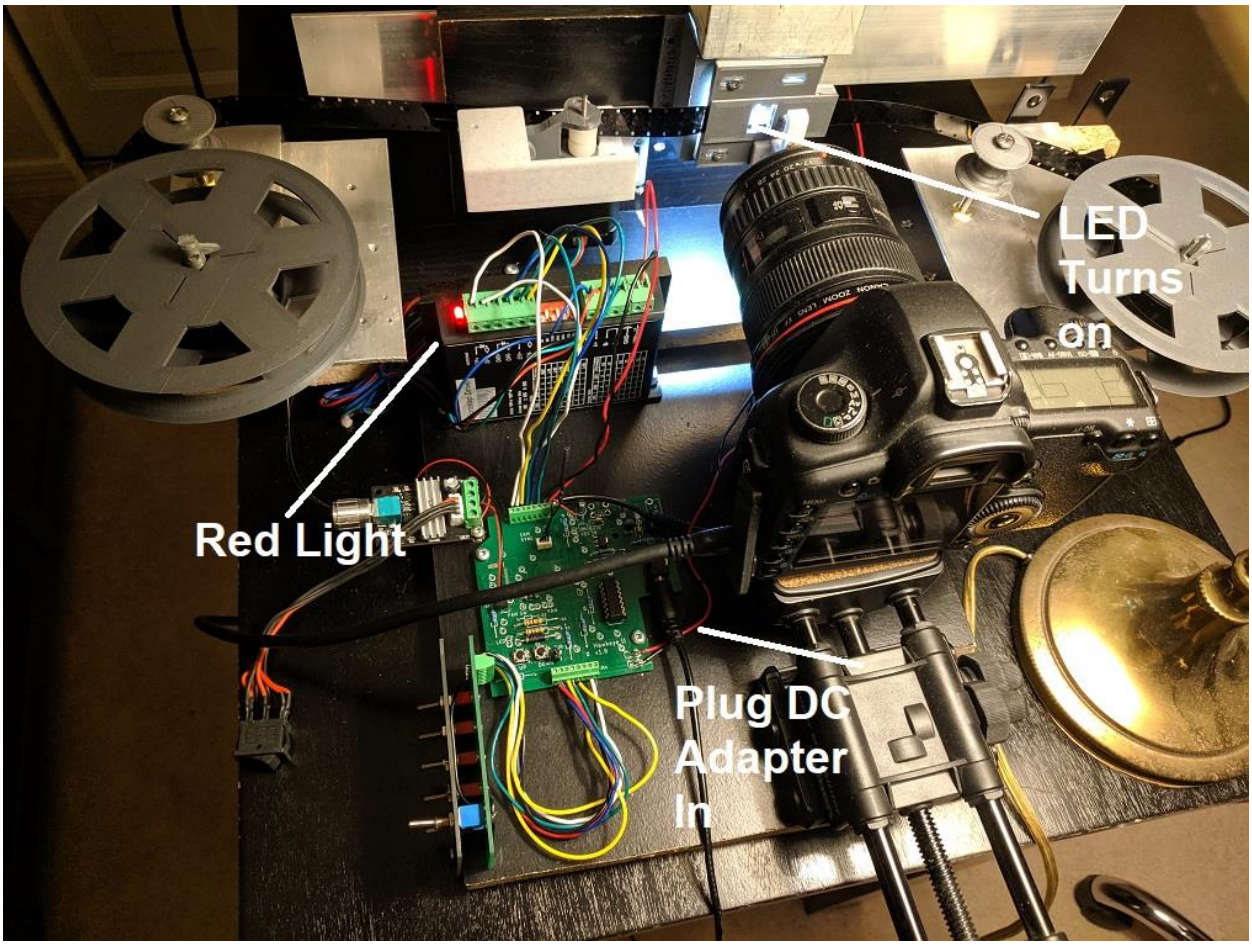
REW UP, F2 Memory4

Power the board up by plugging in the DC Adapter.

Note, the memory bank selection is done right at the power up and is not read again by the controller until next power cycle.

If the REW switch is set to UP for the purpose of memory election, make sure to turn it off right after the power up to prevent the takeup motor from running and possibly damaging the film.

After the power up, the stepper controller red light and the gate LED should turn on.



Tuning the Stepper

After the film is threaded properly turn the camera on. It is preferable to connect camera to a large monitor in order to facilitate the tuning.

Turn the RUN switch on. Make sure all other switches are off.

The film will start advancing. It may advance continuously or in smaller steps or not advance at all depending on the memory initial content.

Turn the ALIGN switch on.

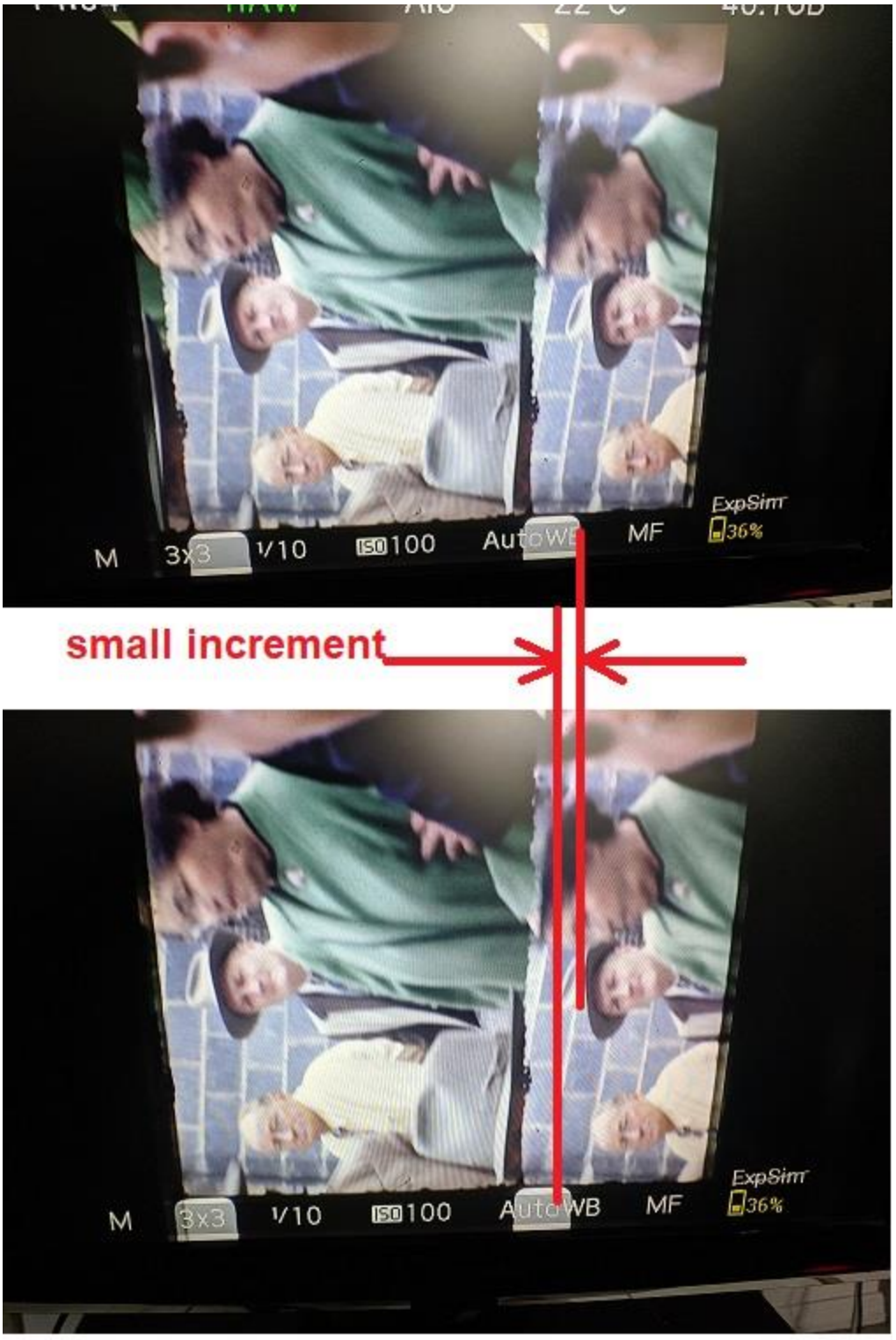
This puts the controller into tuning mode.

Turn on the DOWN switch. This makes the stepper step one count longer.

While the DOWN switch is still on, turn on the UP switch.

This will cause continuous stepper count increase. You will see that the film advance becomes larger and larger.

In the beginning the steps film advances will be small, as shown.


small increment
M
3x3
1/10
ISO100
AutoWB
MF
ExpSim
36%

But after several minutes the increments will become large enough to be close to the full Iframe advance as shown.

Once the alignment is close to a full frame, turn the UP switch off and then the DOWN switch off.

For may continue with fine tuning by using the controller board mounted UP and DOWN buttons. The UP button will make the advance shorter by one count and conversely, the DOWN button will make it longer by one count.

Once the alignment is done DO NOT POWER THE BOARD DOWN.

Turn the ALIGN switch off and then the RUN switch off. Then you can power the board down.

Power the board up again with the correct memory selected.

Turn the RUN switch on and check that the alignment was restored properly from the memory.

It is to be noted that the UP and DOWN sequence can be reversed if the Film advance is too large. In that case turn the UP switch on first followed by the DOWN switch. Once done, turn the DOWN switch off first followed by the UP switch.

Film Scanning Procedure

Once the alignment is complete you are ready to start with the scan.

Tuning Memory Check

The tuning values are stored into the nonvolatile memory located at address 0x1000

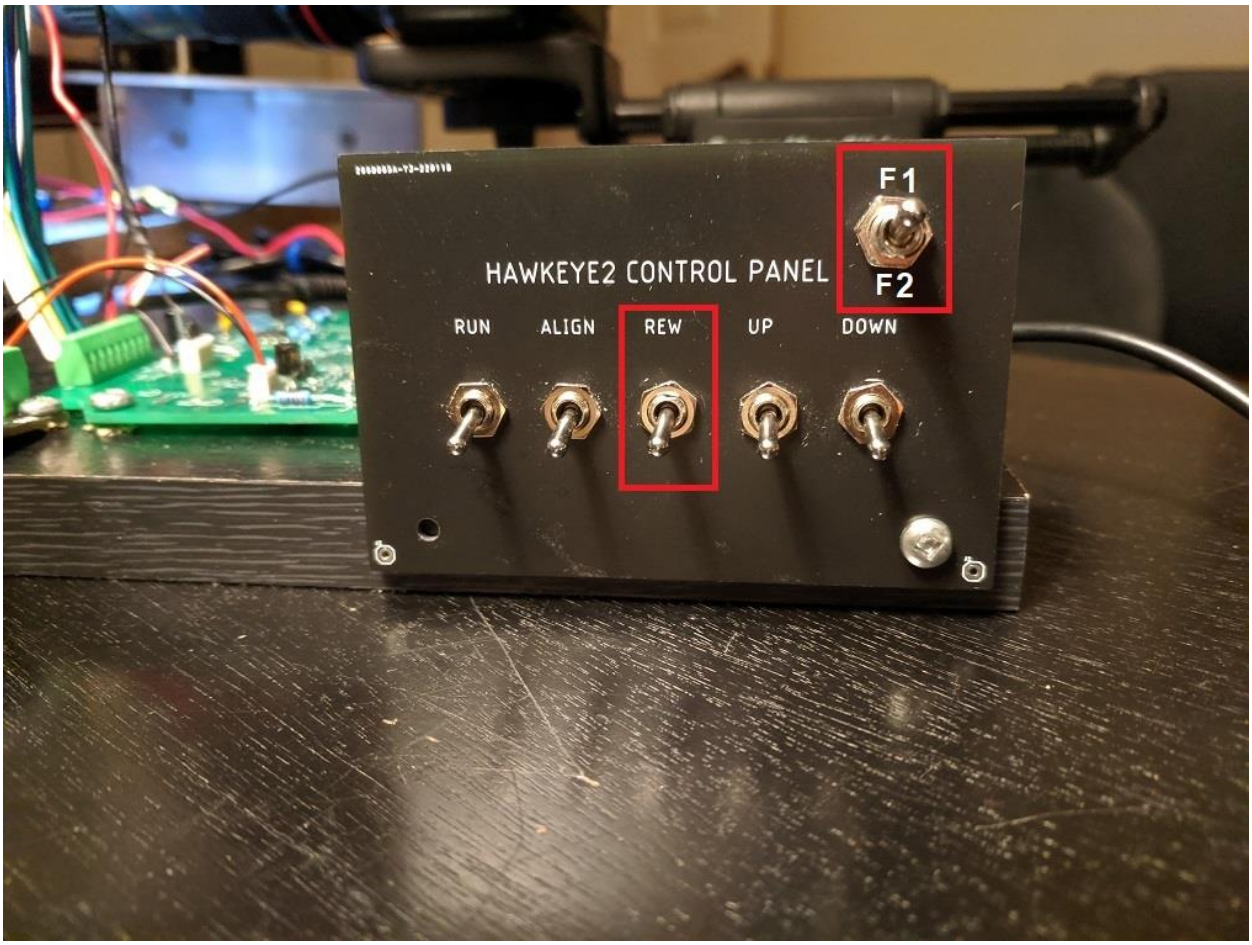
0x1001, 0x1000 MS and LS values of memory1

0x1003, 0x1002 MS and LS values of memory2

0x1005, 0x1004 MS and LS values of memory3

0x1007, 0x1006 MS and LS values of memory4

The memory is selected at power up by reading the status of the memory selection switches.



REW OFF, F1 Selected Memory1

REW OFF, F2 Selected Memory2

REW ON, F1 Selected Memory3

REW ON, F2 Selected Memory4

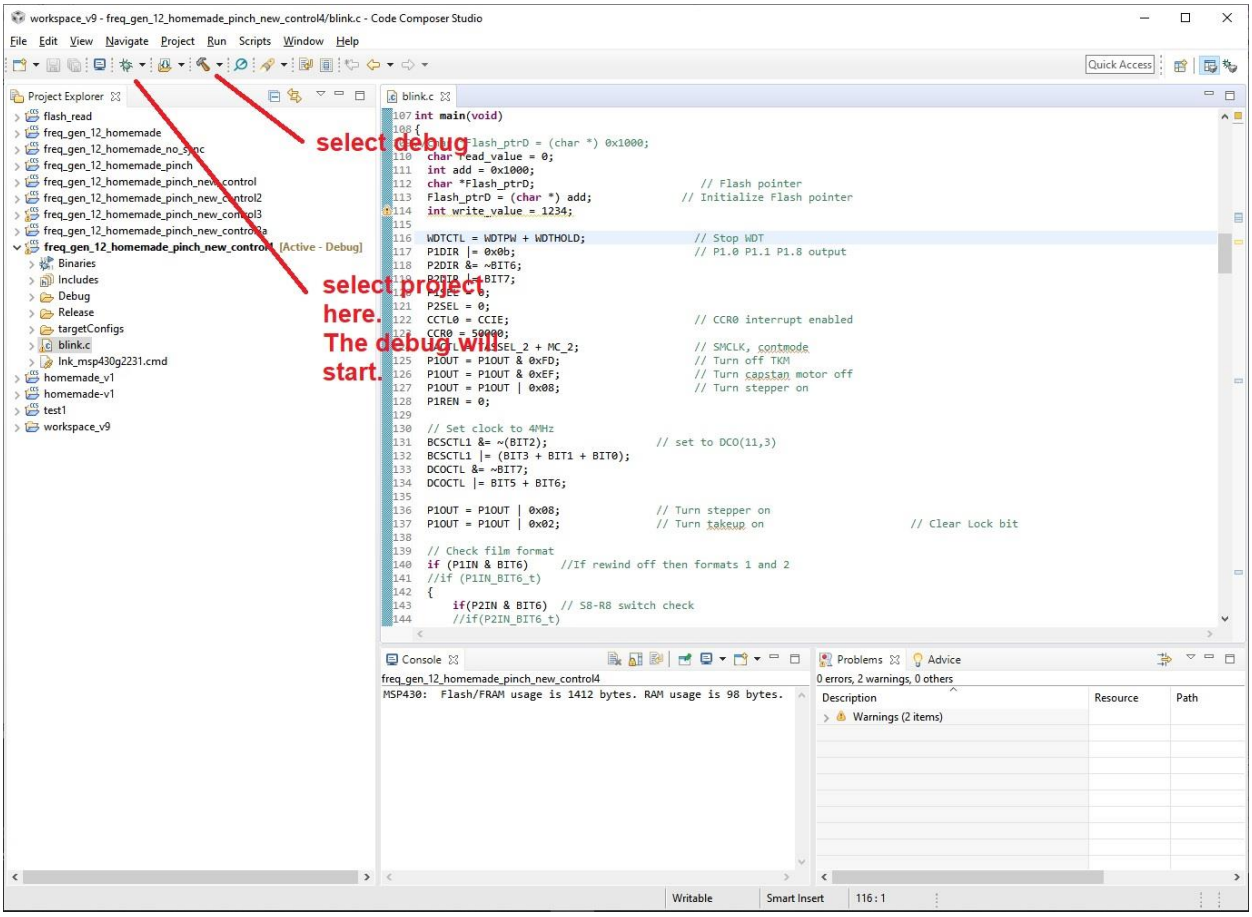
If the memory values do not seem to be right then the following debug procedure can be used to fix the issue.

Check the memory in CCS (Code Composer Studio)

Remove the MSP from the target and plug it into Launchpad.

Make sure the right project is selected and highlighted in the left pane.

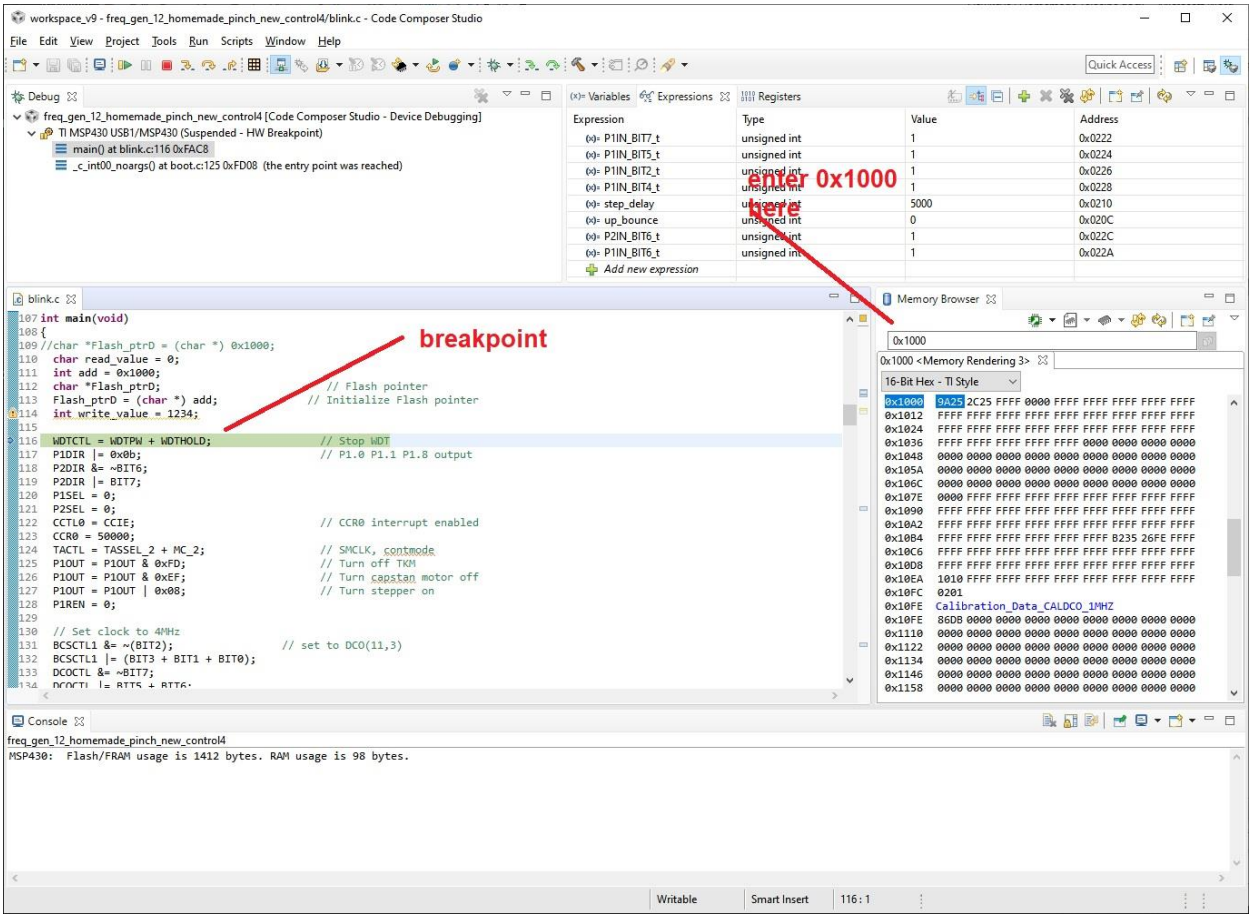
Then click on the hammer symbol and switch to debug mode and rebuild the project.



The debug window will open and display the breakpoint highlighted in green.

Click on View and select Memory Browser.

The memory will be displayed in the pane on the right.



In this example the following values can be observed.

Memory1 0x259A

Memory2 0x252C

Memory3 0xFFFF

Memory4 ox0000

If the MSP is plugged into the target the following behavior is observed.

Memory1 (mode1) The stepper makes roughly 1/5 of the full rotation every 10 seconds

Memory2 (mode2) About the same as mode1

Memory3 (mode3) The stepper never stops

Memory4 (mode4) The stepper ever moves but the takeup kicks in every 10 seconds

This all makes sense. The stepper used has 1.8 degrees per step and each step 128 microsteps.

For 360 degrees (full rotation) 360/1.8 = 200 steps or 200*128 = 25,600 microsteps.

Memory 1 value of 0x259A = 9,626

That represents the number of toggles or 9626/2 = 4813 pulses which is

4813/25600 roughly 1/5 of the full revolution and that is what the test shows.

Mode2 is about the same.

Mode3 is larger than the full rotation. The code has a bug that results in the stepper running continuously if the value is larger than 30,000. This will be fixed version 5 of the code.

Mode4 memory is 0. That does not send any pulses to the motor but will still trigger the camera and the takeup.

If the memory value is 0 then the tuning with the DOWN switch will increase the value to the desired delay. Just select the desired bank that has the zero value and then turn the RUN switch on followed by the ALIGN switch and then DOWN switch followed by the UP switch. After a while the stepper will start running and the delay will slowly start increasing. Once it is close to the desired ballpark, turn the switches off in the following sequence:

UP

DOWN

ALIGH

RUN

Recycle power and fine tune by using the UP and DOWN switches.

If the memory value is set to 0xFFFF then using the UP and DOWN switches tuning will work but will take up to an hour to get close to the desired delay.

A better way is to use the nonvolatile memory write utility.

Download the latest CCS project code:

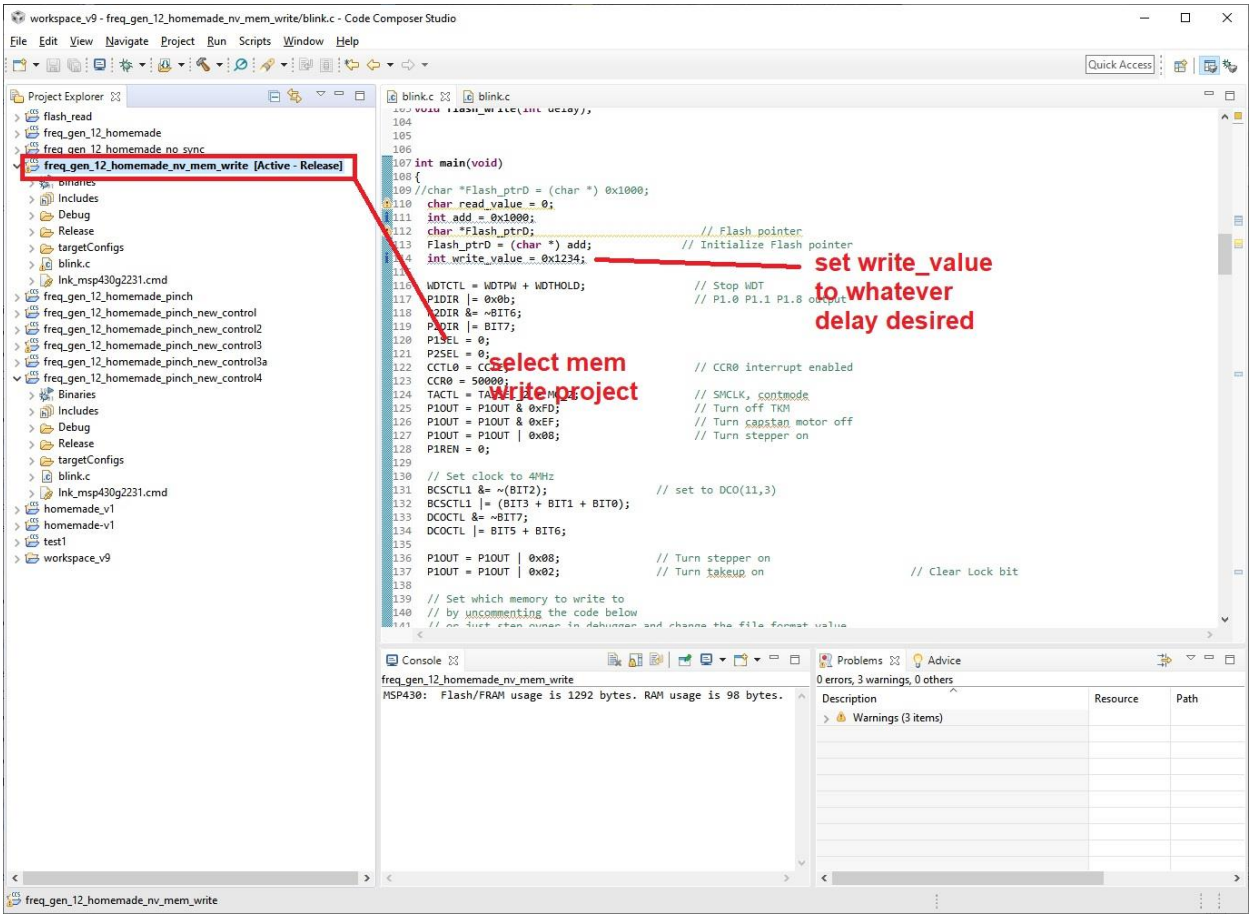
https://github.com/vintagefilmography/homemade-telecine/tree/main/MSP

Plug the MSP into Launchpad and plug Launchpad into the computer.

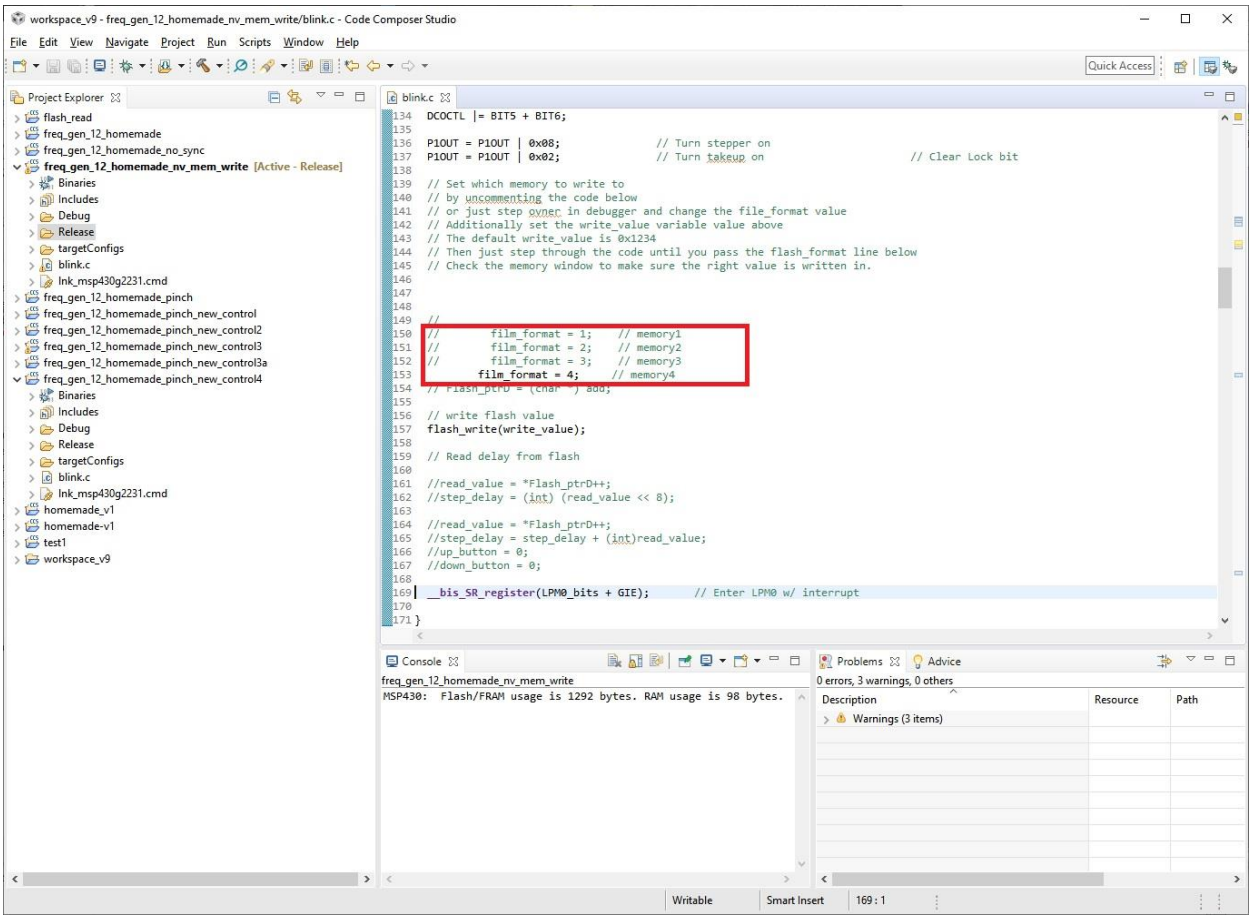
Open up workspace_v9 in CCS

Select

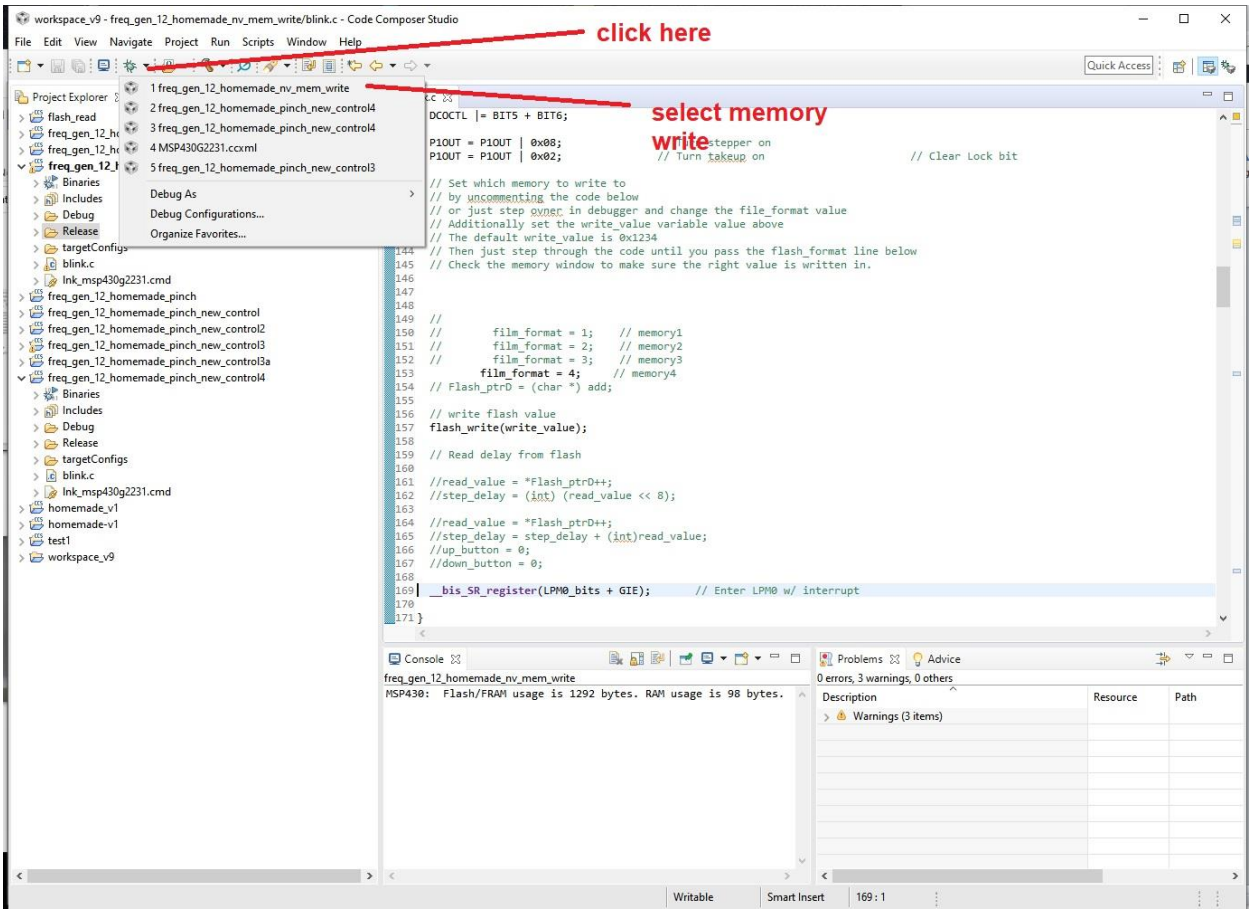
freq_gen_12_homemade_nv_mem_write



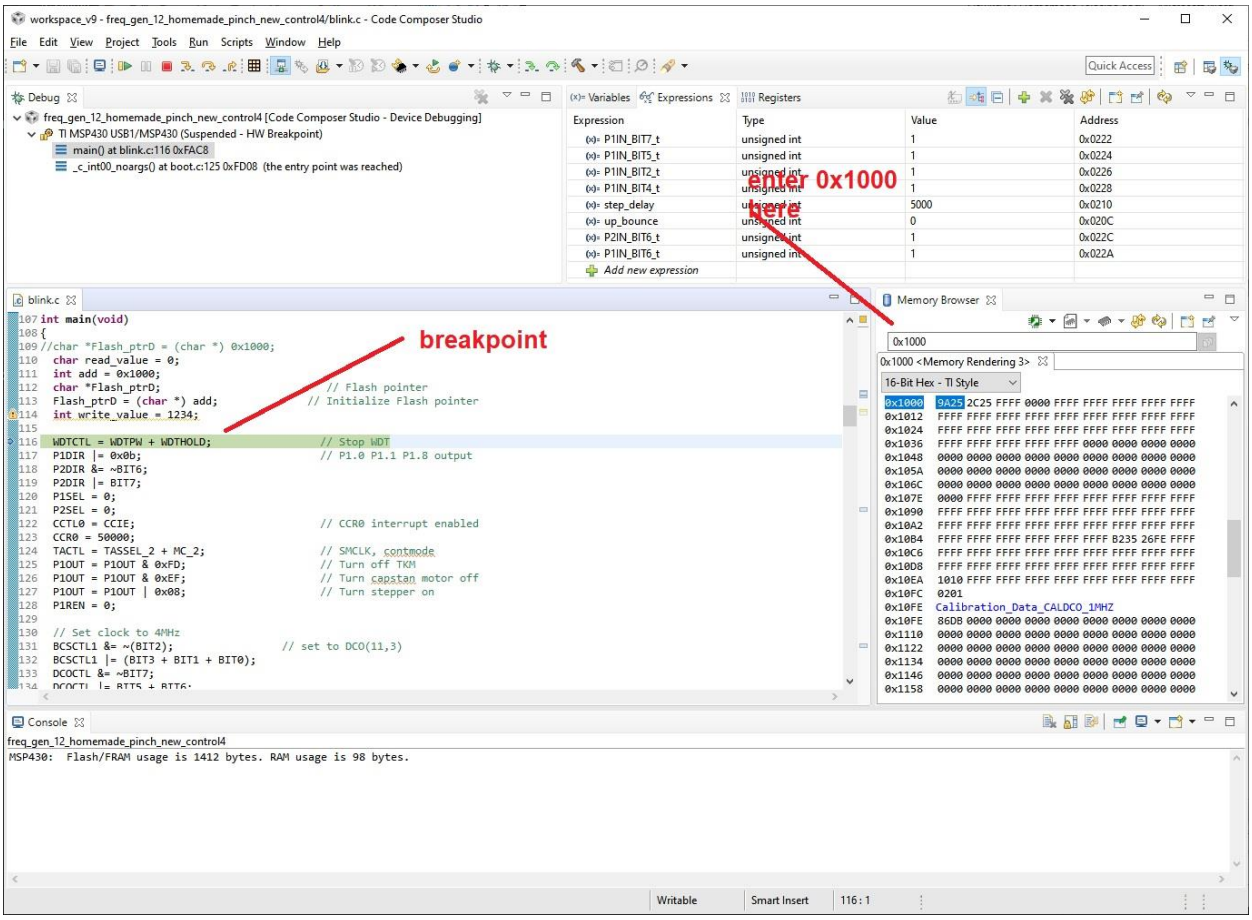
Change write_value to whatever delay required.



Select debug config



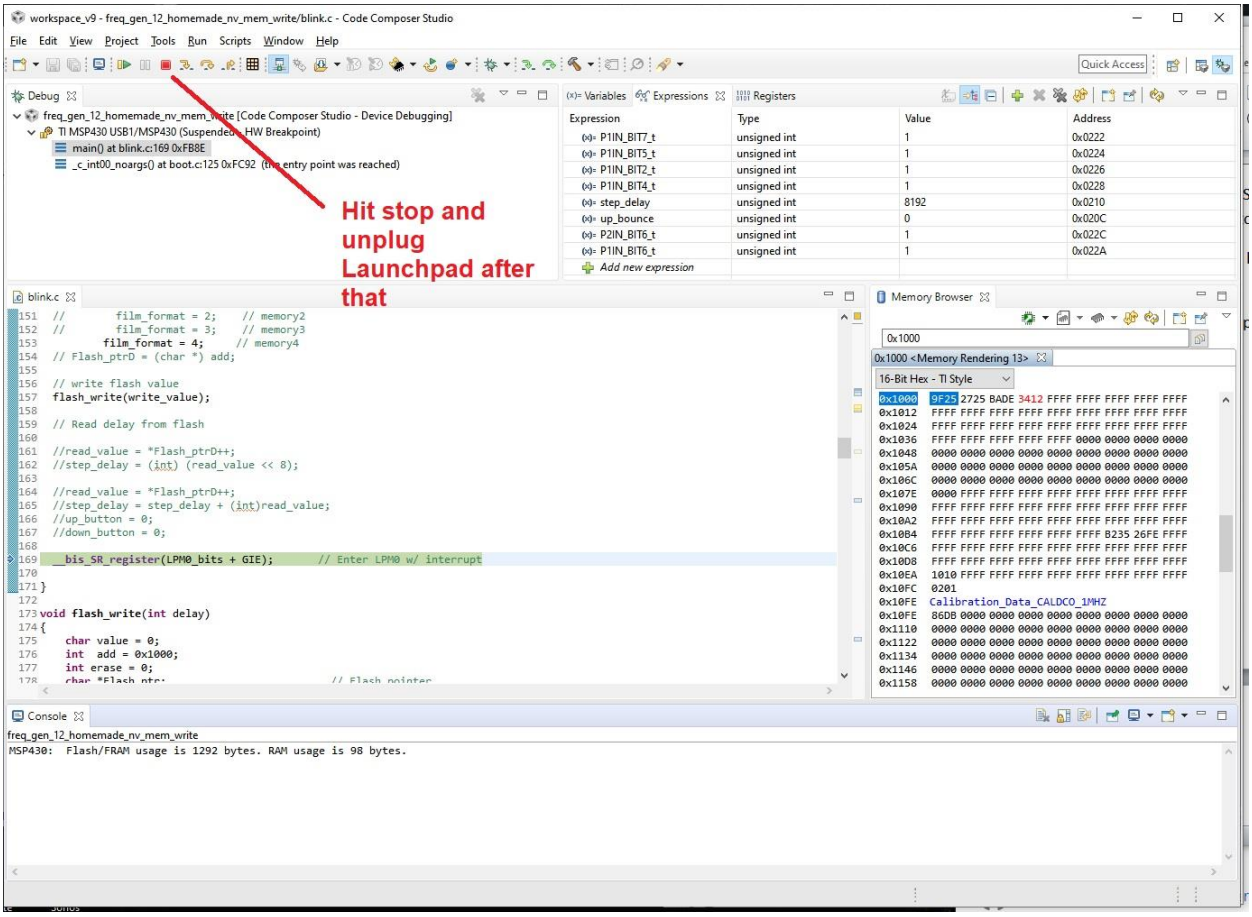
The debug session will start and the code will stop at the breakpoint, highlighted in green. Set the address in the memory window to 0x1000. If the memory window is not open, open it from the View pulldown. Select Memory Browser.

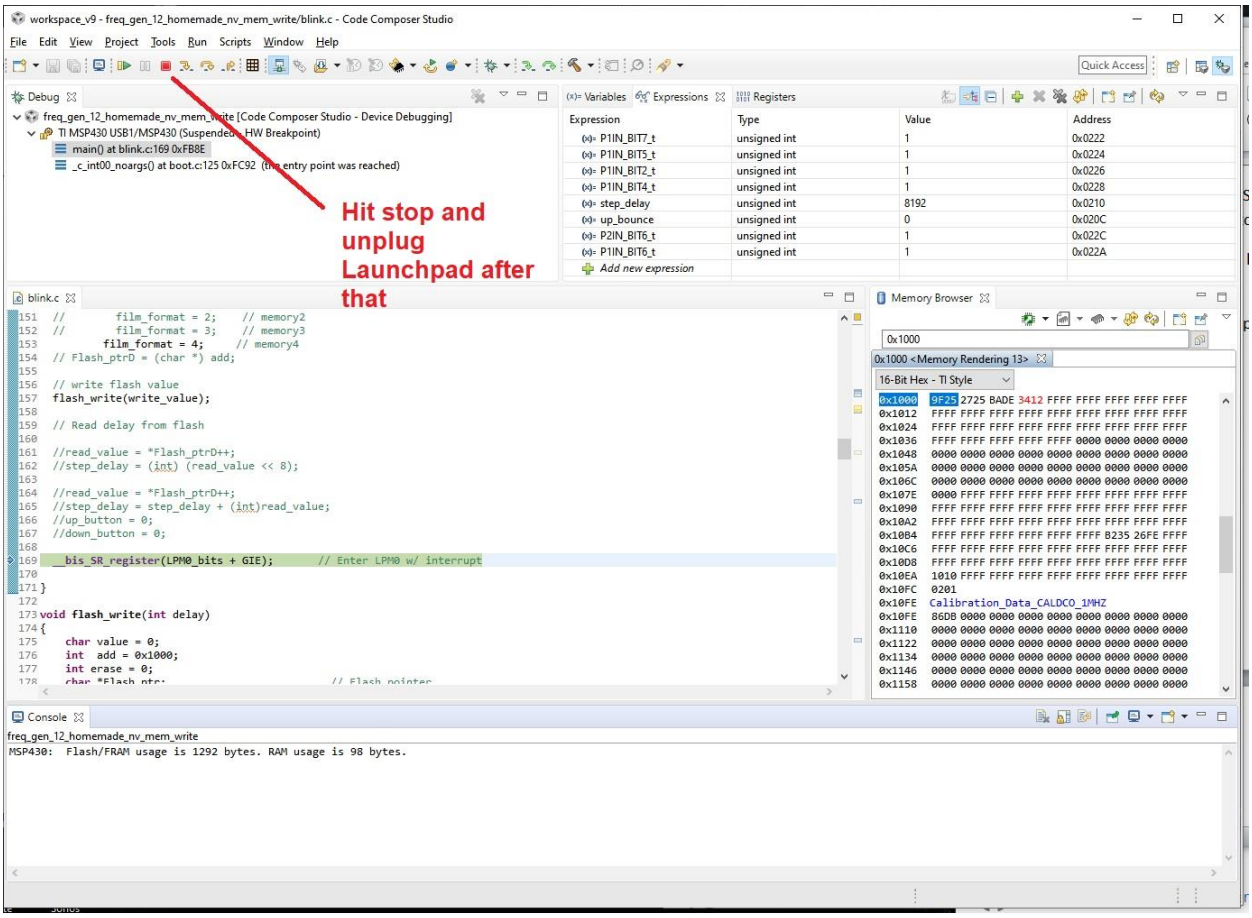


Press F6 key (Step Over) repetitively until the code execution pointer passes the flash_write line. Check the memory to make sure the right value is written into the memory.

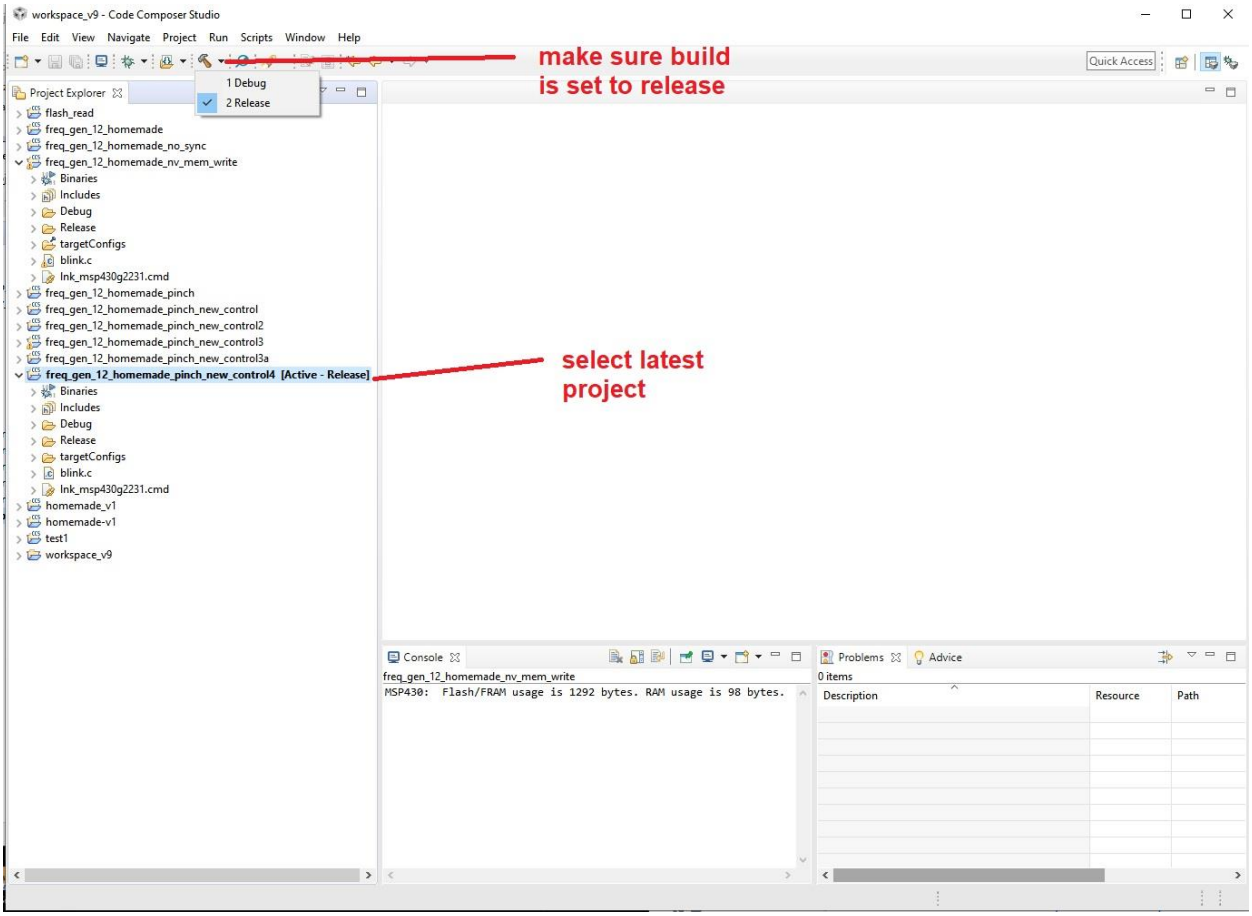
Note that the bytes are in reversed order i.e. the value of 0x1234 will be displayed in the memory as 3412.

And that completes the memory initialization. At this point hit the RED stop button on the top of the CCS window and unplug the Launchpad.

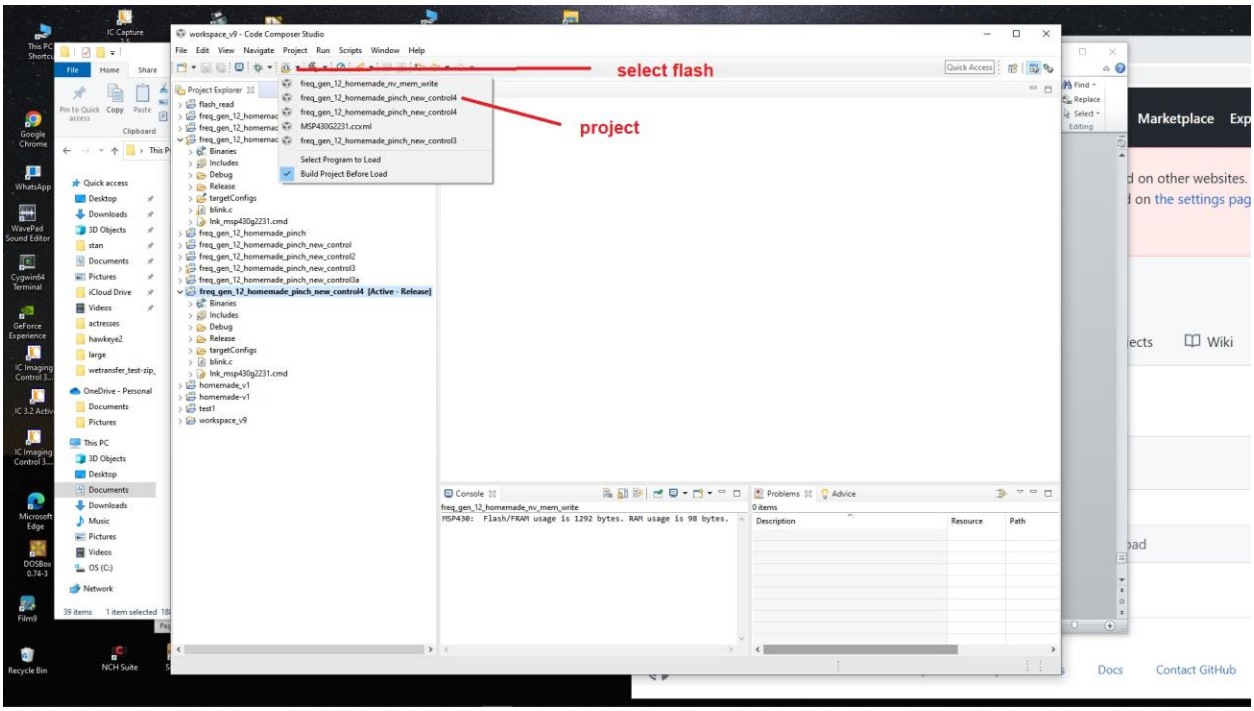



workspace_v9 - freq_gen_12_homemade_nv_mem_write/blink.c - Code Composer Studio
File Edit View Project Tools Run Scripts Window Help
Quick Access
Debug
freq_gen_12_homemade_nv_mem_write [Code Composer Studio - Device Debugging]
TI MSP430 USB1/MSP430 (Suspended - HW Breakpoint)
main() at blink.c:169 0xFB8E
_c_int00_noargs() at boot.c:125 0xFC92 (this entry point was reached)
Hit stop and unplug Launchpad after that
Variables
Expressions
Registers
Expression
Type
Value
Address
P1IN_BIT7_t unsigned int 1 0x0222
P1IN_BIT5_t unsigned int 1 0x0224
P1IN_BIT2_t unsigned int 1 0x0226
P1IN_BIT4_t unsigned int 1 0x0228
step_delay unsigned int 8192 0x0210
up_bounce unsigned int 0 0x020C
P2IN_BIT6_t unsigned int 1 0x022C
P1IN_BIT6_t unsigned int 1 0x022A
Add new expression
blink.c
// film_format = 2; // memory2
// film_format = 3; // memory3
film_format = 4; // memory4
// Flash_ptrD = (char *) add;
// write flash value
flash_write(write_value);
// Read delay from flash
//read_value = *Flash_ptrD++;
//step_delay = (int) (read_value << 8);
//read_value = *Flash_ptrD++;
//step_delay = step_delay + (int)read_value;
//up_button = 0;
//down_button = 0;
__bis_SR_register(LPM0_bits + GIE); // Enter LPM0 w/ interrupt
}
void flash_write(int delay)
{
char value = 0;
int add = 0x1000;
int erase = 0;
Memory Browser
0x1000
0x1000 <Memory Rendering 13>
16-Bit Hex - TI Style
0x1000 9F25 2725 BADE 3412 FFFF FFFF FFFF FFFF FFFF
0x1012 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x1024 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x1036 FFFF FFFF FFFF FFFF FFFF 0000 0000 0000 0000
0x1048 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x105A 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x106C 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x107E 0000 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x1090 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x10A2 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x10B4 FFFF FFFF FFFF FFFF FFFF FFFF B235 26FE FFFF
0x10C6 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x10D8 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x10EA 1010 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0x10FC 0201
0x10FE Calibration_Data_CALDCO_1MHZ
0x10FE 86DB 0000 0000 0000 0000 0000 0000 0000 0000
0x1110 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x1122 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x1134 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x1146 0000 0000 0000 0000 0000 0000 0000 0000 0000
0x1158 0000 0000 0000 0000 0000 0000 0000 0000 0000
Console
freq_gen_12_homemade_nv_mem_write
MSP430: Flash/FRAM usage is 1292 bytes. RAM usage is 98 bytes.

Plug the Launchpad back in. Make sure the build is set to release.



Flash the MSP.



Unplug the Launchpad. Remove the MSP and plug it into the target and retest the delay values.

IMPORTANT NOTE: The memory write utility writes a value only to one memory at a time. Make sure that you unplug and plug in the Launchpad after every write.

Appendix

Stepper used:

https://www.amazon.com/gp/product/B00PNEQKC0/

Stepper cable:

https://www.amazon.com/gp/product/B0776J8CCN/

Wiring:

Motor	Controller
Red	Red
Blk	Grn
Grn	Blu
Blu	Blk