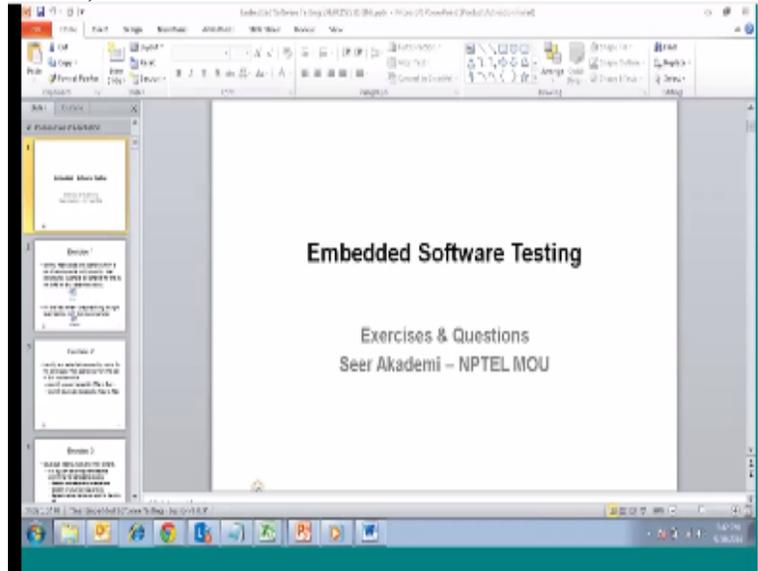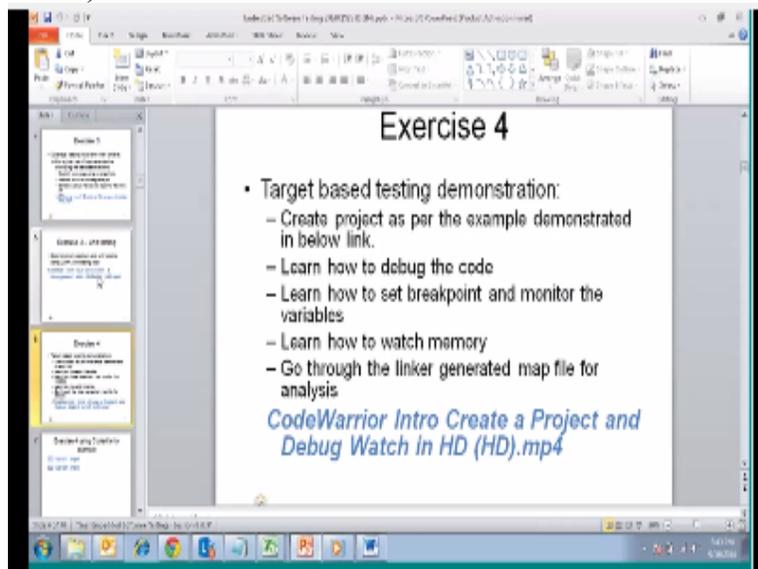Hi all welcome to a next practical session of an embedded software testing.
(Refer Slide Time: 00:09)

And today will go through a session about code warrior.
(Refer Slide Time: 00:22)

Intro creating a project and debug because we know in interesting we need to execute the embedded software not to target using equipment called IDE. So we have intranet development informant so is basically used for connecting the embedded target with help of the target base connectors such has étagère are any debug reports medium excreta. Then we have an actual target and we have the complier go to the code warrior code IDE code warrior basically IDE and we take understand how we can used the debug.

So basically with help of that we can see of the break points are starts the view memory and monitor the variables how to watch the memory modify. So like these are several aspects of that

we could study that in a couple of sessions today will try to go through a video of code warrior basically we have code warrior connect with the micro controller the most of this frees scale mailed control used they were.

A code warrior imposed the specially in there initial service of HC08, Hc12 serious process these are popular process since almost couple of debugs are the using on the automotive industrial and such micro control applications so let us stately understand and how to create the project with an example of demonstration does not to the debug the code does not to set break point and the how to watch the memory.

So this was the exercise with try to understand and we continue the exercise in the next session okay so basically we HD circle one of the embedded expert video, call processor expert software,

(Refer Slide Time: 03:06)

So this serious will referring to understand so in terms of all session okay, this YouTube video form of processor expert software serious will specify to go through and the feature fiction will try to understand to works. So this is called has a connate curser sagest with the help of that we will try to understand of the code warrior tool. Okay so we will begin with the example project work space this is are window you can see the code warrior project first space wherever are the components category project listed here.

And in terms of managing the projects the build setting debugging the all that you can because you will console were debugging output will be provided. So we will take to open the example project. So you click the input project and there is an input project in some of the reregister is to

we have to import winder we can goes to the group to the example project here we have one try to see project.

We have method to go through that and we will select that use it. You can see the project got selected the left hand side were at list the file names with the different folders.

(Refer Slide Time: 05:18)
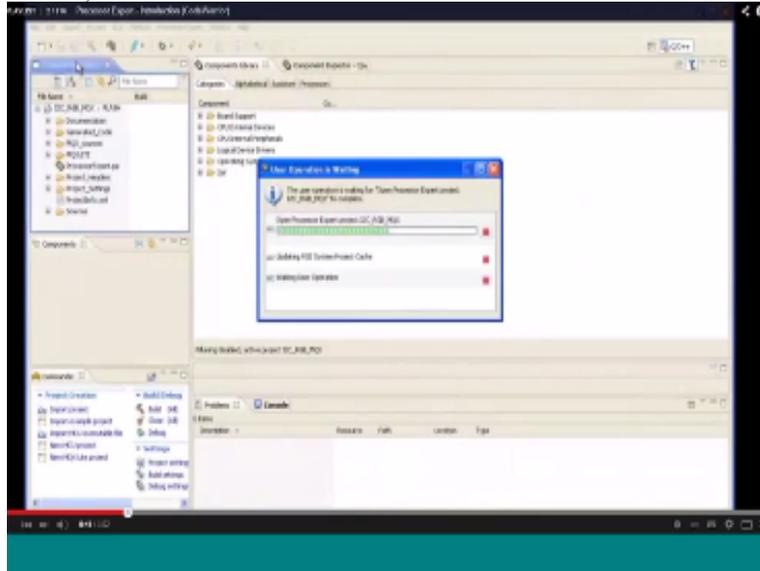
So we have trying to build the project what is saying? The processor expert is the project file to open the free file is the project processor expert the project file. We can see several options coming up in terms of congregation like flash, memory and the components what to want to use in the project. So you can see these are the basically the registers part of the processor you can see that internal maculated clocks and all that and all that these are basically components of the cross what this processor expert this.

Which have library to components those ask is to select the CPU external, internal clock and all stock. So privet of that we can choose these options and build the perfect. Okay these are some of the categories that components library that is build in the processor expert of the code warrior were he can steel about support information CPU external value CPU internal service for else logical to the drivers.

If the board adds the connections try to see as driver then we add operating system and small carnal or then we held the software, software application configuration. So we can see closely refuse there is one wires selected has MC, X slide other component we can select with the multiple task what to want to have it.

So there are several task and communize wire components so we are just tried to understand how to the Ul walks. So what components are used in the OS that is building us part of the processor

except, so this as several methods which in the components you want can congugarate it is one library it is being selected here.

In this for that library TSSI you can see the library having the difference complex components of the library can be selected or deselected used so basically with the help of that project is built okay.
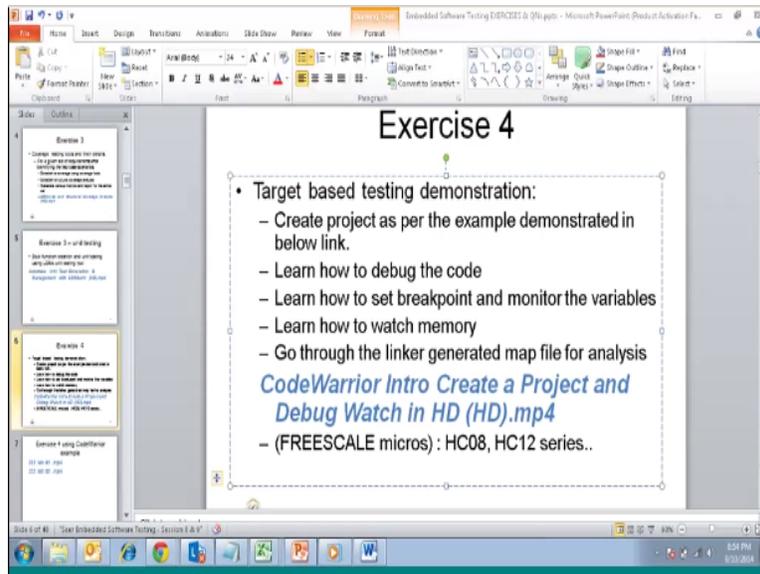
(Refer Slide Time: 08:58)
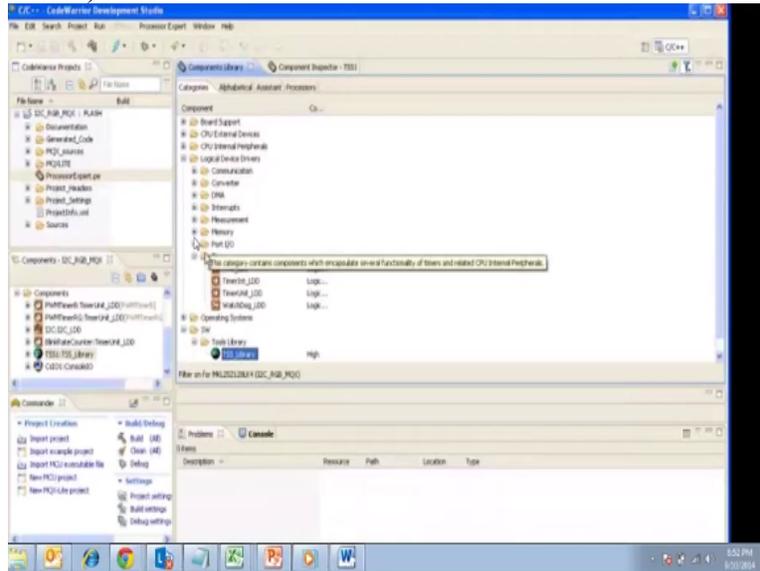
So you can see logical device, dimensional in terms of communication, converter, many drivers are there multiple presale of use like timer what should have timer ,so basically this video is for building project similarly we can use it for testing in terms of what is used after loading this project are at the use for development see.

What configuration it is using what is a behavior something like what Doc we have source we can remove if you do not want that component so what processor except does is generates the code for this and builds the project you can click on the generate button and generate the whole project okay. We will try to study another project.

(Refer Slide Time: 10:00)

Similarly created in a code warrior
(Refer Slide Time: 10:06)
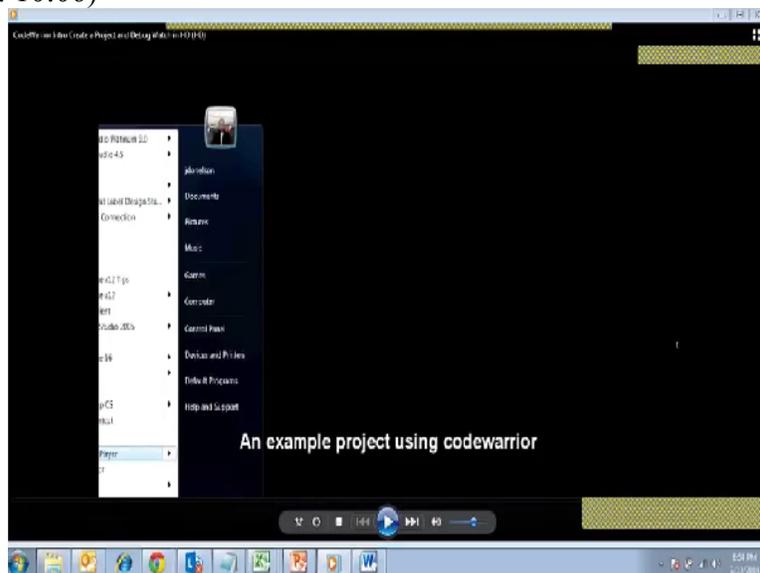
This is the more simpler one built it study this what it does okay, today in terms of code warrior development studio alert works this video showed the hell this work space we can select it in terms of lab example path so you can create a directory name it and you can store projects in a folder called works.

So workspace will have one more projects will be available but better have a separate workspace for each project so if there are complicated projects having multiple files and all that is attainability one workspace under any part we can multiple projects, similarly you can built test project having instrumented code, or the unit which you are interested to feed more values concur at the code.

And test it what is the behavior of that we can build in project this way. To the code warrior studio irritating setup this is a first page basic thing.
(Refer Slide Time: 11:34)
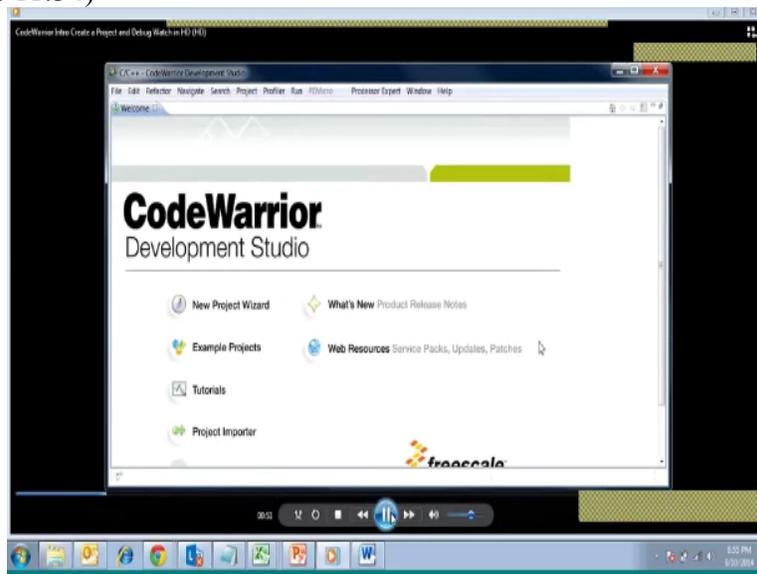
Is called the perspective it is a set of development projects settings you can see you can use this come strain create a new project so we can click that, new project result and you can name the E project here remake a control that is being specified here you can called it as a lab one, can press next and you can choose what sort of a word.

We have process of a like a C you just V1 called V1 V2 V3 so one of it is process we can selected that whole V1 used being is a family of all the code for persons for another one, this skill so there are is a processor called MCR 51 CNR 28 that is being a selected and what could of that a connection that we have with the N target from this ID we can selected, there are different options like USB based PDM debugger or a serial code or a TCP or any data as etc. Is a P&E USB PDM multiple is used here?

You can have also open source PDM which is already available on the evaluation code many files you want to add we can add it here.
(Refer Slide Time: 13:26)
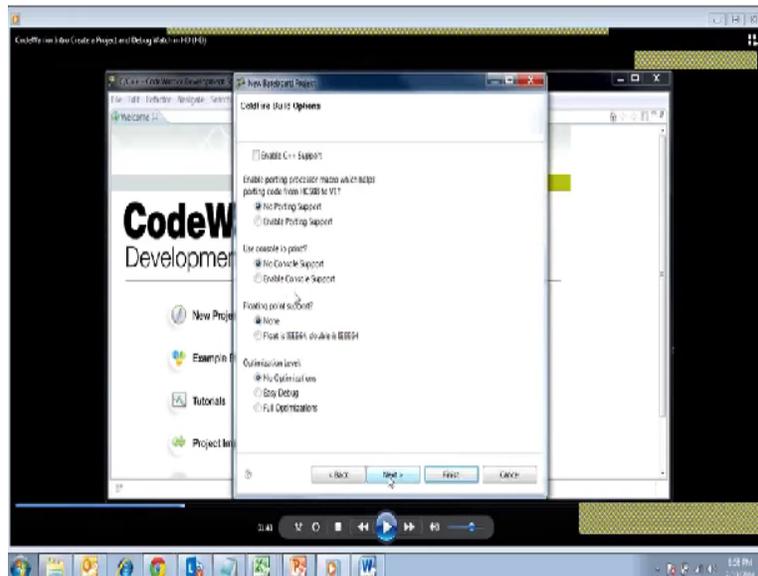
So code file builds options you can particular what certain of grade you want to have, C++ support, you want to enable closer macro which helps the code form any skilled on process something like that use to this code warrior so service port is selecting here any input output, in terms of inter statement you want to have a account, so long any float point support we need any optimization or debug etc.

All this options part of the built to inters elects that the embrace of the program after program is creating build okay, next then we have rapid applications development and in terms of building the entire workspace from the project we can select either device insulation ID is using was expert what are assign earlier in terms of conferring the values of the registers devices and all that whether if you can ask processor expert visualize or you use can initialize or one of the you influence.

So here it is not use the proper export it is no this is used option, now you should have one project example you can see a different folders that clear that in terms of projects things library, files sources so we can see that we C++ on the right side to side so you can just go head and select the project window at the method you can build the project with built all so it is going to built the entire and BBR builder.

(Refer Slide Time: 15:43)
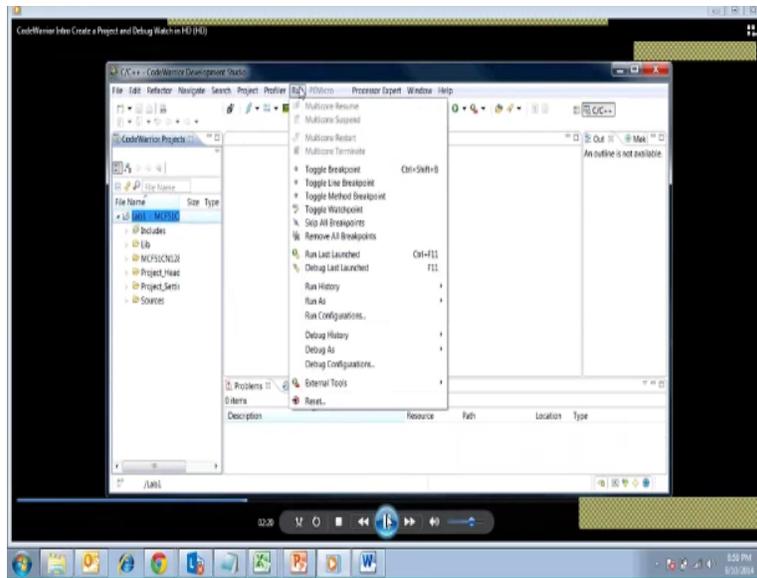
We can use then you are going to have a creator configuration now you want able the project after you are built.
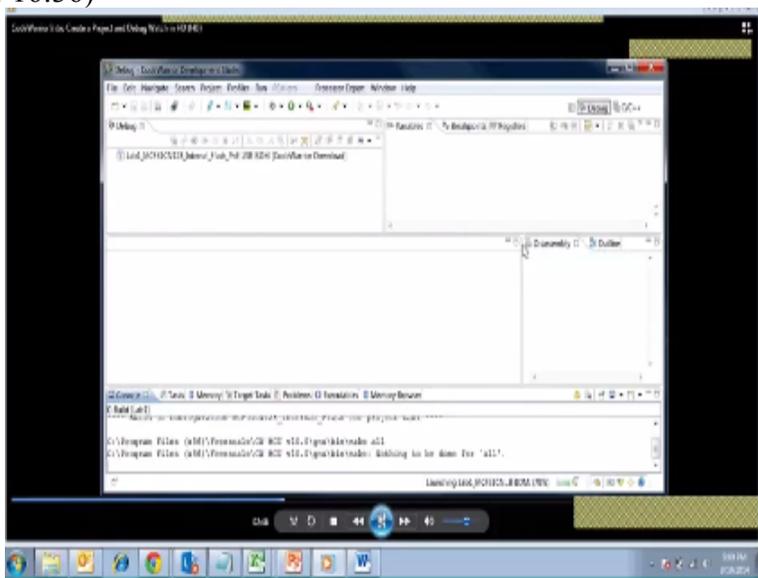
(Refer Slide Time: 15:51)
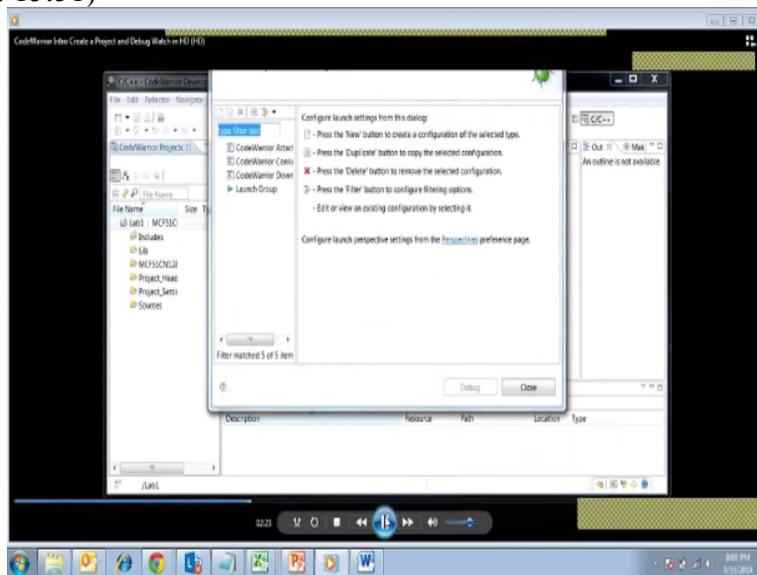
A company here with this window you can have the debug configuration. There is an example code word is downloaded used to have that is way, is has been chosen.
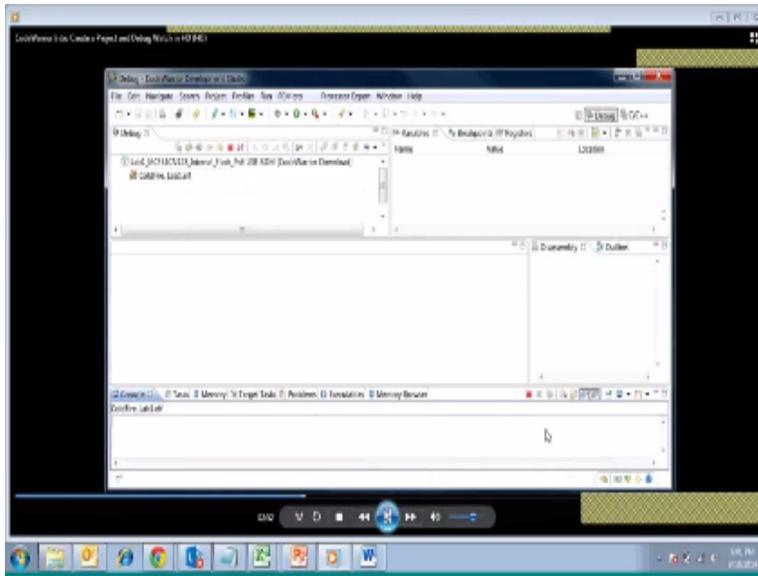
(Refer Slide Time: 16:17)
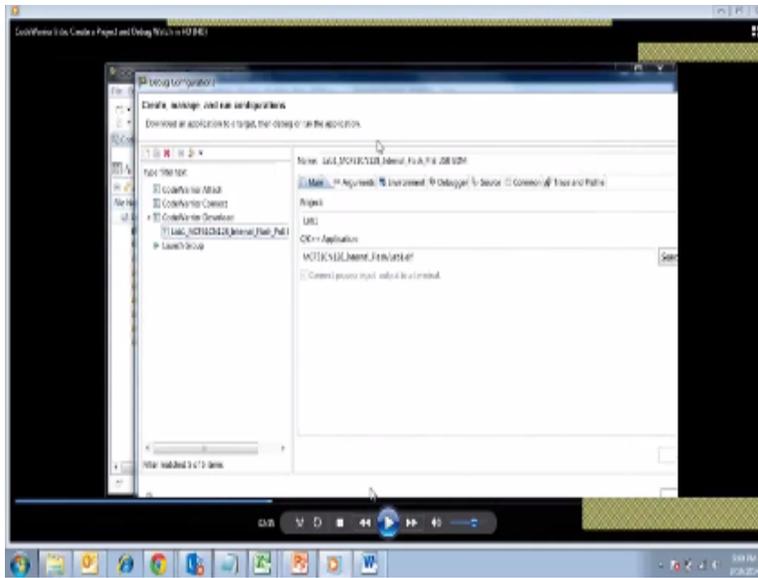
So basically the program what we are built is trying to that flashed on to the code we can see that l format this a concentrated on that built we have written same thing can we flash on to the target code.
(Refer Slide Time: 16:36)

And clearly we can go for the debug option it will automatically open the debug prospective here we can see that.
(Refer Slide Time: 16:51)

Now it is a downloading order the target, it turning into the flash executes the debug it showing the window after it is loaded.

(Refer Slide Time: 17:05)
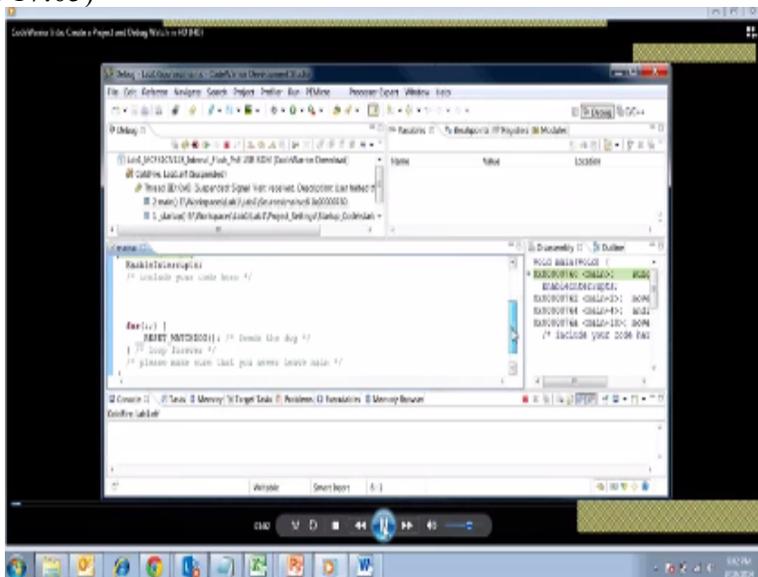
And here we can see the debug, operations what are the things that we have her we see the file which file is been currently program or currently being a executed instead a consoling the input output this is dis assembly of that main file what is resource, so the executable code you can see just simple code main interrupts.

And a loop some recent process we are getting. Then this why because it has I wash bug circuitry and words are should be again and again certified so that no result is going to happen so any in their software you know that we should be call in while 1 and a for 1 loop for while 1, while 1 is

referred so for this example which we see for loop so this is the project and we can see, we can get the resume of can you can use.

(Refer Slide Time: 18:16)
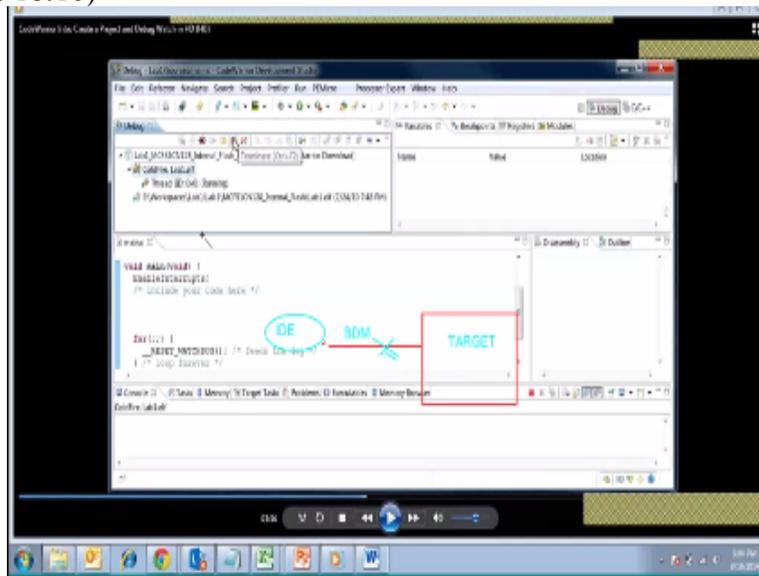
Now the program is running on the board you can pause it, it will show It will show where it is stop and where it is running and the sort were the project till the executed so it is continues running around the for loop, you can resume it we can stop for terminate the project execution we can disconnect the target which is connected to this IDE the debug and normal, what happens need to disconnected because are flashed and the earlier option said that resume or execute. So though you are disconnected actually flash.

The program and asking it to run dependent running of the target but the PDM of the target that has been connected will not have any debug option so, so here we have PDM debugger module and actual target is here, so what we trying to do here it is I did try to load the example program flash out to the flash circuit of the target mode and we said let it run disconnect the, target still the target.

(Refer Slide Time: 20:30)

Similarly we can connect backend start executing it, so now we can see the source code C, C++ review this display if you want we can add code for example let see this is a code getting added the variable counter is added here, and for loop the this a counter getting implemented in a for loop, so you save it as a control S then since you change the source it will be first source it will be rebuilt so you have the debug connection set up then you can source to the debugger can I said set up then make debugger.

I mean a getting art the when debugger and the target this is the progress it shows as it was showing an idea for a new code. We see here a new code getting displayed here and the resource similar the same shown here and, you can see that table that showing this let us progress the okay.

(Refer Slide Time: 21:52)

Now shows the different act trays difference of what has been called first second and what has been called trace it is showing so you can see that used regulate that start, have that is have before remaining is getting called the basic start has been used stand the assembly had used start up resolve the assemble.

(Refer Slide Time: 22:18)
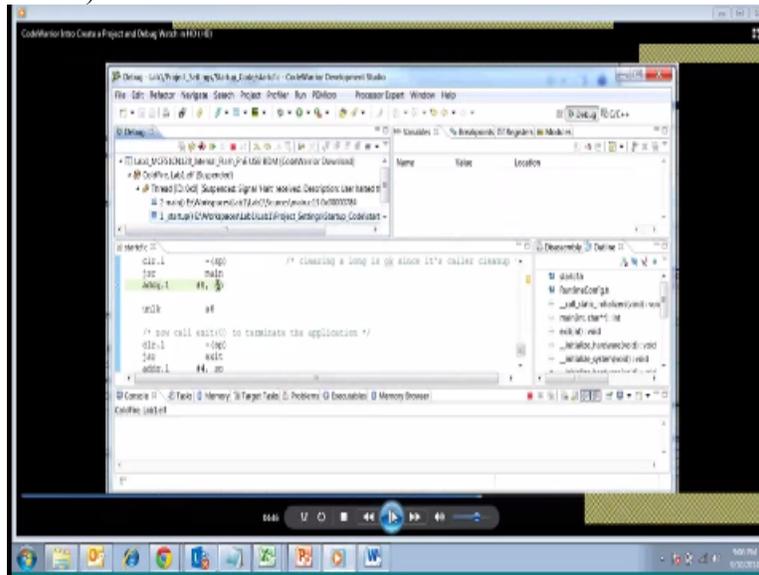
All that the basic quality of processor declaration allocation buffered flash all that instructions part of the start of any before many actually Getting call are executed, you can that is a main has come.

(Refer Slide Time: 22:40)
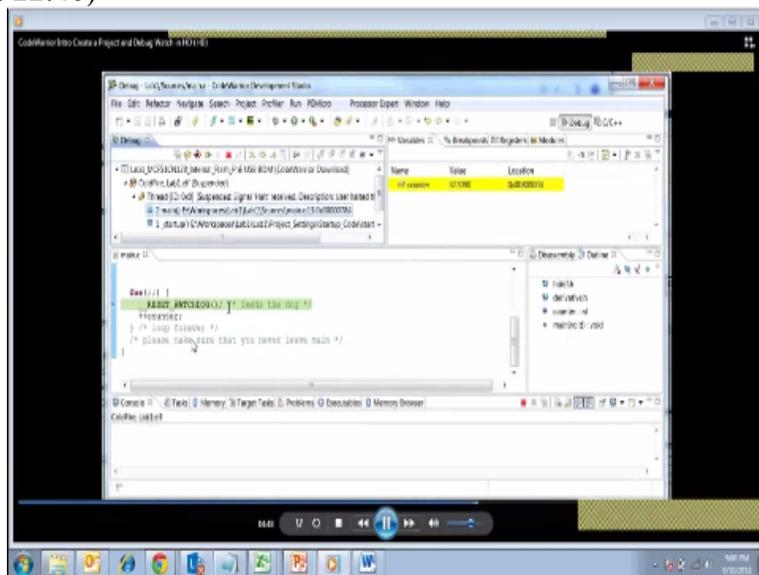
Is a execution of this there is a counter you can see which I have used call a global variable similarly using this display this will be are the continuously in demanding is every cycle so the

what we like a value that got operated and also using the memory rare the countries playing on the target, we can again this project see.

(Refer Slide Time: 23:15)
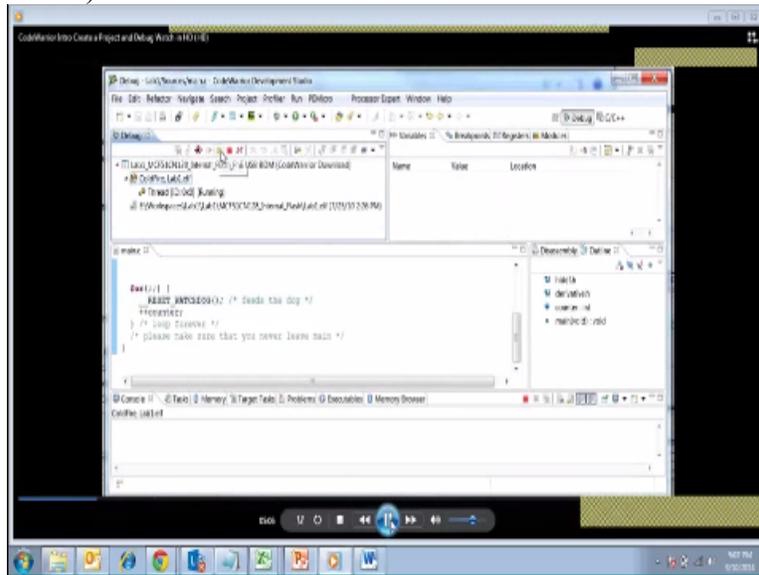
You can see the connect or rejecting updated or intermitted We can also see by showing the cursor on the particular variable of the content we can also that declaration as well can see the values of there what is ratio it is so we can see the values of the variables to the debugging, so now we can try to said break, fonts and try to see what is going on recycle we have to look like here user where plain were we can click and see that like what is being label and of the stop at the break point.

We can see every run is blinked here and is going to stop at the break point, you can see every run has been clicked here, and it is going to stop, because we have set the break point, every time, the breakpoint is enabled and you can see the counter, it is getting incremented right, so if you remove the breakpoint, by double clicking and run it.

(Refer Slide Time: 24:23)
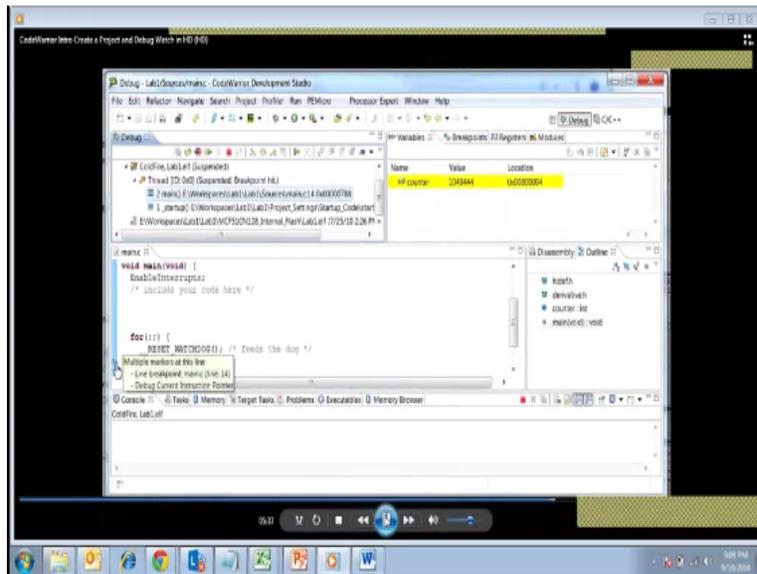
And you can go the breakpoint button, and you can disable it, with this option.
(Refer Slide Time: 24:29)
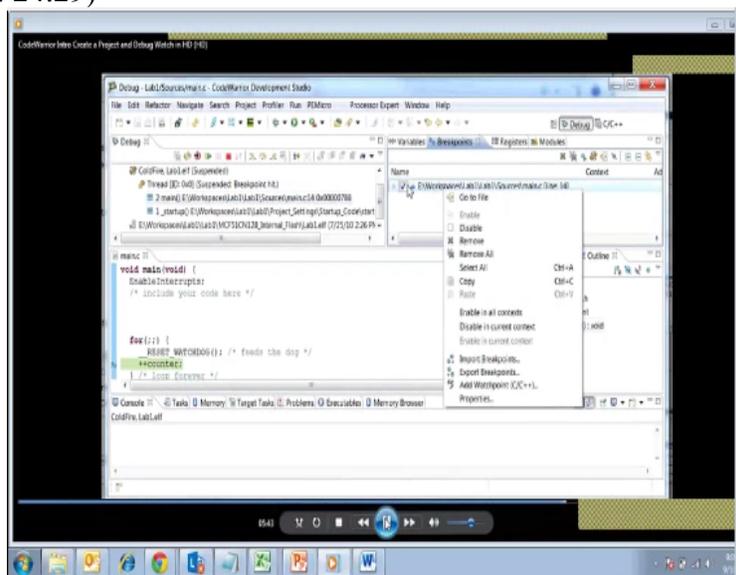
And run , it will run a free execution, and while doing the execution ,you can modify the value, or the location what we have seen, it will be useful for testing, however the behavior of the program ,to ensure that different aspects, of the project variables, how it is getting used, that is what the example shows, you can have a break point at different places, likewise, some of the embedded systems ,will allow, 3 hardware breakpoint, and unlimited software.
(Refer Slide Time: 25:10)
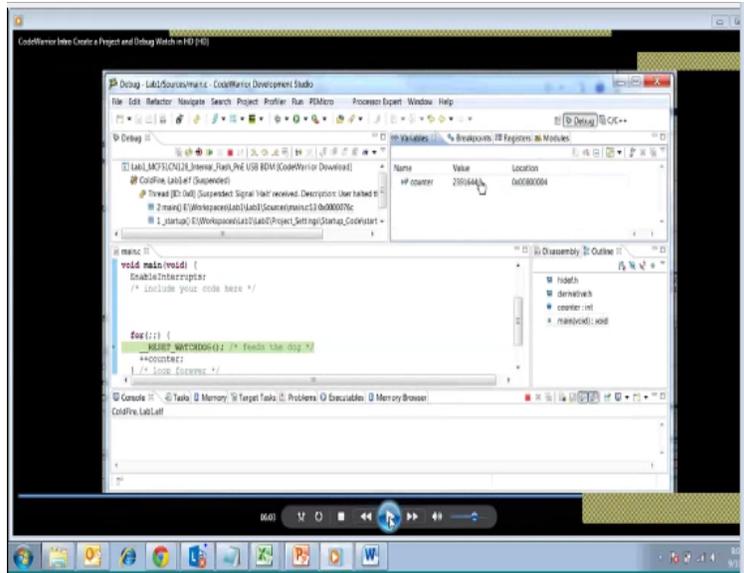
The hardware breakpoint means ,the breakpoints that are getting, loaded, for the target directly, the actual target memory, and which is executed to the memory of the RAM, so you see that value, being changed, from something to 0, so that a new variable, value is being loaded,0 is being loaded.

(Refer Slide Time: 25:40)
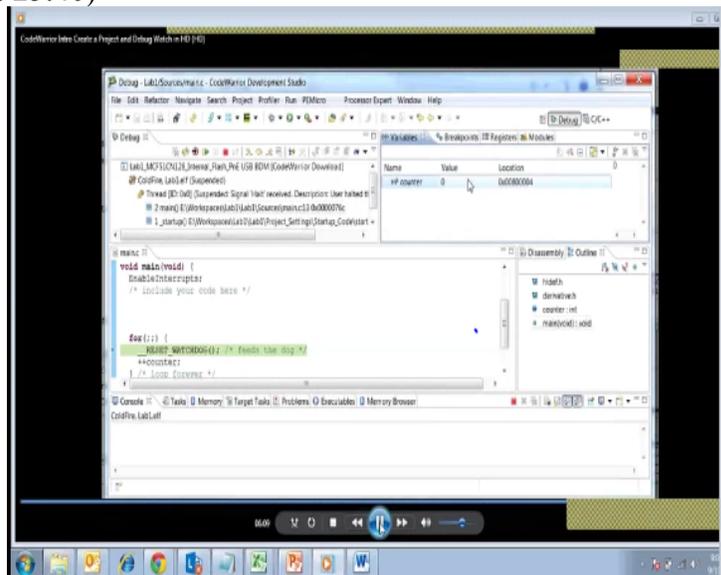
So you can create a watch point, you can click it.

(Refer Slide Time: 25:49)
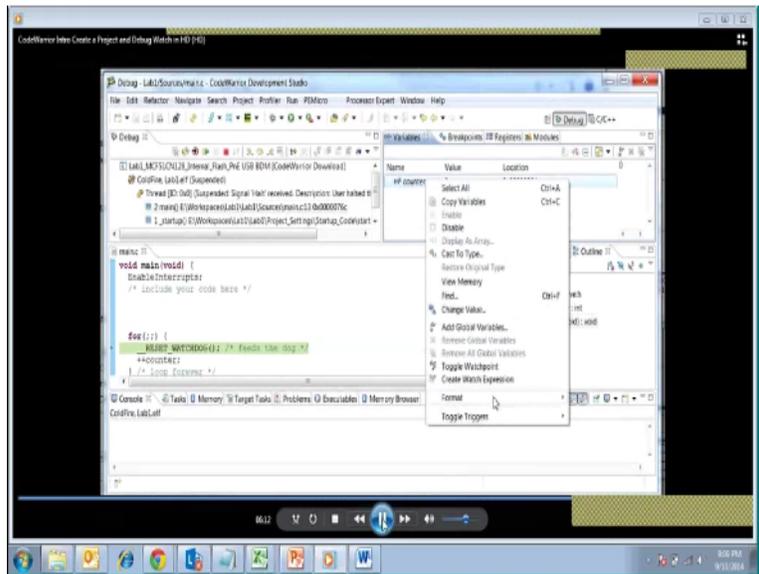
And the counter will want to have a watch, and the breakpoint is enabled here.
(Refer Slide Time: 25:54)
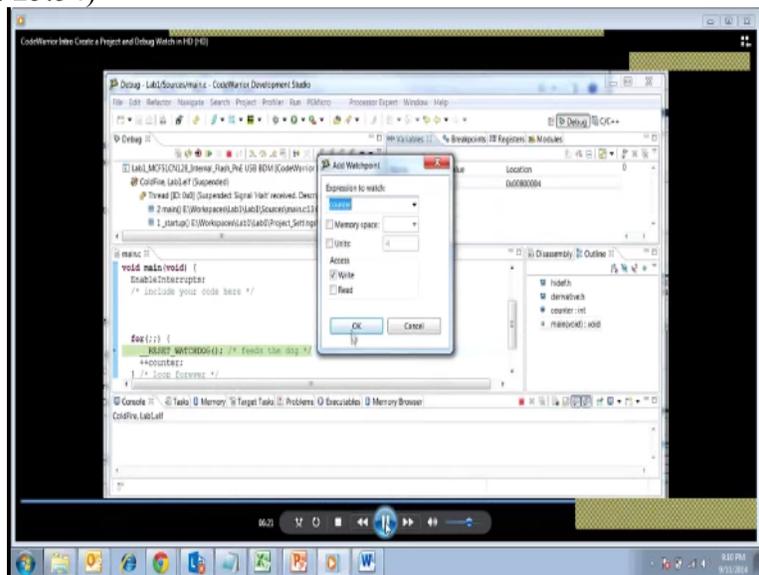
And the properties, condition for the watch point.
(Refer Slide Time: 26:01)
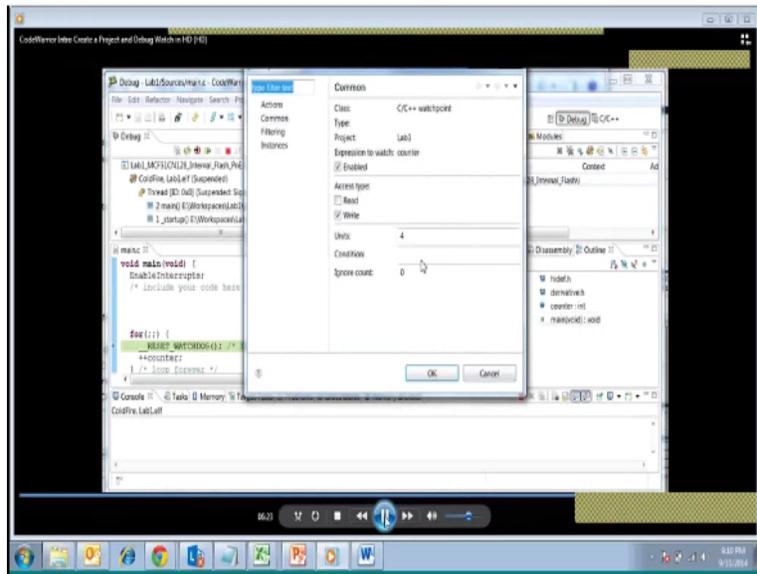
The counter, when it reaches and we should be able to stop and see the watch point.
(Refer Slide Time: 26:10)
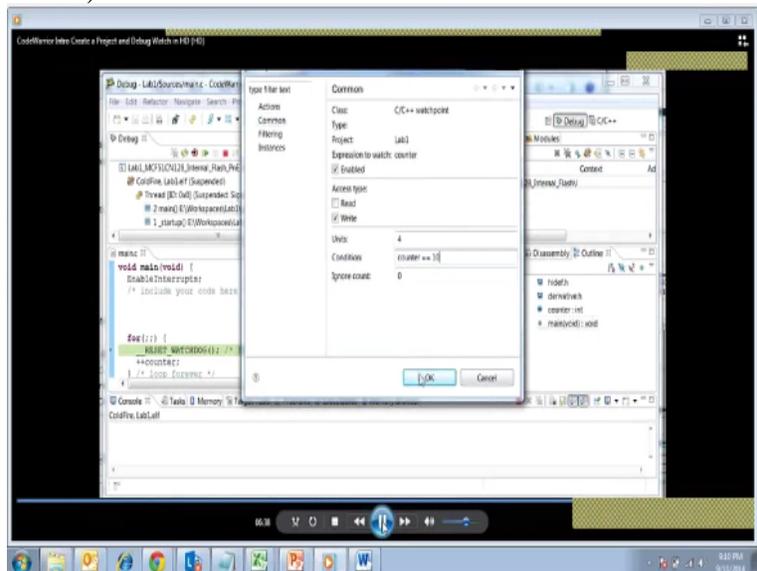
It is 0 now, so we will be able to resume it, and the moment is going, so the value of m and it has

hit, and execute continuously, it will be incremented continuously, and it stops at 10, we can take

it out again and try.
(Refer Slide Time: 27:05)
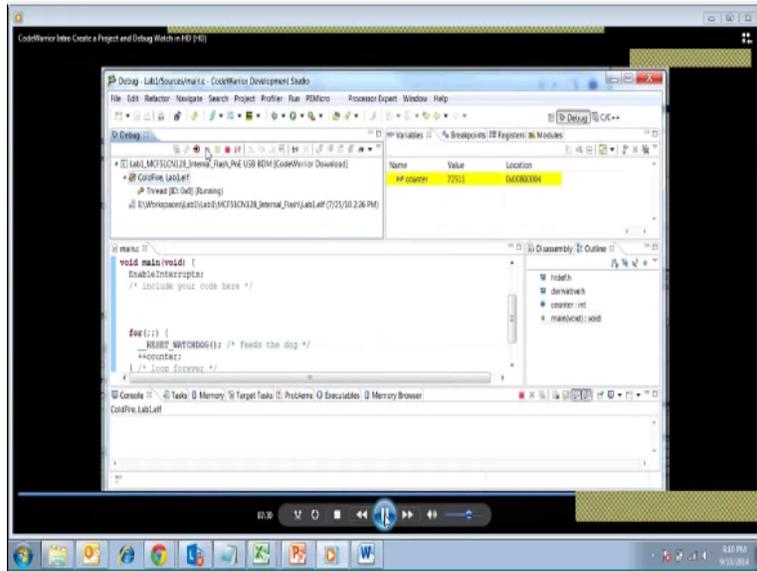
Now you can see the value is getting updated, so this is what the debugging aspects of the code warrior one of the popular debugging that is used in the industry, okay.
(Refer Slide Time: 27:36)
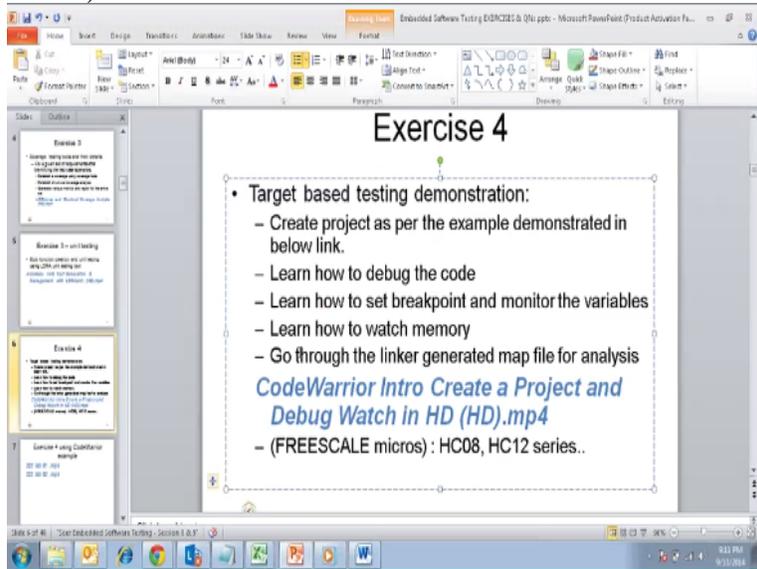
So in the next session, we will try and watch, the next type of code warrior debugging, for different settings, okay so in the next video, we will try to understand, other type of project creation.
(Refer Slide Time: 27:55)
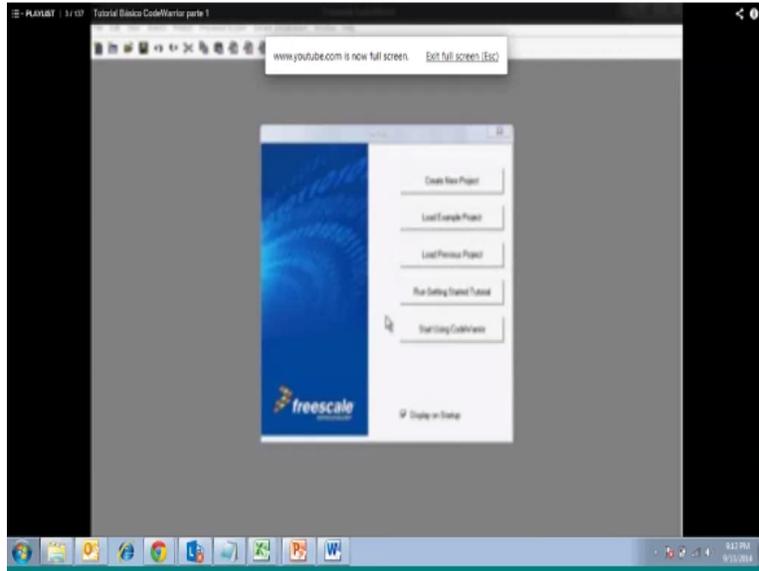
And here same code warrior, has a different version, is being implied to create new project.
(Refer Slide Time: 28:05)
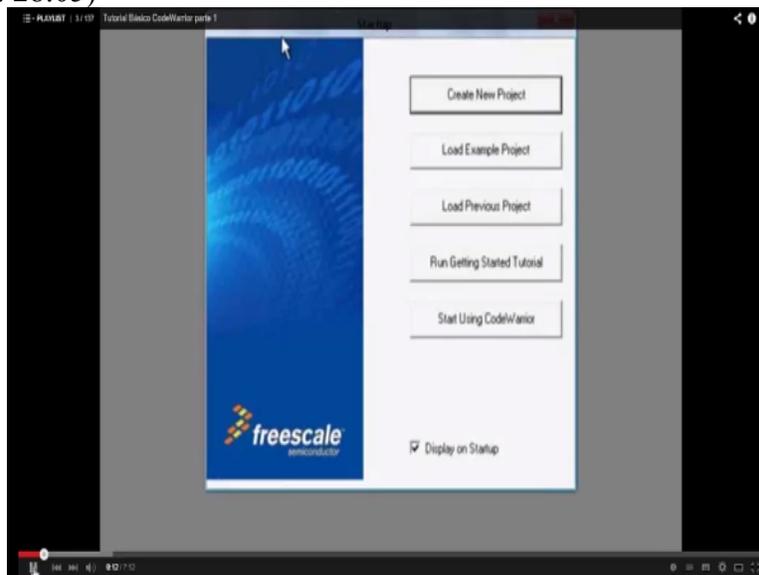
And here you can see the different options like, create a new project, we can load project, load previous project ,run getting started tutorials are also there, with examples, okay s we will try to create a new project, click on that new project. , here you can see this code warrior supporting, this process memory like these are controllers, we can choose, it will give basic information about that processor in this.
(Refer Slide Time: 28:45)
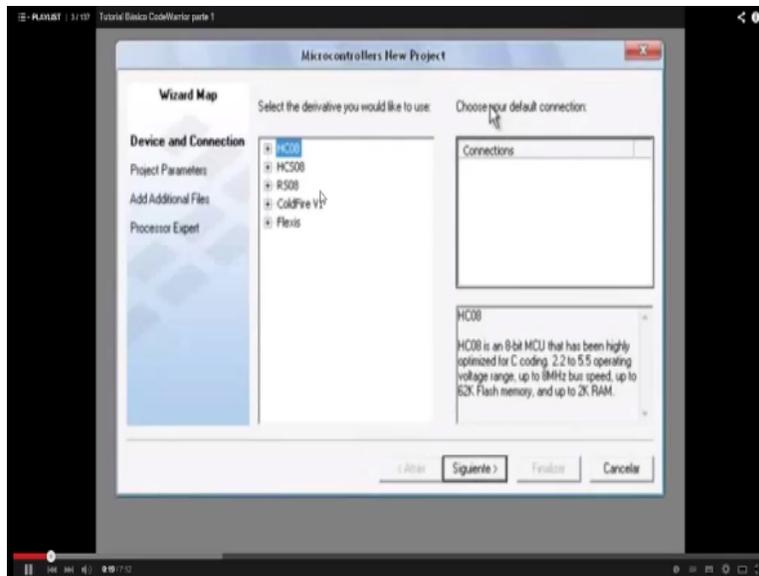
And the connections that are used, so we can have multiple connections for this processors, one

of the HC08 we will try to, select we can see in the window, so the code that got selected.
(Refer Slide Time: 29:10)
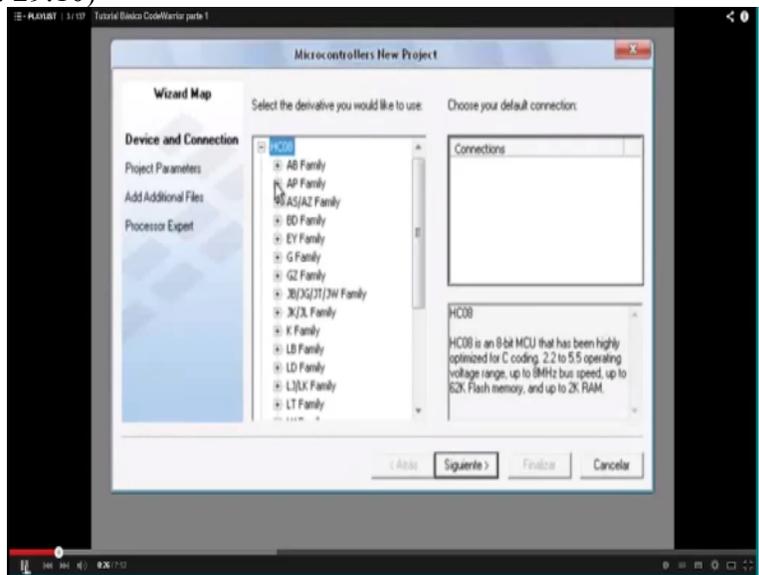
So we can have connections.
(Refer Slide Time: 29:17)
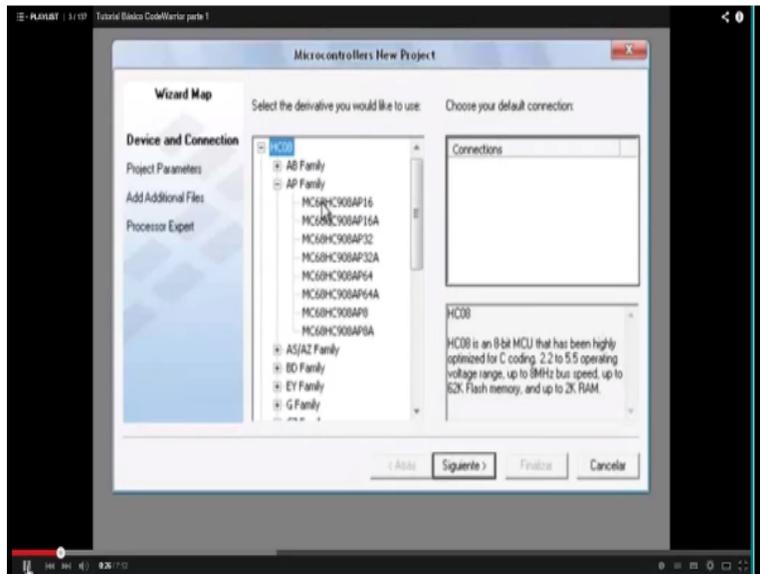
And chip simulation, where simulation is done without the target connection.
(Refer Slide Time: 29:20)
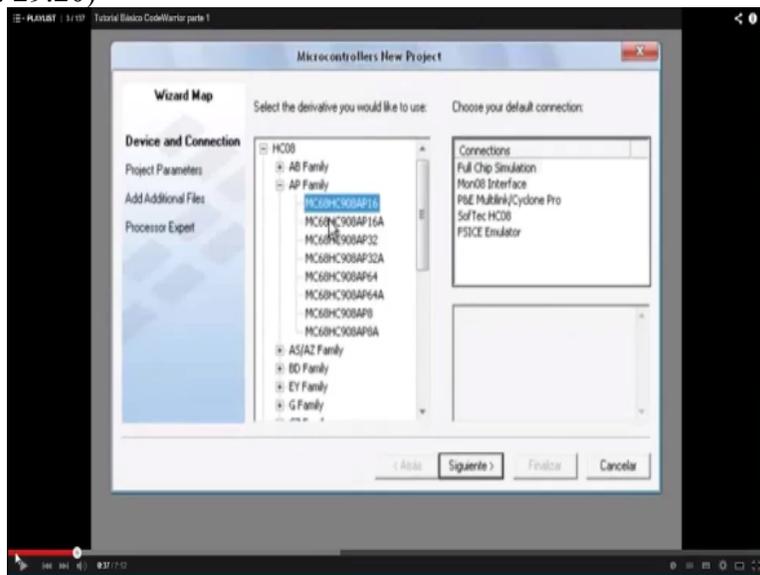
Then end of the process if you select, then we can have languages ,what you want to have, like
this project, to be created like in C, C++, assembly or relocate the assembly platform , so C is
being selected here, the project name, it usually make the project under disk ,we select the folder.
(Refer Slide Time: 29:55)
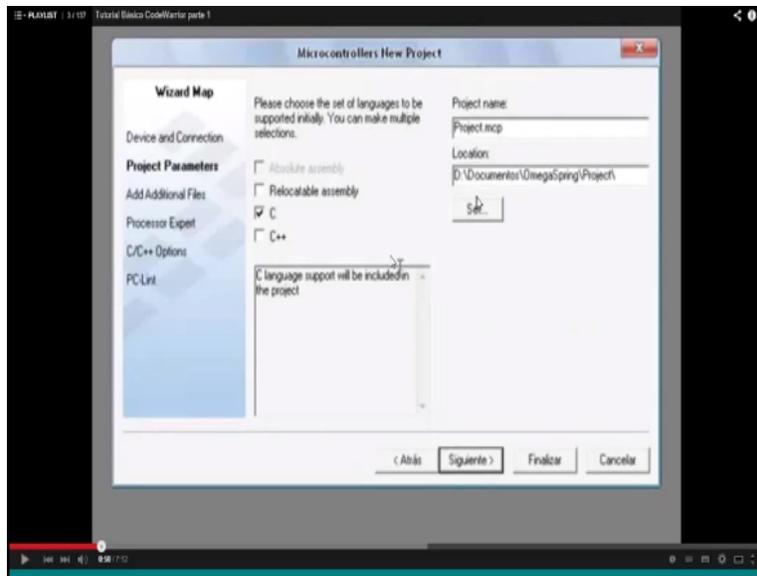
Now if you want to use, the assembly is used for the entire project, this example assembly project, will also have an assembly code.

(Refer Slide Time: 30:30)
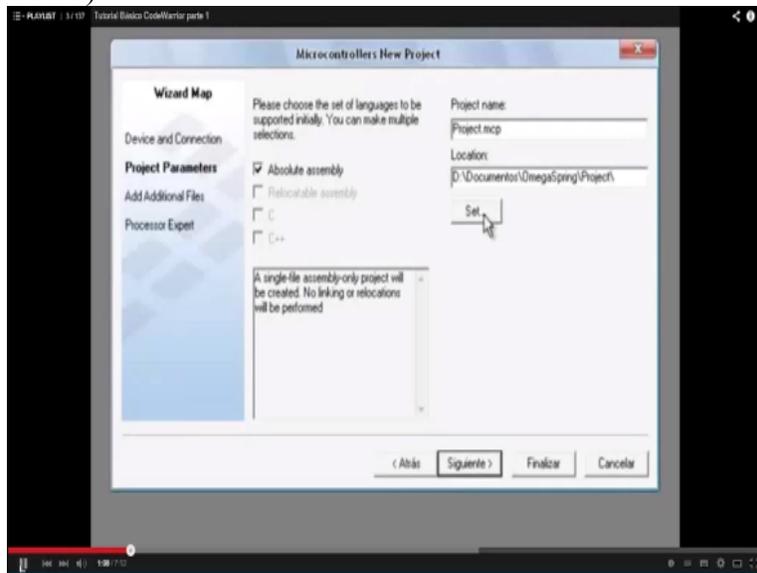
This will also have an assembly code, so you may need to test it, so in order to do that, when all the assembly code has been organized, into the project, so there is some led project that has been crated here.
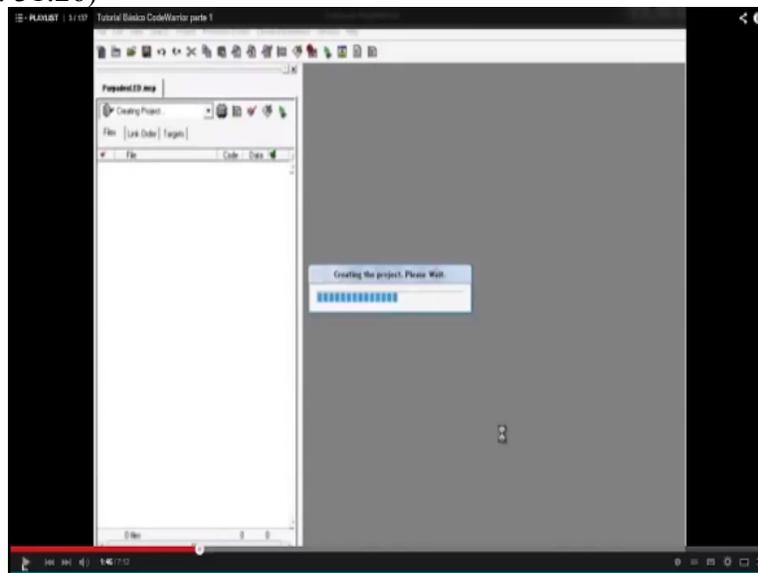
(Refer Slide Time: 30:39)
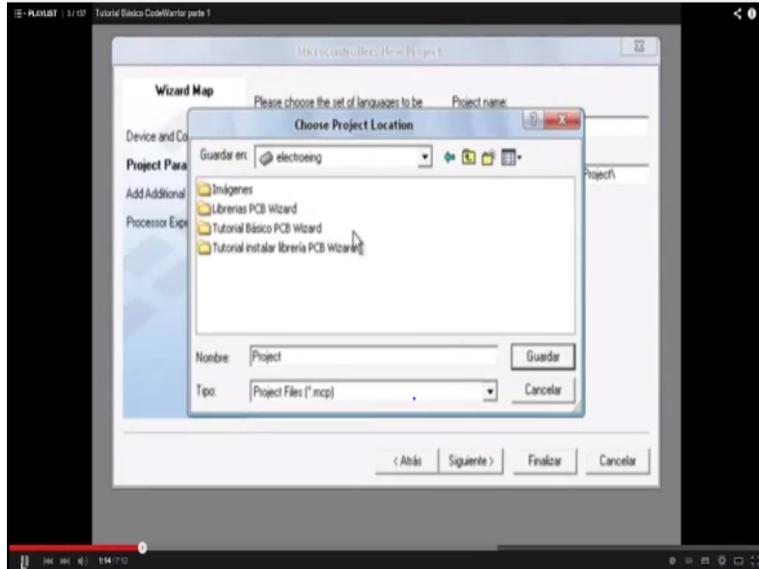
So any files, you want to add, you can add it, in that folder, so any rapid application development option you want, or initially you want to have your own, or you want to select the project explorer you want to select.
(Refer Slide Time: 31:02)
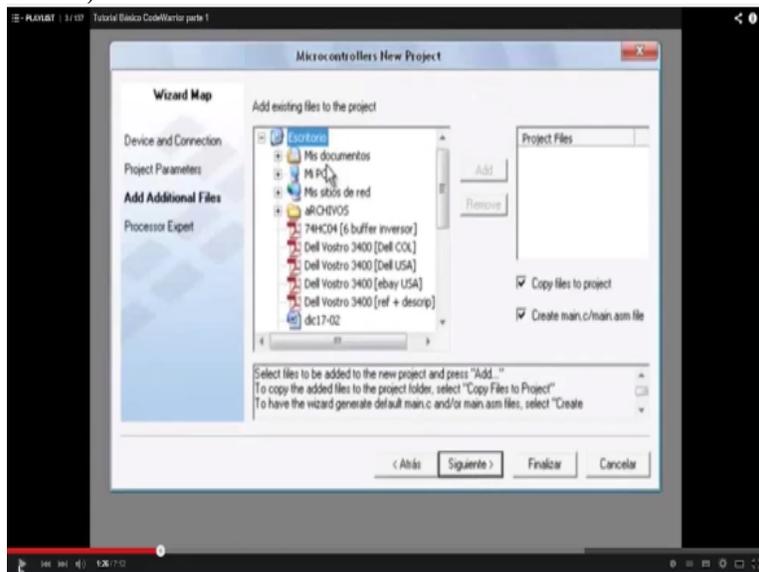
And for assembly that option is not there.
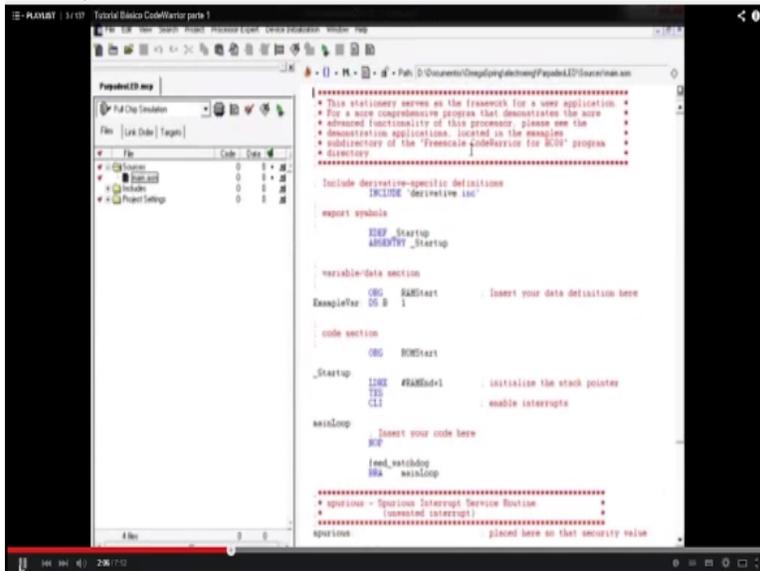(Refer Slide Time: 31:04)
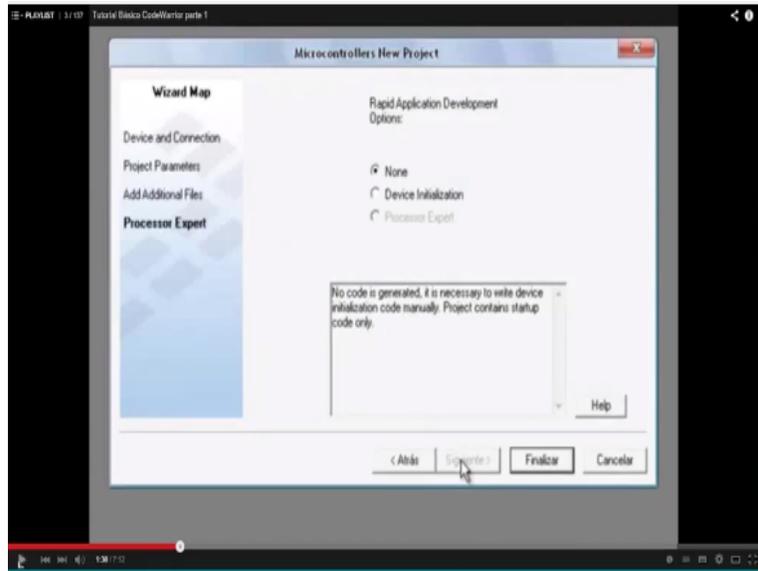
So none is selected here, so now the project is being created, we can see here, the source

main.asm, project libraries and al l is being displayed here, so this is the debugger window.
(Refer Slide Time: 31:26)

And on the left hand side, there are different debug options available, so when you click on that
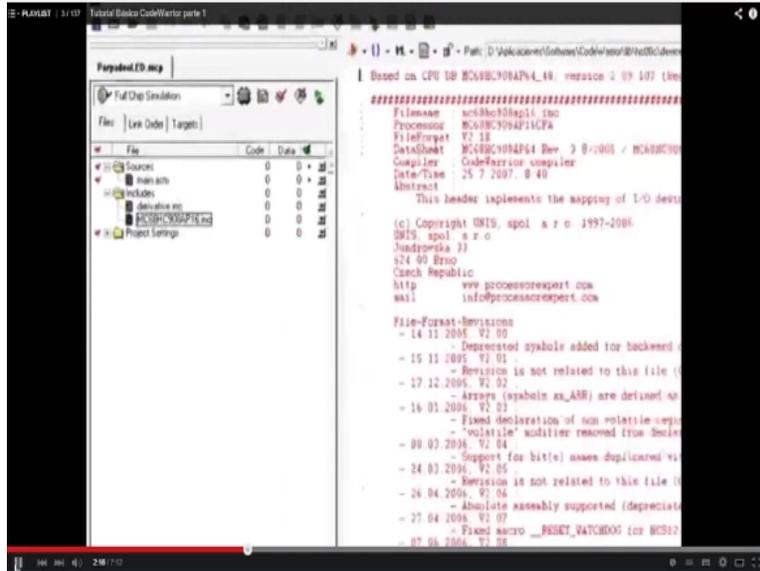
main. Assembly, there is an assembly that is displayed.
(Refer Slide Time: 31:38)
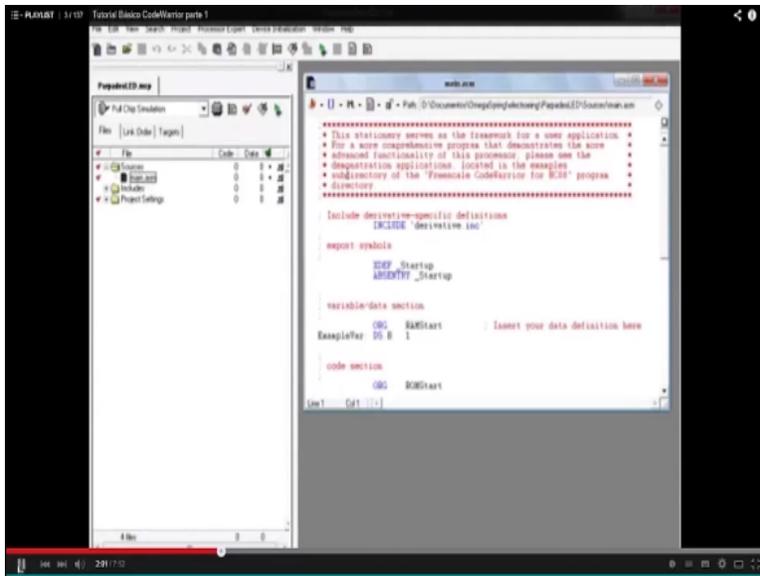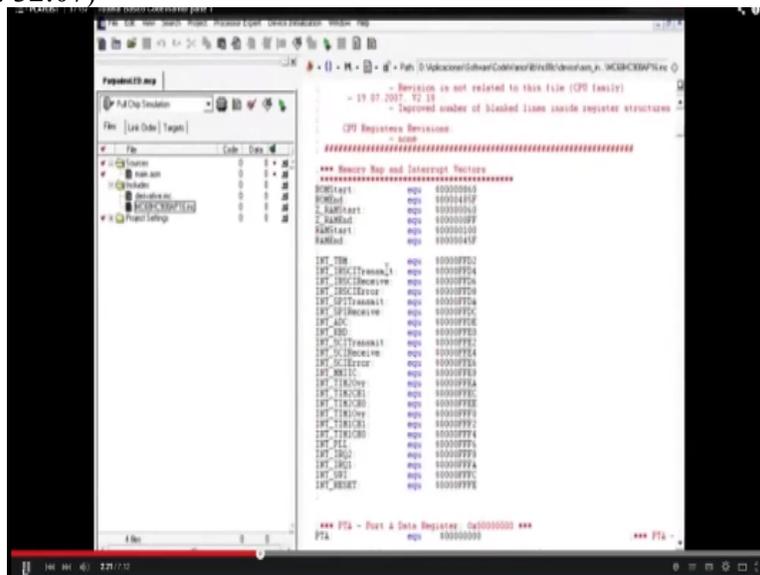
(Refer Slide Time: 31:45)

(Refer Slide Time: 31:46)

So it includes the 2 libraries, .inc, these are under process specific library.
(Refer Slide Time: 31:51)

And you can see that, and try to see the entire micro controlled memory map.
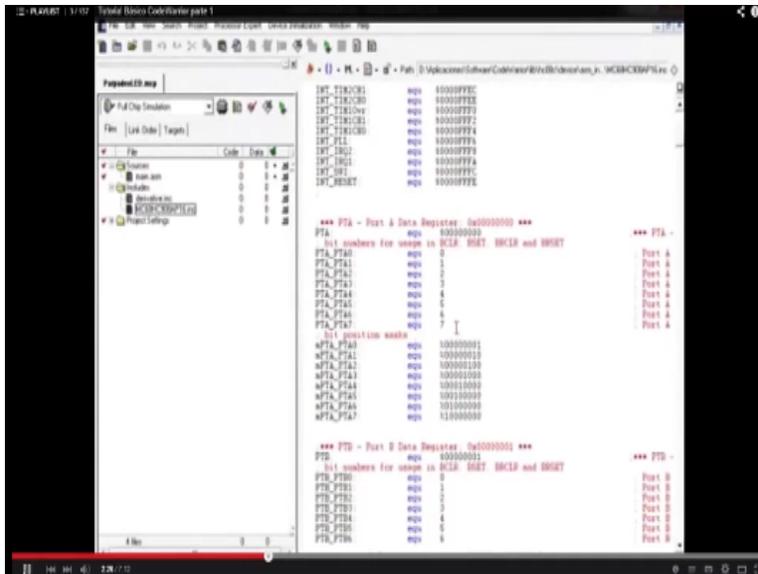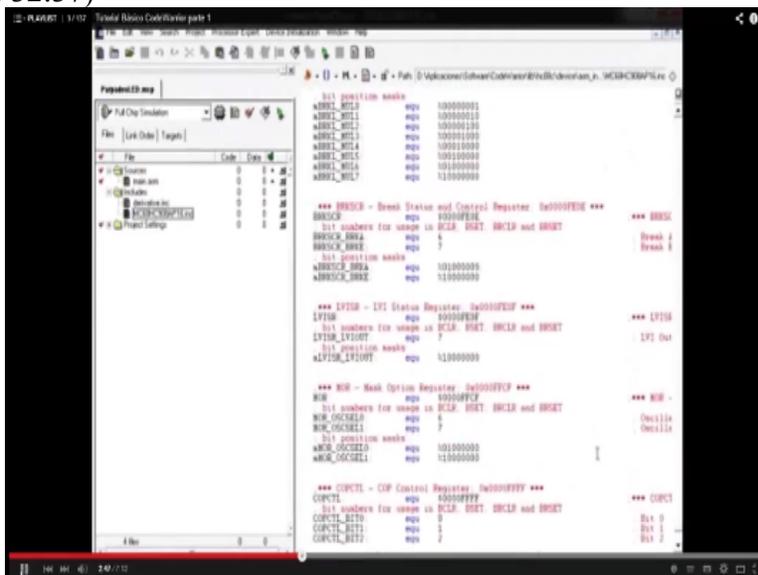(Refer Slide Time: 32:05)

Interpreters
(Refer Slide Time: 32:07)

And all that are defined here in this library. We can use any of these variables of sub advantage.
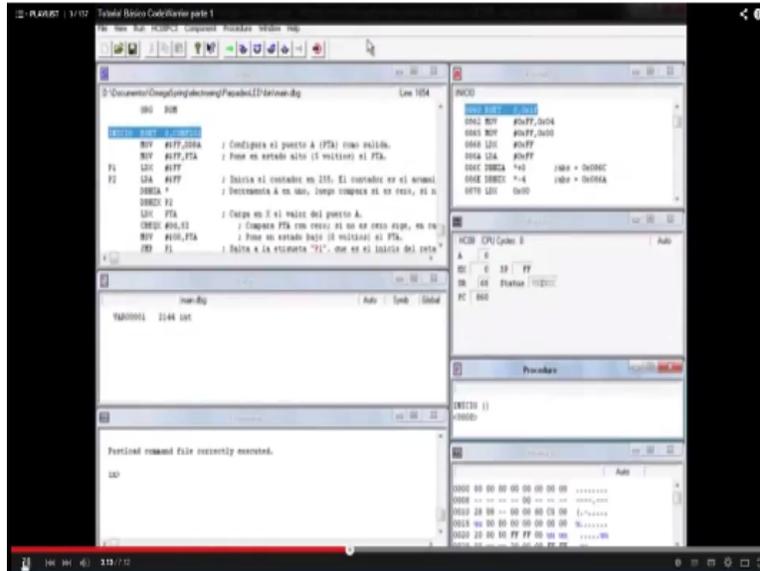(Refer Slide Time: 32:14)

Going to be used
(Refer Slide Time: 32:37)


Okay.
(Refer Slide Time: 32:41)

So main.O we can see, main.O being used.
(Refer Slide Time: 32:54)

And main.O includes other library included as usual we have in a C similar to that.
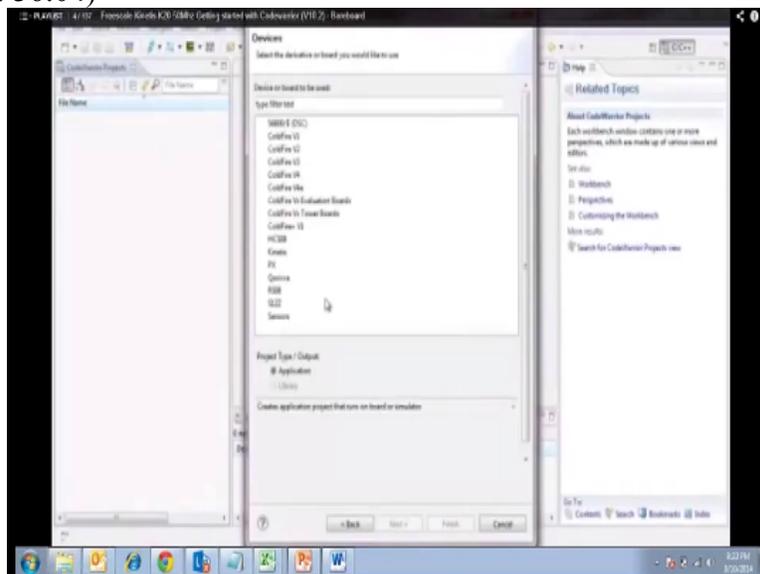(Refer Slide Time: 33:05)

Okay, you can see the assembly and disassembly on the conserved you can see the actual source and you can see that CPU cycles how much is being used for pointer, accumulator, strategy status of the particular micro printer in described a indicate serial memory that is being used which memory address 16bit that specific memory addresses with that MSB first and LSB last being displayed here.

And, we can have a continuation of that single step you want to have, you can have step out, step in, Ault and final execution of the target we can do. We can do the reset here single step we can do, we can see step by step lines have been executed. We can set the brake point modify any of these while use or any testing that you want to have and done it for assembly project to be used in the code warier.

We can see the break points if you want to set for a particular line we can set it and watch the accumulator what when it is coming especially this assembly debugging it is still when you want to have a boot software ended testing related things were different registers want to verify, with the right click option you can set the break point at watch window you can see that run try cursor (Refer Slide Time: 35:05)
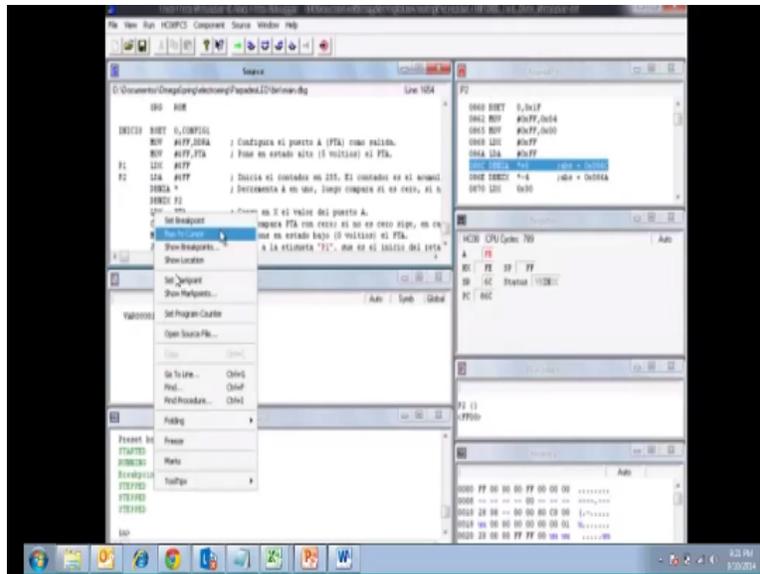
We can have that with that value modified and see behavior of the CPPU what is going to describe, load, store, select, and start of the instructions will be there and, start a pointer reference that is loaded in the memory.

So let us try to understand next type of a project how that is being debugged and used to k20 family project is being the cursor family is being used particularly km, k20 bits and 20 bit, so it is depends on the development byte whatever you have that particular cursor in usually selected.

(Refer Slide Time: 36:04)

We can see the several options like we assign last time.
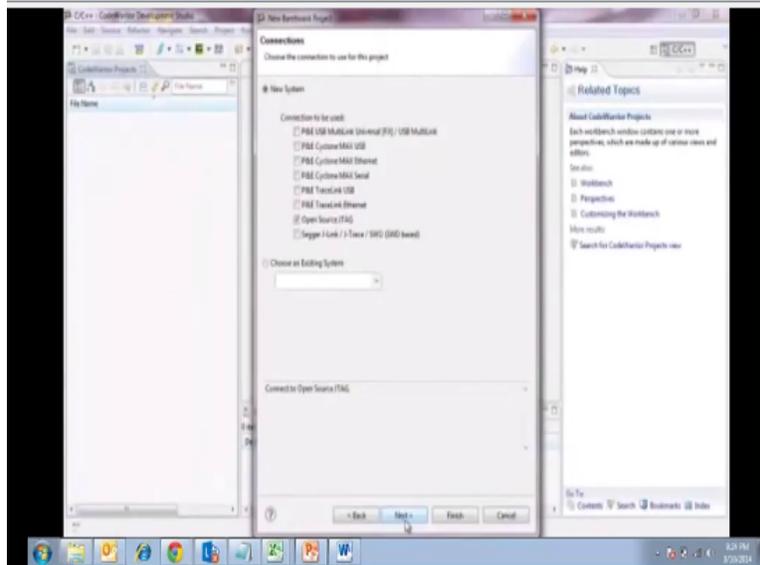
(Refer Slide Time: 36:12)
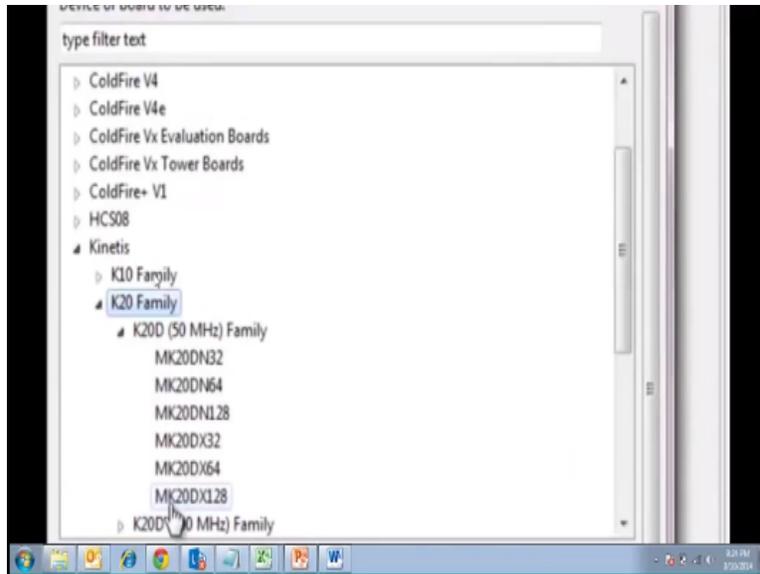
The PADSP multiline Kinetis law connection is being used.
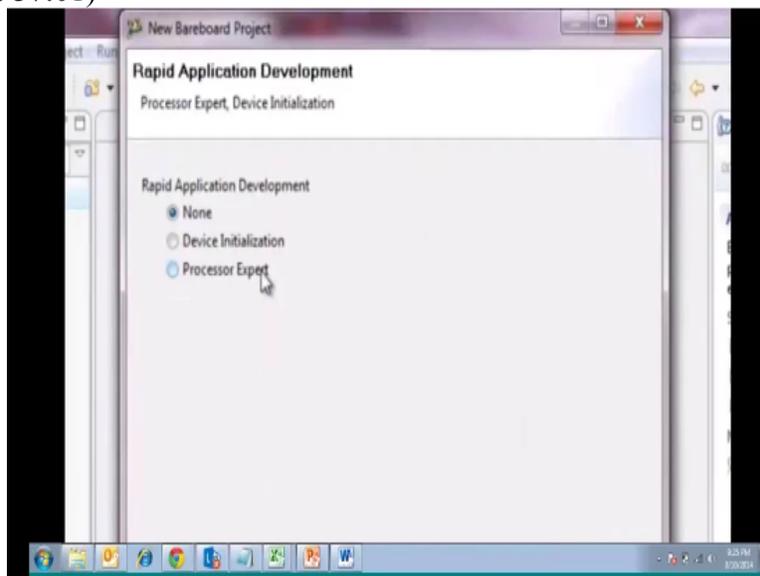(Refer Slide Time: 36:22)
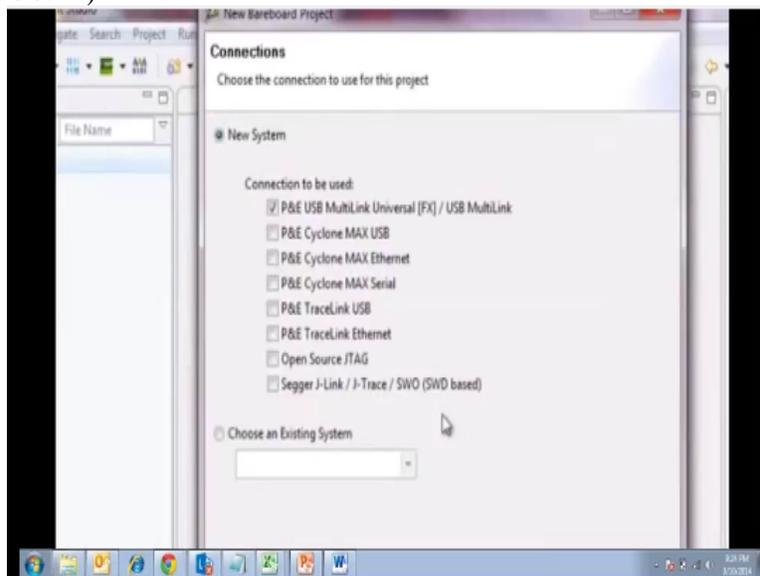
If you want to use the onward data audios open source we can use here it is being used, so you

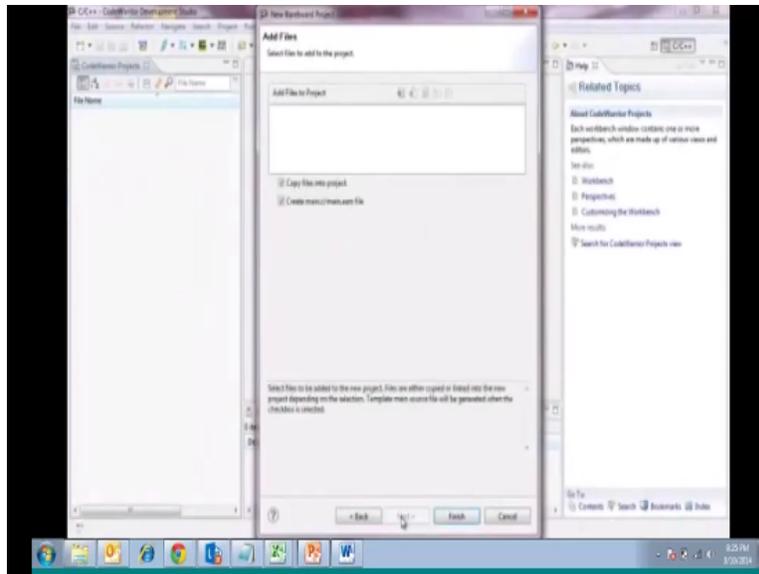can select any of these languages.
(Refer Slide Time: 36:56)
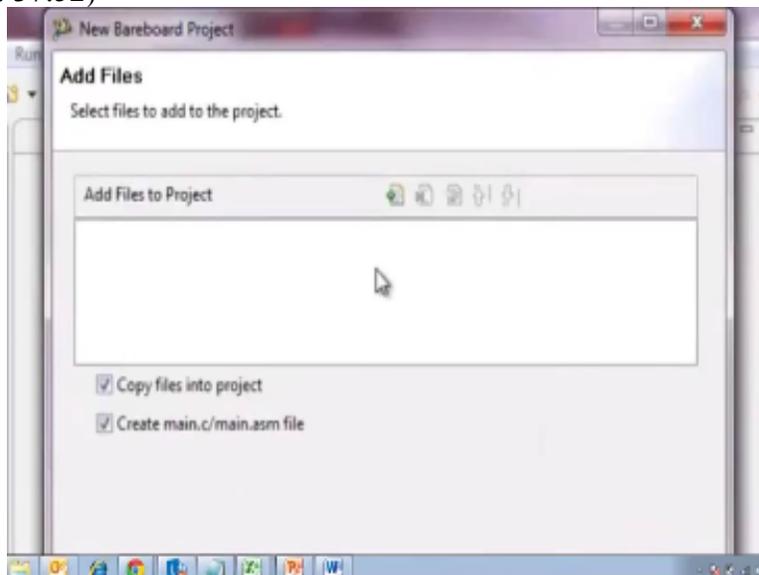
(Refer Slide Time: 37:01)

Here in this project device initialization being used for add in a file to a project you can search and describe option.
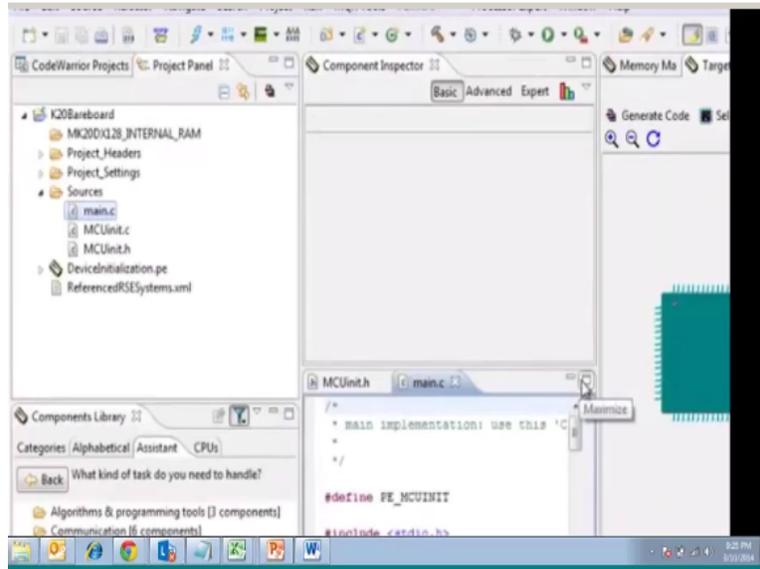
(Refer Slide Time: 37:15)

You want to copy the plus three project any use option you can do create main.C, this environment being a created here if you target CPU project K20.
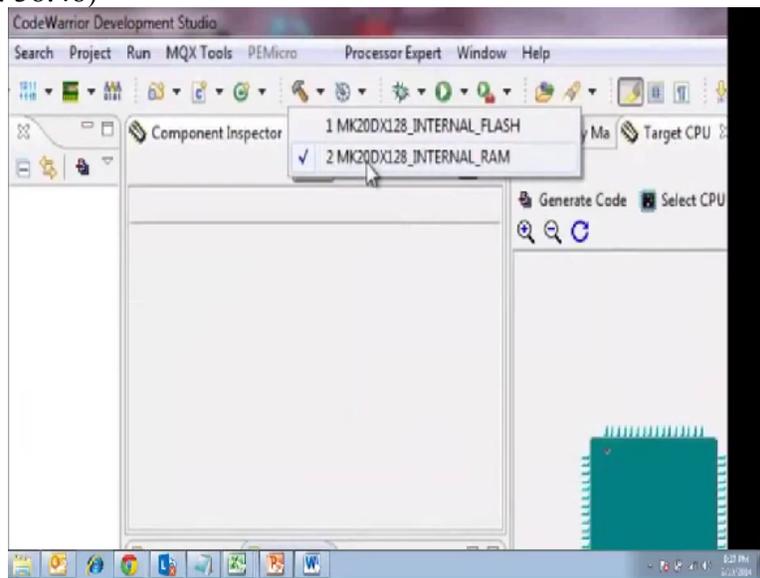(Refer Slide Time: 37:52)

Sources we can see this may couple of 3 files feeder is the insulation code generate automatically Main. C we need .C you generate by tool itself so you should not modify this basically Main.C.
(Refer Slide Time: 38:21)

These libraries yesterday were used have this is simple counter.
(Refer Slide Time: 38:31)

Similarly what we have seen scenario being used here.
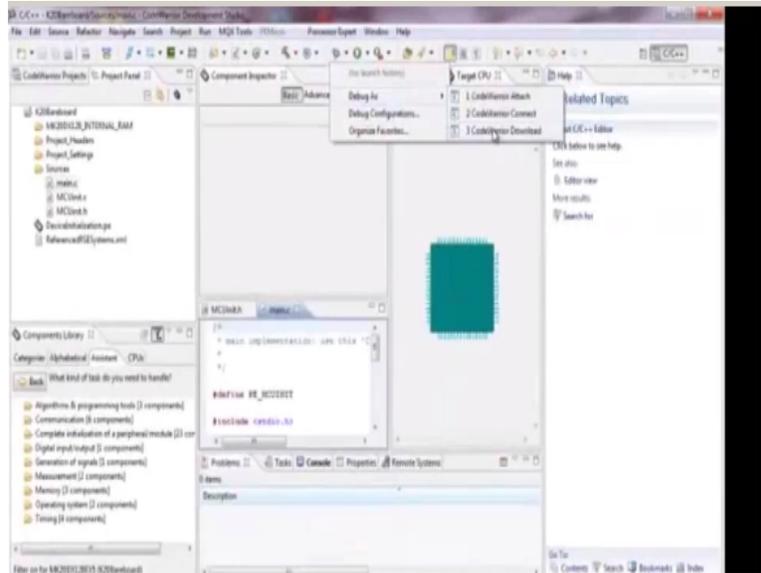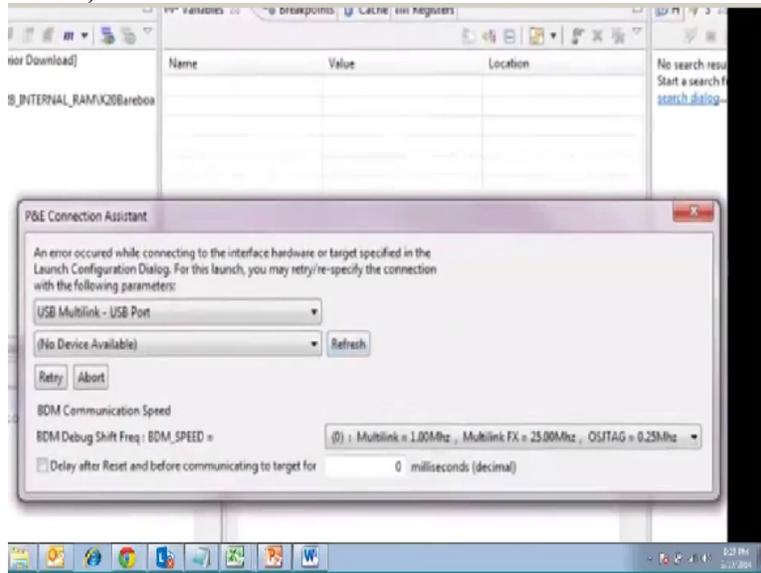(Refer Slide Time: 38:38)

It can built a with this amour option dropped down internal flash model in order to RAM.
(Refer Slide Time: 38:48)

You know the difference flash is different misterger RAM is a one that is going power up we said this sectionally RAM here and the debug here it is option code warrior connect or download selected the option and debugged.
(Refer Slide Time: 39:18)

It will switch to the debugged perspective here C multiple windows what we seen easier in terms

of the traces and source.
(Refer Slide Time: 39:38)

So the target that is being plugged in here, it is right to start here we say break point we cut the

counter the break point used here ,is being run of.
(Refer Slide Time: 40:09)
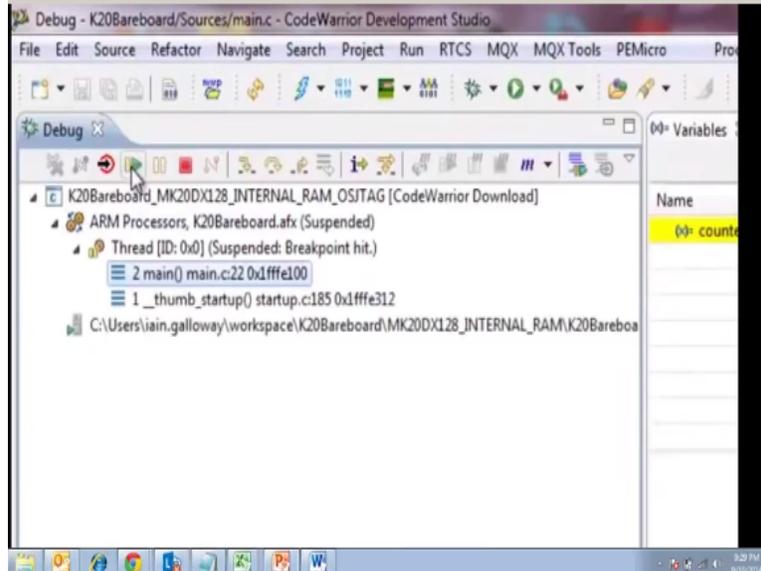
There are stepping commands like step in, step up. Step return of the option you can is a break point the project has stopped here control value you can seat here again you RAM 8 here seated at control value because control C is getting call by code warrior is the code warrior control version that is being used.

Okay this how we can use debugger loading the project we using project adding the files cutting the break point making the changes into the counter of the variables, watched in the memory likewise we can have it.

And we can skill it out for a complex project this way okay, we will continue in the next session we touch up more.

(Refer Slide Time: 41:38)
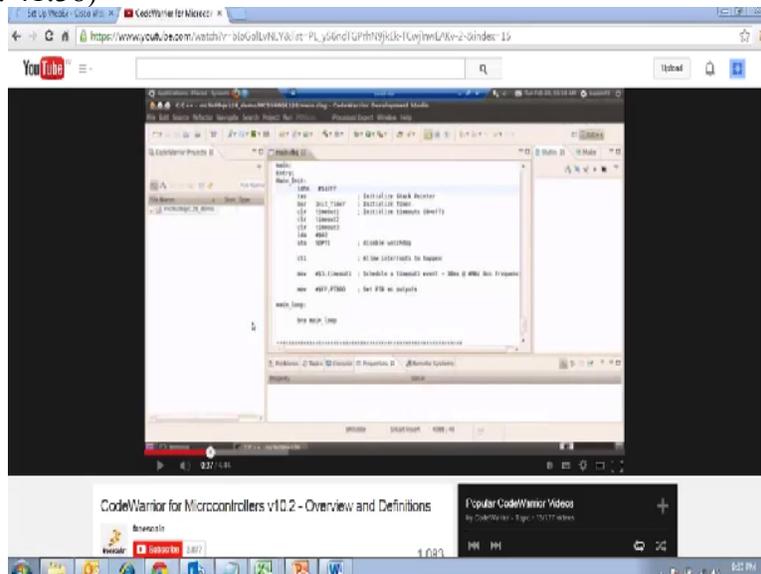
Exercises and till then see you, thank you bye, and bye.