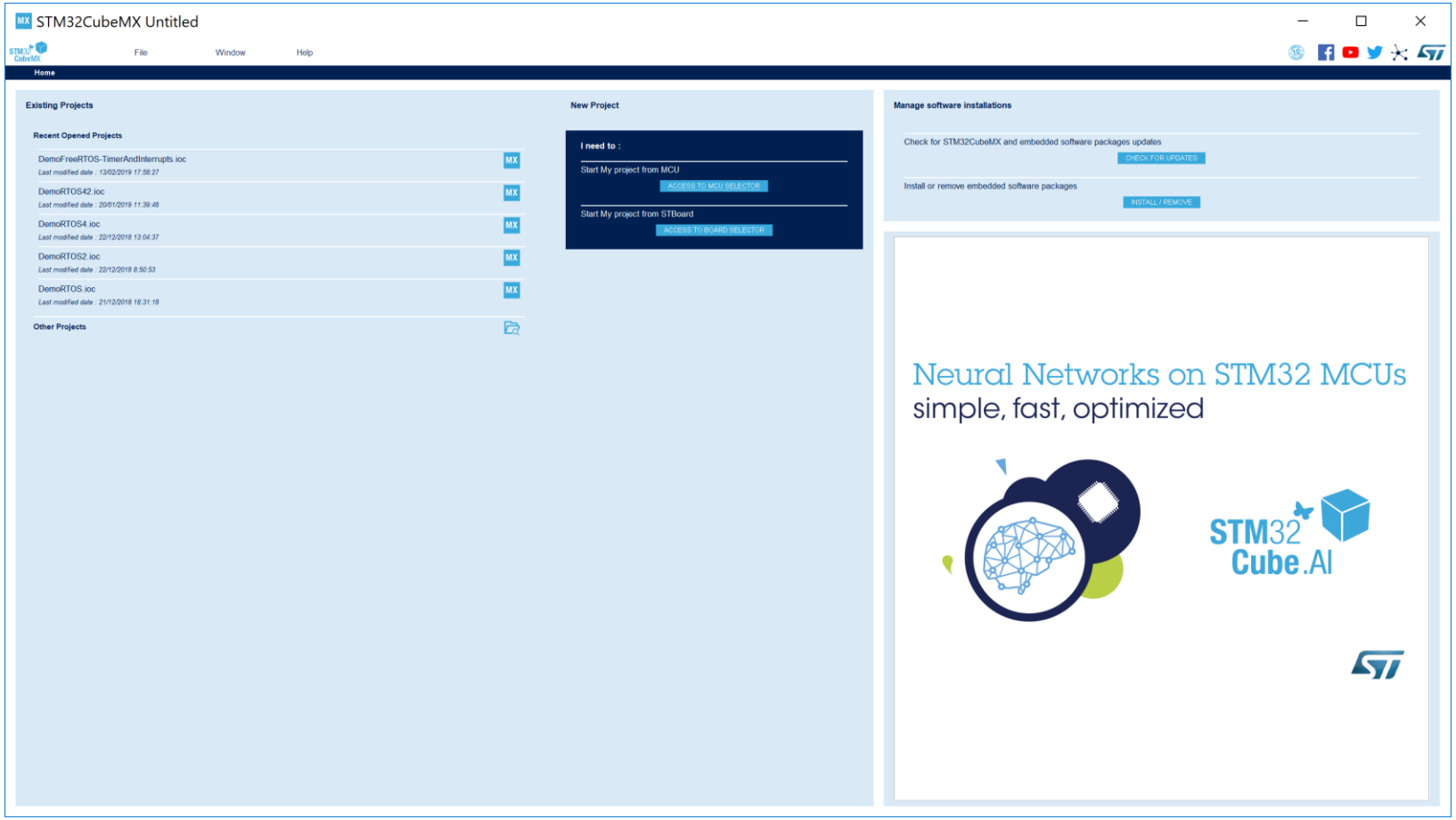
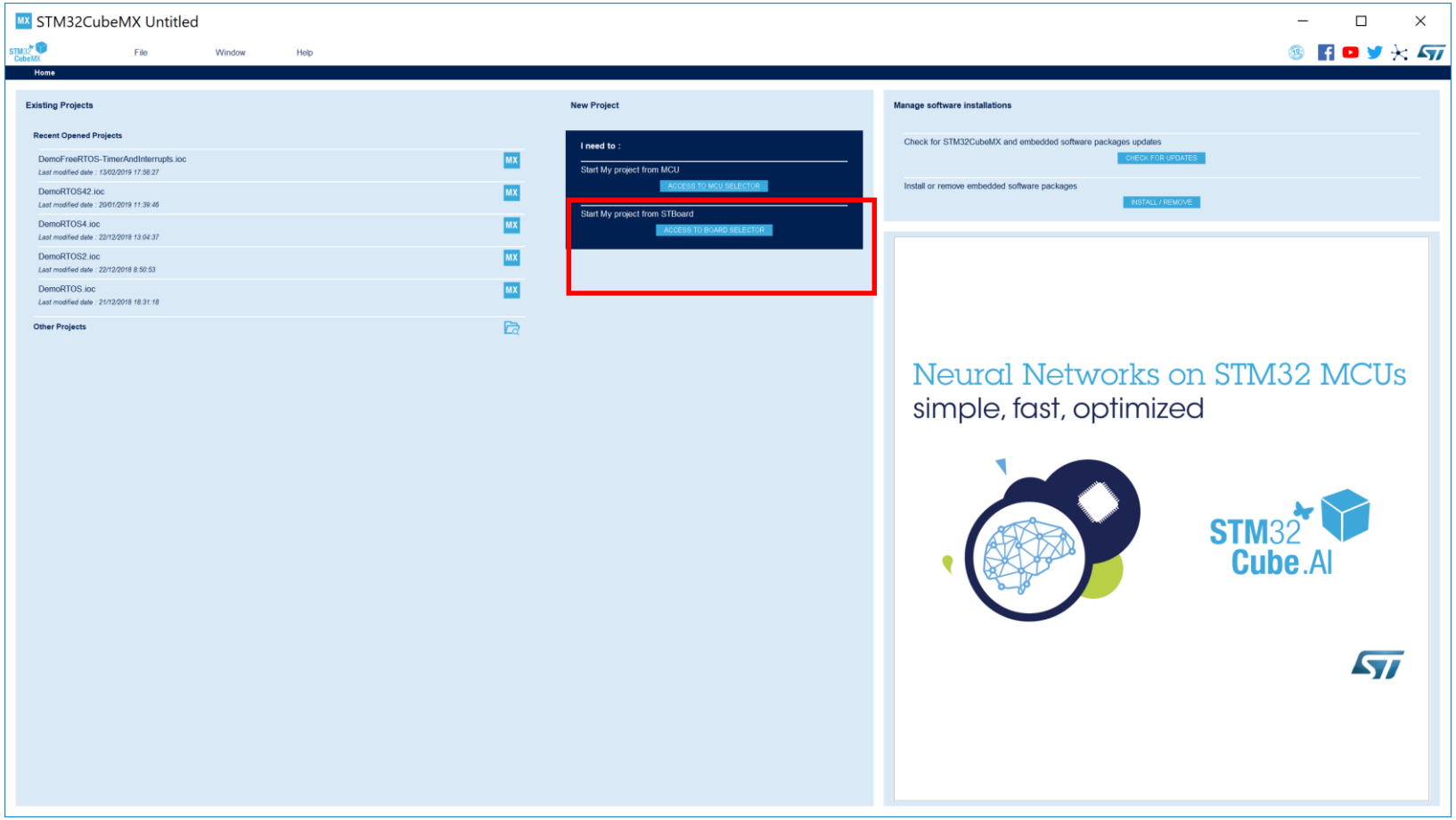
USCD Embedded C Assignment 6

By
Norman McEntire
Norman.mcentire@gmail.com

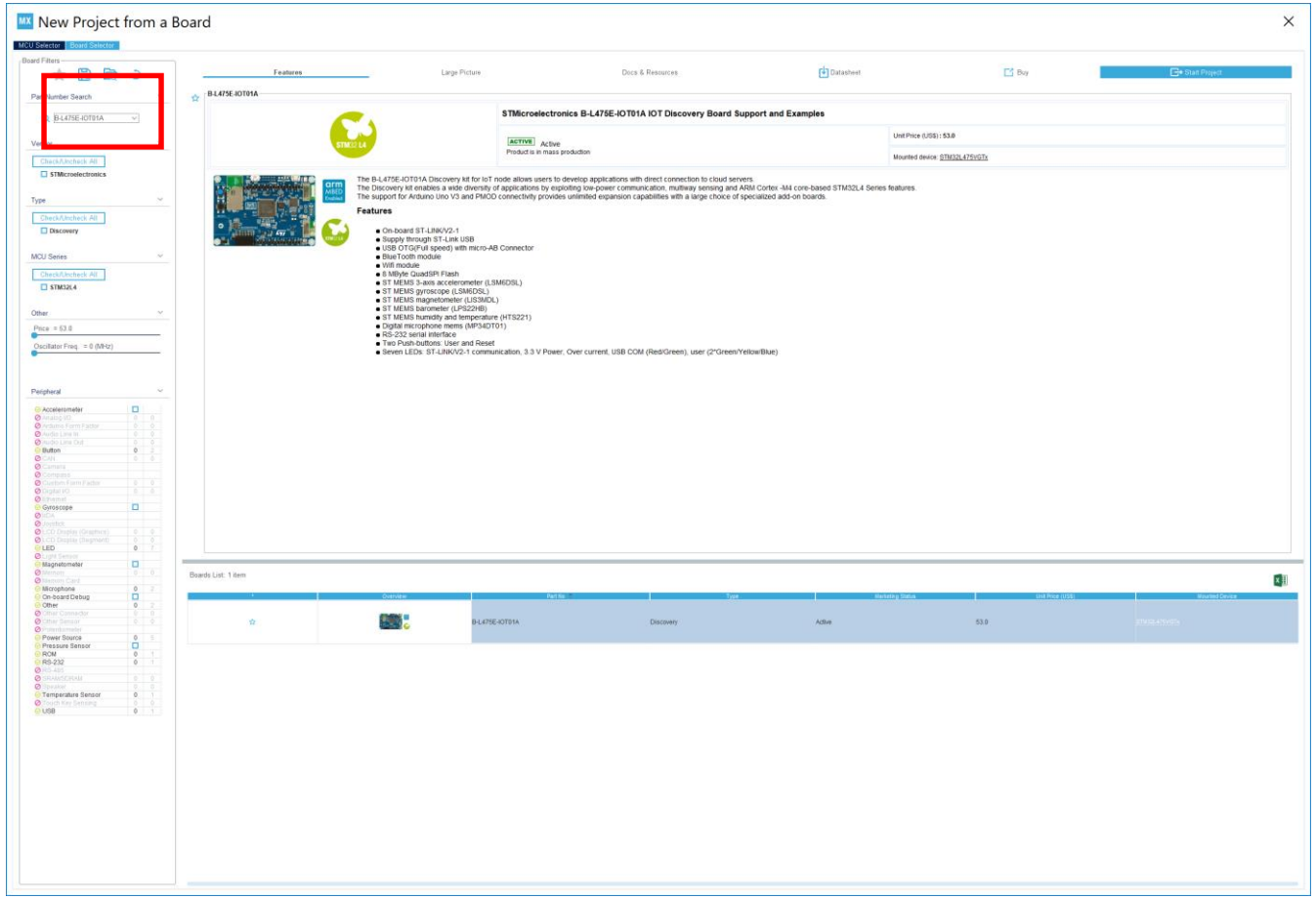
Step 1. Startup STM32CubeMX



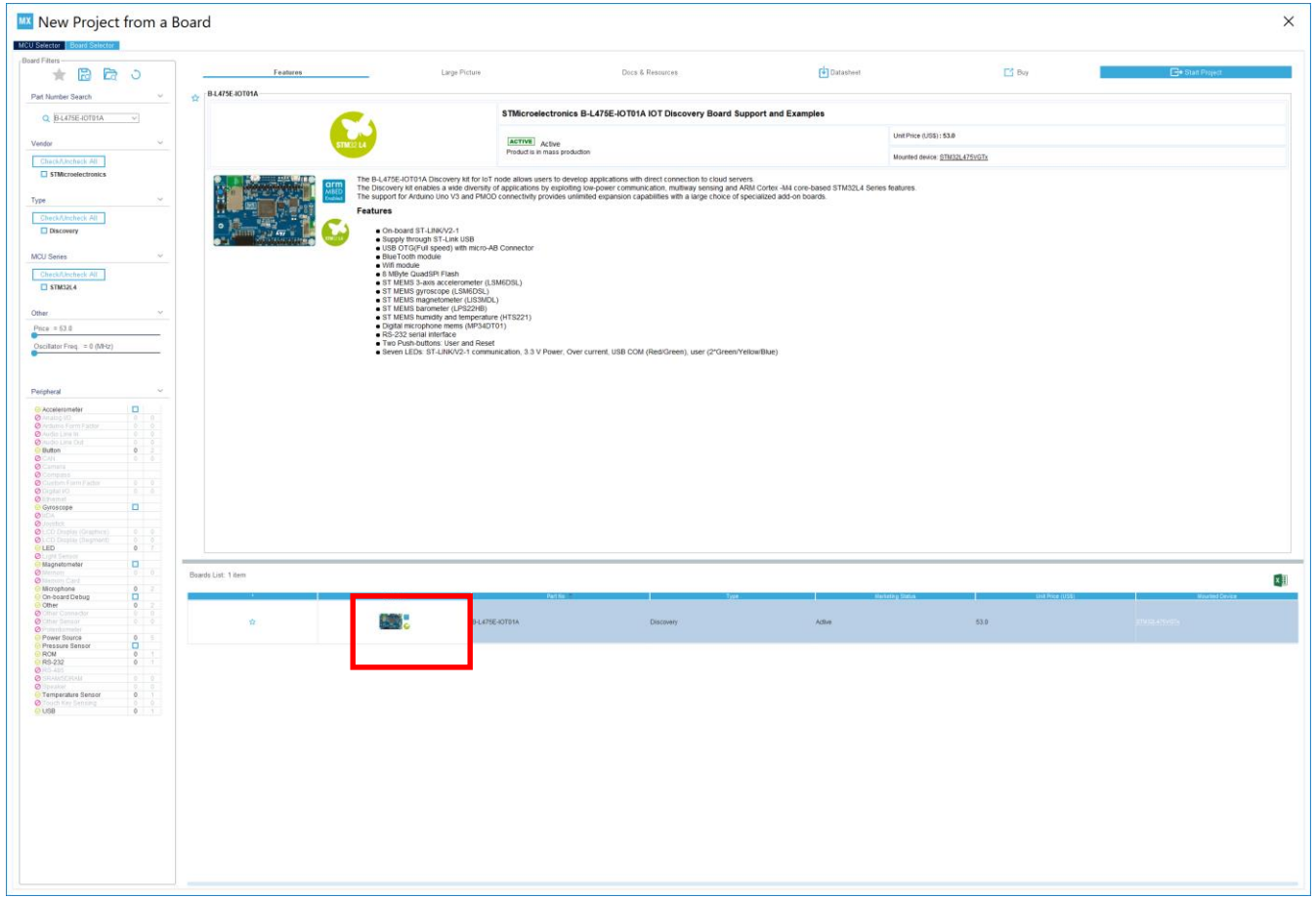
Step 2. Access Board Selector



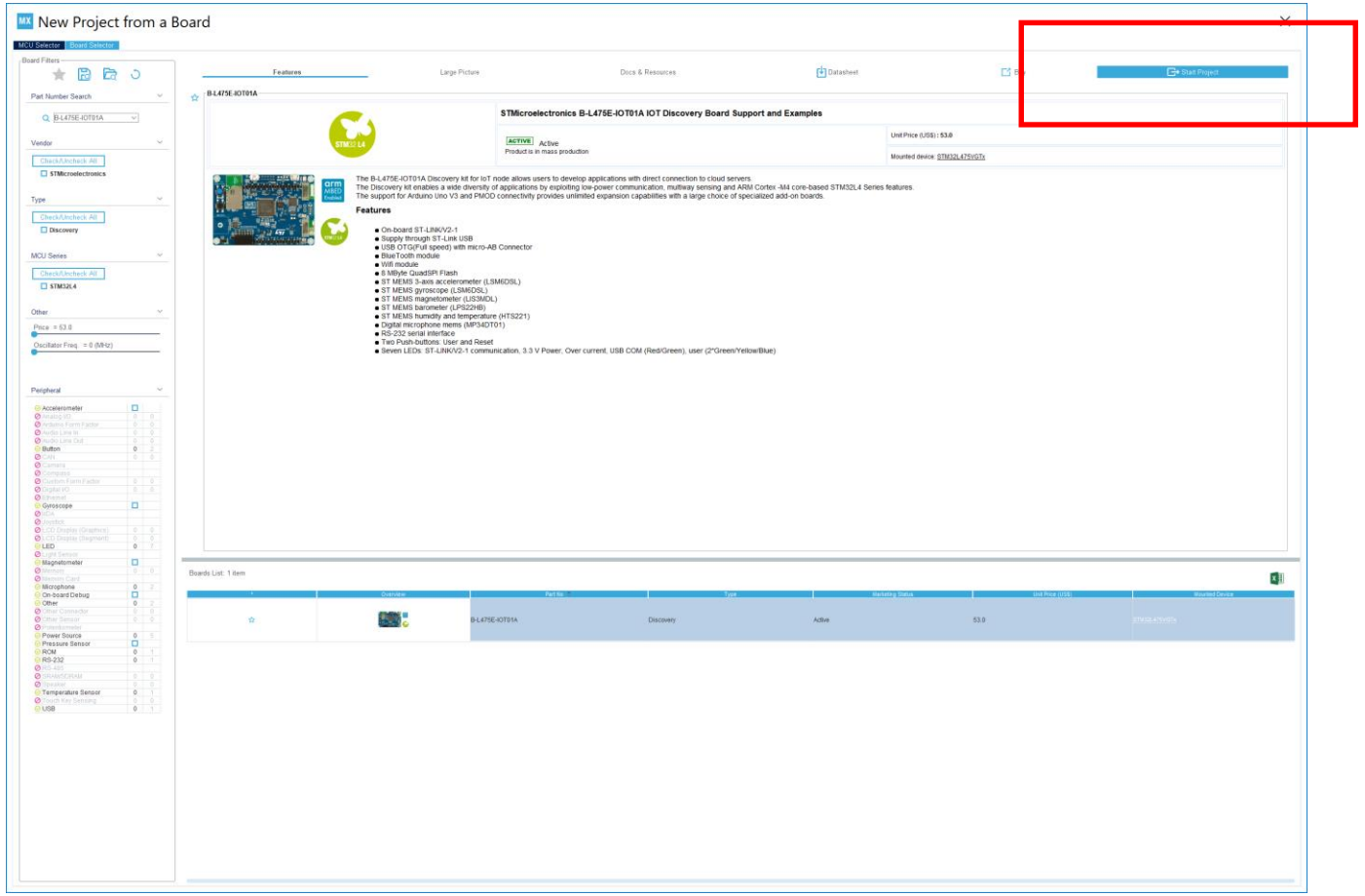
Step 3. Enter “B-L475E-IOT01A” Board



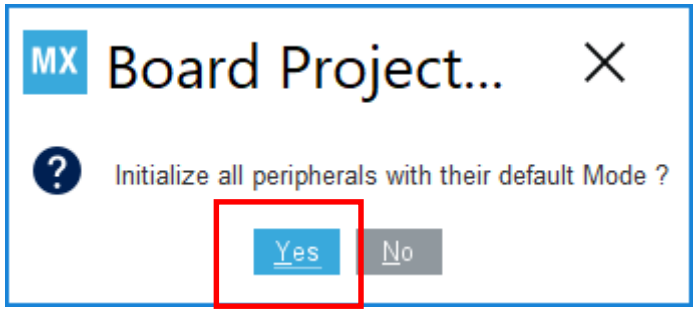
Step 4. Select Board Photo



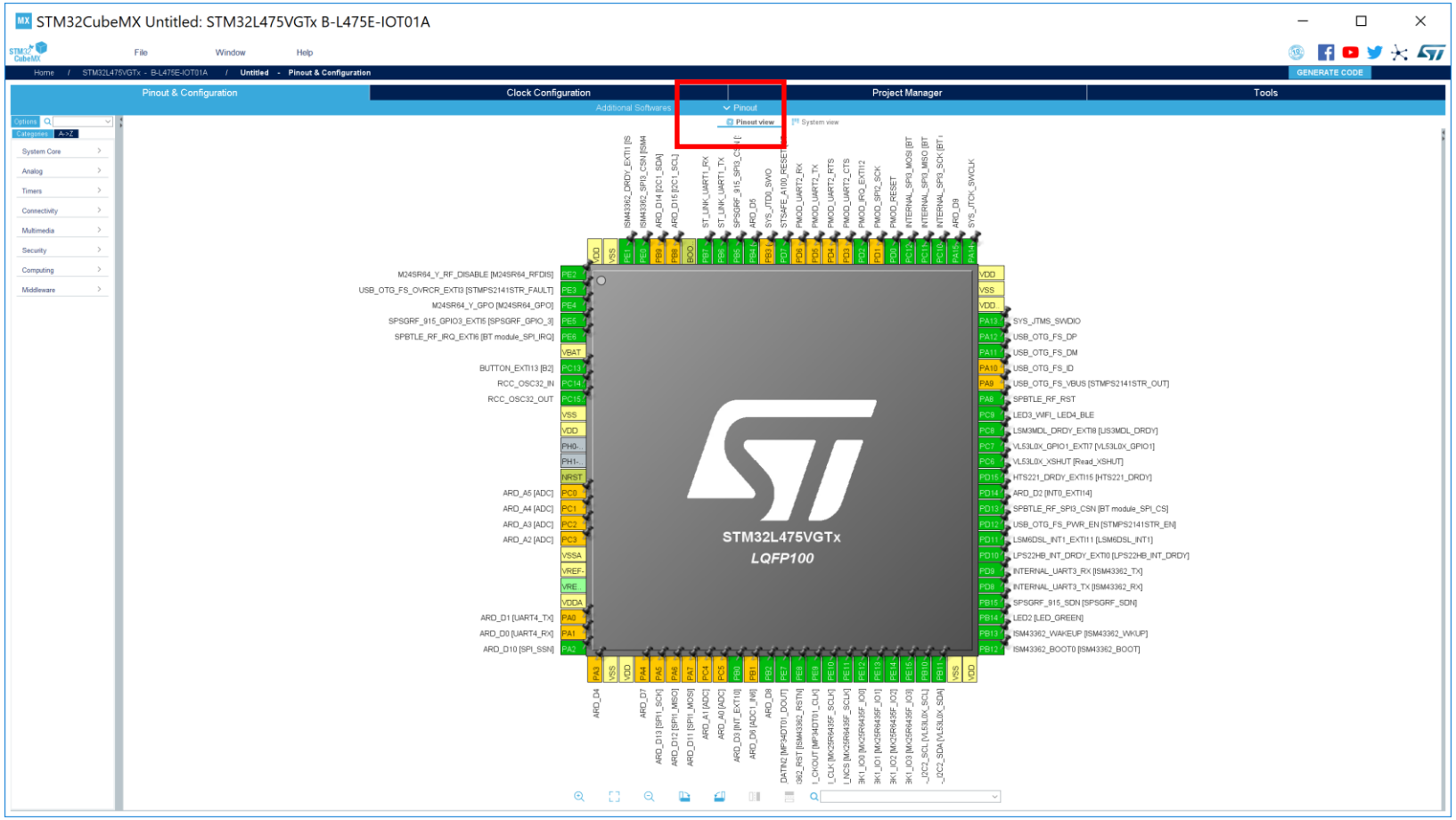
Step 5. Select “Start Project”



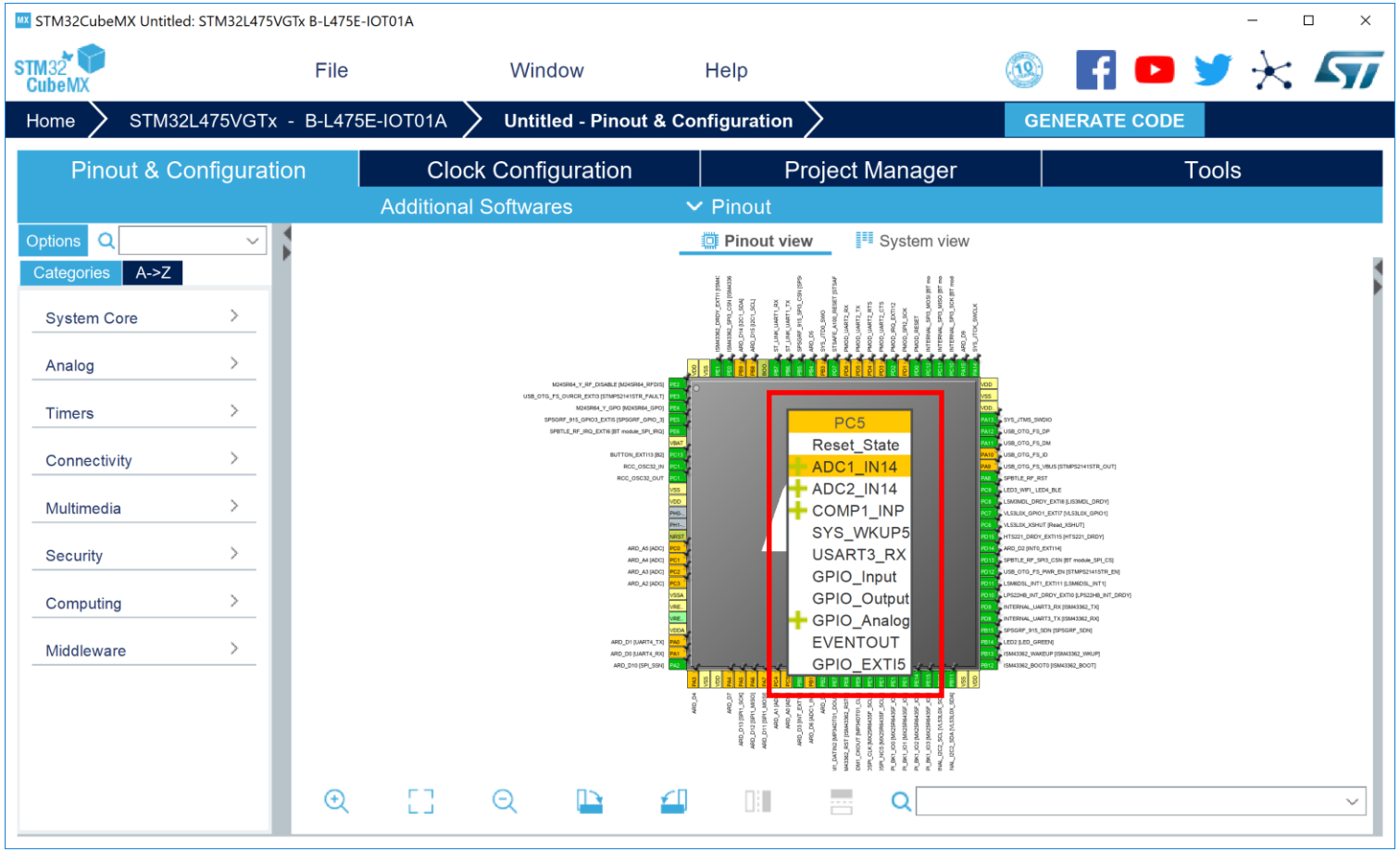
Step 6. Select **YES** (initialize all peripherals with the default mode)



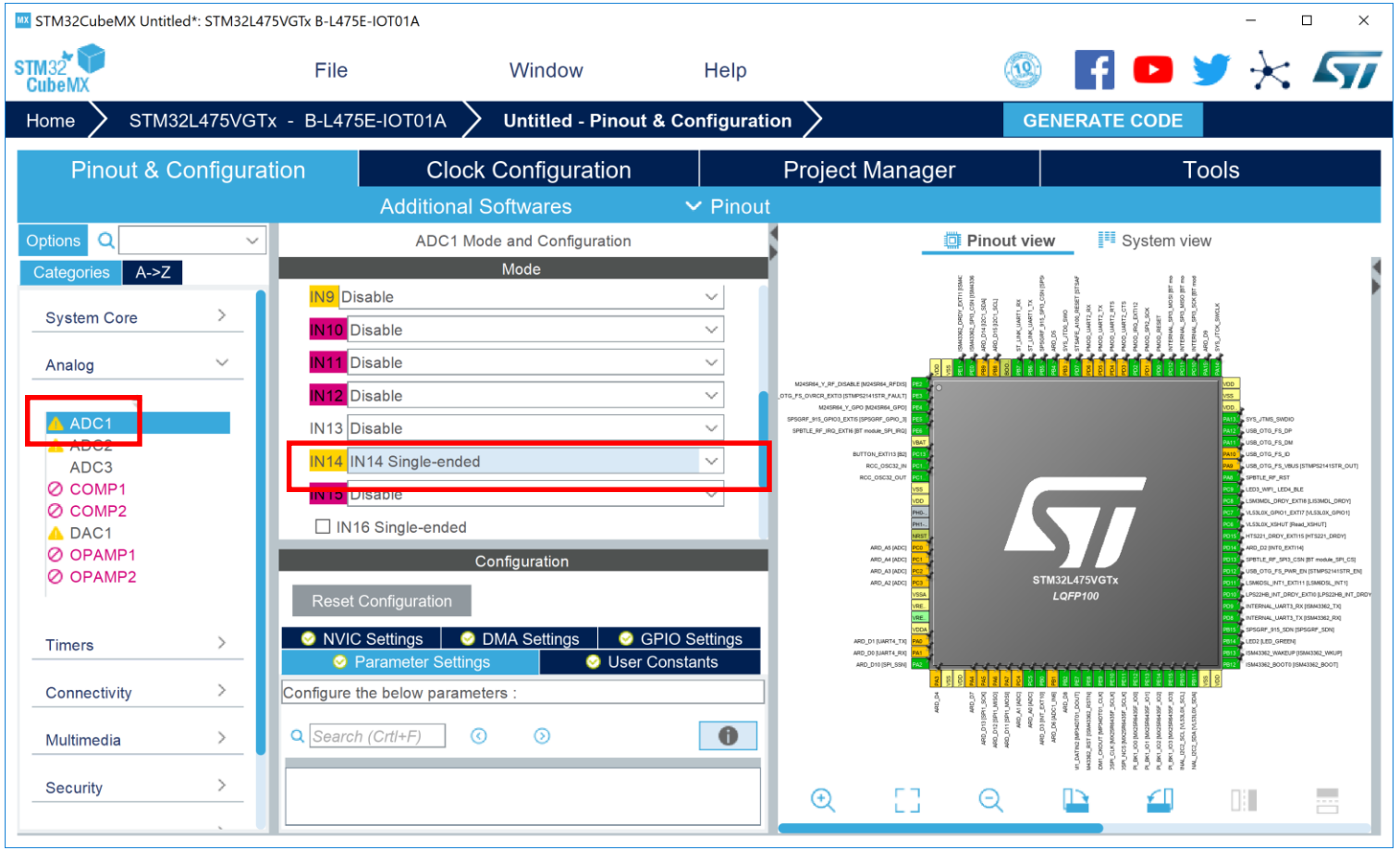
Step 7. Observe Results (Pinout View)



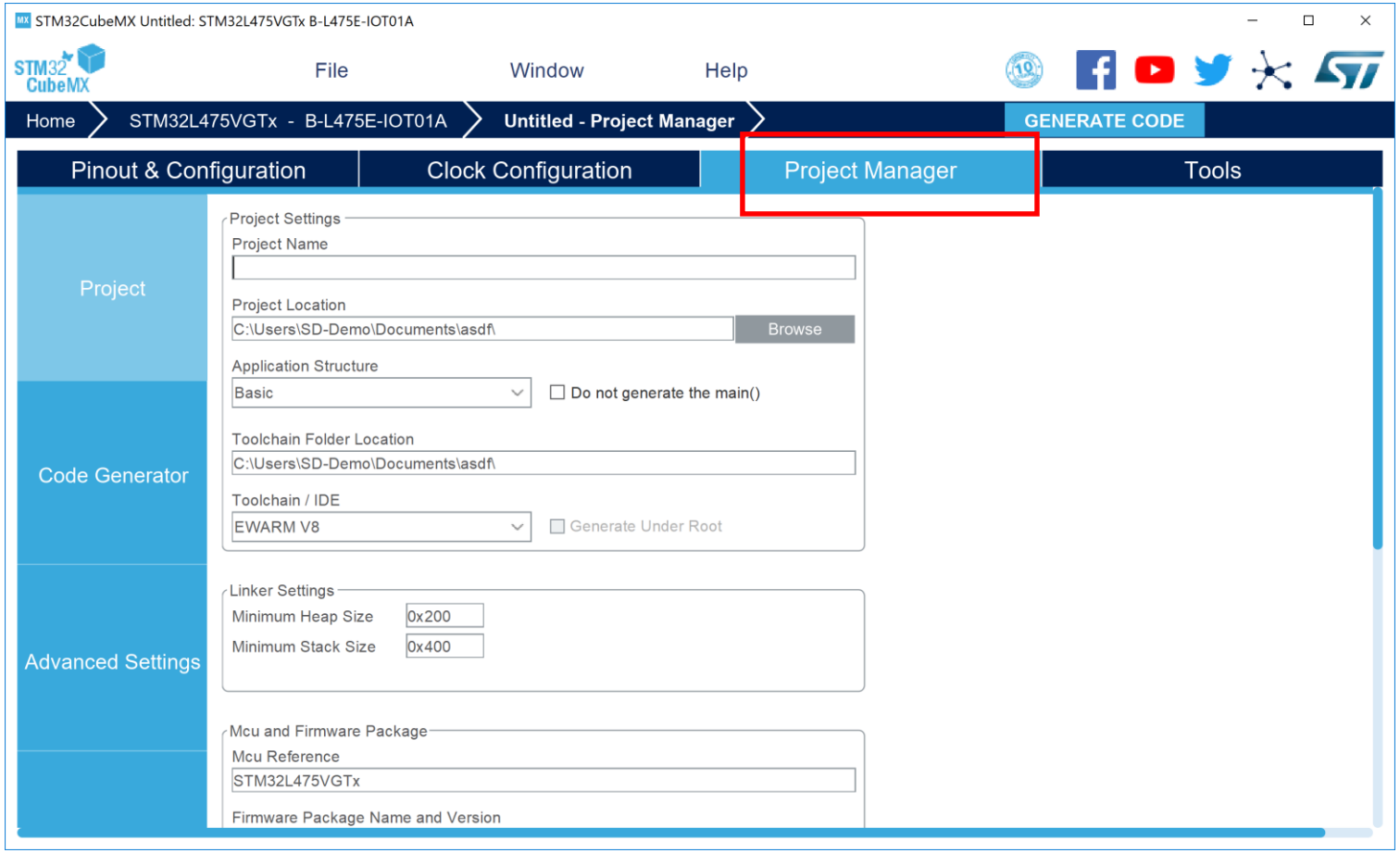
Step 8. Observe PC5 – ADC1_IN14 – ARD_A0



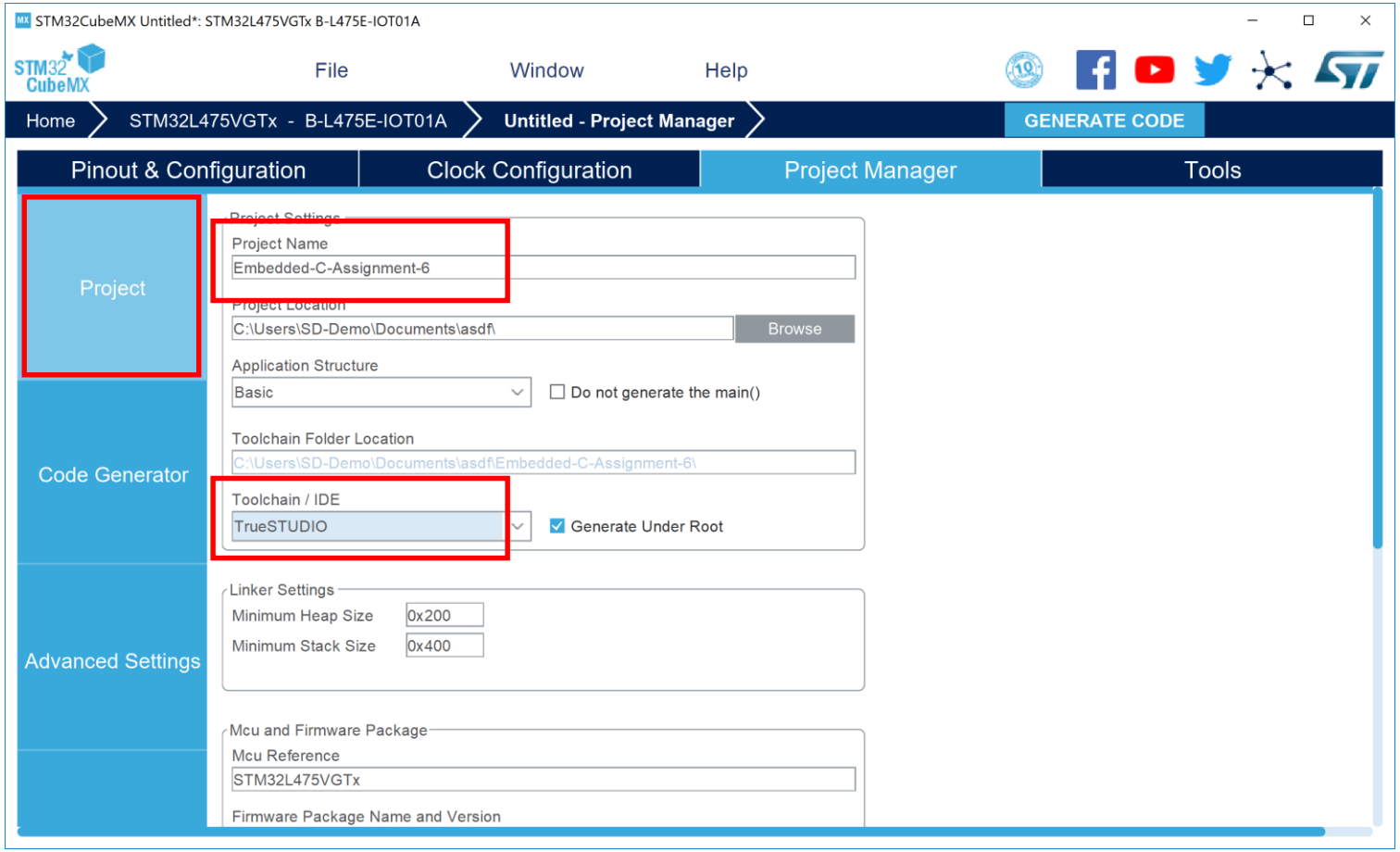
Step 9. Set ADC1, IN14 to Single-ended



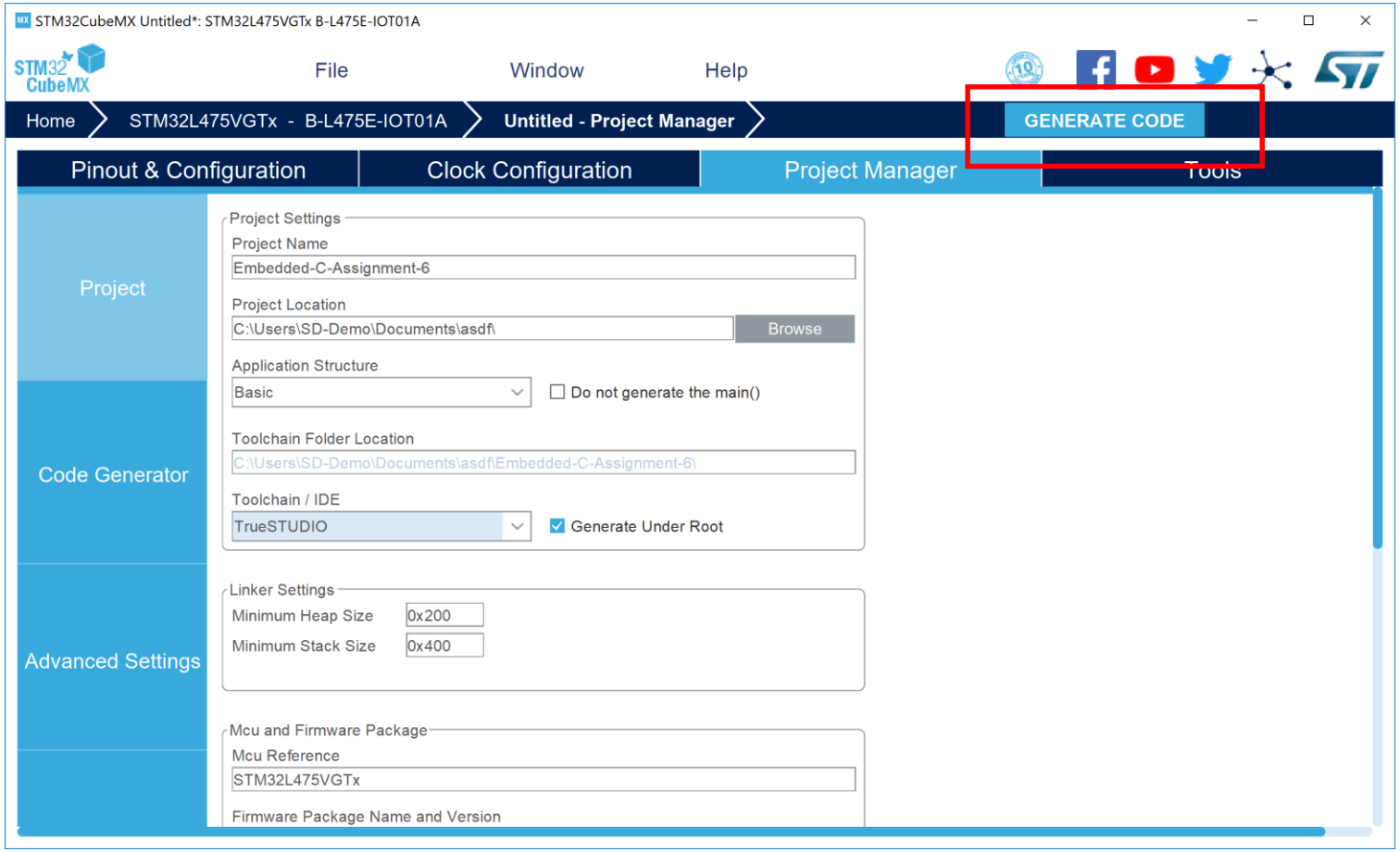
Step 10. Select Project Manager Tab



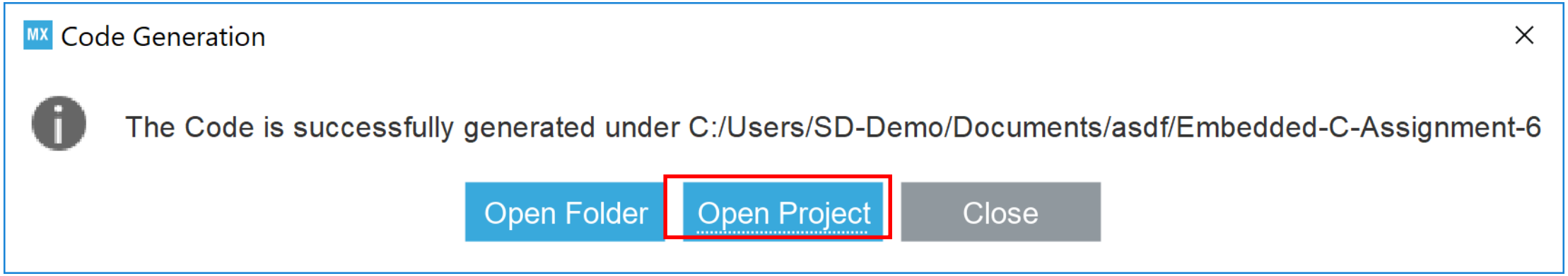
Step 11. Enter “Embedded-C-Assigment-6” and select TrueStudio as IDE



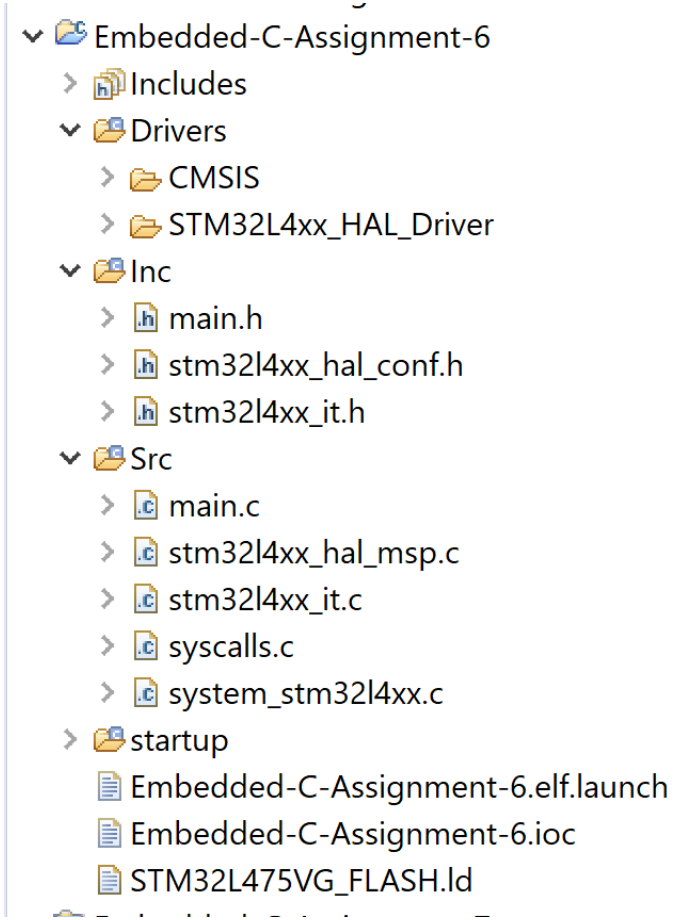
Step 12. Select “Generate Code”



Step 13. Select “Open Project”



Step 14. Resulting Project



Step 15. In main.c, find code that initializes the ADC1

```
216 static void MX_ADC1_Init(void)
217 {
218
219   /* USER CODE BEGIN ADC1_Init 0 */
220
221   /* USER CODE END ADC1_Init 0 */
222
223   ADC_MultiModeTypeDef multimode = {0};
224   ADC_ChannelConfTypeDef sConfig = {0};
225
226   /* USER CODE BEGIN ADC1_Init 1 */
227
228   /* USER CODE END ADC1_Init 1 */
229   /** Common config
230   */
231   hadc1.Instance = ADC1;
232   hadc1.Init.ClockPrescaler = ADC_CLOCK_ASYNC_DIV1;
233   hadc1.Init.Resolution = ADC_RESOLUTION_12B;
234   hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
235   hadc1.Init.ScanConvMode = ADC_SCAN_DISABLE;
236   hadc1.Init.EOCSelection = ADC_EOC_SINGLE_CONV;
237   hadc1.Init.LowPowerAutoWait = DISABLE;
238   hadc1.Init.ContinuousConvMode = DISABLE;
239   hadc1.Init.NbrOfConversion = 1;
240   hadc1.Init.DiscontinuousConvMode = DISABLE;
241   hadc1.Init.ExternalTrigConv = ADC_SOFTWARE_START;
242   hadc1.Init.ExternalTrigConvEdge = ADC_EXTERNALTRIGCONVEDGE_NONE;
243   hadc1.Init.DMAContinuousRequests = DISABLE;
244   hadc1.Init.Overrun = ADC_OVR_DATA_PRESERVED;
245   hadc1.Init.OversamplingMode = DISABLE;
246   if (HAL_ADC_Init(&hadc1) != HAL_OK)
247   {
248     Error_Handler();
249   }
250   /** Configure the ADC multi mode
```

Step 16. In main.c, find code that initializes the Channel IN14

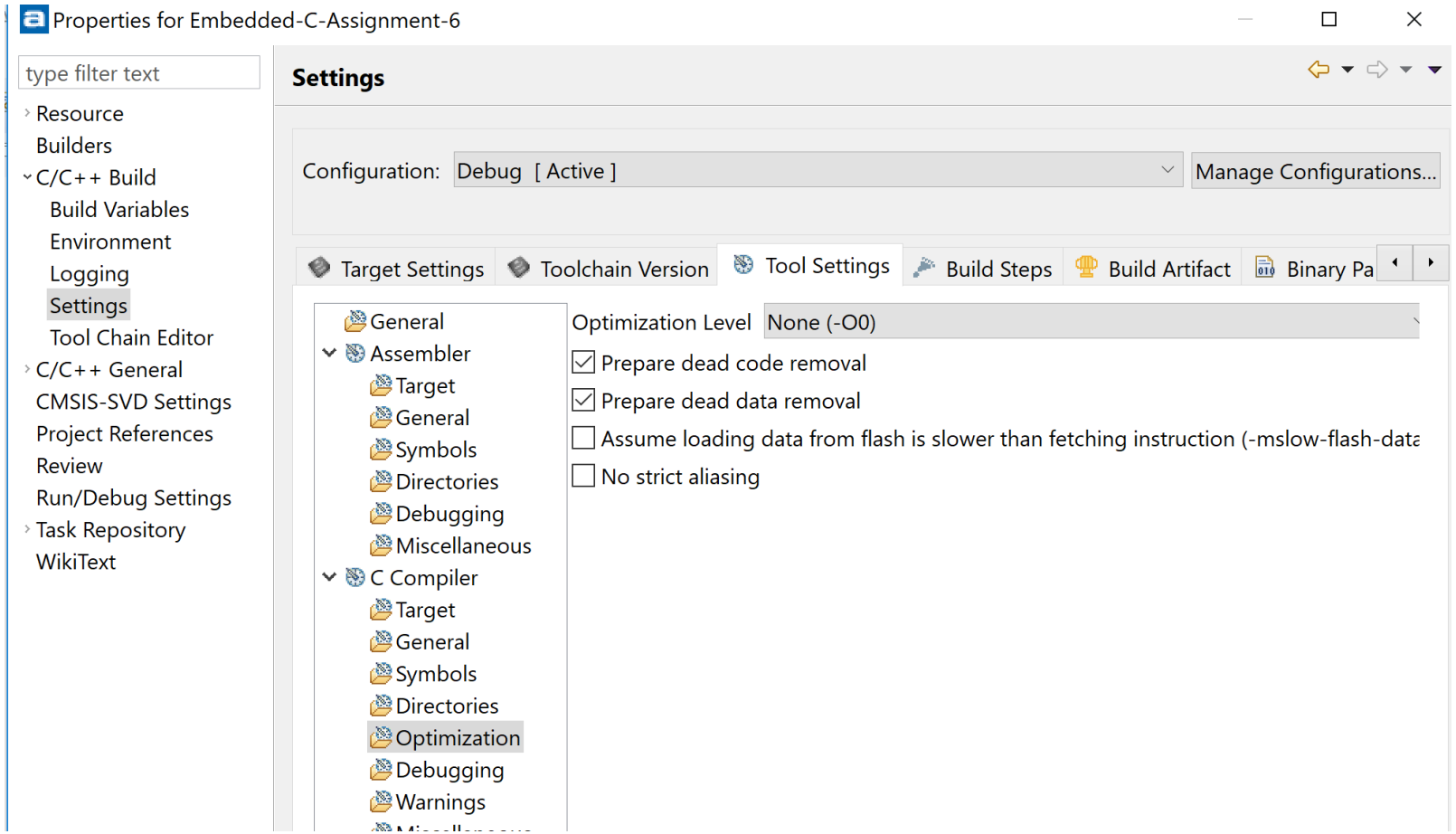
```
257   /** Configure Regular Channel
258   */
259   sConfig.Channel = ADC_CHANNEL_14;
260   sConfig.Rank = ADC_REGULAR_RANK_1;
261   sConfig.SamplingTime = ADC_SAMPLETIME_2CYCLES_5;
262   sConfig.SingleDiff = ADC_SINGLE_ENDED;
263   sConfig.OffsetNumber = ADC_OFFSET_NONE;
264   sConfig.Offset = 0;
265   if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
266   {
267     Error_Handler();
268   }
269   /* USER CODE BEGIN ADC1_Init 2 */
270
271   /* USER CODE END ADC1_Init 2 */
272
273 }
```

Step 17. Add code to main.c to read from ARD_A0

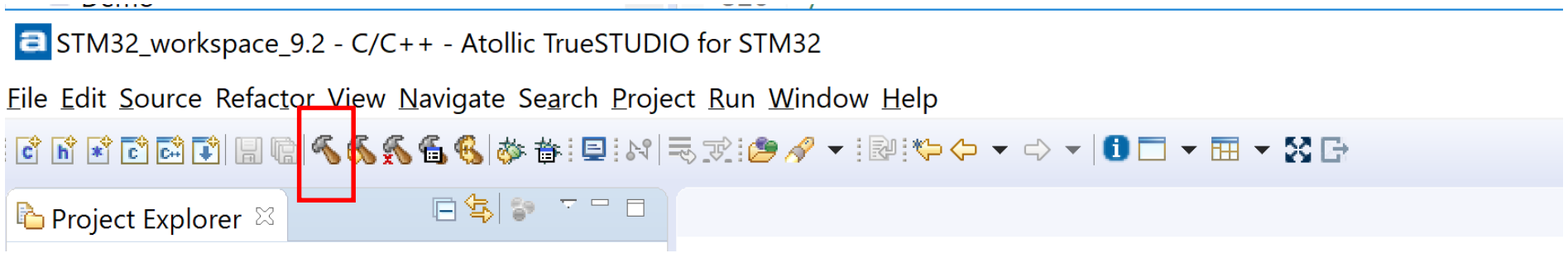
main.c

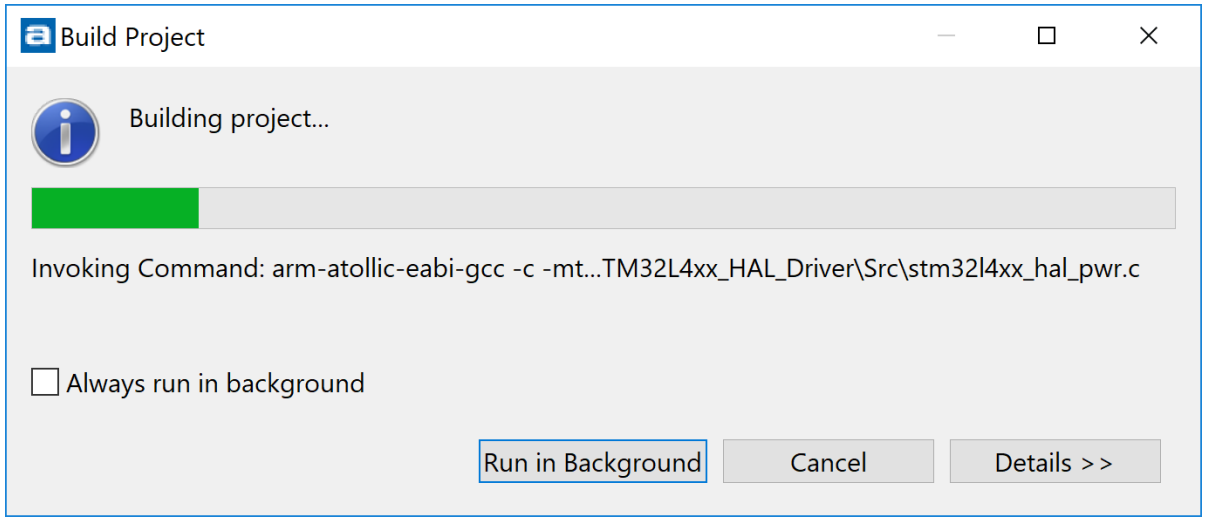
```
127   /* USER CODE BEGIN WHILE */
128   uint32_t adcResult;
129
130   while (1)
131   {
132       HAL_ADC_Start(&hadc1);
133
134       HAL_ADC_PollForConversion(&hadc1, 100);
135       adcResult = HAL_ADC_GetValue(&hadc1);
136       printf("adcResult: %lu - 0x%lx\n", adcResult, adcResult);
137
138       HAL_ADC_Stop(&hadc1);
139
140       HAL_Delay(1000);
141
142     /* USER CODE END WHILE */
143
144
145     /* USER CODE BEGIN 3 */
146   }
147   /* USER CODE END 3 */
148 }
```

Step 18. Set optimization to none

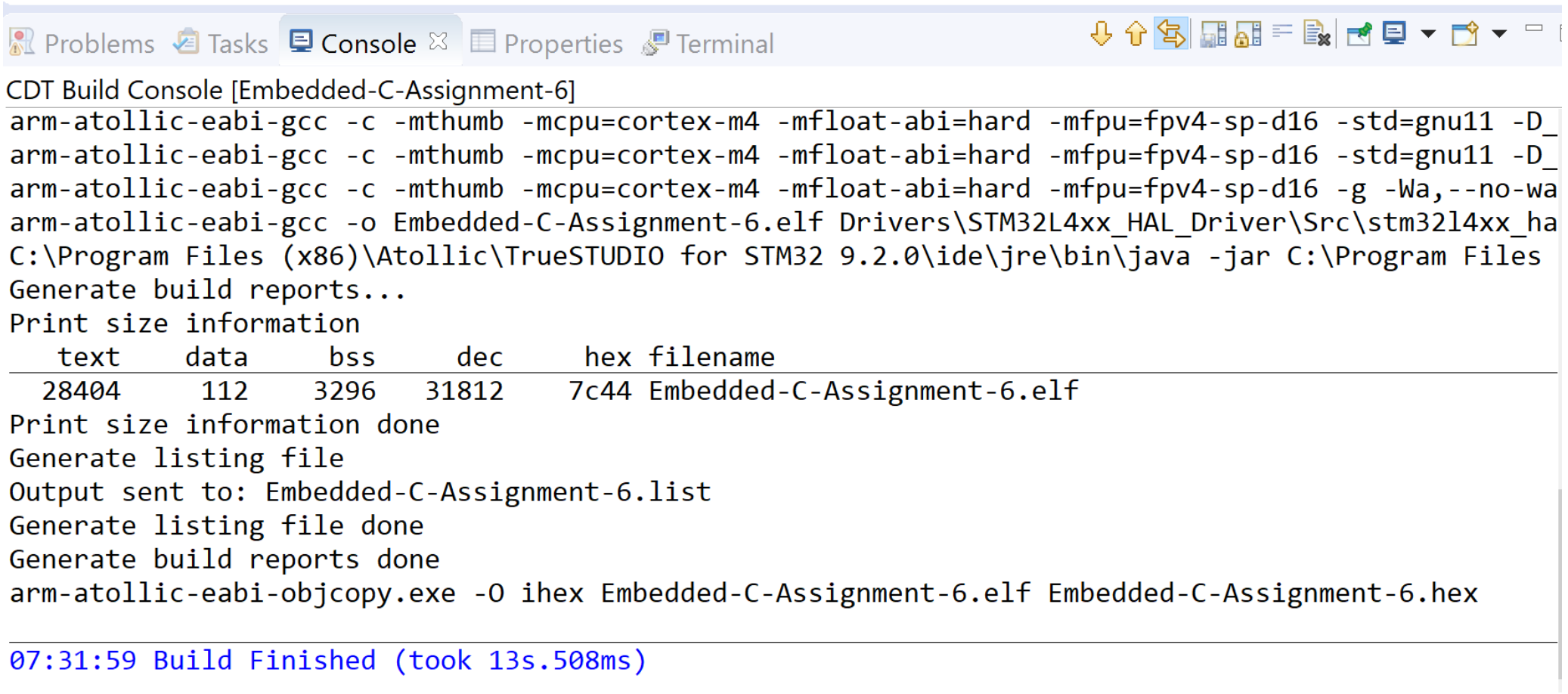


Step 19. Build Project





Step 20. Results of Build – Part 1



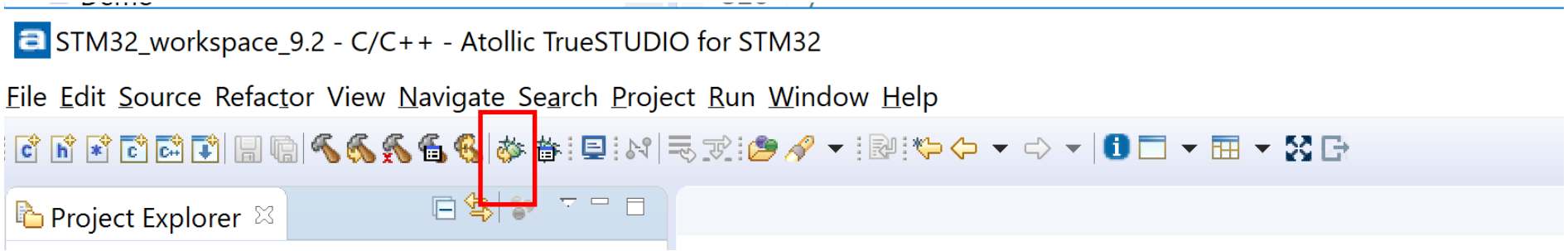
Step 21. Set Breakpoint

Double-click
Here to
Add “green”
breakpoint

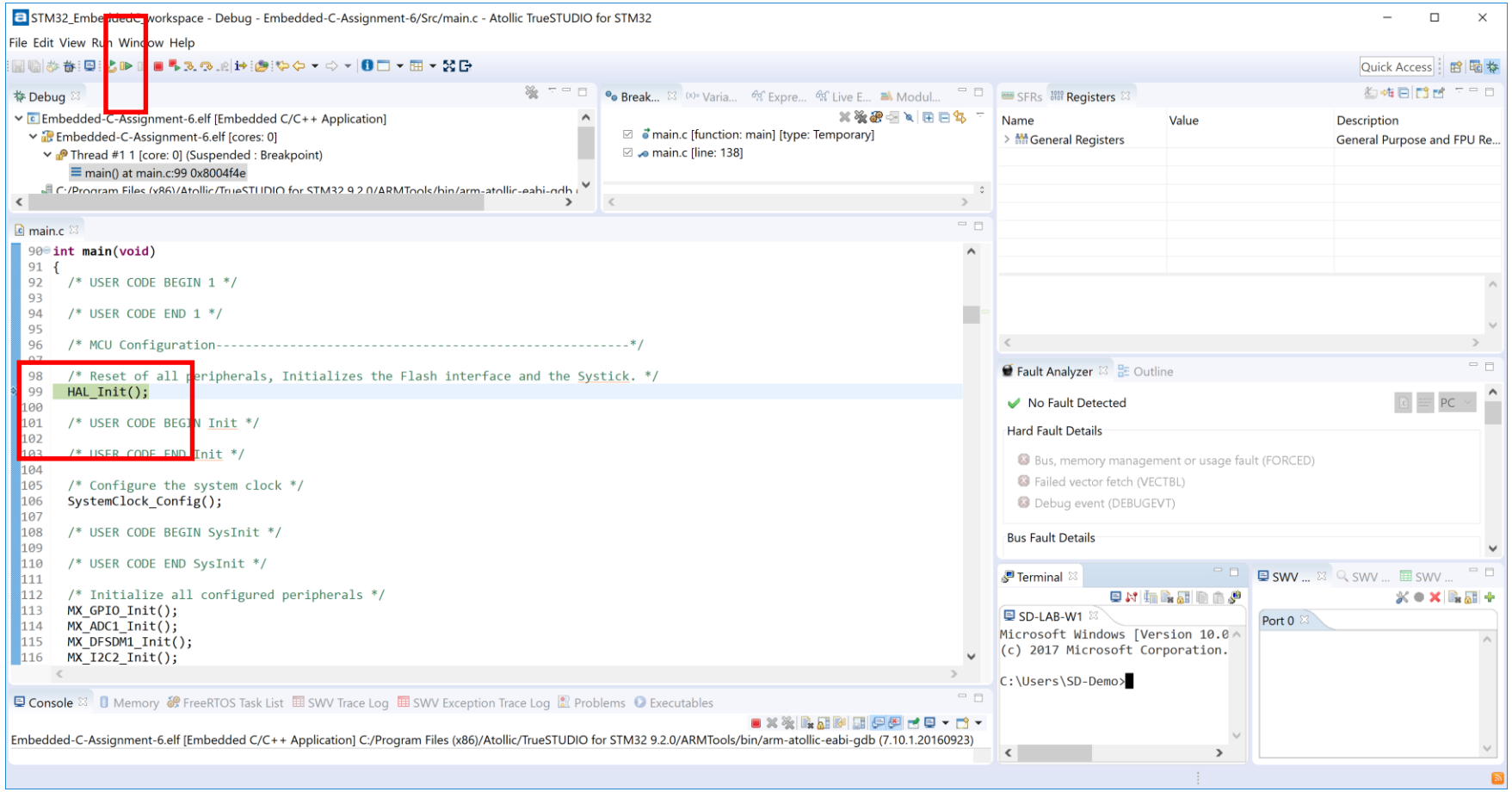
main.c

```
127   /* USER CODE BEGIN WHILE */
128   uint32_t adcResult;
129
130   while (1)
131   {
132       HAL_ADC_Start(&hadc1);
133
134       HAL_ADC_PollForConversion(&hadc1, 100);
135       adcResult = HAL_ADC_GetValue(&hadc1);
136       printf("adcResult: %lu - 0x%lx\n", adcResult, adcResult);
137
138       HAL_ADC_Stop(&hadc1);
139
140       HAL_Delay(1000);
141
142     /* USER CODE END WHILE */
143
144
145     /* USER CODE BEGIN 3 */
146   }
147   /* USER CODE END 3 */
148 }
```

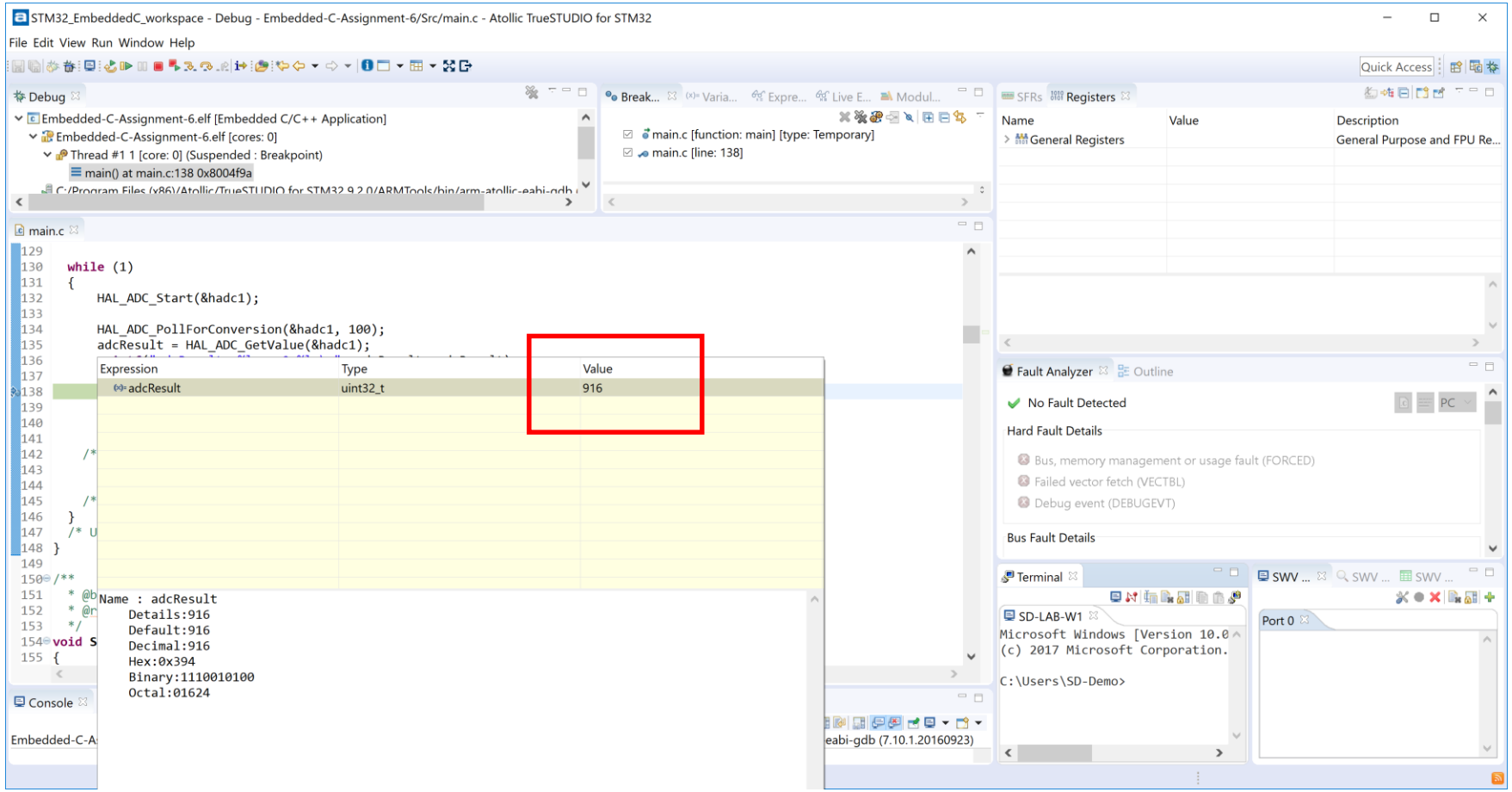
Step 22. Run in Debug



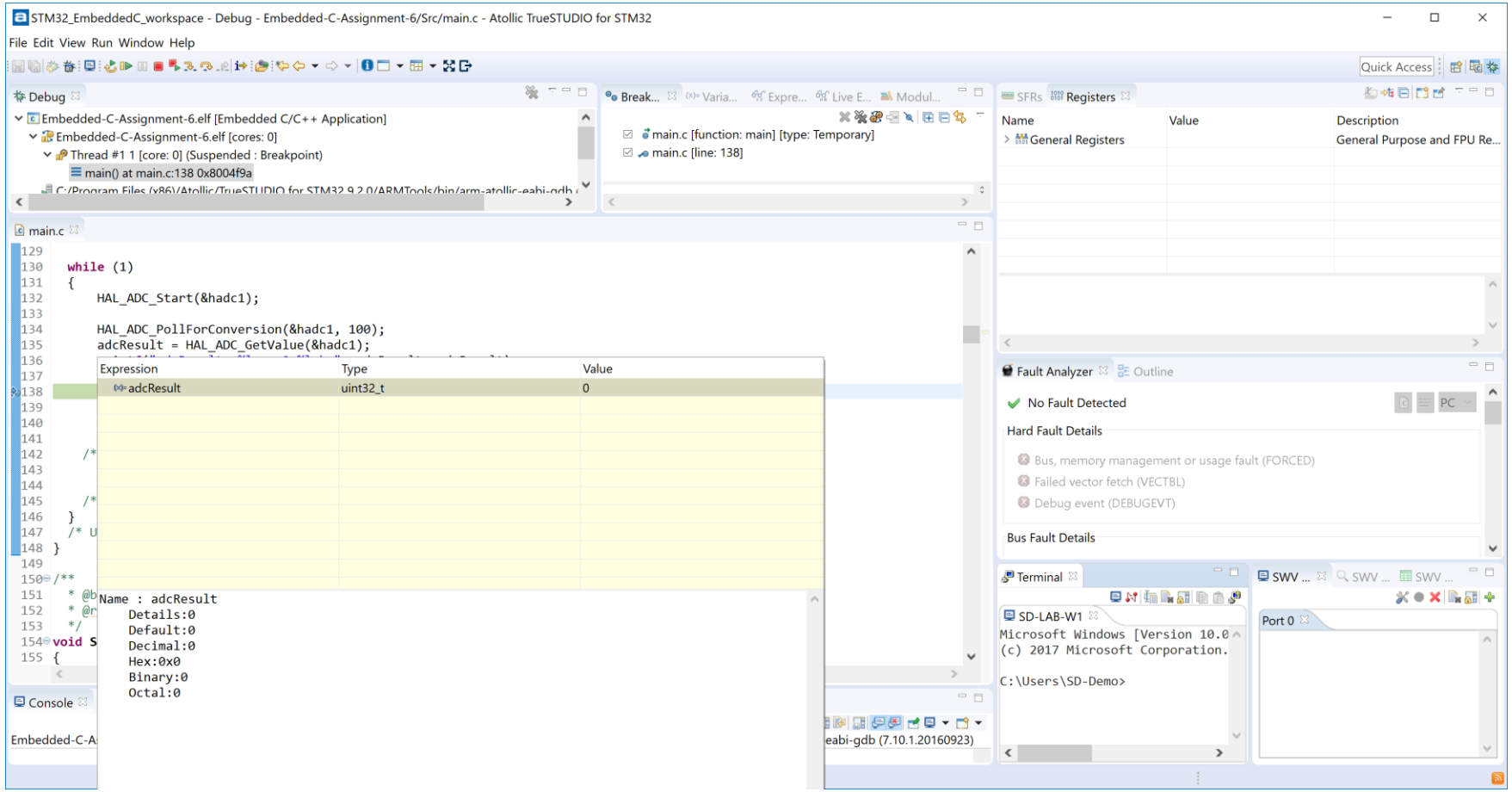
Step 23. Hit Breakpoint, then click Resume



Step 24. Read value when nothing connected to ARD_A0 – In my case adcResult = 916



Step 24. Connect ADR_A0 to GND (two pins over). Then read value of ARD_A0 – In my case adcResult = 0



Step 25. Connect ADR_A0 to 1.5VDC. Then read value of ARD_A0 – In my case adcResult = 1845