

# Beaglebone Black with CODESYS

Madi Nurmanov  
Robotics and Mechatronics  
Nazarbayev University  
Astana, Kazakhstan  
madi.nurmanov@nu.edu.kz

Dinmukhammed Mukashev  
Robotics and Mechatronics  
Nazarbayev University  
Astana, Qazaqstan  
dinmukhammed.mukashev@nu.edu.kz

**Abstract**—This report contains information about establishing connection to Beaglebone Black(BBB) through CODESYS and using it as Programmable Logic Controller(PLC) accessing its Input/Output(IO) ports.

## I. STARTING UP

Firstly, the board should be connected through USB port to PC and open up CODESYS. In order to be able to run CODESYS programs on BBB it is necessary to install the special Plug-in from CODESYS Store Figure 1.



Fig. 1. BBB Plug-in for CODESYS

When creating new Project in CODESYS, from the Device list, Control for **BeagleboneBlack SL** should be chosen as working device (Figure 2).

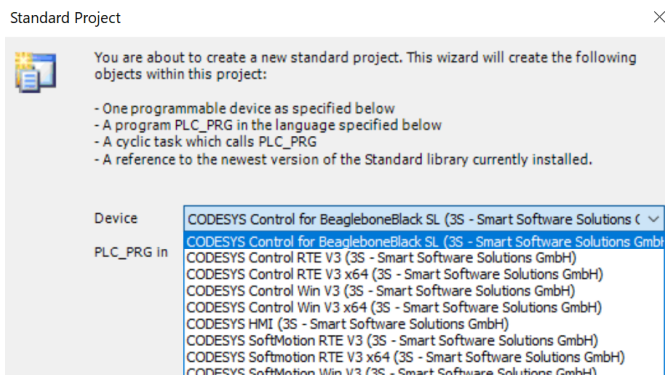


Fig. 2. Device when creating project in CODESYS

After the project is created, in the Project view Setting for BBB window will appear as in Figure 3. If it not appeared, go to **Tools** and click **Update Beaglebone Black**. Where we need to write username (**debian**) and password (**temppwd**) which are default for BBB. Also, the IP adress **192.168.7.2** is default for the board.

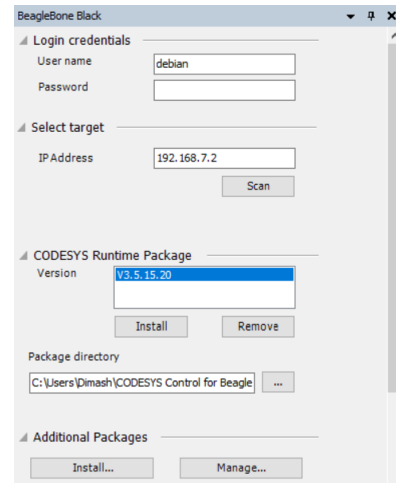


Fig. 3. Device when creating project in CODESYS

After entering all necessary information, we press **Scan** button and choose the BBB from the list. If there is only one device connected to PC it will be seen from the list. We choose it and press **Install** button. System will report that packages on BBB are installed (Figure 4).

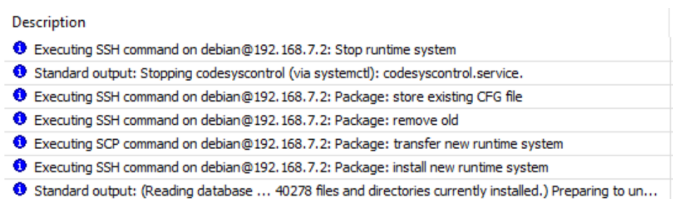


Fig. 4. CODESYS messages list when Installing on BBB

## II. SETTINGS AND ESTABLISHING CONNECTION

When device is installed, we can plug device in order to access its ports and addresses. Right Click on **< Empty >** in Device list and click **Plug Device** Figure 5.

Window with accessible devices will appear Figure 6

Choose the **GPIOs P9/P8** and click **Plug Device** button. After which it will be in the list of the devices.

As devices are settled, the PC can be connected to PLC from Device setting. Where **Scan network...** button should

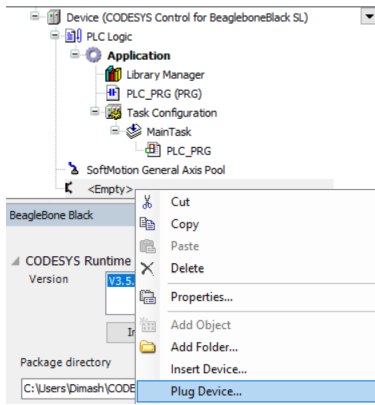


Fig. 5. Plugging device

P9				P8			
DSND	1	2	DSND	1	2	DSND	
VDD_3V3	3	4	VDD_3V3	GPIO_38	3	4	GPIO_39
VDD_5V	5	6	VDD_5V	GPIO_34	5	6	GPIO_35
5V_5V	7	8	5V_5V	GPIO_66	7	8	GPIO_67
PWR_BTN	9	10	SYS_RESETN	GPIO_69	9	10	GPIO_68
GPIO_30	11	12	GPIO_60	GPIO_45	11	12	GPIO_44
GPIO_31	13	14	GPIO_40	GPIO_23	13	14	GPIO_26
GPIO_46	15	16	GPIO_51	GPIO_47	15	16	GPIO_46
GPIO_4	17	18	GPIO_5	GPIO_27	17	18	GPIO_45
I2C2_SCL	19	20	I2C2_SDA	GPIO_22	19	20	GPIO_63
GPIO_3	21	22	GPIO_2	GPIO_62	21	22	GPIO_37
GPIO_49	23	24	GPIO_15	GPIO_36	23	24	GPIO_33
GPIO_117	25	26	GPIO_14	GPIO_32	25	26	GPIO_61
GPIO_125	27	28	GPIO_123	GPIO_66	27	28	GPIO_66
GPIO_121	29	30	GPIO_122	GPIO_67	29	30	GPIO_69
GPIO_120	31	32	VDD_ADC	GPIO_10	31	32	GPIO_11
AIN4	33	36	GPIO_ADC	GPIO_9	33	34	GPIO_61
AIN6	35	36	AIN5	GPIO_8	35	36	GPIO_60
AIN2	37	38	AIN3	GPIO_78	37	38	GPIO_79
AIN0	39	40	AIN1	GPIO_76	39	40	GPIO_77
GPIO_39	41	42	GPIO_7	GPIO_74	41	42	GPIO_75
DSND	43	44	DSND	GPIO_72	43	44	GPIO_73
DSND	45	46	DSND	GPIO_70	45	46	GPIO_71

Fig. 8. BBB pin outs

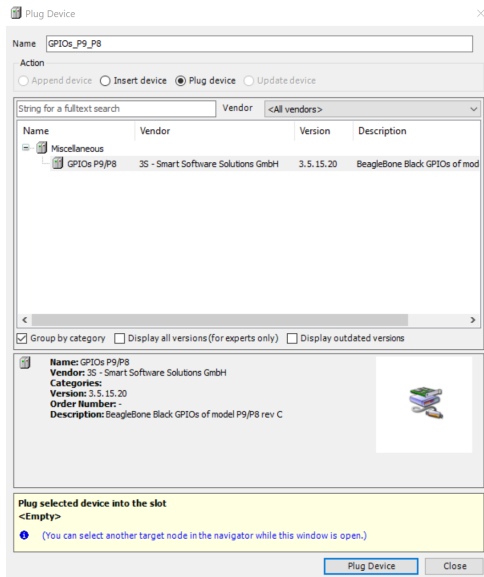


Fig. 6. List of devices to plugin

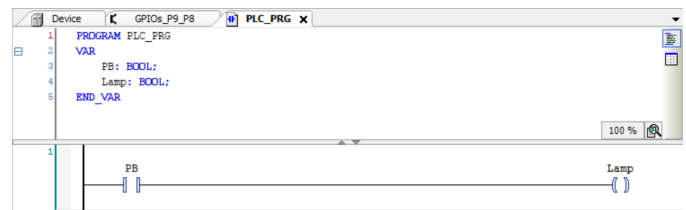


Fig. 9. BBB pin outs

Simple program with one input and one output is created in LD language as example (Figure 9).

As the variables are created. Those should be associated with pins of BBB. Proceed to **GPIOs\_P9\_P8** tab. Where in the parameters of the pins we can set the values of the pins that are going to be used as **Input** or **Output**. For example, we want to use **pin 8** in slot **P8** to get input from the button. So, we set GPIO\_67 as **Input** and GPIO\_68 as output for the lamp which is located on **pin 10** (Figure 10).

GPIOs Parameters		Parameter	Type	Value	Default Value	Unit	Description
GPIOs I/O Mapping		GPIO_66	Enumeration of BYTE	not used	not used		P8 Pin 7
		GPIO_67	Enumeration of BYTE	Input	not used		P8 Pin 8
GPIOs IEC Objects		GPIO_68	Enumeration of BYTE	Output	not used		P8 Pin 10
		GPIO_69	Enumeration of BYTE	not used	not used		P8 Pin 9

Fig. 10. Setting up Input/Output pins

be pressed. From the list, choose device and click **Ok** button Figure7.

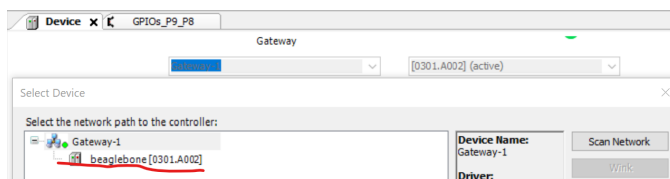


Fig. 7. Connecting to BBB

### III. CODDING THE BOARD AS PLC

The last step is to create code and run it on BBB. Simple program can be crated as an example. Beaglebones Pins can be used as Inputs as long as Outputs. The pin-outs can be found from Figure 8. All the pins which are tagged as **GPIO\_NN** can be used for programming.

After that, variables should be associated with correspond-ing pins. In the same tab, we go to **GPIOs I/O Mapping** where is separate lists for inputs and separate lists for outputs. As our GPIO for input is 67. We go to **digital inputs 64-95** and extend it, from the list choose respective pin (GPIO\_67) and double click to choose variable (Figure 11). The same is done for output.(IMPORTANT to not mix up **digital outputs** list with **digital inputs** list)

After the pins are associated. The program can be ran and tested on hardware.

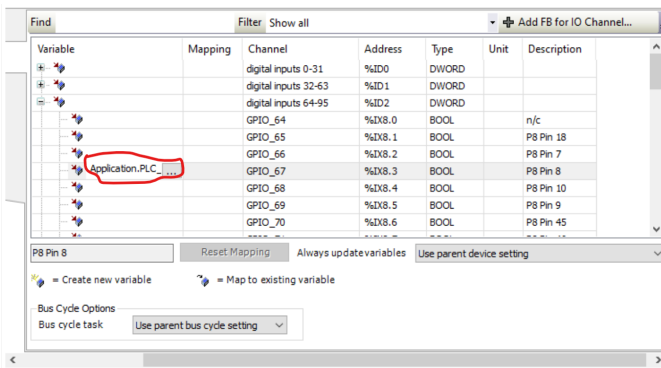


Fig. 11. Associating variables with pins

#### IV. WEB VISUALIZATION

Firstly, the visualization of the system should be created. Then, **WebVisu** is automatically created. So it can be accessed by entering the following address

**192.168.7.2:9090/webvisu.htm**

in browser. The board can be connected to the internet network so it will be possible to access its web visualization from anywhere in the world. However, University WiFi blocks such connections which is why we are restricted by only local network from own router.

#### V. CONCLUSION

It is turns out that CODESYS software can be used not only with different PLCs but with boards such as Arduino or Raspberry Pi. It is very useful when it comes to learning the PLC programming. It is good opportunity to see all processes in hardware. Moreover different implementation for IoT can be made tested and implemented.