Auto Generation of Embedded C Code from Transfer Function

Koti Mudela

Abstract—Nowadays the implementation of control systems in the embedded world is becoming more prevalent. All the processes handled by the control systems are performed by the designed embedded controllers. The continuous or analogue control systems are converted to discrete control systems because discrete control systems can perform logical manipulation and complex mathematical computations by using appropriate programming techniques.

Index Terms— Embedded controllers, discrete control systems, control systems.

I. INTRODUCTION

The embedded system has specific requirements (specifically designed) and performs pre defined tasks (particular function), unlike a general purpose computer. Embedded systems control action primarily dependent on the microcontroller or microprocessors and software in fulfilling the tasks set on them. Almost all the embedded controllers are software-intensive systems; due to increase in the complexity of the embedded systems software has led to greater interest in improving the automation of software production "The automatic code generation could introduce major productivity improvements in automotive software generation by the ability to have industrial software code available target processor with a reduced amount of development time" [4][5]. Many do not realize that every time while boarding a plane in airport, the code inside the aircraft's flight control computer is not always hand written, but is often rather code designed through automated code generation techniques[6].

The Auto code generation software Solves the present problems in coding techniques and offer major advantages:

1) The development time is reduced for the concerned industries in the development process by shortening the coding and testing phases.

2) The quality of the code is increased and reduced the need of low-level testing and validation, as the transformation from transfer function to code is proven to be correct by means of modern formal verification techniques.

3) Complete control over the maintenance of the code generation which is platform independent and open source. This is an essential aspect for industries where the lifetime of the product is very long, which is the case in the automotive industry.

4) The mathematical derivations (example: related to control design algorithms and dynamic states of the system) are avoided by making the GUI (Graphical User Interface) access to developers with no back ground in the field of control engineering.

II. WORKFLOW

The control system is implemented on the embedded microcontroller by using the system transfer function because the transfer function of a control system is a mathematical model and is an operational method expressing the differential equation that relates the output variable to the input variable. For a system if transfer function of the system is known, the output response of the system can be studied for various forms of inputs with a view toward understanding the nature of the system. The continuous transfer function generated from the numerator and denominator coefficients is converted to discrete domain .The bilinear transformation is applied on the transfer function .The Embedded C code is generated from the discrete (Z domain) system and saved in the destination file.

III. CODE GENERATION AND TESTING

For description purpose of the developing system a simple low pass filter transfer function was used .In the first step the system is designed for a low pass filter which is a simple SISO (Single Input Single Output) system.

The transfer function of the low pass filter

$$\text{Transfer\_function} = \frac{S}{S+10}$$

For the convenience the" transfer function "can be divided into two parts

1. Numerator part and
2. Denominator part

$$Transfer\_function = \frac{\text{Numerator}(=S)}{\text{Denominator}(=S+10)}$$

The performances specifications are obtained for the SISO transfer function from the MATLAB and compared with the results of testing phase. For a step input the waveform of mat lab simulation (a) is compared with the testing phase output (b) obtained at 200 Hz.

Manuscript Received October 30, 2011
Koti Mudela, Royal Saudi Air Force for developing security systems.

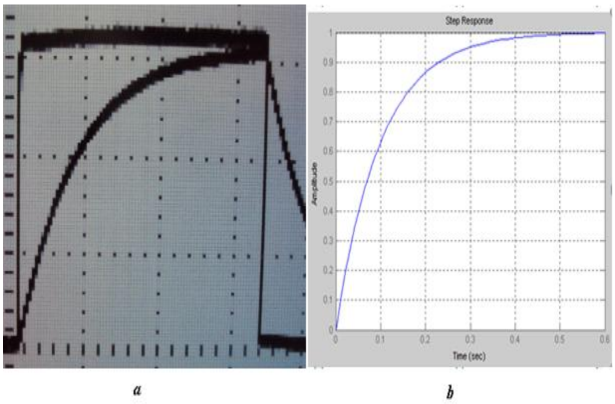


Figure.1 comparing the output for a pulse.

The shape of the both the results are nearly equal, the tested SISO transfer function is low pass filter. Therefore for low frequency the i.e. at 100 Hz the wave form reflects the simulation results of mat lab.

The typical performance parameters of testing phase are compared with the mat lab simulation results. The comparisons of values are illustrated in table (1).

Table. 1 comparison of performance parameters.

Parameters	MATLAB SIMULATION	Test results for 800 Hz
Rise Time	0.2198	0.8899
Settling Time	0.3912	0.999==1
Overshoot	0	0
Undershoot	0	0
Peak	1	0.9
Peak Time	1.0491	1

The testing result comparison of output waveforms at different frequency is shown in figure (3).

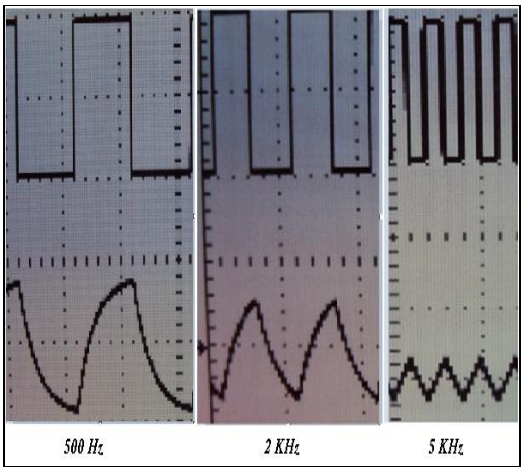


Figure.3 comparison of wave forms at different frequencies

In the testing results as the frequency increases the peak values of the testing phase are decreased but the remaining parameters remains same because as the frequency changes for a filter only the output amplitude changes . Therefore the Auto Generated Embedded C code of SISO transfer function is working as a low pass filter.

In case of MIMO systems transfer function coefficients are in the form of cell arrays. The Numerator and Denominator are cell arrays of row vectors with as many rows as outputs and as many columns as inputs. The MIMO system with 2 outputs and 1 input:

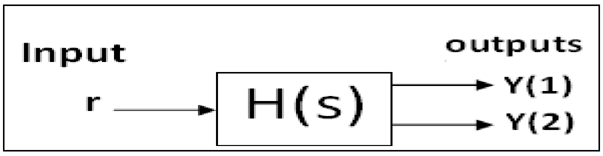


Figure. 4 A 2 input 1 output MIMO system.

The transfer function can be represented as:

$$H(s) = \begin{bmatrix} \dfrac{s+1}{s^2+2s+2} \\ \dfrac{1}{s} \end{bmatrix}$$

IV. TESTING RESULTS AND DISCUSSION

The testing result comparison of output waveforms at different frequency is shown in figure (4).

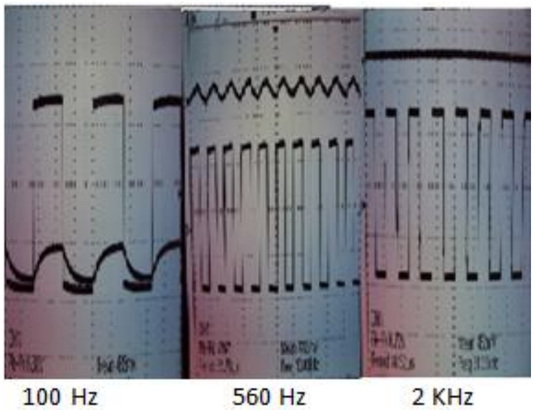
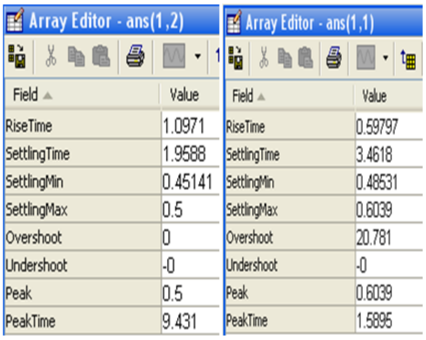


Figure.5 comparison of wave forms at different frequencies

The performances specifications are obtained for the MIMO transfer function from the MATLAB and compared with the results of testing phase. The obtained performance parameters are the average performance output values for both the inputs of tested MIMO system. The typical performance parameters of testing phase are compared with the mat lab simulation results and are illustrated in table (2).

Table.2 Performance parameters comparison for simulated and obtained results for MIMO system.



Array Editor - ans(1,2)

Field	Value
RiseTime	1.0971
SettlingTime	1.9588
SettlingMin	0.45141
SettlingMax	0.5
Overshoot	0
Undershoot	-0
Peak	0.5
PeakTime	9.431

Array Editor - ans(1,1)

Field	Value
RiseTime	0.59797
SettlingTime	3.4618
SettlingMin	0.48531
SettlingMax	0.6039
Overshoot	20.781
Undershoot	-0
Peak	0.6039
PeakTime	1.5895

Parameters	Test results for 3 KHz
Rise Time	Large value (around2.56)
Settling Time	Large value >3
Overshoot	0
Undershoot	0
Peak	0.3
Peak Time	1.0

The average performance results of the testing phase are comparable with the mat lab simulation results.

V. CONCLUSION

The auto Generated Embedded C code by the developed software system has the characteristics of professionally hand written code. The generated code can be implemented in the industrial environment for any control systems if the transfer function of the system is known. The software generates the embedded C code for any higher order SISO and MIMO systems. The developed system is not restricted to integer coefficient transfer functions but also generates the code for complex coefficient transfer functions. The above information lead to a final conclusion which is the designed software system (Auto Generation of Embedded C code from control system transfer function) at this project was successful.

ACKNOWLEDGEMENTS

The work presented in this paper was partially supported by the embedded systems and control laboratory University Of Leicester, Leicester, United Kingdom.

REFERENCES

1. Modern Control Engineering Katshuko OGATA.
2. Control system fundamentals William S Levine.
3. Signals and Systems, 2nd Edition Simon Haykin Barry Van Veen.
4. Weigert T., Weil F., van den Berg A., Dietz P., Marth K. Automated Code Generation for Industrial-Strength Systems // Computer Software and Applications COMPSAC '08. 3 2nd Annual IEEE International Conference. – July 28, 2008 – Aug. 1, 2008. – P. 464–472.
5. Toeppe S., Bostic D., Ranville S., Rzemien K. Automatic code generation requirements for production automotive powertrain applications // Computer Aided Control System Design. Proceedings of the IEEE International Symposium. – 22–27 Aug., 1999. – P. 200–206.
6. http://www.eetimes.com/design/signal-processing-dsp/4007112/Why-is-Autocode-Generation-Slow-to-Adoption.
7. Sampling Moments and Reconstructing Signals of Finite Rate of Innovation: Shannon meets Strang-Fix Pier Luigi Dragotti and Martin Vetterli and Thierry Blu IEEE Transactions on Signal Processing, SP EDICS: DSP-WAVL (Wavelet theory and applications) and DSP-SAMP (Sampling, Extrapolation and Interpolation)
8. RCGES: Retargetable Code Generation for Embedded Systems_ Trong-Yen Lee, Yang-Hsin Fan, Tsung-Hsun Yang, Chia-Chun Tsai, Wen-Ta Lee, and Yuh-Shyan Hwang Institute of Computer, Communication, and Control, National Taipei University of Technology, Taipei, Taiwan.
9. Pont, M.J. (2001) *"Patterns for time-triggered embedded systems: Building reliable applications with the 8051 family of microcontrollers"*, ACM Press / Addison-Wesley. ISBN: 0-201-331381. [Reprinted 2003]
10. "Embedded C", Pont, M.J. (2002) Addison-Wesley. ISBN: 0-201-79523X.
11. Class notes and material from design of discrete systems module Prof N B Jones.

AUTHORS PROFILE

Koti Mudela received the B.Tech and MSc degrees in Electronics and communication engineering and embedded systems and control engineering from Vignan institute of technology and sciences and University of Leicester U.K in 2008 and 2011 respectively. During Masters he stayed in embedded systems laboratory (ESL) to develop auto code generation tool for embedded systems from control system transfer function and programming on ARM LPC2129 network devices. Presently working with the Royal Saudi Air Force for developing security systems.