# FPGA-Based Real-Time Color Tracking for Robotic Formation Control

Ying-Hao Yu[1], N. M. Kwok[2] and Q. P. Ha[1]

[1]School of Electrical, Mechanical and Mechatronic Systems, University of Technology, Sydney, Broadway, NSW 2007, Australia, Email: {yinghao.yu, quangha}@eng.uts.edu.au
[2]School of Mechanical and Manufacturing Engineering, University of New South Wales, Sydney, NSW 2052, Australia, Email: nmkwok@unsw.edu.au

***Abstract***

In construction automation, the tracking of worker location or a moving object is important for labor monitoring, resource management, and machine coordination. For object tracking, a camera is often utilized to obtain information of construction vehicle motion which can then be employed in coordination controls. Real-time tracking of autonomous vehicles, particularly for the control of multiple targets in formations, still suffers from constraints imposed in computation resources. Here the field programmable gate array (FPGA) technology is applied in a prototypical tracking system for vehicles by using a CMOS camera to detect their color-tags. The raw image from the Bayer color pattern is used to indicate the 2-dimensional position of vehicles and encrypted infrared commands are issued to deploy them in a leader-follower formation. It is shown that the novel system-on-programmable-chip with parallel control cores design can efficiently handle color recognition and multi-vehicle control while significantly reduces memory requirement and computation time.

## Introduction

The digital camera has become a popular sensor for monitoring and surveillance systems. By improvement of the semiconductor technology, modern digital cameras with high pixel number can provide greater image detail in various applications. A digital image sensor usually utilizes an array consists of the charged-couple device (CCD) or complementary metal oxide semiconductor (CMOS). By passing light through the lens and color filter array (CFA), the real image is transformed into the mosaic-like RGB image projected on the digital image sensor array (Lukac and Plataniotis, 2007). The raw RGB image then is reconstructed as a meaningful image for human perception by de-mosaicking (Lee, 2005) and color correction (Gonzalez and Woods, 2002). For sensing applications, the additional image processing to extract object's color strength or contour (Nixon and Aguado, 2008) is required for every image pixel. Moreover, it should be noted that imaging processing consumes a lot of memory space and most of computation resources in a computer system.

The digital camera is also a popular sensor in a multi-robot system (Fierro et al., 2002) for obstacle detection and environment learning. Such a system is a very valuable tool in the development of an automated coordination system for construction vehicles. In a real-time object tracking and detection system, particularly for moving targets, the picture acquisition and image processing speed critically determines the system performance. Low computation speed will cause the loss of precise driving control for the multi-robot system and results in collisions. When the robot system needs a high resolution camera to improve imaging details of the object, the coherent time delay with massive computation for image processing will occur. On the other hand, even the powerful microprocessor can mitigate time delay, the higher power consumption is also unsuitable for a mini robot system with battery powered operation (Bräunl, 2006).

Thus a compromise scheme for the multi-vehicle system in image processing issues is sharing the image processing and strategy making tasks in an external server (Fierro et al., 2002). Therefore, the vehicle carries only a simple embedded system which captures and sends image data to the server by wireless in order to get a balance between power consumption and execution speed. Based on the same idea, we propose a new colour-based tracking system for vehicle coordination using color-tag recognition for identification with an external server. This scheme will achieve higher performance of power management and real-time control speed.

In our design, each vehicle will have an embedded system for motion control and communicate by the external server by an encrypted infrared signal. Here, to avoid the access to a large external storage device when running the whole program and acquiring information from outside for the control purpose, we propose use the Field Programmable Gate Array (FPGA) technology (Yu et al., 2008). The server is designed by using an FPGA chip with parallel logic control groups (cores) in order to achieve such benefits of low power, low cost, and flexible speed control as compared to a general purpose computer. The designed external server will monitor the vehicle by the captured 2-dimentional object positions with a single digital camera equipped on the server. The target tracking by using raw image, window of interest (Carvalho et al., 2000), and the dynamic trust region (Wang et al., 2006) designs is obtained for a low computation burden in the FPGA chip. Finally, the low logic gate usage and the multi-object tracking with the basic leader-following strategy in formation control will demonstrate the feasibility of this novel system.

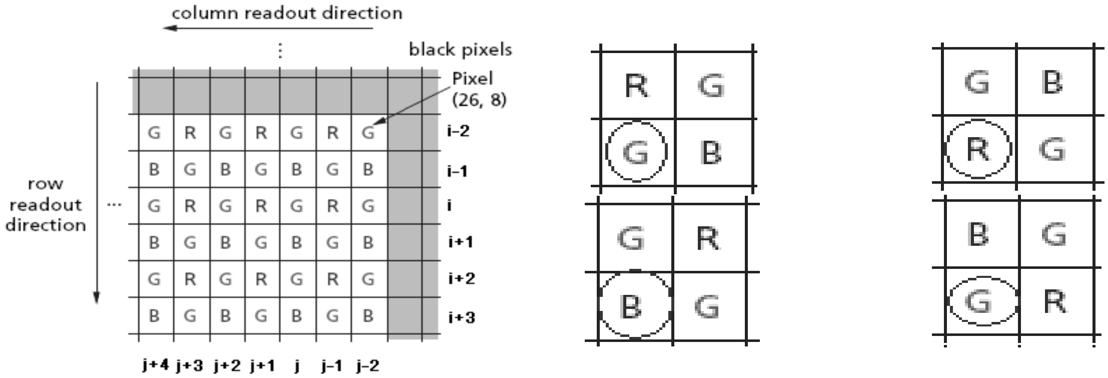
The paper is organized as follows. Section II presents the raw image processing with Bayer pattern, color prediction, and noise filtering. The color tracking mechanism is described in Section III. The FPGA design, resource usage, and the experimental result captured from the video images are illustrated in Section IV. Some discussion of the development is given in Section V. Finally, a conclusion is drawn in section VI.

## Processing of Raw Image

Image processing with large matrix calculation will deteriorate the system performance due to limitations in computation speed. The simpler image processing such as the threshold (Hu et al., 2006) could be the better choice for lower demand for computation resource. Based on this strategy, we develop the color prediction and noise filter algorithm with the raw image to achieve real-time color recognition.

### *Bayer Pattern*

The mosaic-like RGB image in the Bayer pattern (Lukac et al., 2005) is shown in Figure 1.



(a). The Bayer pattern (b). Possible combinations for every four pixels

Figure 1. Bayer pattern of a typical digital image.

The color pixels are obtained by scanning the image line by line. We determine the color components of the circled pixels, Figure 1(b), by a local neighborhood of every four scanned pixels. In our system, we only need to recognize whether the label of object's surface belongs to a blue or green color range. Hence, we define the basic threshold range for blue- and green-like color as:

$$G \Leftrightarrow (g_1 \cap g_2) > (r \cup b) \quad (1)$$

$$B \Leftrightarrow b > (r \cup g_1 \cup g_2), \quad (2)$$

where g, r, and b denote the RGB pixel strength for the combination of four color pixel units.

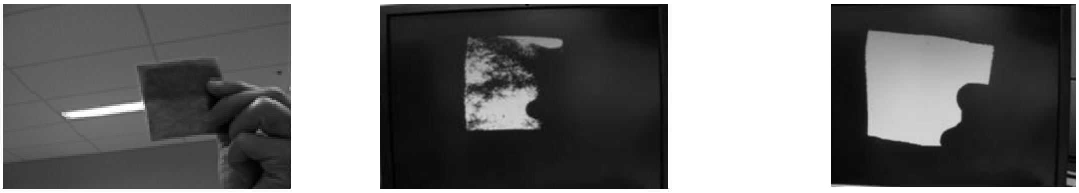
### *Color Prediction*

In real world problems, the object's color always changes due to effects of light source, shadow, and reflection on surface. The basic color definition above is, therefore, not sufficient for an arbitrary environment. Thus, the color prediction is designed to compensate for the incurred color error. We rewrite the color definition as:

$$G \Leftrightarrow (g_1 \cap g_2) > (r \pm \Delta r_n \cup b \pm \Delta b_n) \text{, if } G \Leftrightarrow g_{1(i-1,j)} \cap g_{2(i-1,j)} \quad (3)$$

$$B \Leftrightarrow b > (r \pm \Delta r_n \cup (g_1 \cup g_2) \pm \Delta g_n) \text{, if } B \Leftrightarrow b_{(i-1,j)} \text{, } (4)$$

where $\Delta r_n$, $\Delta b_n$, and $\Delta g_n$ are the maximum color strength errors that are measured from the object's surface in the test environment by the digital sensor, n denotes the different color strength level below the saturation value. The comparison between color prediction and image captured in our laboratory is given in Figure 2. The system shows the full scale of grey if the green color is confirmed. Figure 2(b) shows the grey levels of detecting green where uncertainties exhibit due to the shadow in image 2(a). In the contrary, Figure 2(c) illustrates a good detection result with color prediction by using (3).



(a) (b) (c)

Figure 2. The operation and comparison of the color prediction approach.

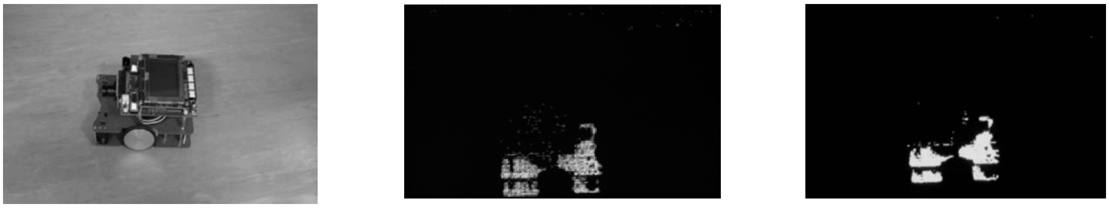
*Noise Filter*

Since color prediction can make the full grey level image more complete, the background noise can be eliminated by examining the continuity of pixel groups. We add an extra criterion defined by:

$$G \Leftrightarrow G_{l1(i,j)} \cap G_{l2(i,j)} \cap G_{l3(i,j)} \quad (5)$$

$$B \Leftrightarrow B_{l1(i,j)} \cap B_{l2(i,j)} \cap B_{l3(i,j)} \text{, } \quad (6)$$

where Gl(i,j) are the active pixel on scan line "l". Figure 3 shows the performance of noise filtering combined with the color prediction function. The mini robot's grey image is improved for color tracking tasks with the proposed color prediction and noise filter as shown in Figure 3(c). Here, a mini-robot, the Eyebot (Bräunl, 2006), was used in laboratorial experiments to test the development for color detection purposes on a construction site.



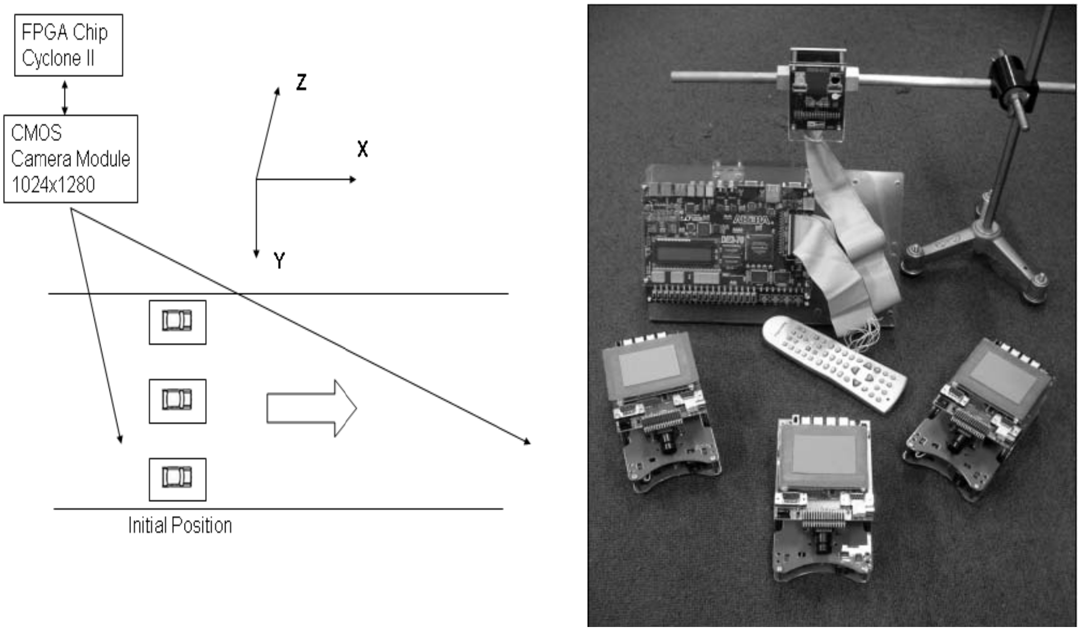
(a) Eyebot (b) Robot image surrounded by noise (c) Image with noise filter.

Figure 3. The performance of noise filtering.

## Color Tracking Mechanism

The color tracking system is controlled by an external server which is designed on a FPGA development platform, the DE2-70 with Cyclone II FPGA, from Altera®. Figure 4 shows the scheme and the real system components. The FPGA chip monitors the mini robot, via a single camera above the robots. For activating the tracking mechanism, every Eyebot needs firstly to pass through the initial area and then monitored by the FPGA chip. The chip will recognize the robot via a 6 x 4 cm2 square greenish blue paper adhered on top as the window of interest (Carvalho et al., 2000); the robot will be detected and tracked if the counted blue pixel amount is over the threshold level in the initial docking area.

The green cloth attached with the greenish blue paper is used to initialize the start and end point of pixel counting. After the Eyebot is locked by the FPGA chip, the chip can track the object by defining the dynamic trust region. In every dynamic trust region, the FPGA chip checks the threshold level of sampled blue pixel amount and drawing the new red trust region by the moving centre point in blue pixel area. Here we assume the maximum speed of centre point is moving for 1/2 length on x or y coordinate of last tracked blue area in rectangular shape then set the new trust region for four times bigger than the blue label as the minimum tracking dimension with the Eyebot's speed 3.8 km/ hr. Finally, the FPGA platform contacts the Eyebots via the encrypted infrared commands of a TV remote.



(a) Color tracking scheme

(b) Components of the vehicle/robot tracking system

Figure 4. Color tracking system and components.

## FPGA Chip Design and Experimental Scenario

Unlike the general purpose processor which adopts fixed hardware design for the maximum compatibility and flexibility, we embed the logical programming into the hardware circuit design directly. This kind of design helps the low speed system to satisfy the demand for sufficient real-time control with highly parallel architecture, shown in Figure 5. The raw image is distributed to three control groups (cores), each group controls one Eyebot and the leader core endowed with the extra ability to lead the followers.

An Altera® FPGA platform supplied with Terasic® auxiliary VGA demo program is used in our experiments. The surveillance camera is installed up from the ground for 1.2 m with 45 degrees inclined to the Eyebot on ground, and the frame rate is 12 fps for 1024×1280 resolution. The monitoring area under stationary camera monitoring is 1.2×1.6 m2.

Figure 6 presents the test results captured from the camera. The test scenario is shown in Figure 6(a). The FPGA chip locks three Eyebots immediately after the robots enter initial docking areas, Figure 6(b) to 6(c). In our test scenario, the leader drives forward automatically after a period of time delay, then the two followers move forward immediately after they received the commands from the leader, Figure 6(d). In the end, the leader waits for the followers at the goal area and asks them to stop after having reached the goal area, Figure 6(e). This test scenario demonstrates the use of color tracking, remote wireless polling, parallel driving control, and basic multi-robot formation (a wedge), performed by hardware design in a single chip.

The chip resource usage given is summarized in Figure 7. It is shown that the novel color tracking system including the VGA demo interface only consumes 7% of the total logic elements (LEs). This saving in resources would enable us to put in more intelligent control functionalities in the single chip. This advantage is rather important in the robotic formation control in the presence of obstacles and also for construction automation where the tracking of worker location or a moving object is important for labor monitoring, resource management, or machine coordination.

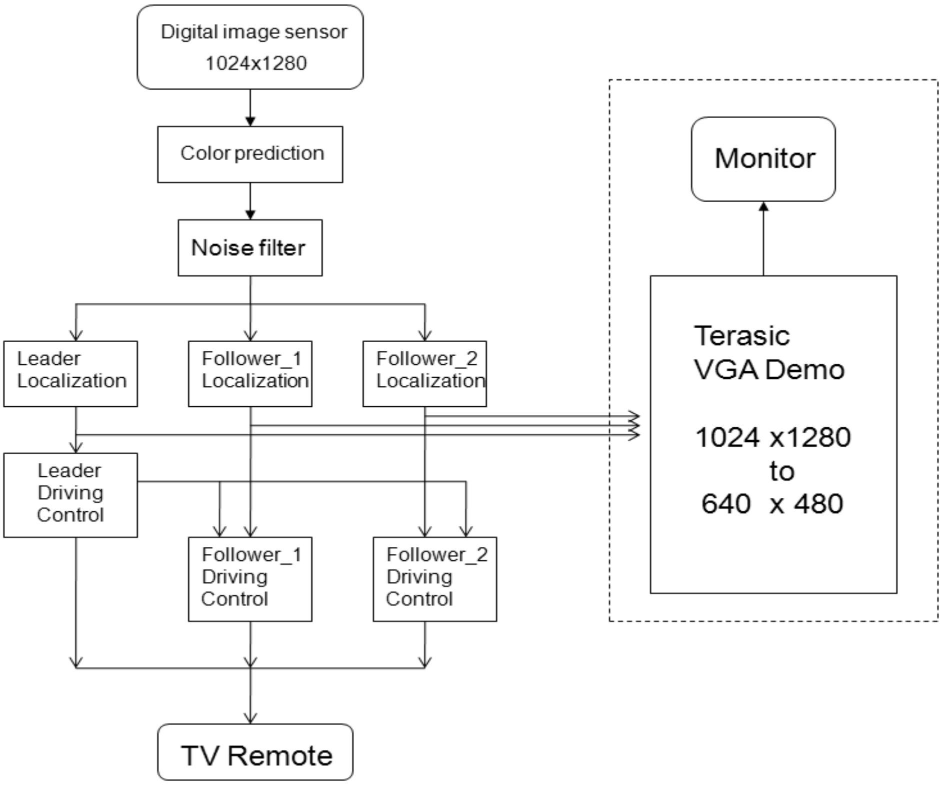


Figure 5. Internal function design of the FPGA chip.

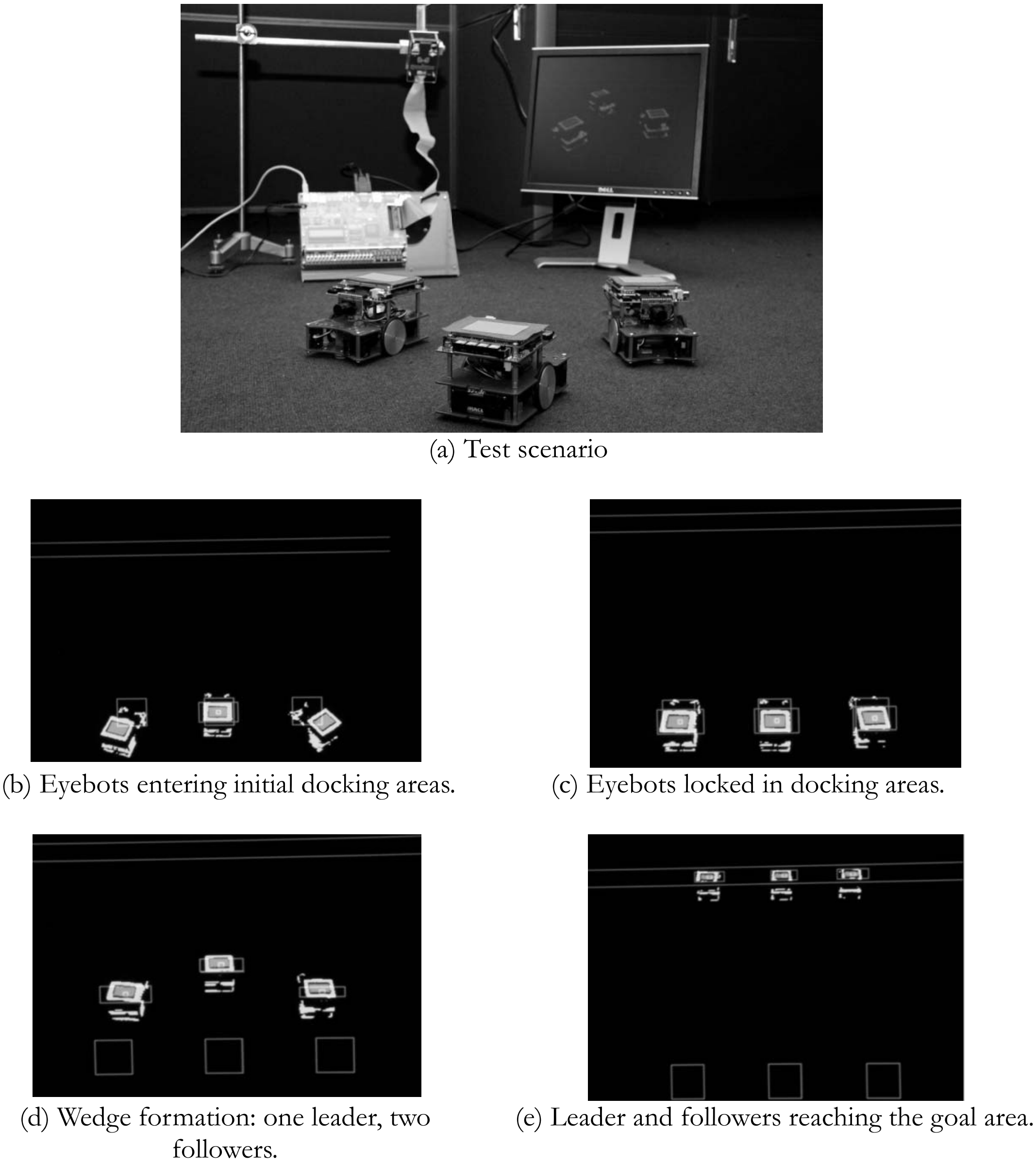
## Discussion

The proposed color tracking approach using raw image, initial docking position, and dynamic trust area has contributed to a good performance in the proposed color tracking system. The color discrimination is useful in detecting the worker's vest, moving vehicle, and the position of conveyer for operation or security reasons, which is essential for construction automation.

Meanwhile, the color prediction plays a critical role by providing a reliable image for object recognition, and the real-time raw image processing is not effected by the complexity of high resolution digital sensor array. Notwithstanding the automatic facilities usually need to operate in the adverse circumstance in construction site, the system-on-programmable-chip design with FPGA chip can provide the superiority for easy installation, maintenance, and stability. The savings in chip resources allow more room for further incorporation of decision making and control algorithms.

## Conclusions

This paper has presented a real-time color tracking system with low cost and low power consumption. The refined circuit design with raw image, initial docking detection, dynamic trust area tracking, and parallel control guarantee satisfactory performance in FPGA chip design. The effectiveness of the approach has been demonstrated by experiments conducted using mini-robots. Finally, the least LE usage in FPGA chip also inspires more intelligent control design in the future. The development is useful in the tracking of worker location or a moving object labor monitoring, resource management, or machine coordination for construction automation.



(a) Test scenario

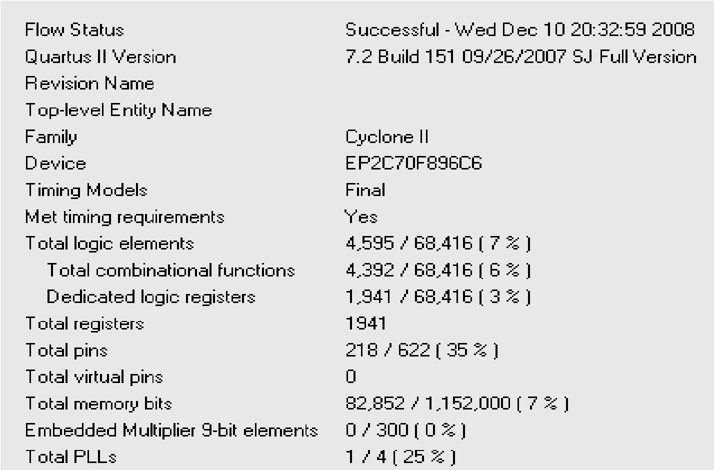
(b) Eyebots entering initial docking areas.

(c) Eyebots locked in docking areas.

(d) Wedge formation: one leader, two followers.

(e) Leader and followers reaching the goal area.

Figure 6. Color tracking with three Eyebots.



```
Flow Status                          Successful - Wed Dec 10 20:32:59 2008
Quartus II Version                   7.2 Build 151 09/26/2007 SJ Full Version
Revision Name
Top-level Entity Name
Family                               Cyclone II
Device                               EP2C70F896C6
Timing Models                        Final
Met timing requirements              Yes
Total logic elements                 4,595 / 68,416 ( 7 % )
   Total combinational functions     4,392 / 68,416 ( 6 % )
   Dedicated logic registers         1,941 / 68,416 ( 3 % )
Total registers                      1941
Total pins                           218 / 622 ( 35 % )
Total virtual pins                   0
Total memory bits                    82,852 / 1,152,000 ( 7 % )
Embedded Multiplier 9-bit elements   0 / 300 ( 0 % )
Total PLLs                           1 / 4 ( 25 % )
```

Figure 7. Resource usage in the FPGA chip.

## Acknowledgement

This work is supported by UTS research grant 2006000848 and, in part, by the Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

## References

[1] Bräunl T. (2006). Embedded Robotics, Springer-Verlag New York, Secancus Inc., NJ, USA.

[2] Carvalho José R.H., Rives P., Ailton S.B, and Bueno Samuel S. (2000). "Visual Servo Control for a Class of Mobile Robot," Proc. 2000 IEEE Intl. Conf. on Control Application, Anchorage, AK, USA, pp. 431-436.

[3] Fierro, R., Das, A., Spetzer, J., Esposito, J., Kumar, V. , Ostrowski, J., Pappas, G., Taylor, C., Hur,Y., Alur, R., Lee, I., Grudic, G. and Southall, B. (2002). "A Framework and Architecture for Multi-Robot Coordination," International Journal of Robotics Research, Vol.21, No. 10-11, pp. 977-995.

[4] Gonzalez R.C. and Woods R.E. (2002). Digital Image Processing, Prentice Hall, Upper Saddle River, NJ, USA.

[5] Hu Q., Hou Z., and Nowinski Wieslaw L. ( 2006). "Supervised Range-Constrained Thresholding," IEEE Trans. on Image Processing, Vol. 15, No.1, pp. 228-240.

[6] Lee X. (2005). "Demosaicing by Successive Approximation," IEEE Trans. on Image Processing, Vol. 14, No. 3, pp. 370-379.

[7] Lukac R. and Plataniotis K.N. (2007). Color Image Processing: Methods and Applications, CRC Press.

[8] Lukac R., Plataniotis K.N. and Hatzinakos D. (2005). "Color Image Zooming on the Bayer Pattern," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 15, No. 11, pp. 1475-1492.

[9] Nixon M.S. and Aguado A.S. (2008). Feature Extraction and Image Processing, Academic Press.

[10] Wang Z., Zhang K., Wu Y., and Lu H. (2006). "Tracking Dim Target in Infrared Imagery Using the Trust Region Embedded Particle Filter." Proc. 2006 8th Intl. Conf. on Signal Processing, doi 10.1109/ICOSP.2006.345667.

[11] Yu Y.-H., Kwok N.M. and Ha Q.P. (2008)., "A New System-On-Programmable-Chip (SOPC) for Smart Car Braking Control," Proc. Int. Conference on Intelligent Technologies (InTech 2008), Samui, Thailand, 7-10 Oct., pp. 34-39.