FPGA Based Hardware in the Loop Test Platform of Small Size UAV

Ta-ming Shih, Ho-chung Chang

Abstract—**Recently, there has been a need of small size UAV for vast applications in military and civilian applications as local area surveillance reconnaissance in hostile condition, damage assessment in natural disaster and remote sensing of harmful materials. UAV can finish missions without risking human life. However, applying low-cost sensors for flight control is of extreme challenge due to less accuracy nature of sensors. Therefore a Hardware-in-the–Loop(HIL) is most necessary for test and evaluation purpose to reduce the risk of flight testing. In this paper, a HIL system is built up to test the autopilot hardware performance, control parameter tuning. By building the dynamic model of airplane and combining the microcontroller and FPGA(Field Programmable Gate Array) based hardware interfaces, a 3D visualize HIL platform is built to improve test efficiency and reducing time and cost.**

I. INTRODUCTION

The UAV has currently provide an efficient and economic platform for civilian and military applications as disaster control, local environment detection, security surveillances, field wireless network deployment, etc. Due to the varieties of mission need, low cost UAV specially using low cost sensors as gyros, accelerometer and GPS is considered to be the solution to satisfy different mission requirements [1,2].

In order to reduce the developing time and cost, it is normally using commercial available flight control system like MicroPilot or Cloud Cap to provide complete guidance solution. However, these systems are focus on higher lever control rather for lower level customer adjustment. It is the disadvantage of commercial product, which is easy of use but restrictive to functional design and hard to be adjusted in control structure while facing mission requirements change. As to be adapted to different vehicle configuration and also reducing the development barrier of UAV in a sense of time and cost, an efficient HIL development platform is developed to verify the vehicle flight control scheme. After through testing, it can later be put on the real vehicle for experimental test.[3,4]

The purpose of this research is to build a customized UAV HIL system. Issues as system integration design of flight computer, sensor, and vehicle attitude control will be tested on this platform. The research procedures are arranged as four parts. First, a real UAV model plane is customized for installing sensors, servo motor, flight computer and power control module. Manual and autonomous switching function is also designed through the control key of RC controller to initiate remote or autonomous command. The next step is to build the hardware communication interface between the software simulation environment and real attitude sensors, servomotors for control and sensor signal output/input. Then, the customized UAV software simulation based on fly condition and operational design by NI Simulation and Control toolkit will be built. Next, the control design scheme and integrate with the vehicle simulation will also be built and the control scheme is embedded into the fly control controller. Lastly, by using the reconfigurable nature of FPGA to simulate the hardware interface of analog and digital input output type as serial, SPI and PWM connection of sensors and motor. The final system is able to demonstrate the communication between the software simulated vehicle and flight control board through FPGA customized interface.

Ta-ming Shih is with the Department of Power Vehicle and System Engineering, NDU, CCIT, Tao Yuan, Taiwan, R.O.C(Stm0137 @gmail.com).
Ho-chung Chang is with the Graduate School of Defense Technology, NDU, CCIT, Tao Yuan, Taiwan, R.O.C(G980101@ccit.edu.tw).

II. HIL SYSTEM INTRODUCCTION

A. HIL Structure Design

The focus of this project is to build a HIL fly control evaluation system. The hardware system is using an embedded NI PXI 8186 module as real-time controller, it is also integrated with the NI LabVIEW Control and Simulation software module to build the vehicle model on remote host computer and deployed to the PXI 8186 through real-time simulation module. By assuming the operating conditions such as wind speed, sensor noise, wing surface offset, control loop and gain design, it is able to approaching the real system dynamic response, as shown in fig 1.

Originally, a FP2000 embedded system from NI instrument was used for preliminary concept design and verification. However, it was soon abandoned for limited capacities of hardware as low program volume and with no supportability in configurable FPGA I/O expansion. The PXI 7833R module with reconfigurable FPGA I/O was later used for SPI, serial, PWM, analog, digital I/O interface configuration. The major subsystem is as follows:

(1) PXI 8186: used as PXI controller to provide a platform for simulation and controller development,
(2) PXI 7833R FPGA module: mostly used for multiple data interface configuration. However, the vehicle model can be deployed from simulation module as bitmap format,
(3) PXI Real-Time core used for real-time control of PXI system to perform process management,
(4) PXI 7833R FPGA digital input/ output used for PWM sensing and actuating. It is also configured as SPI, RS232 for data communication digital sensors,

（5） PXI 7833 FPGA Analog in/out used for analog sensor retrieving and switching control.

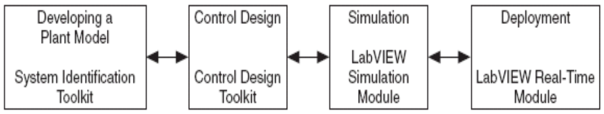


Fig. 1. Software development process

The developing concept can be further explained as follows. Firstly, mathematical model of vehicle system and associated controller is built in the simulation environment and the simulation results are compared with the system specifications to exam the fidelity of simulation models. Then, a real-time controller verified from the previous process is deployed to the embedded real-time controller by using automatic code generation function, and the prototype controller will later be used to test the dynamical responses of virtual vehicle in the simulation environment. Lastly, the verified embedded controller will be fitted on the real vehicle to test the control performance of real vehicle.

Fig. 2. System structure of HIL test system

The finished HIL is demonstrated as Fig 2. The FPGA module uses 4 different time sequence loop to retrieve sensor and control data. PID controller is responsible for the pitch and yaw control of vehicle at the rate of 50 hz. PWM interface output command to the speed and attitude controller to regulate the aileron and rudder. Also a SPI interface is configured to acquire gyro and accelerator data, and an RS232 interface is used to retrieve GPS information form a receiver at 4800 bps rate. Compared to the traditional real-time HIL system, the FPGA module will provide better flexibility and ensure deterministic data stream received by the software.

B. *On Board Flight Control System*

For the design of on-board UAV flight control system, most often it includes five parts as microcontroller, sensor board module, actuator module and communication module to provide full capacity of autonomous control. The flight control able should be able to perform data acquisition and processing and execute flight control command based on the received sensor data

C. *Software Developing Process*

In this research, instead of using traditional C language, the graphical control software LabVIEW developed by National Instrument was chosen to develop the overall system because of providing advantages in interoperability and compatibility of various interfaces. It also expedites programming process by using graphic design concept and subprograms associated in the program [10]. The LabVIEW software and NI PXI hardware is initially integrated and initialized by using NI MAX software to test and configure the interoperability between the hardware and software.

The process for the software design includes three stages as rapid control prototyping (RCP) and hardware in the loop (HIL). Therefore, in the first stage, the vehicle model is approached by lineralized model and the parameters are approached by using system identification method; the process has data acquisition and parameter estimation. Then, the whole system including vehicle and control model will be built by NI Simulation module(as shown in Fig. 3.)及FPGA software module, and deployed to the PXI 8186 system to do the simulation in the PXI system.

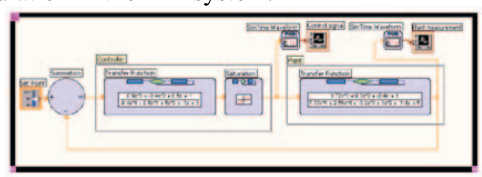


Fig. 3. Vehicle and controller simulation

In the RCP stage, the design controller in the first stage will be deployed to real-time controller to finish up the design process and verified with the real vehicle model, as shown as Fig. 4.

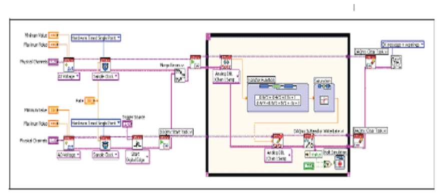


Fig. 4. RCP program block diagram deployment

In the HIL stage, the process in the previous stage is reversed. The controlled is deployed to a real embedded controller and verified with the software based vehicle model to judge whether it satisfies the control requirements as shown in Fig. 5. After cross verification, the system is admitted to behave as anticipated system response.

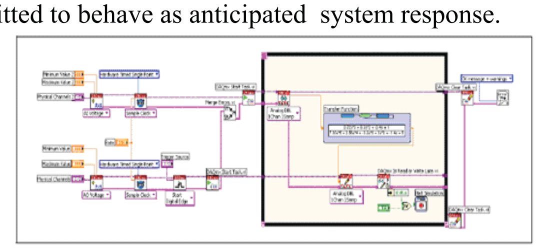


Fig.5. HIL program block diagram deployment

III. INTRODUCTION OF EXPERIMENTAL VEHICLE

A. *System Description of Experimental Vehicle*

In order to be easy of use for the entree level user, in this research, low cost solid state sensors with median lever precision are used for the fly control system in this project.

The high/low lever software were also developed after going through the simulation process.

In the beginning, a commercial off the shelf wooden model airplane is use for vehicle test as shown if Fig. 6. The system specifications are listed in Table. 1.

The hardware structure of flight control system is shown as Fig. 7. The left side is a flight control board, which includes both power module and sensor module, each individually responsible for vehicle control and sensor data acquisition. The data can also perform down link through a radio module from Aerocomm, and will be transmitted to a ground station developed by LabVIEW for Windows XP on a laptop computer. The remote GUI has two frames; one for virtual pilot and one for data received from radio module.。

Fig. 6. Experimental Vehicle

TABLE1 Model Vehicle Specificatrion

Wing span	1600 mm
Total weight	800 g
Wing surface	372 sq in
Length	920 mm
Wing load	29.1~35.5g/dm2(9.79~12.0oz/sq ft)
Propeller	9inch. 3 leaves
Servo motor	Mid range load 4EA
Speed controller	15Amp.
Receiver	40 M HZ
Battery	11.1V 1.5Ah Li-Po
RC controller	5CH PPM control

B. *Sensor and Actuator modules*

（1） Micro processor: two STAMP SX processors from Parallax are used as sensor and actuator modules. The attitude data are acquired by the sensor modules, and the analyzed control parameters are sent back to actuator module and output to the servo motors to do attitude correction.

（2） GPS: A GPS receiver is used to provide guidance correction of flight path. 10 way points can be set for UAV flight path comparison to correct the vehicle attitude.

（3） 3-axis accelerator: An accelerator H48C form Parallax is used for providing UAV attitude information.

（4） 2-axis Gyro: An IDG 300 gyro from parallax is used for providing angular speed sensing of roll and pitch rate.

（5） Electric compass: A HM55B compass from parallax is used for provide UAV heading information.

（6） Power unit: A Li-Poly 11.1 Volt battery is used to provide speed motor control and servo motor power of aileron, rudder and elevator.

（7） Altitude meter: An absolute pressure gage MPXHH6115 form Freescale is used for low altitude estimation.

（8） Speed sensor: A differential pressure gage MPXV5004 is used for vehicle speed measurement.

（9） Environment sensors: SHT11 from parallax is used for temperature and humid sensing and recording.

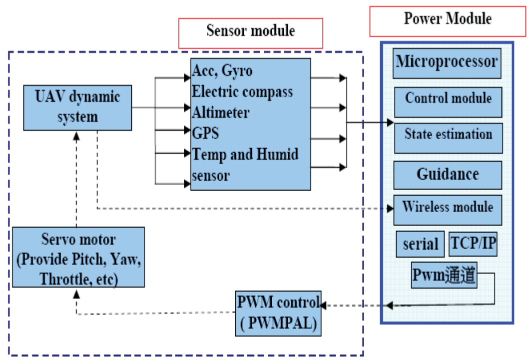


Fig 7. Hardware structure of flight control system

C. *Initial Fly Computer Test*

The PCB board is of two layer designs as shown in Fig. 8. The power is connected with an on board 11.1 volt battery regulated to 5 volts to provide operating voltage. In the currently design, two Stamp SX processors are chosen, the major reasons are that it provides synchronous/an synchronous serial communication, analog/digital I/O. Most of all, Parallax provide a free IDE Basic programming software development tool, which provides most benefits for user not familiar with C or C++. Therefore, it is to much easier to incorporate all interfaces to realize the flight control computer.

The test including three parts as serial communication test, sensor test and PWM signal test to ensure it is functioning as designed.

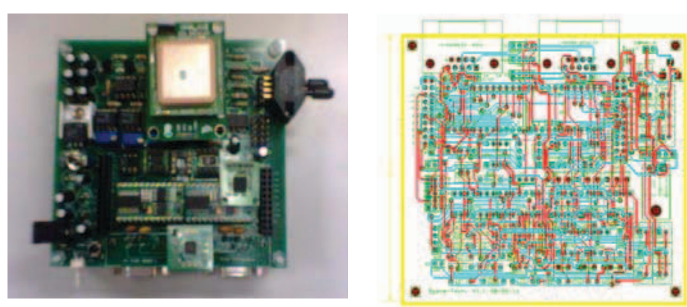


Fig. 8 PCB prototype board

IV. SYSTEM DYNAMIC SIMULATION

A. Vehicle Dynamic Model[5]

The relationship between the body axis and ground inertial axis is shown as Fig. 9.

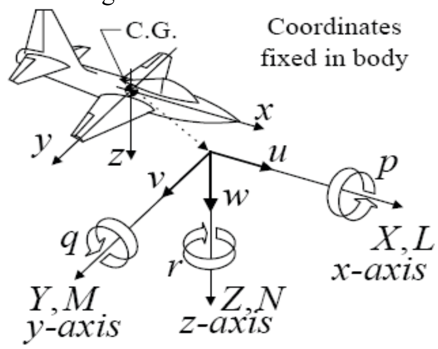


Fig. 9. Axis Definition

$$\dot{V}_T = A_{x_w}$$

$$\dot{\alpha} = q - (p\cos\alpha + r\sin\alpha)\tan\beta + \frac{A_{zw}}{\cos\beta V_T}$$

$$\dot{\beta} = -(r\cos\alpha - p\sin\alpha) + \frac{A_{Yw}}{V_T}$$

$$\dot{p} = \frac{I_Y - I_Z}{I_X}qr + \frac{L}{I_X}$$

$$\dot{q} = \frac{I_z - I_x}{I_y}rp + \frac{M}{I_y}$$

$$\dot{r} = \frac{I_x - I_y}{I_z}pq + \frac{N}{I_z}$$

$$\dot{\psi} = (r\cos\phi - q\sin\phi)/\cos\theta$$

$$\dot{\theta} = q\cos\phi - r\sin\phi$$

$$\dot{\phi} = p + \dot{\psi}\sin\theta$$

V_T, α, and β stands for total velocity, attack angle and sideslip angle, p, q, r are vehicle angular rate of body axis, Ψ, θ, Φ are Euler angles, I_x, I_y, I_z are angular momentum of body axis, all the cross coupled terms (IXZ, I_{yz}, I_{xy}) are assumed to be 0. Aerodynamic parameters A_{Xw}, A_{Yw}, A_{Zw}, L, M, and N are drag force and side force.

$$A_{xw} = \left(\left\{\frac{T}{m} + g_x\right\}\cos\alpha + g_z\sin\alpha - \frac{\tilde{q}S}{m}C_D\right)\cos\beta + \left(g_y + \frac{\tilde{q}S}{m}C_Y\right)\sin\beta$$

$$A_{Yw} = -\left(\left\{\frac{T}{m} + g_x\right\}\cos\alpha + g_z\sin\alpha - \frac{\tilde{q}S}{m}C_D\right)\sin\beta + \left(g_y + \frac{\tilde{q}S}{m}C_Y\right)\cos\beta$$

$$A_{Zw} = -\left\{\frac{T}{m} + g_x\right\}\sin\alpha + g_x\cos\alpha - \frac{\tilde{q}S}{m}C_L$$

$$L = S\tilde{q}b\left(C_l\cos\alpha - C_n\sin\alpha\right)$$

$$M = S\tilde{q}bC_m$$

$$N = S\tilde{q}b\left(C_l\sin\alpha - C_n\cos\alpha\right)$$

Where C_D ,C_L ,Cy ,C_l ,Cm, C_n are non-dimensional force and moment coefficients, which are function of aircraft state vector x and control input of wing deflection δ_e ,δ_a ,δ_r ; g_x, g_y, g_z are gravity components in body axis, ，S is wing surface area, *c* is wingspan and *b* is mean aerodynamic cord.

The linearized lateral/directional model is derived as :

$$\begin{bmatrix}\dot{\beta}\\ \dot{p}\\ \dot{r}\\ \dot{\phi}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}\frac{Y_\beta}{V_{T_0}} & \frac{Y_p}{V_{T_0}} & -1+\frac{Y_p}{V_{T_0}} & \frac{g}{V_{T_0}} & 0\\ L_\beta & L_P & L_r & 0 & 0\\ N_\beta & N_p & N_r & 0 & 0\\ 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\end{bmatrix}\begin{bmatrix}\beta\\ p\\ r\\ \phi\\ \psi\end{bmatrix} + \begin{bmatrix}0 & \frac{Y_{\delta_r}}{V_{T_0}}\\ L_{\delta_a} & L_{\delta_r}\\ N_{\delta_a} & N_{\delta_r}\\ 0 & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\delta_a\\ \delta_r\end{bmatrix}$$

δ_a, δ_r are aileron and rudder control inputs respectively. L_P, L_r ,r ,Y_β,Y_P,Y_R, N_β,N_P,N_r $Y_{\delta r}$, $Y_{\delta a}$, $N_{\delta a}$, $N_{\delta r}$ are dimensional stability and control derivatives of the lateral/directional dynamics.

B. Dynamic System Simulation

The software used in this project are listed as follows:[6]

1. LabVIEW 8.5,
2. LabVIEW Real-Time 8.5,
3. LabVIEW Simulation Interface Toolkit 3.0.2,
4. Labview Simulation 8.5,
5. LabVIEW Conrol Toolkit 802,
6. Matlab Simulink 2008.
7. NI FPGA 2.5.2

Two types of simulation; one using item 1, 2, 4, 5 and the other one using 1, 2, 3, 6 are studied for comparison. The first one is using item 4 to build the model and deploy the model through item 1,2. The next one is using LabVIEW 8.2 Real-Time ETS module to transform PC as a real-time simulator. PXI 8186 module acts as target PC. In addition, NI-VISA module and real-time module will be the interface program over the TCP/IP standard network connection. All the initial setup and adjustment can be adjusted through MAX system management program. In addition, using simulation Interface toolkit module to convert the model built in Matlab Simulink to LabVIEW. The simulation interface toolkit module in the host PC will add nidll.tlc program in Real Time (RTW) to compile the Simulink model to DLL format and downloaded to PXI PC with embedded ETS system. The Simulation interface toolkit is able to do the connection between LabVIEW graphic I/O with ETS system Simulink model to do the simulation. In addition, the interface can be set up by using RT-FIFO technique to build the variable buffer size and priority of time loop. Both type studies show good agreement.

The controller performance properties as steady state error, rise time, max overshoot are examined here.

C. HIL— Integration Test between the PXI and Flight Computer

The host PXI system which is responsible for distributing the received data to the onboard flight computer system (Stamp microprocessor) through NI realtime network to control elevator, aileron, and rudder servo. The avionics data from simulation is connected to the HIL system to emulate the actual measurement. The outputs of the sensors are directly feed into the PXI 7388R DAQ module, and the hardware compatible I/O mode will connect interface I/O(AI, AO,DI,DO, PWM, etc) to do the interactive simulation between the physical vehicle and window based simulator, as shown as Fig. 10.

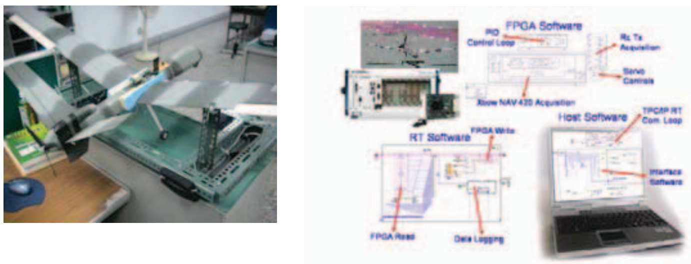
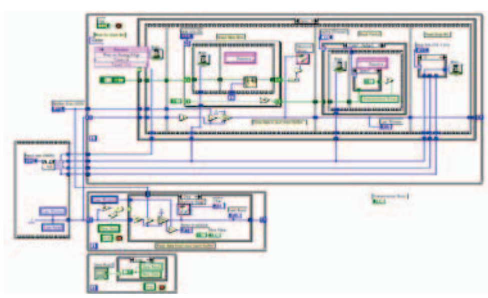


Fig. 10. Integrated HIL system test system

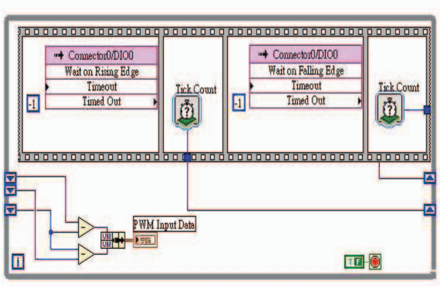
D. *FPGA Based Interface Hardware interfaces [7,8]*

Since the traditional communicational interval, delay and emulated measurement content are operated by software, which is easily delayed by the operating system. To further improve the precision of simulation timing constrains, and in consideration of both flexible and convenient. FPGA based hardware is used to define different type interface by software design. The effects of sensor and actuator packet delay as well as communication constraints in the navigation and controls can be easily investigated.

Currently, FPGA based analog I/O, Digital I/O and RS232, SPI and PWM interface are used for HIL interface. Fig. 11 demonstrates the RS232 and PWM interface signal simulation.



(a) FPGA based RS232 interface



(b) FPGA based PWM interface

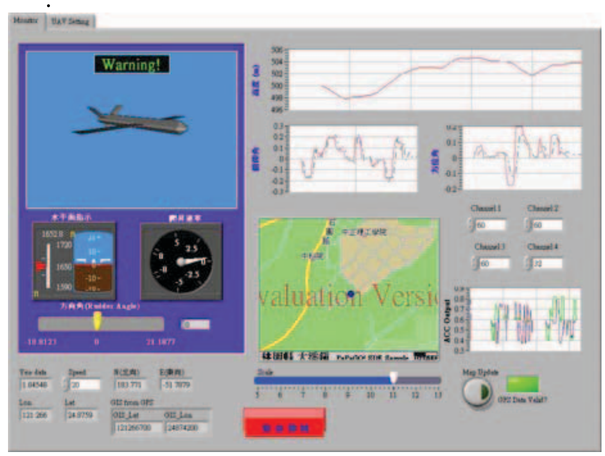
Fig. 11. FPGA based signal interface

E. *Graphical Human Interface Integration Test*

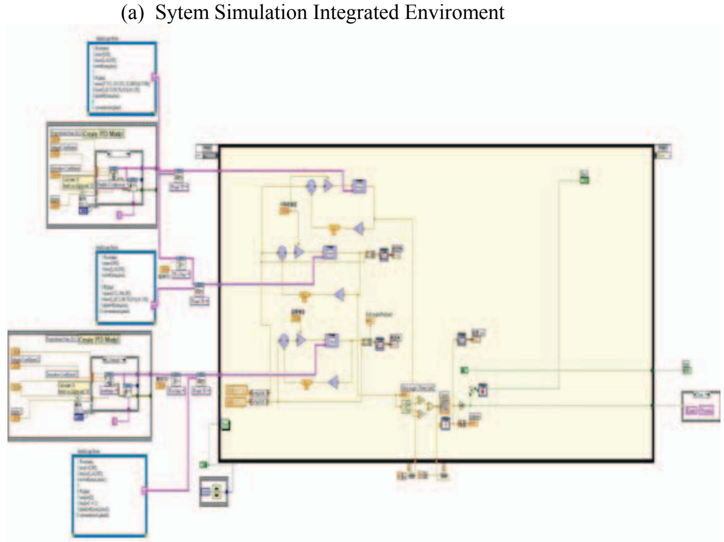
The graphical environment is built according to the testing procedure, specification. To closely demonstrate the attitudes of vehicle, 3D picture control software from NI is used to connect a vehicle model built by SolidWork with the simulation toolkit as sown in the integrated operating environment Fig.11. Besides, GPS data is down linked from simulation module to commercial GPS navigation software to give live GPS information. Also, the joystick input, control parameters, and vehicle response are clearly shown in the picture frame; the operator can compare the difference between the inputs with the simulated vehicle response.

Fig. 12(b) is the simulation program of vehicle motion analysis and control. Fig.12(c)(d) is the program of sensor and joystick, which is also acting as the connection between the flight computer and simulator. Fig.12(e) is computer graphic 3D modeling and GPS data simulation. The program needs to load a vehicle built by VRML and embedded into the picture control software to interact with the joystick commands

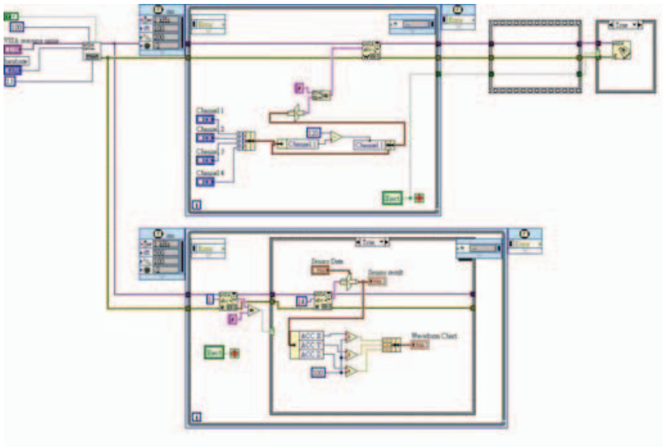
.



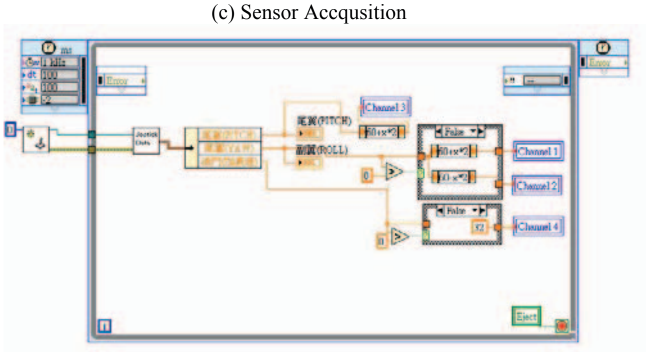
(a) Sytem Simulation Integrated Enviroment



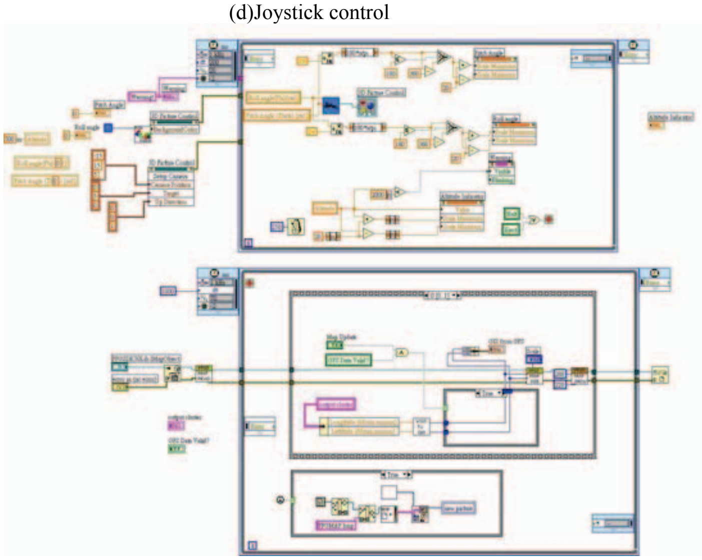
(b) Vehicle Simulation



(c) Sensor Accqusition



(d)Joystick control



(e) VRML 3D Graphical Simulation and GPS Data Simulation
Fig. 12. Hardware in the Loop Test Program

V. CONCLUSION

In this paper, we have presented a FPGA Based Hardware in the Loop Test Platform of Small Size UAV HIL test. The system is focusing on visualizing the control effect of vehicle attitude change. The vehicle dynamical model was built bases on the physical plane. By incorporating the microprocessor, the flight computer was built and integrated with the HIL system. The FPGA based interface configuration is able to give more flexible and precision approach to the real system. The most advantage of the above system is to improve flight test and save test cost.

ACKNOWLEDGMENT

The authors acknowledges the supported by National Science Foundation under the Industry-University-Research grant NSC 97-2622-E-606 -002 -CC3..

REFERENCES

[1] Grega,Wojciech, “Hardware-in-the-Loop Simulation and its Application in Control Eeducation,” November 10 - 13, 1999 San Juan, Puerto Rico 29th ASEE/IEEE Frontiers in Education Conference.

[2] Kendall,I.R., Jones, R.P.,” An Investigation into the Use of Hardware-in-the-Loop Simulation Testing for Automotive Electronic Control Systems,” Control Engineering Practice, 1999.

[3] Hanselman, H., “Hardware-in-the- Loop Simulation for Development and Test of Electronic Control Units,” Proceedings of the Real-Time System’93 Conference, January 13-15, 1993, Paris, France.

[4] Sudiyanto,T. A., Budiyono and H.Y. Sutarto, “Hardware In-the-Loop Simulation for Control System Designs of Model Helicopter,” Aerospace Indonesia Meeting, Bandung, Indonesia 27 July 2005.

[5] Jang, J., Tomlin, Claire,“Autopilot Design for the Standford DragonFly UAV:Validattion through Hardware-in-the Loop Simulation,” AIAA GNC Conference, Montreal, Canda, Aug, 2001.

[6] Control and Simulation User Manual, National Instrument , Aug., 2005.

[7] Http:/www.ni.com/Using High-Level Protrotyping Hardware and Software in Machine Control Applications.

[8] Visser, Peter M., Marcel, Groothuis, A. and Jan Broenink, F., “FPGAs as versatile configurable I/O devices in Hardware-in-the-Loop Simulation,” RTS 2004 Work-In-Progress Proceedings, Lisbon, 5-8 December 2004).