

Using Summit Reset Controllers in Multi-Microcontroller Systems

Introduction

In many systems today, the control algorithm is divided among more than one microcontroller. This distributed processing approach provides greater overall processing bandwidth as well as a modularization of the design for various versions of the end product. In these systems care must be taken to insure that all microcontrollers in the system are properly reset during power-up, power-down and brownout conditions; especially in systems which utilize a nonvolatile memory such as a serial E²PROM. In addition to insuring that the individual microcontrollers are reset properly, the microcontrollers must also be synchronized to insure proper operation. This application note will outline the utilization of the SUMMIT S24042 Precision Reset Controller and nonvolatile memory in a dual MCU television chassis.

Requirements

In many television chassis today, a separate MCU is used to control the on-screen display for control of the television parameters through the use of user-friendly menus. This MCU is often referred to as the On Screen Display processor or OSD MCU.

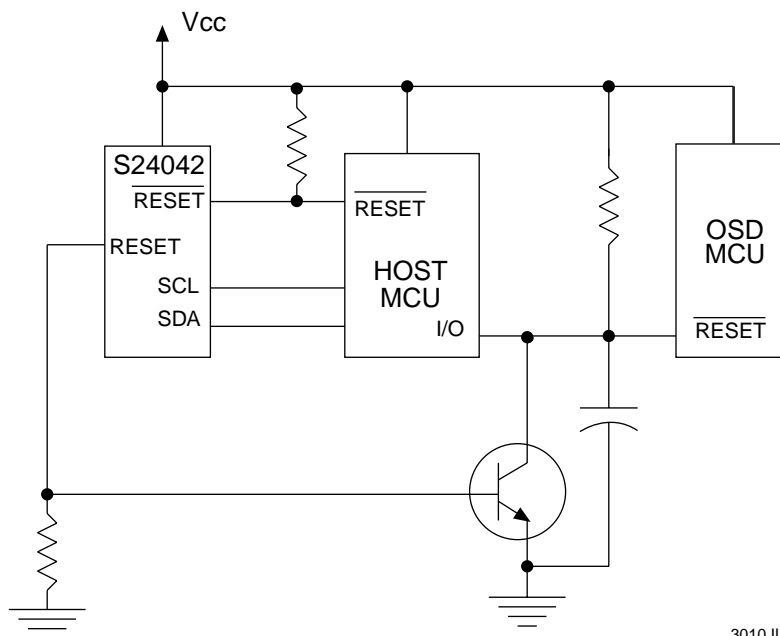
The main MCU in the television controls the operation of the tuner, display and user interface and is often referred to as the Master MCU, since it can control all other circuitry in the television including the OSD MCU. The Master MCU uses an I²C serial E²PROM to store various user

and factory programmed setup parameters including those controlled through the menus displayed by the OSD MCU.

During power up, both MCUs must be reset and it is often desirable that the Master CPU is released from the reset after the supply voltages are stable, but before the OSD MCU is released from reset. In addition to this, the Master MCU must be able to generate a reset condition to the OSD MCU via software control, and once this condition is triggered, the OSD MCU must be held in the reset state for the proscribed interval. During power-down, both MCUs should be held in reset as soon as V_{CC} drops below the operational voltage of the chassis; and, during a brown-out situation, both CPUs will need to be reset and re-synchronized. During a brown-out situation, it is imperative that the V_{CC} sensor used to generate the RESET signals to the MCUs is the same, in order to prevent a situation where one MCU might get reset while the other does not.

Implementation

Figure 1 [on the following page] shows a solution to this multi-MCU design. The S24042 precision reset controller is connected directly to the RESET input of the master microcontroller. The I²C interface from the S24042 is also tied to two I/O ports of the master microcontroller. The S24042 RESET and RESET outputs will be active on power-up as soon as V_{CC} is above 1.0 volts.



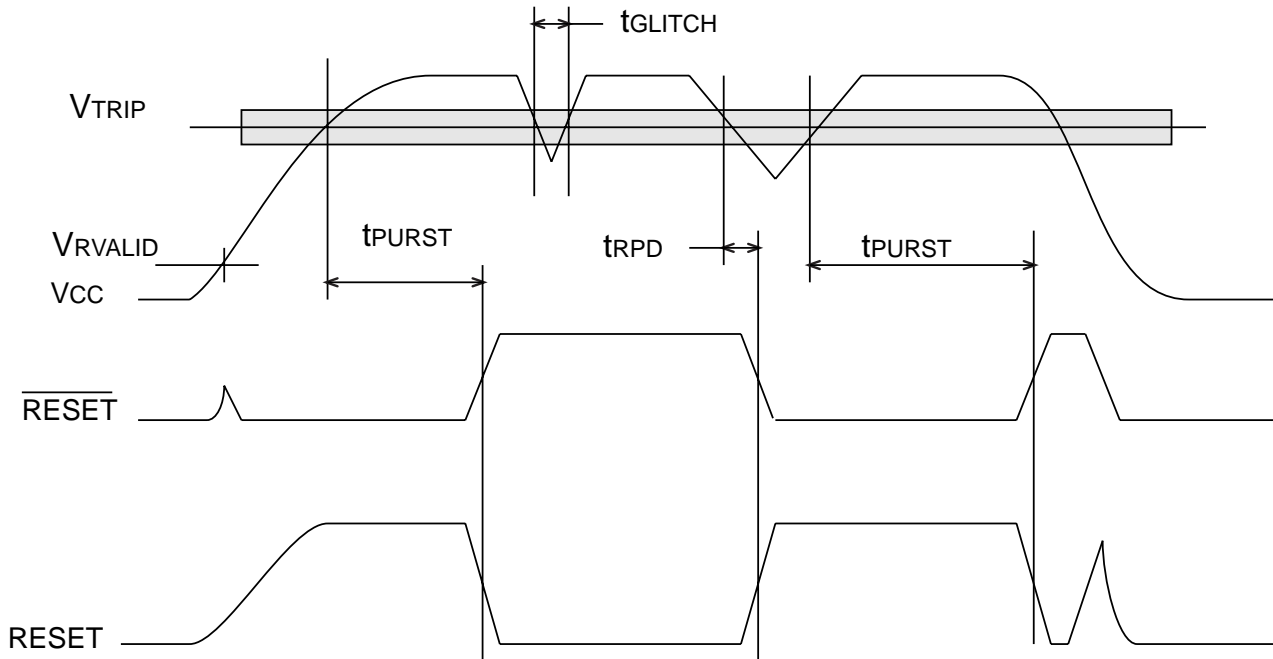
3010 ILL1.0

Figure 1. Multi-MCU Reset Circuit Solution



The S24042 will hold RESET low until V_{CC} reaches the reset controller trip point of 4.4 volts. An internal timer then begins to time the reset interval of t_{PURST} which is typically 120 msec. After this time has elapsed, the Master MCU

will be released from the Reset state. IF V_{CC} drops below the reset trip point for longer than 50 nsec, the S24042 will again assert the reset signal. Figure 2 shows the output of the S24042 Reset Controller at the various supply conditions.



3010 ILL2.0

Figure 2. S24042 RESET Output Timing

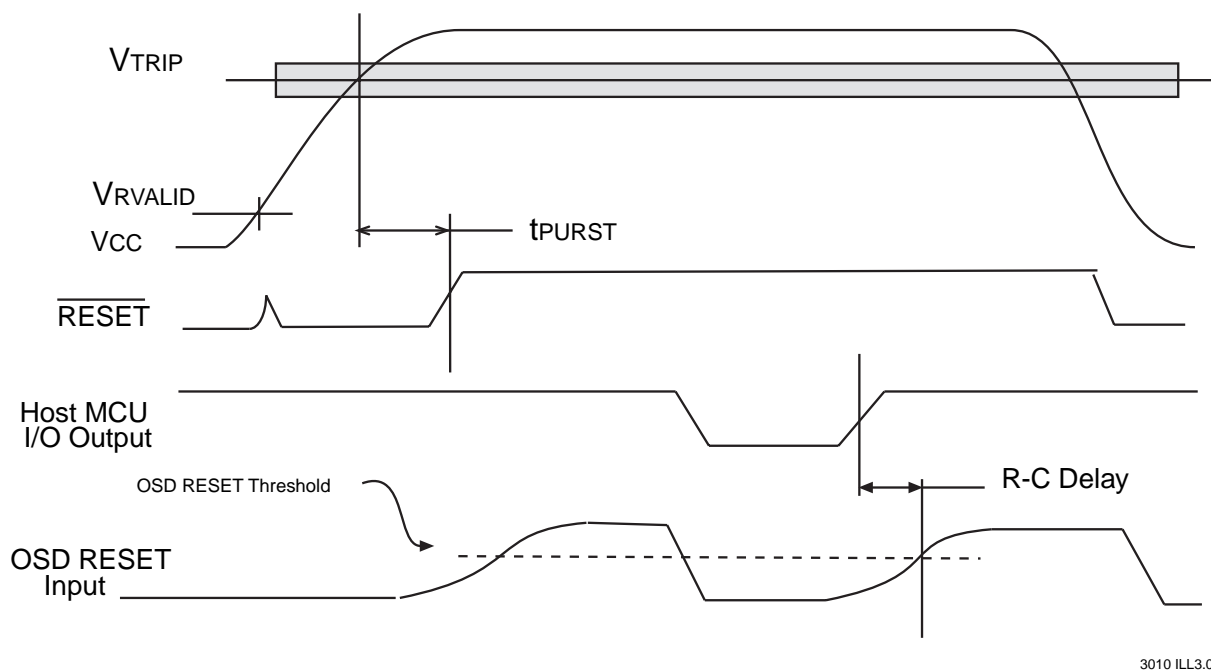


The slave MCU's reset signal is further delayed by the capacitor on the reset line. This capacitor and its related pull-up resistor should be chosen to insure that the RC time constant will give the adequate reset pulse time for the slave MCU. Reset signals for the slave which are generated by the S24042 reset controller will have a duration of t_{PURST} + the rise time of the RC network.

The Host MCU resets the slave MCU by driving the I/O line low. Once the I/O line is released, the capacitor will time out a reset interval based upon the RC time constant. Care should be taken to insure that the I/O controlling the

reset line is driven low long enough to discharge the capacitor, and in no case should this I/O should be configured to output a High logic level. Rather, to de-assert the reset, this I/O should be left in high impedance state or configured as an input, see Figure 3 below.

During a brown-out situation, the S24042 will drive the master reset line low for 120 msec to insure that the timing capacitor is fully discharged. Once the master reset line is driven high, the timing capacitor will generate the proper reset timing for the slave MCU.



3010 ILL3.0

Figure 3. Delay Reset Timing to the OSD MCU

