

SpaceWire を用いた CCD 駆動実験

梅津里香、村上弘志、北本俊二、山本育（立教大学）

1 概要

SpaceWire は、人工衛星でのデータ通信を行うための通信インターフェイスおよび通信プロトコルの仕様である。本研究では、SpaceWire のインターフェイスを搭載している SpaceWire DIO ボード、SpaceWire ADC/DAC ボードを用いて CCD の駆動と読み出しを行う Module 開発を進めている。作成した Module を使って実際に CCD の駆動を行い、模擬波形による読み出しと画像化に成功した。

2 Introduction

SpaceWire は、人工衛星でのデータ通信を行うための通信インターフェイスおよび通信プロトコルの仕様である。人工衛星の通信インターフェイスは、その人工衛星ごとに開発されることが多かった。しかし、人工衛星の製作コストの削減、製作期間の短縮などの理由により、様々な利点を持つ SpaceWire による通信インターフェイスの統一化、標準化作業が進められている。本研究では、SpaceWire のインターフェイスを搭載している SpaceWire DIO ボードとさらに ADC を搭載している SpaceWire ADC/DAC ボードを用いて、CCD の駆動を行う Module を開発する。宇宙観測によく用いられる CCD の SpaceWire での駆動は、上述のような人工衛星間通信の標準化に沿う物である。また、FPGA を用いて自由度の高いシステムを組み、様々な CCD の駆動パターンに適用できるようにすれば、CCD 読み出し方法の柔軟な変更や改良ができるであろう。

3 開発の全体像

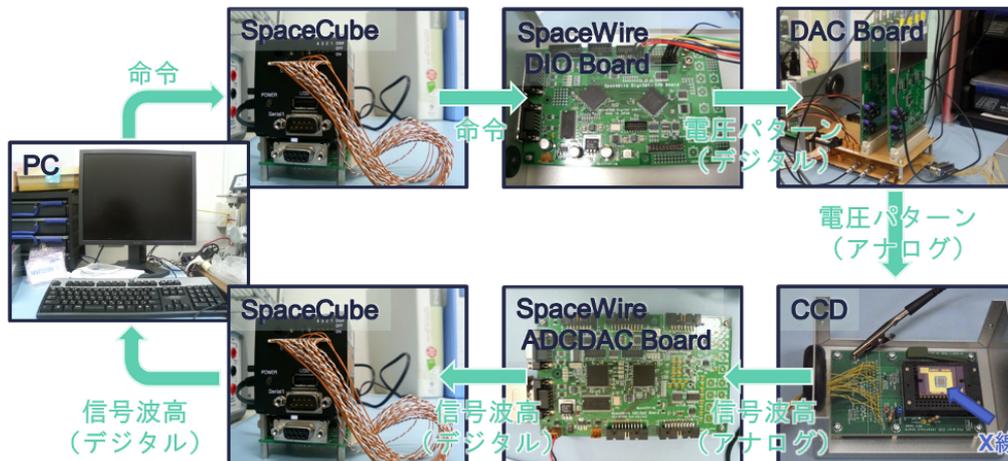


図 1 開発の全体像

この実験の開発の全体像を図 1 で示す。

まず親機から SpaceWire のインターフェイスをもっているコンピュータである SpaceCube を通して、SpaceWire DIO ボードの SpaceWire FPGA へと命令を出す。CCD を駆動するためには制御信号 (Clock) が必要となるため、電圧パターンを SDRAM に書き込み、そのパターンを読み込むための Module を User FPGA に書き込む。そして電圧パターンをデジタル-アナログ変換 (DA 変換) して CCD に送り、駆動する。

CCD から得られたデータはアナログ信号であるので、これを SpaceWire ADC/DAC ボードを用いてアナログ-デ

デジタル変換（AD変換）を行う。AD変換を行うためのModuleや、SDRAMへの書き込みを行うModuleなどは前もってUser FPGAに書き込んでおく。CCDから得られた電圧データをModuleで信号波高に計算し直し、SDRAMに書き込む。これをSpaceCubeを通して親機に送り、得られた信号波高をPNM方式を使うことにより画像とする。

4 開発状況

4.1 CCD駆動のための開発

駆動側の全体像が図2である。

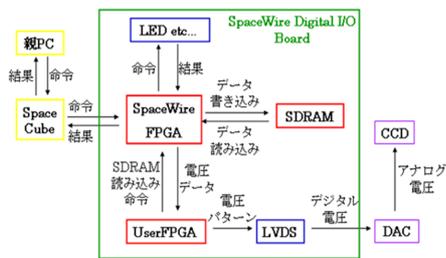


図2 駆動用開発像

CCDを駆動するためには複数の制御信号が必要となるため、CCD駆動のための電圧パターンのModuleを開発した。このModuleは様々なCCDに対応できるように汎用性を持たせてある。また、SDRAMに書き込まれたcsvファイルから電圧値を読み込み、各種Clockに対応した電圧値データを出力するModuleも作成した。電圧値データを送る際は、シリアルデータでDACに送らなければならないため、パラレル-シリアル変換を行うModuleも作成した。縦転送と横転送のための電圧パターンをDACボードから出力したものが図3、図4である。

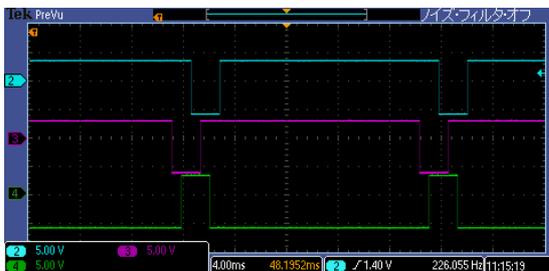


図3 縦転送

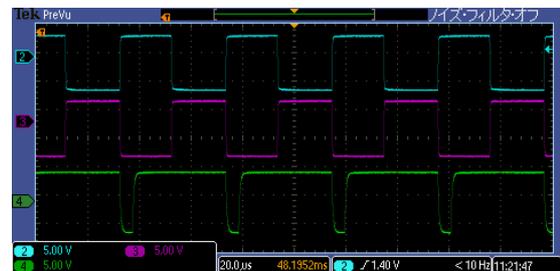


図4 横転送

4.2 CCD読み出しのための開発

読み出し側の全体像が図5である。

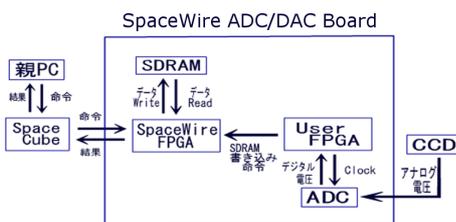


図5 読み出し用開発像

CCDから信号波高を受け取るためのModuleを開発した。CCD模擬波形のFloating LevelとSignal Levelのタイミングにあわせて信号波高をAD変換し、それぞれパラレル変換したものをういて差をとることので一つのCCDピクセル信号の出力と考える。これをSpaceCubeを通してピクセルの信号として取り込み、PNM形式で画像とする。図6のようにCCDの出力を模擬した波形を画像化したものが図7である。横方向にOverclockingをして得られる仮想イメージ領域をHOC領域(Horizontal Over-Clocked region)と呼び、図6ではSignal Levelが低くなっている部分がこれに相当する。図7では、右端の黒い部分がHOCである。図8の実際のCCDで得られた画像と同じように確認することができた。

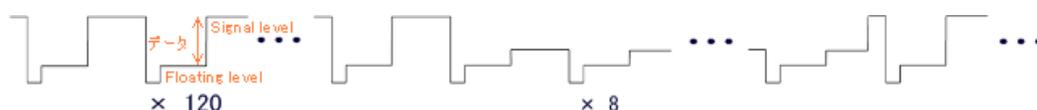


図6 CCD模擬波形



図 7 CCD 模擬波形画像

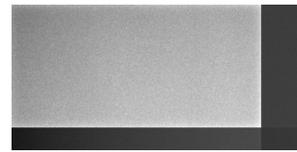


図 8 実際の CCD で得られた画像

4.3 CCD の駆動

今回は SpaeCube を使わず、手動スイッチで駆動開始のタイミングを与えている。蓄積時間やデータ取得枚数も Module で与えている。図 9 から図 12 が CCD の出力をオシロスコープで確認したものである。図 9 が CCD に光を当てている時の横転送 1 行分、図 10 がその 1 ピクセル分を確認できるスケールである。図 11 が CCD に光を当てていない時の横転送 1 行分、図 12 がその 1 ピクセル分を確認できるスケールである。水色の波形が CCD の出力を表している。緑色の波形はデータ取得のためのタイミングである。

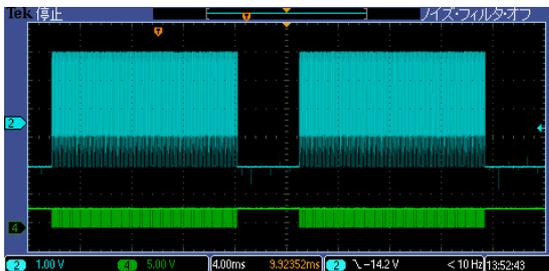


図 9 CCD 横転送 1 行 (光あり)



図 10 CCD 横転送 1pix (光あり)



図 11 CCD 横転送 1 行 (光なし)

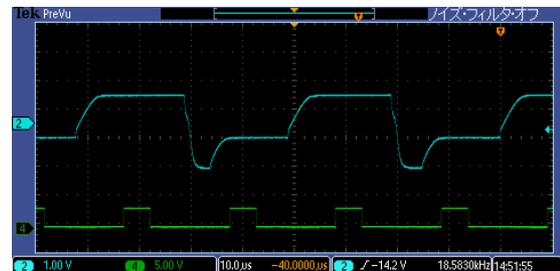


図 12 CCD 横転送 1pix (光なし)

5 まとめと今後の課題

開発した Module で CCD を駆動することができたが、まだ転送がしっかりできていないため、確実に行われるようにしていく。読み出し側も模擬波形では画像化することができているため、転送の問題が解決したら、実際の CCD を用いた画像化を行う。開発の全体像のように CCD の駆動、読み出しを行えるようにしていく。また、冷却、真空中で X 線も利用する。最終的には、この装置を用いて CCD 読み出し方法の柔軟な変更や改良を目指す。

参考文献

- [1] SpaceWire/SpaceCube Tutorial
- [2] SpaceWire/RMAP Library