

開源 LIDAR 原型製作平台

本文將探討 ADI 新推出且擁有廣泛市場的 LIDAR 原型製作平台，以及它如何透過提供完整的硬體和軟體解決方案，讓使用者能夠建立其演算法和自訂硬體解決方案的原型，從而協助客戶縮短產品開發時間；詳細介紹模組化硬體設計，包括光接收和發送訊號鏈、FPGA 介面，以及用於長距離感測的光學元件；介紹系統磁碟分割決策，以突顯良好的系統設計、介面定義和合適的模組化分級的重要性；描述開源 LIDAR 軟體堆疊的元件和平台客製的 API，顯示客戶在產品開發期間如何受益，以及如何將這些產品整合到其最終的解決方案中。

■作者：István Csomortáni / ADI FPGA 設計工程師
Dragos Bogdan/ 軟體發展工程經理
Cristian Orian/ 系統設計工程師
Andrei Cozma 工程經理

簡介

隨著自動駕駛汽車和機器人從想像逐漸變為現實，汽車和工業客戶開始尋求新的環境感知解決方案，力圖讓這些機器能夠自動導航。LIDAR 是該領域中發展最快的技術之一，隨著它越來越成熟和可靠，其應用範圍也變得更加廣泛，帶來了巨大的市場機會。許多新創企業和知名感測器公司都致力於開發更加精準、功耗低、尺寸小，且更加經濟高效的 LIDAR 感測器，但在設計系統硬體、建置軟體基礎設施以及系統中的所有元件通訊時，他們都遇到了同樣的挑戰。正是在這些區域，ADI 能夠透過軟體參考設計和開源軟體堆疊提供價值，令客戶能夠輕鬆將 ADI LIDAR 產品系列、軟體模組和 HDL IP 整合到其產品和 IC 中，從而縮短上市時間。

系統架構

客戶在開發自己的 LIDAR 感測器時，系統設計中些許差異化包括：接收和發送光學元件、雷射器的數量和方向、鐳射發射模式、雷射光束控制，以及光接收元件的數量。但是，不管做出什麼選擇，在接收訊號鏈和雷射器驅動訊號要求方面，所有感測器都高度相似。基於這些假設，ADI 設計出模組化 LIDAR 原型製作平台 AD-FMCLIDAR1-EBZ，以期讓客戶能夠使用他們自己的硬體輕鬆配置或更換

元件；該平台根據特定的應用要求設計，但仍可以用於整個系統。該系統可以分為三個不同的電路板，每個都配備標準化的數位和類比介面：

■資料擷取 (DAQ) 電路板，包含高速 JESD204B ADC、對應的時脈和電源。此電路板上有一個符合 FMC 要求的介面，可以連接至使用者優選的 FPGA 開發板。它充當系統的基板，透過用於在這些板和 FPGA 之間路由控制和回饋訊號的數位連接器，以及用於傳輸類比訊號的同軸電纜，將另外兩個板連接至這個板。

■包含雪崩光偵測器 (APD) 的光感測器和整個訊號鏈的類比前端 (AFE) 電路板，訊號鏈用於調諧 APD 輸出訊號，以便能送入 DAQ 板上的 ADC。

■包含雷射器和驅動電路的雷射器板。

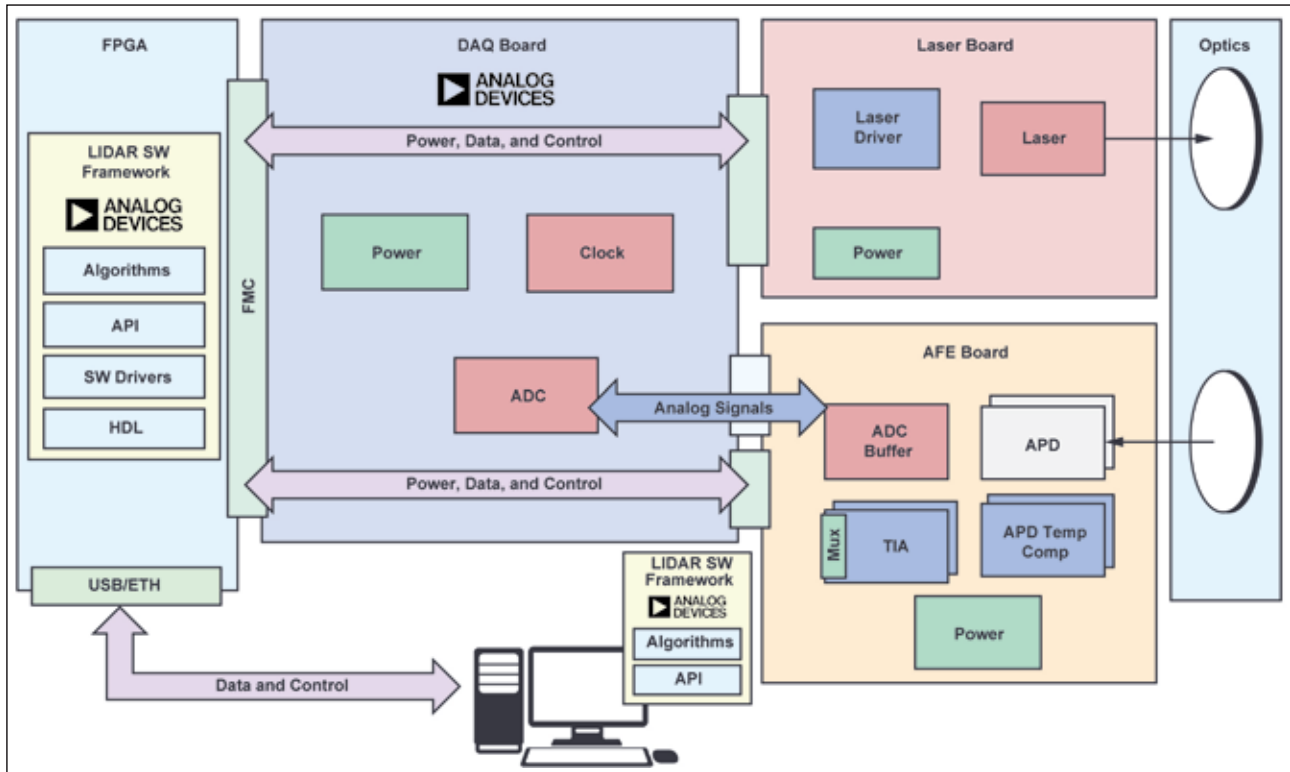
和以往一樣，在系統設計中，模組化表示彈性，但它也具有一些缺點，比如複雜性增加、性能下降和成本增加，在決定系統磁碟分割時必須全面評估這些缺點。在這種情況下，系統被分成三個板，原因如下：

■無論使用哪種類比前端，選擇哪種雷射器解決方案，ADC 和時脈很可能保持不變。

■類比前端硬體設計和尺寸根據所選的 APD、整體的系統接收靈敏度，以及選擇的光學元件而變化。

■雷射器板設計和尺寸根據所選的照明解決方案和

圖 1: LIDAR 平台系統設計。



光學元件而變化。

- 對於接收器和發射器的位置和方向，系統提供很高的彈性，以便它們彼此對應或和其他目標對應，因此使用柔性電纜來傳輸數位訊號，使用同軸電纜來傳輸兩個電路板之間的類比訊號。
- 包含硬體設計的軟體堆疊以分層方法為基礎，以少數幾個層級區分為適用於特定作業系統的驅動和介面、系統特定的 API 和應用層。這使得堆疊的上層可以保持不變，無論軟體是在嵌入式目標上運行，還是在透過網路或 USB 連接與系統通訊的 PC 上運行。如圖 2 所示，在不同的產品開發階段，這一點非常有用，因為這表示將系統連接至 PC 以簡化開發時，在原型製作期間開發的同

樣的應用軟體可輕鬆部署到嵌入式系統中，甚至無需觸碰低層介面。

硬體設計

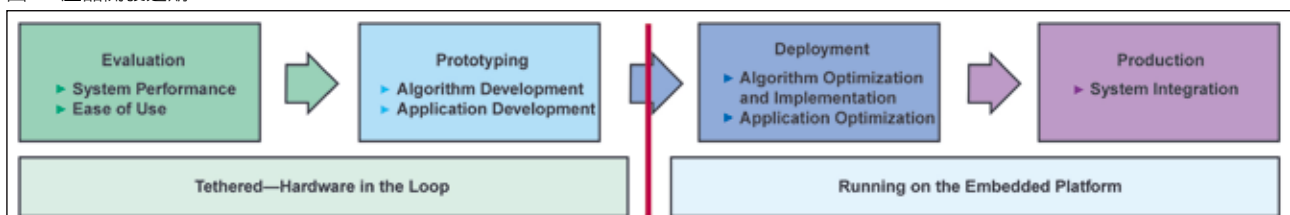
LIDAR 感測器透過測量光脈衝到達目標並返回的時間來計算與目標之間的距離。測量時間時，以 ADC 採樣資料為增量，這裡 ADC 採樣速率決定了系統對接收的光脈衝採樣時的解析度。公式 1 顯示如何根據 ADC 採樣速率計算距離。

$$d = N \times \frac{L_s}{2 \times f_s} \quad (1)$$

其中：

L_s 為光的速度， 3×10^8 m/s

圖 2: 產品開發週期



f_s 為 ADC 採樣速率

N 為光脈衝產生至返回接收期間 ADC 樣本的數量

假設系統使用 AD9094JESD204B 四通道 ADC 的 1 GHz 採樣速率，那麼每個樣本結果相當於 15cm 距離。因此，系統中不能存在採樣不確定性，因為任何樣本不確定性都可能導致巨大的距離測量誤差。傳統上，LIDAR 系統以並行 ADC 為基礎，這種 ADC 本身提供零採樣不確定性。隨著接收通道的數量不斷增加，功率和 PCB 尺寸的要求越來越嚴格，這些 ADC 類型不能良好地擴展。另一個選項是運用具備高速串行輸出的 ADC，例如 JESD204B，以解決並行 ADC 存在的問題。這種選項的資料介面複雜度更高，因此難以實現零採樣不確定性。

LIDAR DAQ 板提供了解決這些挑戰的方案，透過展示為在 Subclass 1 模式下運行的 JESD204B 資料擷取系統的電源、時脈和資料介面設計來確保確定性延遲，以實現零採樣不確定性，同時利用 JESD204B 介面提供的所有優勢，令時脈方案的功耗達到最低。要在 Subclass 1 模式下運行 JESD204B，系統總共要用到 5 個時脈：

- ADC 採樣時脈：驅動 ADC 訊號採樣過程。
- ADC 和 FPGA SYSREF：源同步、高壓擺率時序解析度訊號，用於重置元件時脈分頻器，以確保獲得確定性的延遲。
- FPGA 全域時脈（也稱為核心時脈或元件時脈）：驅動 JESD204B PHY 層和 FPGA 邏輯的輸出。
- FPGA 參考時脈：產生 JESD204B 收發器所需的

PHY 層內部時脈；需要等於，或是元件時脈的整數倍。

所有時脈都由一個 AD9528 JESD204B 時脈產生器產生，因此可以確保它們彼此都同步。圖 3 顯示了時脈方案，以及與 FPGA 的資料介面。

AFE 板接收光學反射訊號，將其轉化成電子訊號，然後傳輸給 DAQ 板上的 ADC。這個板可能是整個設計中靈敏度較高的部分，因為它混合訊號調節電路（使用 16 通道 APD 陣列產生的微安電流訊號），將光學訊號轉化成電子訊號，並採用為同樣的 APD 供電所需的 -120 V 至 -300 V 大電壓電源。16 個電流輸出被饋送至 4 個低雜訊四通道互阻增益放大器 (TIA) LTC6561，具有一個內部 4 合 1 複用器，用於選擇之後向其中一個 ADC 輸入端饋送的輸出通道。要特別注意 TIA 的輸入部分，以實現所需的訊號完整度和通道隔離等級，使得 APD 產生的極低電流訊號中不會摻雜更多雜訊，從而最大化系統的 SNR 和物件檢測率。

AFE 板的設計顯示，要實現較高訊號品質，較好的方法是讓 APD 和 TIA 之間的線路長度盡可能短，並在 TIA 輸入之間增加橢圓孔，以最大化通道間隔離；此外，在部署訊號調節電路時，要保證該電路不會干擾板上的其他電源電路。另一項重要特性是能夠測量 APD 的溫度，以補償 APD 訊號輸出的變化，這種變化是因為在正常運行期間 APD 溫度上升導致的。提供幾個旋鈕來控制訊號鏈的偏置和 APD 偏置，這些偏置轉化成 APD 靈敏度，從而最大化 ADC 輸入範圍，以實現最大 SNR。圖 4 顯示

圖 3: DAQ 板時脈和資料路徑。

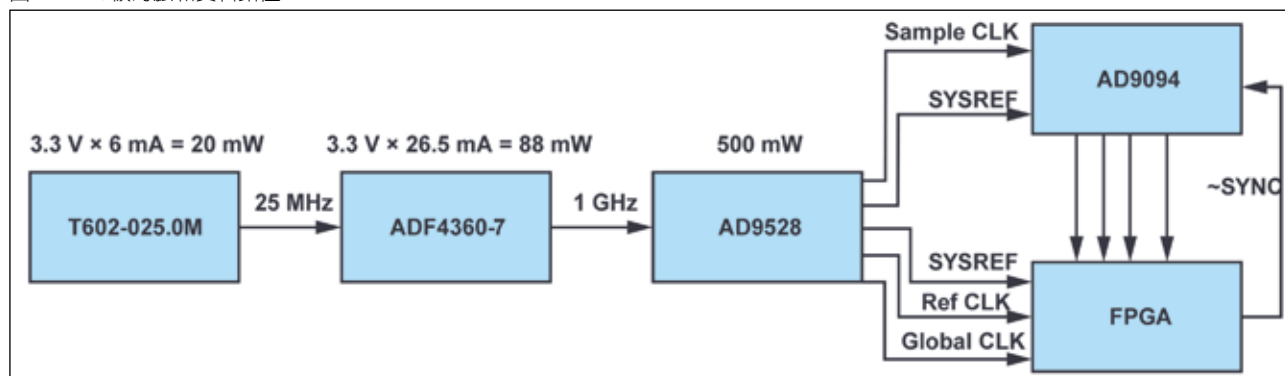
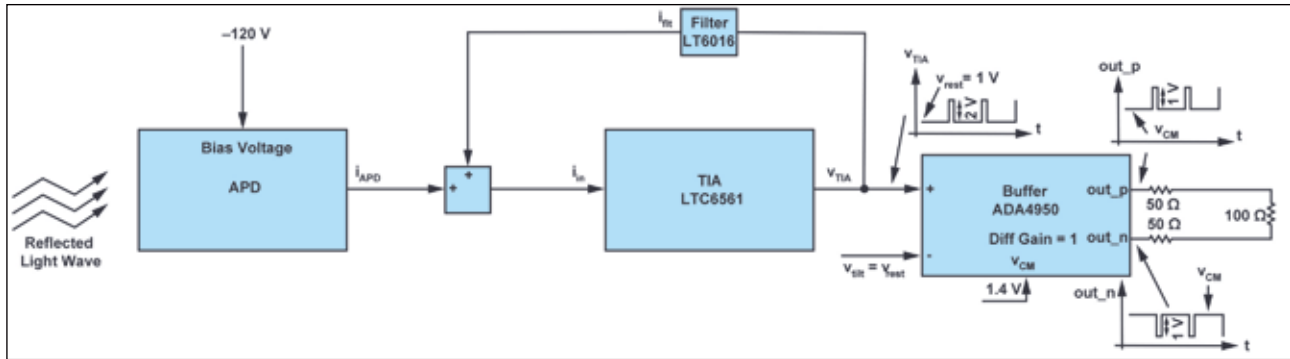


圖 4: AFE 板訊號鏈。



了 AFE 板訊號鏈的框圖。

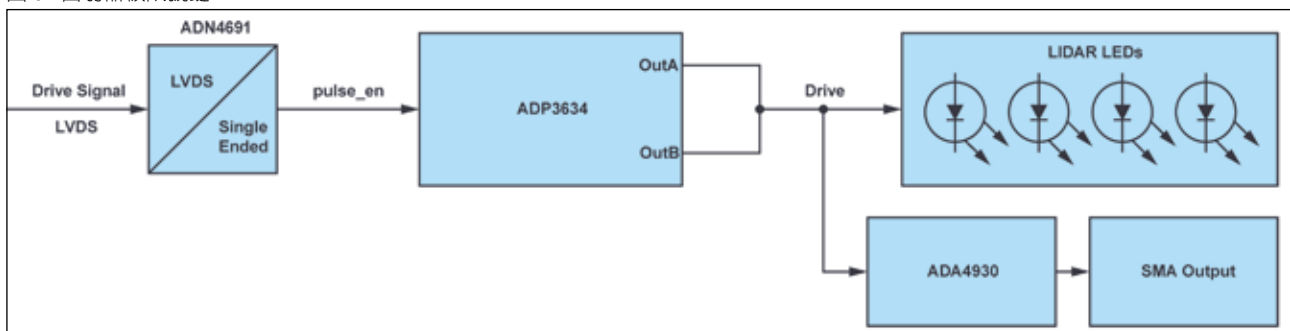
雷射器板產生波長為 905 nm 的光學脈衝。它使用四個雷射器，這些雷射器同時驅動，以增加光束強度，實現更長的測量距離。此雷射器使用由 FPGA 載波板產生的具備可編程脈寬和頻率的 PWM 訊號來控制。這些訊號在 FPGA 上生成，以 LVDS 從 FPGA 傳輸至雷射器板，經過 DAQ 板以及連接 DAQ 和雷射器板的排線期間，不易受到雜訊影響。驅動訊號可以返回至其中一個 ADC 通道，以獲得飛行時間參考。採用外部電源為雷射器供電。其設計符合國際標準 IEC 60825-1:2014 和 IEC 60825-1:2007 中關於 Class 1 級雷射器產品的要求。

AFE 和雷射器板都需要光學元件，以實現長距離測量。事實證明，該系統可在 60m 範圍內測量，使用快速軸準直器¹，幫助鐳射二極體將垂直 FoV 縮小到 1°，同時在保持水平視場不變的情況下，在接收側放置一個非球面透鏡。

HDL 參考設計

HDL 設計包含連接硬體的主要介面，其邏輯電

圖 5: 雷射器板訊號鏈。



路實現了將來自 JESD 連結的資料傳輸至系統記憶體，驅動雷射器，同步接收器和發射器以準確測量飛行時間，並且在所有元件上設計了通訊介面。圖 6 顯示了 HDL 設計的簡化框圖。ADI 的 HDL 參考設計採用了通用架構使得框架可擴展，且更容易連接另一個 FPGA 埠。該設計使用 ADI 的 JESD204B 框架²，以及多個 SPI 和 GPIO 介面來接收來自 AD9094 ADC 的資料，以及控制該原型機平台上的所有元件。

JESD204 連結配置用於支援 4 個資料轉換器 (M)，這些轉換器使用線路速率為 10 Gbps 的 4 條路線來實現 8 位元轉換器解析度。元件時脈與高速收發器的參考時脈相同，被設定為 250 MHz，由 DAQ 板提供。該連結在 Subclass 1 模式下運行，確保高速轉換器和 FPGA 之間具備確定性延遲。

對於 LIDAR 系統，最大的挑戰在於如何同步各種功能和發射脈衝，以及如何處理從高速 ADC 接收的必要數量的資料。為了解決這一個挑戰，HDL 設計中包含了一個 IP，用於提供產生雷射器脈衝所需的邏輯，控制 TIA 的內部多工器，以及為 DMA 提

圖 6: HDL 設計框圖。

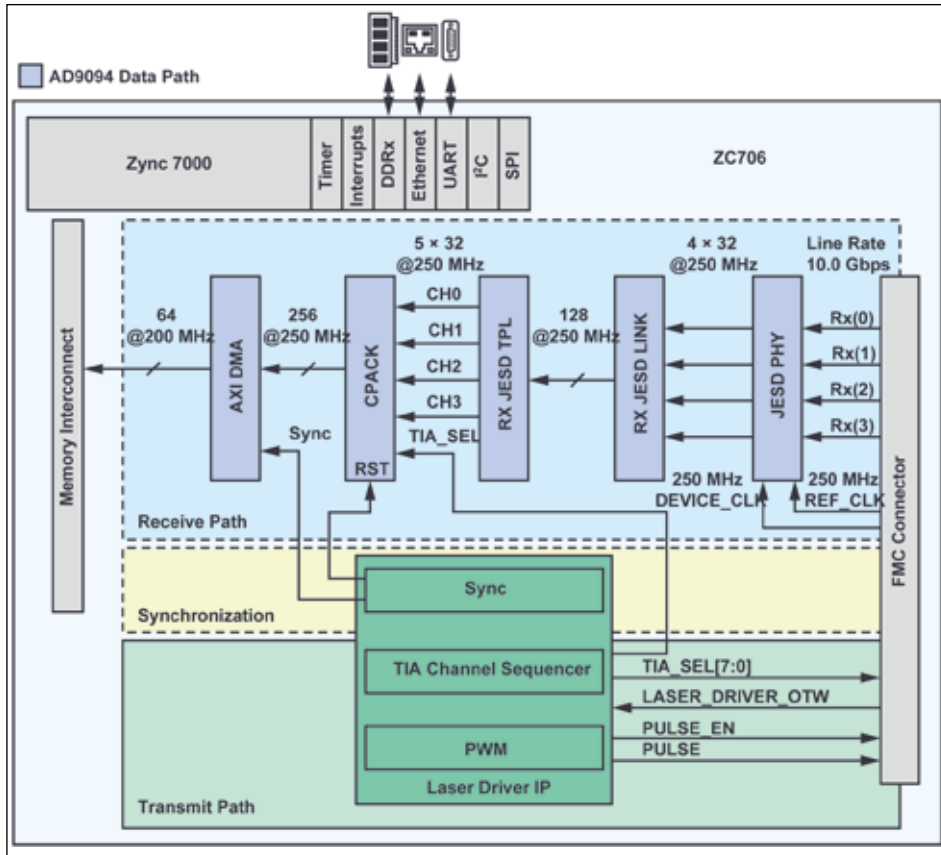
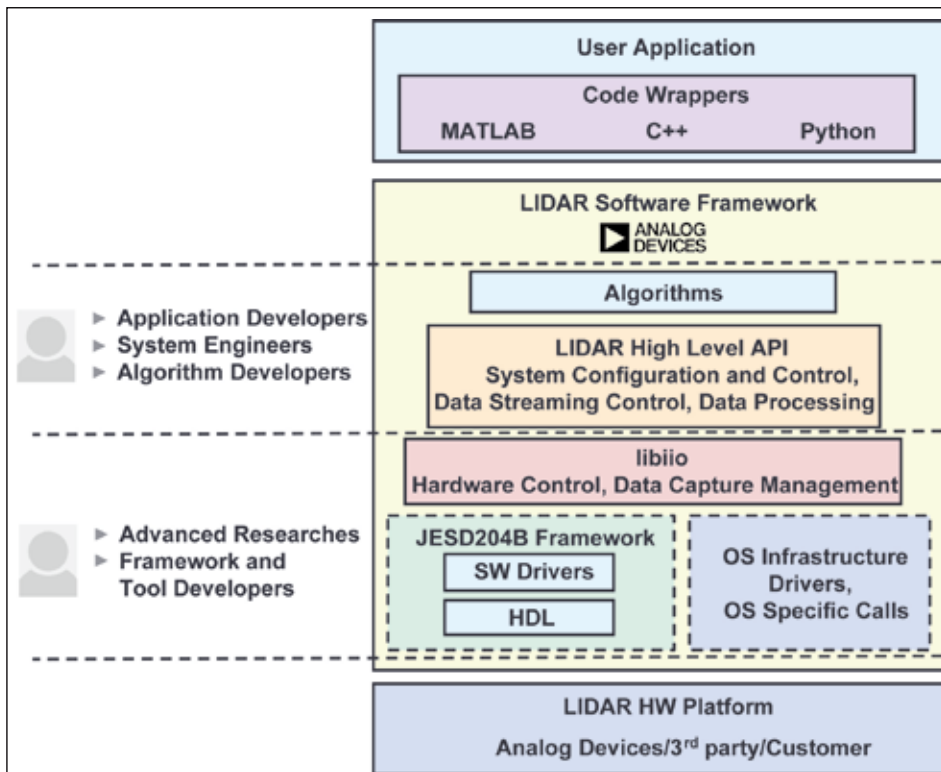


圖 7: 軟體堆疊。



供背壓。所有這些控制函數都與發射脈衝同步，以便系統無需保存所有原始高速量化資料流程。如此，大幅降低系統的總數據速率。

軟體

T 定義 LIDAR 平台的軟體堆疊的幾個關鍵點包括自由和開源。用戶因此能夠自由運行、複製、分發、學習、變更和改善軟體，³ 包括從 Linux 核心到用戶域的工具，以及與此相關的所有代碼。

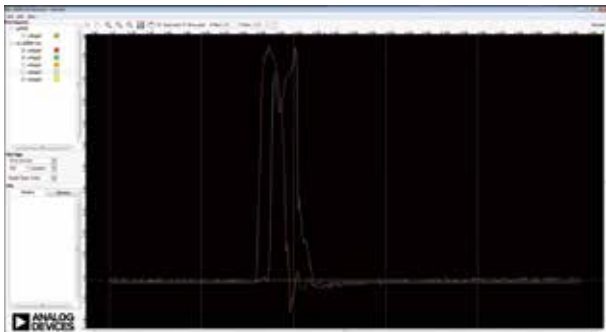
核心中使用的軟體驅動器會啟動硬體元件，向使用者顯示所有可用功能。這些驅動器大部分都是工業 I/O (IIO) Linux 子系統的組成部分。⁴ 這些驅動器都與平台無關，所以無需改變硬體，包括與 FPGA 供應商相關的部分 (例如，從 Xilinx FPGA 遷移至 Intel)。

為了簡化軟體介面 IIO 元件開發，ADI 開發出了 libiio 庫。⁵ 該庫提取硬體的底層詳情，提供簡單但完整的編程介面，可供進階專案使用。多種可用的 libiio 後端 (例如，本地、網路、USB、串列端) 支援在本地使用 IIO 元件，以及遠端在不同作業系統上運行的應用 (包括，Linux、Windows、

macOS) 中使用該元件。

ADI 開發的 IIO 示波器就是這樣一項應用示例，它使用 libiio 連接 IIO 元件，可在系統評估階段使用。該工具可在不同模式下捕捉和圖示資料 (例如，時域、頻域、星座圖、交互相關)、發送資料以及允許使用者查看和修改被檢測元件的設定。

圖 8: 顯示 LIDAR 資料的 IIO 示波器捕捉視窗。



雖然 libiio 提供低層編程介面，但在大多數情況下，使用者期望使用平台相關的集合了低層驅動器調用的 API，來展示一組功能，用於瀏覽和配置各種系統參數和流資料。因此，LIDAR 原型製作平台採用特定的 API，以及適用於常用框架和程式設計語言 (例如 C/C++、MATLAB 或 Python) 的配套元件，⁶ 使使用者能夠使用其優選的程式設計語言與系統連接，集中精力研發對客戶而言極具價值的演算法和應用。

結論

對於系統設計而言，在建立架構和做出設計決定時，存在著一定程度的模糊性。這代表著系統建構完成後無法正常工作或運行的風險，會導致重複的設計週期，增加開發成本，以及延長產品上市時間。參考設計以預設計的、針對彼此交互操作的系統為基礎，與從頭開始的自訂專用設計相比，其風險降低，整體可預測性和可靠性提高。在規劃過程中使用參考設計作為起點，有助於更快將新設計推向市場，並確保出現更少的意外和問題。系統設計人員總是尋求透過參考平台來驗證其設計方案，以降低風險和提高可靠性。啟動專案時，使用清晰

標準的設計選項有助於推動規劃過程的實施。可透過使用通用語言來協助協調目標，鼓勵多個職能部門相互合作和參與來實現，並且簡化在各設計目標之間評估和取舍的難度。LIDAR 原型製作平台試圖透過提供開源硬體和軟體設計來滿足這些需求，這些設計可以提供初始系統架構階段的參考。硬體平台和軟體堆疊可用於整個產品開發階段，從初始系統評估、開發，整合到最終產品中。參考設計的內容 (例如工程圖紙和 BOM) 可建構、合法、當地語系化的系統設計提供了一個良好的開端。可以幫助縮短設計週期，且可能在整個過程中有助於節省資金。模組化硬體設計支援使得各種配置選項滿足特定的應用要求，而基於業界標準框架和程式設計語言，搭配應用示例的開源軟體堆疊則允許客戶側重於開發應用，為產品注入價值，無需將精力耗費在堆疊的低層。

參考電路

- ¹ Fast Axis Collimator Lenses。FISBA，2019 年。
- ² JESD204 介面框架。ADI，2019 年。
- ³ 自由軟體是什麼？Free Software Foundation, Inc.，2019 年。
- ⁴ Linux Driver Implementer 的 API 指南。ADI，2019 年。
- ⁵ 關於 libiio。ADI，2019 年。
- ⁶ “PyADI-IIO: Analog Devices Python Interfaces for Hardware with Industrial I/O Drivers.”GitHub, Inc.，2020 年。

Michael Hennerich 和 Robin Getz。“Free and Open-Source Software—An Analog Devices Perspective.”《類比對話》，第 44 卷第 3 期，2010 年 3 月。CTA