

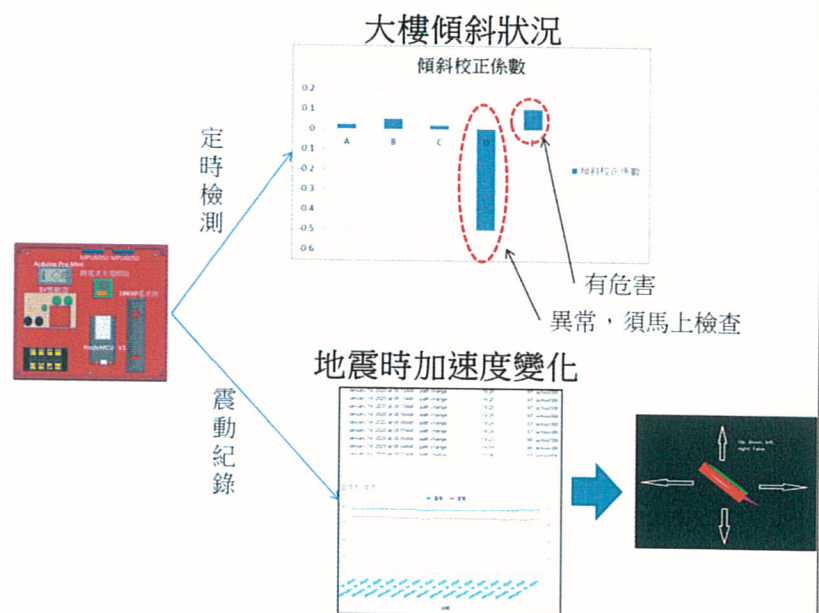
附件二：「2020年花蓮縣夢想起飛-第7屆青少年發明展」作品摘要說明表

作品名稱	大樓傾斜記錄警報系統		隊伍編號	(此編號由主辦單位填寫)
學籍分組	☐ 國小組■國中組			
參賽類組	※作品類組於報名後不得更改之，請再次確認。 ■災害應變 ☐ 農糧技術 ☐ 運動育樂 ☐ 綠能科技 ☐ 安全健康 ☐ 社會照顧			
作品規格	寬： 30 cm	高： 30 cm	深： 30 cm	重量： 2 kg
作品規格長、寬、高上限為1公尺，重量上限為10公斤，若超過上述限制，可利用模型代替之				
摘要說明				
一、 作品名稱： 大樓傾斜記錄警報系統 二、 作品內容與參賽類別的關聯： 本作品主要適用於監控及記錄地震時大樓搖晃傾斜狀況、並且可以在平時作出定時的觀測；若因為地震、工程因素或其他外力因素造成大樓發生傾斜，可以提前預警觀測。而且可以減少人力支出，因此可以達到災害應變的效果。 三、 作品設計/創作動機與目的： 歷經台南大地震、花蓮大地震後，曾看到電視上在報導因為地震倒了幾棟樓、幾個人罹難、幾個人重傷……等等。然而，地震當下，也有發現大樓倒塌的消息其實非常混亂；聽說大樓倒了，結果實際求證時發現它沒倒，或是隔天突然說發現有新的倒塌大樓。再來，地震後，就會有很多人急著要請人診斷，這房子到底有沒有變成危樓等。 聽老師說，有些大樓不是一次地震就會傾斜、就會倒，而是沒有注意到傾斜狀況，因此再一次地震時才會倒；我們平常都是地震後看到大樓有明顯的傾斜狀況，才會去監測它到底傾斜了多少度、或者當大樓倒下時，我們才去看他到底為什麼會倒，這些資料通常都是我們事後在網路上得知，或是事後，我們才去倒塌的大樓裡測量。 而且畢竟我們的土地是在有斷層的地方，地殼的變動是持續不斷的。這些林立的大樓，所在的土地是否有傾斜的問題或是與原先建立大樓的條件不同了，導致這個大樓可能日復一日之下，早就已經傾斜了。 當地震時，該機器會自動啟動，紀錄地震當下時大樓的傾斜角度動態變化，以利設計者觀察，也可以做大樓之後維修的參考、往後建築設計的相關參考、學術資料。 所以如果我們設計一套監測大樓傾斜角度的系統，那可以減少很多的憾事發生！這套系統很省電，只需要一些電力跟網路，就可以一小時監測一次傾斜角度，省電、又顧及了社會大眾的安全！				

四、作品效用與操作方式：

對於這個系統，我們希望它可以每一個小時監測一次大樓的傾斜角度，並且上傳至雲端內，讓相關人員，隨時可以利用機器回傳的雲端圖表，查看大樓是否傾斜，即使他們不在大樓裡。在地震發生過後，如果發現大樓傾斜沒有恢復，可以立即通報，看是系統出了問題，或是大樓真的出狀況了。如果沒有傾斜，就繼續讓機器一天、一天不停的監測下去，而且這樣也可以讓那些工作人員、保全，在地震過後，也不用在大樓盯著螢幕看，可以在家裡利用手機看雲端圖表，不用特別衝去大樓值班觀看。甚至可以直接以 e-mail 或 line 的方式馬上通知相關人員。

除了每小時的傾斜角度偵測以外，我們作品還有地震狀態偵測。ESP8266 每小時會醒來一次，紀錄當下 MPU6050 計錄到的重力加速度，透過 IFTTT(我們利用的是裡面的 Webhooks)，把資料傳到雲端，讓我們可以只要開雲端就可以看到相關資料。當地震時，Arduino 上的 MPU6050 會偵測到重力加速度的角度變化，就會透過低電位腳位立即喚醒 ESP8266，讓 ESP8266 透過 google 雲端，記錄當下建築物頂端的重力加速度的動態變化，做為以後大樓設計者設計的參考以及大樓日後維修參考。



另外，如果遇到地震，當天就可以對照地震的時間表，看看機器回傳的圖表上有沒有大樓傾斜的問題。但若機器不再回傳資料，或者是回傳的角度不符合常理，可以立即通報，看是哪邊出了問題，趕快搶修。

甚至，我們可以透過特定大樓的傾斜狀況建立出圖表，萬一有地形上的變化，我們可能沒辦法立即在外觀上觀看出來，可以透過這些資料統計起來，看它的變化。

我們希望把這個偵測器裝在大樓的頂樓，因為在大樓的頂樓是整個大樓傾斜可能最明顯的地方，所以把偵測器裝在頂樓會特別敏感。至於偵測器的部分，我們希望是用 MPU6050 的 GY-521 6軸感應傾斜角度。這系統，它平常有自己獨立的充電器，如果大樓因地震停電了，它仍然可以運作，只要有網路(可以使用手機，以 SIM 卡網路分享)，就可以持續的偵測大樓傾斜角度。

五、作品傑出特性與創意特質：

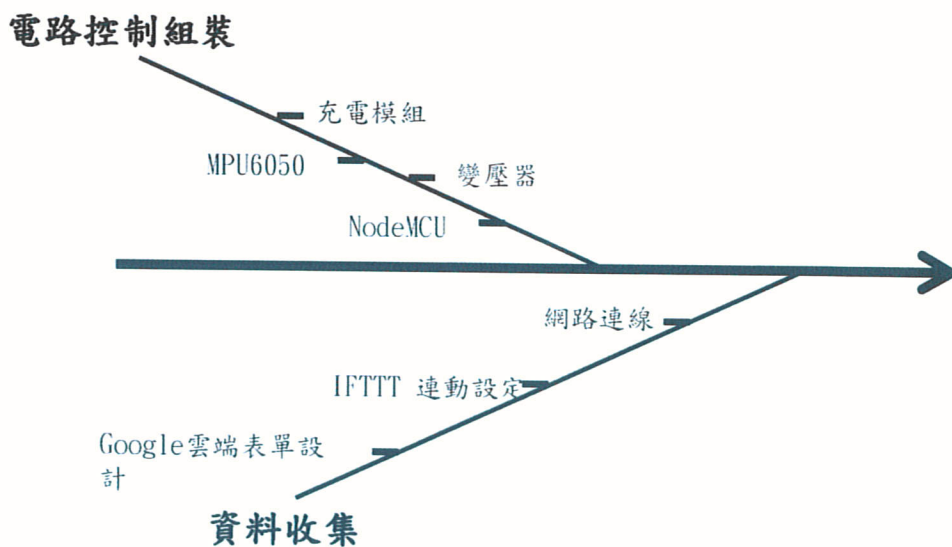
本作品主要用於大樓的地震安全管理系統，很多的測量都必須耗費龐大的資金與人力，我們利用這個機器，不但人力方面只需要維修機器，保全或者是大樓管理者每天也不用為了測量傾斜角度而跑上跑下，只需換用到這台機器，而且這台機器可以自動輸入表單，因此我們在家也可以看傾斜角度，當有地震發生時，有可以盡快的拿到資料，儘管地震把電線扯斷了、停電了，我們還是有辦法拿到這些數據。

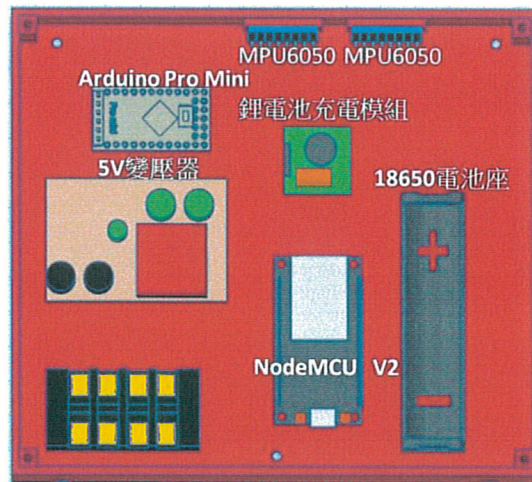
六、其他(創作歷程說明)：

對於 ESP8266 這塊板子，我們選用這塊板子是因為它可以定時「喚醒」來監測某些數據，偵測完後，繼續進入睡眠模式，並且可以連上無線網路，上傳資料到雲端硬碟。我們透過「超圖解 Arduino 互動設計入門」一書，了解了 ESP8266 的線路接法，我們會透過 I2C 與 MPU6050 連線。後來看到了 IFTTT 這個網站，這個網站主要是讓我們上傳資料進入 Google 表單的捷徑，以進行資料記錄。除了上傳固定加速度資料外；其實我們在裝設該偵測器，會故意讓它傾斜，並且在資料處理上提供一個減銷數值來去把它歸零。如果萬一機器可能意外狀況而無法連上網路，我們可以從圖表明顯的正負位置下，判斷它的狀況。我們利用 [ಓಂ](#) [去一又](#) [么、](#) 這個網站來學習如何寫出透過雲端網路監測大樓傾斜角度的程式。

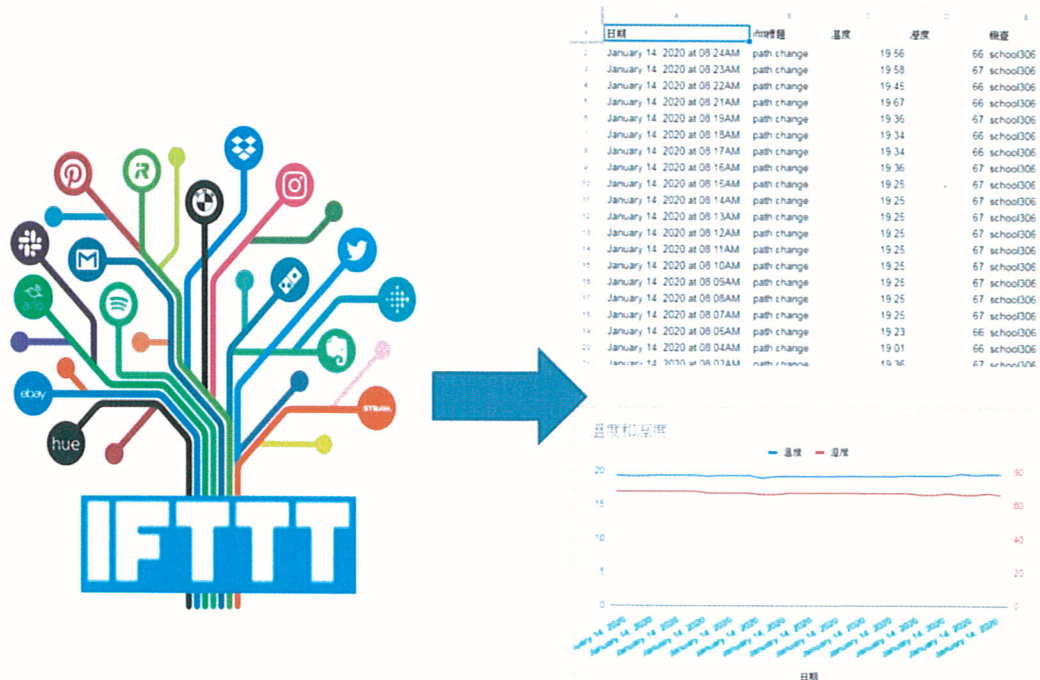
後面利用了 [RANDOM NERD TUTORIALS](#) 這個網站網站裡面的 [WSP8266 Deep Sleep with Arduino IDE \(NodeMCU\)](#)這個分頁，讓我們了解怎麼讓 ESP8266進入睡眠模式的相關資料，並且在固定時間醒來。

我們作品主要分成電路控制、大樓傾斜管控、資料管理。以下是我們作品的魚骨圖。其實我們作品真正的目的是達到了解大樓是否傾斜，至於資料收集的部分，我們將會花非常多的時間學習及研究。以利於使用者可以馬上看出大樓的狀況。





機盒內部圖。如圖示，我們標上了盒內的裝置擺設，從左到右、從上到下分別是:MPU6050-1、MPU6050-2、Arduino pro mini、手機的5V 變壓器、電池充電器、ESP8266，以及電池盒。



我們透過 IFTTT 這個網站來讓 ESP8266連上雲端硬碟，並且在試算表裡每30秒進行一次溫、濕度運算，並且留下日期、時間、以及溫濕度的痕跡，以及讓機器自行製作圖表，以供大家比較容易看出自己要的數據。