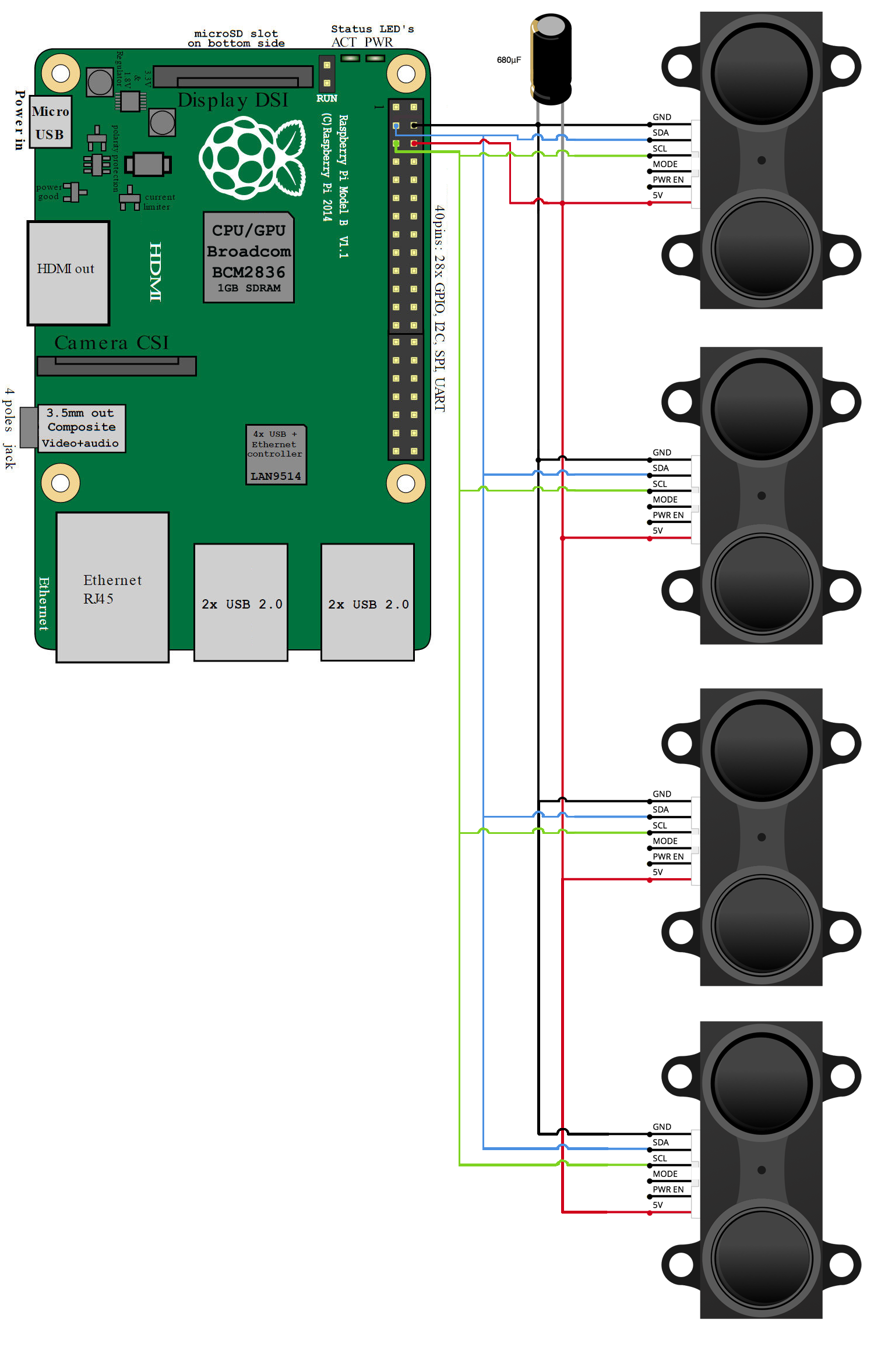
**Developer Manual**

במקרה בו מבוצעת רכישה של כשלושה חיישנים נוספים מסוג Lidar Lite v2, ניתן יהיה להפעיל את המערכת במלואה. על כן, חלק זה במדריך יעסוק בהסבת המערכת לפעילות מלאה.

ראשית, יש לחבר את ארבעת החיישנים למחשב ה- Raspberry Pi 2 כמתואר בתרשים הבא:



לאחר חיבור החיישנים יש לבצע בדיקות קלט לבדיקת פעולתם התקינה של החיישנים. לשם כך, יש ליצור כארבעה אובייקטים מסוג LidarLite המייצגים את כל אחד מן החיישנים ולהעביר אליהם כפרמטר את מספר אפיק ה-Bus המתאים של כל אחד.

לאחר יצירת האובייקטים יש לבצע התחברות לחיישנים באמצעות המתודה connect(self). במידה וישנה תקלה – יש לוודא כי מספר אפיק ה-Bus שנשלח תקין.

לאחר חיבור החיישנים ובדיקתם לתקינות, ישנם שינויים נוספים במערכת התוכנה שיש לבצע להסבת המערכת לעבודה בפעילות מלאה. הטבלה הבאה מתארת את השינויים אותם יש לבצע.

***הערה:*** אין לבצע את השינויים התלויים באובייקט ה-vehicle עד לסיום פיתוח המחלקה על ידי הגורם הרלוונטי בחברה. יש לבדוק את שלמות המחלקה מול גורם אחראי בחברת Airscort.

|  |  |  |
| --- | --- | --- |
| **מחלקה** | **פונקציה/מתודה** | **תיאור השינוי** |
| **Obstacle Avoidance** | **-** | להוציא מהערה את ה-import של מחלקת Vehicle. |
| **-** | לשנות את ערכי קבועי המרחק על פי הכתוב בהערות. |
| \_\_init\_\_ | יש לשנות את חתימת הבנאי של המחלקה כך שיקבל אובייקט vehicle ולהוציא מהערה את בדיקת סוג האובייקט (בדיקה המוודאת את נכונות האובייקט). |
| בעת יצירת אובייקט ה-FlightCommands יש להעביר לו כפרמטר את אובייקט ה-vehicle. |
| בעת יצירת אובייקט ה-FlightData יש להעביר לו כפרמטר את אובייקט ה-vehicle. |
| יש למחוק את השורה  self.\_\_num\_of\_lines = int(self.\_\_get\_number\_of\_line\_in\_file()) |
| run(self) | יש למחוק את חמשת השורות הראשונות, הן יועדו לבדיקת האלגוריתם במצבו הנוכחי במערכת לא מלאה. |
| \_\_get\_number\_of\_line\_in\_file | יש למחוק פונקציה זו. |
| \_\_follow\_safety\_protocol | יש להוציא מהערה את הלולאה, ולהכניס לתוכה את הפקודה הבאה אחריה בלבד. |
| **Flight Commands** | \_\_init\_\_ | יש להוציא את הבנאי מההערה. |
| land | יש למחוק את פקודות ההדפסה ולהוציא מהערה את פקודות ההזזה של הרחפן בעזרת אובייקט ה-vehicle. |
| maintain\_ altitude |
| go\_left |
| go\_right |
| go\_up |
| slow\_down | למחוק את פקודת ההדפסה ולהוציא מהערה את פקודת חישוב המהירות ואת פקודת האטת הרחפן בעזרת אובייקט vehicle. |
| **FlightData** | \_\_init\_\_ | יש למחוק את הבנאי הקיים ולהוציא מהערה את הבנאי הכתוב. |
| get\_current\_ latitude | יש למחוק את השורה ששואבת נתונים מן הסימולאטור ולהוציא מהערה את השורה המקבלת נתוני מיקום דרך אובייקט ה-vehicle. |
| get\_current\_ longitude |
| get\_current\_ height |

|  |  |  |
| --- | --- | --- |
| **Sensors** | - | להוציא מהערה את ה-import של מחלקת LidarLite. |
| \_\_init\_\_ | להוציא מהערה את השורות המגדירות את החיישנים. |
| connect | למחוק את ה-return הנוכחי ולהוציא מהערה את שאר הפקודות הרשומות. |
| check\_ahead | למחוק את השורה השואבת נתונים מן הסימולטור ולהוציא מהערה את זו שמקבלת נתונים מן החיישן. |
| check\_left\_ side |
| check\_right\_ side |
| check\_below |

יש לברר מול גורם רלוונטי בחברה האם פיתוח מחלקת Vehicle הושלם. במידה והיא מוכנה לשימוש, ניתן לחבר את מחשב ה-Raspberry Pi 2 לבקר ה-Pixhawk ולבדוק כי הנתונים שמתקבלים נכונים. במידה ואכן מתקבל קלט תקין מבקר הטיסה של הרחפן ניתן לבצע הרכבה של המערכת על גבי הרחפן ולבצע ניסויים חיים ומבוקרים של המערכת לצורך בדיקתה, ביצוע תיקונים והמשך פיתוח שלה.

במידה ומחלקה זו עדיין לא הושלמה, ניתן להרכיב את החיישנים על גבי מעמד המדמה את הרחפן על פי התרשים שסופק בדו"ח המסכם. לצורך בדיקת המערכת בעזרת מחלקת Simulator יש לבצע את השלבים הבאים:

* תיכנון סימולציות נוספות המדמות מסלולי טיסה עם מכשולים.
* חיבור החיישנים לבקר הארדואינו באותו אופן כמתואר בתרשים החיבור למחשב ה-Raspberry Pi.
* עריכת תוכנת הארדואינו כך שתדפיס ל-Serial Port את נתוני המדידות של כלל החיישנים על פי סדר קבוע.
* הזזת המעמד עם החיישנים במסלול המכשולים שתוכנן תוך כדי שמירת נתוני הקריאה שלהם עם תוכנת CoolTerm, השומרת את ההדפסות המבוצעות על גבי ה-Serial Monitor לתוך קובץ טקסט.
* כתיבת סקריפט פשוט אשר יקבל כקלט את קובץ הטקסט עם נתוני המדידות של כלל החיישנים ויפצל את הנתונים לקבצים נפרדים כאשר כל קובץ יכיל את נתוני המדידות של חיישן אחד בלבד.
* הרצת הסימולציה ובדיקת ההדפסות ל-console לפעולתו התקינה של האלגוריתם תוך כדי השוואת הפעולות לתרשים הזרימה הניתן בדו"ח המסכם.

*במידה ולא מבוצעת רכישה של חיישנים נוספים, ניתן להמשיך לפתח את המערכת באופן הבא:*

ניתן לבצע סימולציות נוספות באופן שבוצעה הסימולציה המתוארת בדו"ח המסכם ולבדוק האם האלגוריתם מתנהג כנדרש על פי תרשים הזרימה המתואר באותו דו"ח.

בנוסף, ניתן להתאים את המערכת לעבודה עם חיישנים אחרים. ראשית, יש לבדוק את התאמת החיישנים למחשב ה-Raspberry Pi 2. לאחר מכן, יש לזנוח את מחלקת ה-LidarLite ולפתח מחלקה חדשה לעבודה עם סוג החיישן החדש. החלפת סוג החיישן לחיישן זול יותר (לדוגמא – חיישן מסוג אולטרא סוני הפועל לטווחים קצרים של עד כ-10 מטרים) יאפשר רכישה של כ-4 חיישנים, את הפעלת המערכת באופן מלא ואף להתקין את המערכת פיזית על גבי הרחפן. בצורה זו ניתן יהיה להגיע לתוצאות טובות יותר כיוון שניתן יהיה לבצע בדיקה מלאה של המערכת עם נתונים ממשיים, בניגוד לבדיקה תחת תנאי סימולציה בהם חלק מן הנתונים נוצרים באופן מלאכותי.

***מודול ה-Simulator***

דבר נוסף אשר ניתן להמשיך לפתח הוא את מודול הסימולאטור של המערכת. מודול זה מעביר את כלל הנתונים הדרושים לפעולתה התקינה של המערכת. להלן פירוט הנתונים שהיא מעבירה לאלגוריתם:

* מדידה מחיישן קדמי.
* מדידה מחיישן ימני.
* מדידה מחיישן שמאלי.
* מדידה מחיישן תחתון.
* גובה הרחפן על פי בקר הטיסה.
* מיקום על קו רוחב על פי ה-GPS המובנה בבקר הטיסה.
* מיקום על קו אורך על פי ה-GPS המובנה בבקר הטיסה.

את נתוני המדידות של החיישן הימני, השמאלי והקדמי מספק המודול לאלגוריתם מתוך קבצי טקסט מתאימים. קבצים אלו מכילים נתוני מדידות שהוזנו אליהם בזמן הזזת החיישן במסלול מכשולים שהוכן כחלק מסימולציית טיסה.

את שאר הנתונים מודול ה-Simulator יוצר בעצמו ומשנה בהתאם לפעולות שונות שהאלגוריתם מבצע. ניתן להמשיך לפתח את אופן היצירה של אותם נתונים במודול זה על מנת לדמות סימולציות אף יותר קרובות למציאות.

דבר נוסף שניתן לבצע הוא חיפוש של כלי עזר חיצוני המדמה נתוני טיסה שונים של הרחפן (לרבות מיקום, מהירות וגובה) ולהיעזר בו לביצוע הסימולציה. ייתכן כי קיים כלי שכזה ובעזרתו ניתן יהיה לבצע סימולציות רבות באופן יעיל יותר לבדיקת האלגוריתם ושיפור הקוד.