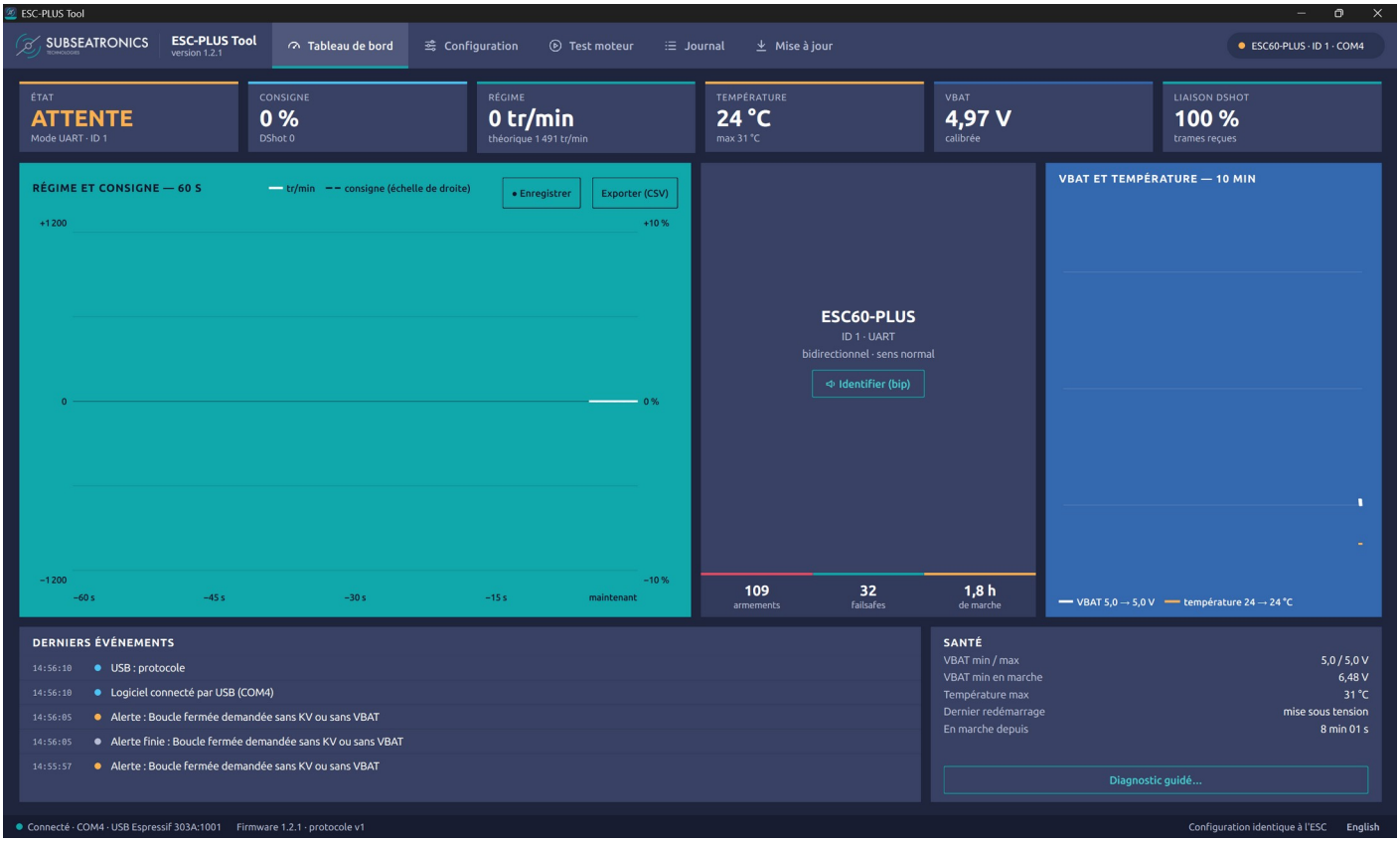
# ESC60-PLUS

## Manuel d'intégration

Piloter l'ESC depuis votre contrôleur : protocoles UART et CAN

Version du document 1.0 - Subseatronics Technologies

Firmware ESC-PLUS 1.2.1- 19 octobre 2026






SUBSEATRONICS
TECHNOLOGIES

# Sommaire

1. Présentation....3
2. Raccordement....4
2.1 Connecteur de commande (connecteur B)....4
2.2 Liaison série (UART)....4
2.3 Bus CAN....4
2.4 Alimentation et mise sous tension....5
3. Principes communs à l'UART et au CAN....5
3.1 Les états de l'ESC....5
3.2 Armer, commander, arrêter....5
3.3 Perte de la consigne : T1, T2 et reprise....6
3.4 Puissance ou régime....6
3.5 Défauts (le moteur s'arrête) et alertes (il continue)....7
3.6 Les commandes doivent être renvoyées si elles restent sans réponse....7
4. Protocole UART....8
4.1 Format....8
4.2 Commandes....8
4.3 Télémétrie TEL:....8
4.4 Événements EVT: et autres lignes émises par l'ESC....9
4.5 Plusieurs ESC sur une même liaison : adressage @ID et bus RS-485....9
4.6 Séquence type (liaison directe)....10
5. Protocole CAN....11
5.1 Identifiants....11
5.2 Trame de commande 0x100 + ID (contrôleur → ESC)....11
5.3 Trame de télémétrie 0x180 + ID (ESC → contrôleur, 8 octets)....11
5.4 Trame d'état 0x1C0 + ID (ESC → contrôleur, 8 octets)....12
5.5 Trames groupées : quatre ESC par trame (option)....13
5.6 Comportements particuliers du bus....13
5.7 Séquence type (ID 3)....13
6. Modes PWM et analogique (rappel)....14
7. Signalisation : LED et bips....14
8. Ce qu'il faut savoir sur le moteur et la télémétrie....15
9. Configuration de l'ESC (ESC-PLUS Tool, par USB)....15
10. Aide-mémoire....16

# 1. Présentation

L'ESC60-PLUS est un variateur pour propulseur brushless de ROV. Il contient deux processeurs : un étage de puissance piloté par le firmware Bluejay (commutation du moteur) et un microcontrôleur de supervision (ESP32-C3) qui reçoit votre commande, applique les sécurités et vous renvoie la télémétrie. C'est ce superviseur que vous pilotez.

Quatre modes de commande, au choix, réglés dans ESC-PLUS Tool :

| Mode | Signal | Pour qui |
|---|---|---|
| PWM RC | Impulsions 1100 / 1500 / 1900 µs (récepteur radio, servo-contrôleur) | Pilotage simple sans retour d'information |
| Analogique | 0 à 5 V (0,5 V arrière, 2,5 V neutre, 4,5 V avant) | Joystick ou sortie analogique d'un automate |
| UART | Liaison série texte 3,3 V, 115 200 bauds (57 600 ou 230 400) | Contrôleur avec un port série ; plusieurs ESC possibles sur un bus RS-485 (chapitre 5) |
| CAN | Bus CAN classique 500 kbit/s (250 ou 1000), identifiants 11 bits | Contrôleur avec un port CAN ; jusqu'à 31 ESC sur le même bus |

Avec l'UART et le CAN, votre contrôleur peut demander une **puissance** (de −100 % à +100 %, boucle ouverte) ou un **régime** (tr/min, boucle fermée régulée par l'ESC), armer et désarmer l'ESC, lire la télémétrie (état, régime, température, tension, défauts) et déclencher des bips d'identification.

### Les cinq règles à retenir

1. **Armer avant de commander** : l'ESC ne suit une consigne qu'à l'état Armé (commande ARM, puis 2 s d'armement).
2. **Entretenir la consigne** : renvoyez-la au moins **10 fois par seconde**. Sans consigne valide pendant 500 ms, le moteur est forcé à 0 ; après 5 s, l'ESC se désarme (failsafe) et il faut réarmer.
3. **Une commande sans réponse se renvoie** : si l'ESC ne répond pas dans les 200 ms (UART) ou si la trame d'état ne reflète pas la commande (CAN), renvoyez-la. Voir 3.6.
4. **Lisez les codes de défaut et d'alerte** : ils sont dans chaque ligne `TEL :` (UART) et dans la trame d'état (CAN) ; un défaut arrête le moteur, une alerte ne l'arrête pas.
5. **Ignorez ce que vous ne connaissez pas** : les versions futures ajoutent des champs en fin de ligne et de nouveaux messages, jamais elles n'en retirent.

# 2. Raccordement

L'ESC porte deux connecteurs JST-SH identiques (pas 1,0 mm, 8 broches) : **CN1** sur la face TOP pour la commande (c'est le connecteur B détaillé en 2.1) et **CN2** sur la face BOTTOM pour l'USB de configuration et la programmation du MCU 2 (étage de puissance, Bluejay). Brochage et couleur des fils :

| Broche | CN1 (face TOP) : commande | Fil | CN2 (face BOTTOM) : USB et programmation | Fil |
|---|---|---|---|---|
| 1 | UART_TX (sortie série de l'ESC) | Violet | C2D (programmation du MCU 2) | Violet |
| 2 | UART_RX (entrée série de l'ESC) | Orange | C2CK (programmation du MCU 2) | Orange |
| 3 | GND | Blanc | VBUS (+5 V de l'USB) | Fiche USB |
| 4 | CAN_H | Bleu | GND | Fiche USB |
| 5 | CAN_L | Vert | NC (ne rien raccorder) | - |
| 6 | GND | Jaune | USB D- | Fiche USB |
| 7 | PWM_IN (entrée PWM RC) | Noir | USB D+ | Fiche USB |
| 8 | 0/10V_IN (entrée analogique) | Rouge | GND | Fiche USB |

**Attention :** ce code couleur ne suit pas la convention des servos RC. Sur CN1, le signal PWM est le fil noir (broche 7) et sa masse le fil jaune (broche 6) ; le fil rouge (broche 8) est l'entrée analogique, pas une alimentation. Sur CN2, la fiche USB est déjà montée au bout du câble.

## 2.1 Connecteur de commande (connecteur B)

Connecteur JST-SH, pas 1,0 mm, 8 broches. Les deux broches de fixation sont à la masse.

| Broche | Nom | Usage |
|---|---|---|
| 1 | UART_TX | Sortie série de l'ESC → entrée RX de votre contrôleur (3,3 V) |
| 2 | UART_RX | Entrée série de l'ESC ← sortie TX de votre contrôleur (3,3 V uniquement, jamais 5 V) |
| 3 | GND | Masse |
| 4 | CAN_H | Bus CAN |
| 5 | CAN_L | Bus CAN |
| 6 | GND | Masse |
| 7 | PWM_IN | Entrée PWM RC |
| 8 | 0/10V_IN | Entrée analogique (0-5 V sur la carte actuelle) |

## 2.2 Liaison série (UART)

- Niveaux logiques **3,3 V**, 8 bits, sans parité, 1 bit de stop (8N1). Débit réglé dans ESC-PLUS Tool : 57 600, **115 200** (défaut) ou 230 400 bauds.
- Point à point : croiser TX et RX (broche 1 de l'ESC vers le RX du contrôleur, broche 2 vers son TX), masse commune.
- Plusieurs ESC sur une même liaison : bus RS-485 avec, pour chaque ESC, un transceiver RS-485 **à commutation automatique du sens** (l'ESC ne pilote pas de broche de direction), terminaisons aux deux bouts, et mode bus activé dans chaque ESC (chapitre 5). Ne jamais relier directement entre elles les sorties TX 3,3 V de plusieurs ESC.

## 2.3 Bus CAN

- CAN classique (pas CAN FD), identifiants 11 bits, 250, **500** (défaut) ou 1000 kbit/s, le même débit sur tout le bus.

- Une terminaison de **120 Ω** à chaque extrémité du bus, et seulement là. L'ESC peut en porter une (résistance R15 sur la carte) : vérifier sur chaque carte si elle est montée, et ne la laisser que sur les ESC en bout de bus. Entre CAN_H et CAN_L, bus hors tension, on doit mesurer environ 60 Ω.
- **Le transceiver CAN est alimenté par la puissance (VBAT)** : sur le câble USB seul, l'ESC répond au logiciel mais le bus CAN est inactif (alerte `CAN_BUSOFF`). Faites vos essais CAN avec la puissance branchée.

### 2.4 Alimentation et mise sous tension

- La puissance (VBAT) alimente le moteur, le superviseur et le transceiver CAN. Sur l'USB seul, le superviseur fonctionne (configuration, essais de protocole) mais le moteur ne tourne pas et aucun bip n'est émis.
- Mettez votre commande **au neutre (consigne 0) avant** de brancher la puissance.
- Après la mise sous tension, l'ESC met environ **5 secondes** à être prêt (initialisation de l'étage de puissance, bips de démarrage). En UART, il l'annonce par la ligne `BOOT:ESC_PLUS_V1.2.1` (version du firmware installé) ; en CAN, la télémétrie commence et indique l'état 1 (attente).

## 3. Principes communs à l'UART et au CAN

### 3.1 Les états de l'ESC

Le même code d'état est utilisé partout (télémétrie UART et CAN, LED, bips).

| Code | État | Moteur | Comment on y entre / en sort |
|---|---|---|---|
| 0 | BOOT - démarrage | Arrêté | Mise sous tension ; dure environ 5 s |
| 1 | WAIT - attente | Arrêté | État de repos. Sortie : commande ARM |
| 2 | ARMING - armement | Arrêté | Après ARM ; dure la durée d'armement (2 s par défaut), un bip aigu à la fin |
| 3 | ARMED - armé | Suit la consigne | Fin de l'armement. Sortie : DISARM, perte de consigne, défaut |
| 4 | FAILSAFE | Arrêté | Consigne absente pendant T2 (5 s). Il faut réarmer (ARM) |
| 5 | SUSPEND - suspendu | Forcé à 0 | Consigne absente pendant T1 (500 ms) ; retour automatique à Armé (avec une rampe de 0,5 s) si la consigne revient avant T2 |
| 6 | FAULT - défaut | Arrêté | Défaut actif (tableau 3.5). Retour en attente quand la cause disparaît ; réarmer ensuite |
| 7 | TEST - test moteur | Piloté par le PC | Depuis ESC-PLUS Tool seulement ; ARM est refusé (BUSY), une consigne aussi (NOTARM) |

### 3.2 Armer, commander, arrêter

1. Envoyez **ARM** (UART : `A` ; CAN : octet `0x02`). L'ESC répond `OK:ARMING` et passe à l'état 2.
2. Attendez la fin de l'armement : état 3 dans la télémétrie, ligne `EVT:ARMED` en UART. Pendant l'armement, une consigne est refusée (`NOTARM`).
3. Envoyez la consigne **en continu**, au moins 10 fois par seconde, même si elle ne change pas : c'est elle qui tient le moteur en vie (voir 3.3).
4. Pour arrêter le moteur sans désarmer : consigne 0. Pour tout arrêter : **DISARM** (UART : `D` ; CAN : `0x03`) - effet immédiat, même moteur en marche.

En bidirectionnel, une consigne positive fait tourner en marche avant, négative en marche arrière. Un changement de sens passe par un arrêt en roue libre (jusqu'à 2 s, réglable) avant de repartir dans l'autre sens : c'est normal, et c'est ce qui évite les surtensions sur votre alimentation.

## 3.3 Perte de la consigne : T1, T2 et reprise

| Délai | Valeur par défaut (réglable) | Effet |
|---|---|---|
| T1 | 500 ms (100 à 2000 ms) | Aucune consigne valide reçue depuis T1 : moteur forcé à 0, état Suspendu (5). Si la consigne revient avant T2, le moteur repart avec une rampe de 0,5 s (état Armé, `EVT:RESUME`). |
| T2 | 5 s (1 à 10 s) | Aucune consigne depuis T2 : état Failsafe (4), l'ESC est désarmé. Il faut renvoyer ARM pour repartir. |

Seule une **consigne valide** (puissance ou régime, dans les bornes, ESC armé) relance ces délais. Une interrogation (`Q`, QUERY), une trame mal formée ou destinée à un autre ESC ne les relance pas.

## 3.4 Puissance ou régime

| | Consigne de puissance (boucle ouverte) | Consigne de régime (boucle fermée) |
|---|---|---|
| UART | `T:<valeur>` de −1000 à +1000 (‰) | `S:<tr/min>` signé, ex. `S:3000`, `S:−2500` |
| CAN | `0x01` SET_THROTTLE, int16 little-endian | `0x06` SET_SPEED, int32 little-endian en tr/min |
| Condition | ESC armé | ESC armé + option « Consigne de régime » activée dans ESC-PLUS Tool + KV du moteur renseigné + tension VBAT mesurée |
| Comportement | La puissance est appliquée telle quelle (après la courbe, les limites et les rampes configurées, neutres par défaut) | L'ESC régule le régime mesuré ; la cible monte avec une rampe (2000 tr/min/s par défaut) et descend sans limite. Au-delà de VMAX (régime atteignable = KV × VBAT × 85 %, recalculé en permanence), la consigne est appliquée à VMAX et une erreur `VMAX` est signalée |

Le dernier type de consigne reçu fixe le mode : une consigne de puissance après une consigne de régime repasse en puissance. En unidirectionnel, une consigne négative est refusée en UART (`ERR:RANGE`) et ramenée à 0 en CAN (erreur 5).

## 3.5 Défauts (le moteur s'arrête) et alertes (il continue)

**Défauts** - code dans le 9e champ de `TEL:` (UART) et dans l'octet 0 de la trame d'état (CAN) :

| Code | Défaut | Cause et sortie |
|---|---|---|
| 0 | Aucun | |
| 2 | VBAT basse | Tension sous le seuil réglé pendant 2 s (seuil désactivé par défaut). Sortie : tension revenue 0,5 V au-dessus du seuil |
| 3 | Température | Température de l'étage de puissance au-dessus du seuil (80 °C par défaut) pendant 2 s. Sortie : 10 °C en dessous |
| 4 | Hélice bloquée | Consigne au-dessus de 15 % et moteur à 0 tr/min pendant 1 s, trois essais. Sortie : acquittement par ARM ou DISARM |
| 5 | Conflit d'identifiant CAN | Un autre équipement émet la télémétrie `0x180 + ID` de cet ESC. Sortie : 10 s après la dernière trame étrangère |
| 6 | VBAT haute | Tension au-dessus du seuil réglé pendant 2 s (désactivé par défaut) |

En défaut, l'ESC est à l'état 6 ; un ARM reçu pendant un défaut est refusé (`ERR:FAULT:<code>`). Quand la cause disparaît, l'ESC repasse en attente (1) : réarmez.

**Alertes** - bits additionnés dans le 10e champ de `TEL:` (UART, en décimal) et dans l'octet 2 de la trame d'état (CAN) :

| Bit | Valeur | Alerte | Signification |
|---|---|---|---|
| 0 | 1 | TELEM | Télémétrie interne du moteur perdue en marche : le régime affiché n'est plus fiable |
| 1 | 2 | VBAT_SAT | Tension hors de la plage mesurable |
| 2 | 4 | CFG_RESET | Configuration illisible au démarrage : valeurs par défaut chargées (à reconfigurer) |
| 3 | 8 | CL_FALLBACK | Régulation de régime sans mesure de régime : anticipation seule |
| 4 | 16 | CL_NO_KV | Consigne de régime autorisée mais VMAX inconnue (KV à 0 ou VBAT absente) |
| 5 | 32 | CAN_BUSOFF | Bus CAN en erreur, récupération automatique en cours |
| 6 | 64 | NO_TEMP | Pas de température confirmée depuis 7 s |
| 7 | 128 | DERATE | Puissance réduite avant une coupure (température à moins de 10 °C de la coupure, ou tension à moins de 1 V de la coupure basse) : plafond de 100 % à 25 % |

## 3.6 Les commandes doivent être renvoyées si elles restent sans réponse

### Règle de robustesse

Moteur arrêté et ESC désarmé, l'ESC enregistre de temps en temps ses compteurs et sa boîte noire en mémoire flash (à partir de 2 s après l'arrêt). Pendant un tel accès (en général moins d'une milliseconde, exceptionnellement quelques dizaines de millisecondes quand la mémoire se réorganise), la réception est suspendue : une commande **en cours de réception** à cet instant précis peut être perdue, **en UART sans aucune réponse** (ni `OK:` ni `ERR:`) ; en CAN, la trame peut être ignorée. Une commande reçue avant ou après l'accès est toujours traitée (elle peut répondre avec quelques dizaines de millisecondes de retard).

Règle pour votre contrôleur : **toute commande d'action (ARM, DISARM, consigne) sans réponse dans les 200 ms est renvoyée** ; en CAN, vérifiez dans la télémétrie que l'état a changé (ARM → état 2 dans les 100 ms) et renvoyez sinon. Une consigne entretenue 10 fois par seconde se corrige d'elle-même à l'envoi suivant.

Cette règle reste une bonne pratique en toutes circonstances (parasites, bus RS-485 partagé) : écrivez votre contrôleur pour qu'il ne suppose jamais qu'une commande a été exécutée sans en avoir vu la réponse ou l'effet dans la télémétrie.

# 4. Protocole UART

## 4.1 Format

- Texte ASCII, une commande par ligne, terminée par CR, LF ou CR LF. L'ESC termine ses lignes par CR LF.
- Majuscules ou minuscules acceptées. Au plus 32 caractères utiles par ligne.
- Réponses : `OK:…` (commande acceptée), `ERR:<code>…` (refusée), `TEL:…` (télémétrie), `EVT:…` (événement), `DIAG:…`, `LOG:…`, `BOOT:…`. Votre contrôleur doit **ignorer toute ligne qu'il ne connaît pas**.

## 4.2 Commandes

| Commande | Effet | Réponses possibles |
|---|---|---|
| `T:<v>` | Consigne de puissance, v de −1000 à +1000 (‰). Appliquée seulement si l'ESC est armé | `OK:T:<v>` ; `ERR:NOTARM` ; `ERR:RANGE` ; `ERR:BADFMT:<ligne>` |
| `S:<tr/min>` | Consigne de régime, signée (voir 3.4) | `OK:S:<v>` ; `ERR:VMAX:<vmax>` (appliquée, mais limitée à VMAX) ; `ERR:CLOFF` (option non activée) ; `ERR:NOKV` (KV ou VBAT manquants) ; `ERR:NOTARM` ; `ERR:RANGE` ; `ERR:BADFMT:<ligne>` |
| `A` ou `ARM` | Armer | `OK:ARMING` puis, 2 s plus tard, `EVT:ARMED` ; `OK:ARMED` si déjà armé ; `ERR:FAULT:<code>` ; `ERR:BUSY` (démarrage ou test en cours) |
| `D` ou `DISARM` | Désarmer (arrêt immédiat) | `OK:DISARMED` |
| `Q` ou `QUERY` | Télémétrie immédiate | Une ligne `TEL:` |
| `I` ou `IDENT` | Bips d'identification (ESC armé ou non, consigne nulle, moteur arrêté) | `OK:IDENT` ; `ERR:RUNNING` |
| `F` ou `DIAG` | Diagnostic complet | Une ligne `DIAG:` |
| `L` ou `LOG` | Journal des 48 derniers événements | Lignes `LOG:<ms>,<code>,<détail>` puis `LOG:END,<nombre>` |

## 4.3 Télémétrie TEL:

Envoyée automatiquement à **10 Hz** par défaut (5, 20 Hz ou coupée, réglable), et sur `Q`.

```
TEL:ARMED,626,6000,30,100,2400,9599,6000,0,0
```

| Champ | Contenu | Exemple |
|---|---|---|
| 1 | État (nom : BOOT, WAIT, ARMING, ARMED, FAILSAFE, SUSPEND, FAULT, TEST) | ARMED |
| 2 | Consigne en ‰ (0 hors état armé). En puissance : la consigne reçue ; en régime : la puissance calculée par la régulation | 626 |
| 3 | Régime mesuré en tr/min | 6000 |
| 4 | Température de l'étage de puissance en °C | 30 |
| 5 | Qualité de la télémétrie interne (0 à 100 %) | 100 |
| 6 | Tension VBAT en centivolts (2400 = 24,00 V) | 2400 |
| 7 | VMAX en tr/min (0 = inconnue) | 9599 |
| 8 | Régime visé en tr/min, signé (0 en consigne de puissance) | 6000 |
| 9 | Code du défaut actif (tableau 3.5 ; 0 = aucun) | 0 |
| 10 | Alertes : somme des bits (tableau 3.5 ; 0 = aucune) | 0 |

Un contrôleur qui ne lit que les premiers champs restent compatibles : les champs sont ajoutés en fin de ligne au fil des versions.

## 4.4 Événements EVT: et autres lignes émises par l'ESC

| Ligne | Quand |
|---|---|
| `BOOT:ESC_PLUS_V1.2.1` | Première ligne après la mise sous tension, quand l'ESC est prêt (environ 5 s) ; la version est celle du firmware installé |
| `EVT:ARMED` | Fin de l'armement : les consignes sont acceptées |
| `EVT:SUSPEND` | Consigne absente depuis T1 : moteur forcé à 0 |
| `EVT:RESUME` | Consigne revenue avant T2 : reprise avec rampe |
| `EVT:FAILSAFE` | Consigne absente depuis T2 : désarmé, réarmer |
| `EVT:FAULT:<code>` / `EVT:FAULT_END:<code>` | Apparition / fin d'un défaut (code : VBAT_LOW, TEMP, STALL, ID_CONFLICT, VBAT_HIGH) |
| `EVT:ALERT:<nom>` / `EVT:ALERT_END:<nom>` | Apparition / fin d'une alerte (noms du tableau 3.5) |
| `DIAG:ST=…,FLT=…,LAST=…,NFLT=…,ALR=…,ERR=…,MODE=…,VMAX=…,TGT=…,OUT=…,RPM=…,KV=…,VBAT=…,T=…,RX=…,CAUSE=…,UP=…,DRT=…` | Réponse à `F` : état, défaut actif, dernier défaut et leur nombre, alertes (séparées par `\|`), erreur de la dernière commande, mode (POWER / SPEED), VMAX, régime visé, puissance réellement envoyée (‰), régime, KV, VBAT, température, qualité télémétrie, cause du dernier changement d'état, secondes depuis le démarrage, plafond de puissance en % (100 = aucune réduction) |

## 4.5 Plusieurs ESC sur une même liaison : adressage @ID et bus RS-485

- **Adressage** : préfixez la commande par `@<ID>:` (ID de 1 à 31, réglé dans ESC-PLUS Tool). Exemples : `@3:A`, `@3:T:+250`, `@3:Q`. L'ESC concerné répond avec le même préfixe : `@3:OK:T:250`, `@3:TEL:…`. Les autres ESC ne répondent pas.
- `@0:<commande>` s'adresse à **tous les ESC** à la fois, sans aucune réponse (sinon ils parleraient en même temps). Exemple : `@0:D` désarme tout le bus.
- **Mode bus** (option « Mode bus (@ID) » de chaque ESC) : l'ESC ne traite que les commandes adressées et **n'envoie plus rien de lui-même** (ni `BOOT:`, ni `TEL:`, ni `EVT:`) : votre contrôleur interroge chaque ESC (`@<ID>:Q`) et entretient chaque consigne (`@<ID>:T:…` au moins 10 fois par seconde). Hors mode bus, la liaison directe reste inchangée et les commandes adressées sont aussi acceptées.
- **Tour de parole sur RS-485** (un seul émetteur à la fois) : une commande à la fois ; attendre la réponse de l'ESC adressé, ou à défaut un délai (15 ms + le temps d'émission de la réponse : comptez 80 caractères, 240 pour `F`) avant la commande suivante ; laisser **3 ms de silence** après chaque réponse (retournement des transceivers) ; après `@0:`, attendre 15 ms. Pendant un journal (`@<ID>:L`), la ligne reste à cet ESC jusqu'à `@<ID>:LOG:END`.
- Arrêt d'urgence : videz votre file de commandes et envoyez `D` (ou `@0:D`) avant toute autre commande.
- Un écho de vos propres réponses relu sur le bus est ignoré par l'ESC.

## 4.6 Séquence type (liaison directe)

```
← BOOT:ESC_PLUS_V1.2.1            (l'ESC est prêt, ~5 s après la mise sous tension)
← TEL:WAIT,0,0,25,100,2400,9599,0,0,0
→ A
← OK:ARMING
   ... 2 s, bip aigu ...
← EVT:ARMED
→ T:300                           (puis T:300 toutes les 100 ms)
← OK:T:300
← TEL:ARMED,300,2900,26,98,2395,9580,0,0,0
→ T:0                             (arrêt, moteur en roue libre)
← OK:T:0
→ D
← OK:DISARMED
← TEL:WAIT,0,0,26,100,2400,9599,0,0,0
```

Si `A` reste sans réponse pendant 200 ms : renvoyez `A` (règle 3.6).

# 5. Protocole CAN

## 5.1 Identifiants

Chaque ESC a un identifiant **ID de 1 à 31** (réglé dans ESC-PLUS Tool). Les trames sont au format CAN 2.0A (identifiants 11 bits) ; les trames étendues et les trames RTR sont ignorées.

| Identifiant | Sens | Contenu | Cadence |
|---|---|---|---|
| `0x100 + ID` | Contrôleur → ESC | Commande (5.2) | À votre rythme ; consigne au moins 10 fois par seconde |
| `0x180 + ID` | ESC → contrôleur | Télémétrie (5.3) | 10 Hz par défaut (20 ou 50 Hz réglables), et sur QUERY |
| `0x1C0 + ID` | ESC → contrôleur | Trame d'état : défauts, alertes, erreur de la dernière commande, VMAX, régime visé (5.4) | 5 Hz par défaut (10 Hz ou coupée), et dès qu'un défaut, une alerte ou une erreur change, et sur QUERY |
| `0x0F0 + g`, `0x0F8 + g` | Contrôleur → ESC | Consignes groupées de 4 ESC (5.5), option désactivée par défaut | Au moins 10 fois par seconde |

Exemple : l'ESC d'identifiant 3 écoute `0x103`, émet sa télémétrie sur `0x183` et son état sur `0x1C3`.

## 5.2 Trame de commande 0x100 + ID (contrôleur → ESC)

| Octet 0 | Commande | Octets suivants | Exemple pour l'ID 3 (identifiant 0x103) |
|---|---|---|---|
| `0x01` | SET_THROTTLE : consigne de puissance | 1-2 : int16 little-endian, −1000 à +1000 ‰ (hors bornes : refusée) | +300 : `01 2C 01` ; −500 : `01 0C FE` |
| `0x02` | ARM : armer | - | `02` |
| `0x03` | DISARM : désarmer (arrêt immédiat) | - | `03` |
| `0x04` | QUERY : télémétrie et trame d'état immédiates | - | `04` |
| `0x05` | IDENT : bips d'identification (consigne nulle, moteur arrêté) | - | `05` |
| `0x06` | SET_SPEED : consigne de régime (voir 3.4) | 1-4 : int32 little-endian, tr/min signé | 6000 tr/min : `06 70 17 00 00` ; −2500 tr/min : `06 3C F6 FF FF` |

Une trame trop courte ou une commande inconnue est refusée (erreur 8 dans la trame d'état). La consigne n'est appliquée qu'à l'état Armé (sinon erreur 4, NOTARM).

## 5.3 Trame de télémétrie 0x180 + ID (ESC → contrôleur, 8 octets)

| Octet | Contenu |
|---|---|
| 0 | État (code du tableau 3.1) |
| 1-2 | Consigne en ‰, int16 little-endian (0 hors état armé). En puissance : la consigne reçue ; en régime : la puissance calculée par la régulation |
| 3-4 | Régime mesuré en tr/min, uint16 little-endian |
| 5 | Température de l'étage de puissance en °C |
| 6 | Qualité de la télémétrie interne, 0 à 100 % |
| 7 | Tension VBAT × 4 (pas de 0,25 V : 96 = 24,0 V) |

## 5.4 Trame d'état 0x1C0 + ID (ESC → contrôleur, 8 octets)

| Octet | Contenu |
|---|---|
| 0 | Défaut actif (tableau 3.5 ; 0 = aucun) |
| 1 | Dernier défaut apparu depuis la mise sous tension (reste affiché après la fin du défaut) |
| 2 | Alertes : bits du tableau 3.5 |
| 3 | Bits 0-3 : erreur de la dernière commande (tableau ci-dessous) ; bit 4 : consigne de régime en cours ; bit 5 : régime limité à VMAX ; bit 6 : régulation sans régime mesuré ; bit 7 : identification en cours |
| 4-5 | VMAX en tr/min, uint16 little-endian (0 = inconnue) |
| 6-7 | Régime visé en tr/min, uint16 little-endian (valeur absolue ; 0 en consigne de puissance) |

| Code (octet 3, bits 0-3) | Erreur de la dernière commande |
|---|---|
| 0 | Aucune : commande appliquée |
| 1 | VMAX : régime demandé au-delà de VMAX, appliqué à VMAX |
| 2 | CLOFF : consigne de régime non autorisée (option à activer dans ESC-PLUS Tool) |
| 3 | NOKV : KV inconnu ou tension VBAT absente |
| 4 | NOTARM : consigne reçue alors que l'ESC n'est pas armé |
| 5 | RANGE : hors bornes (puissance hors ±1000 : refusée ; négatif en unidirectionnel : ramené à 0) |
| 6 | FAULT : ARM refusé, un défaut est actif (ARM sert aussi d'acquittement d'une hélice bloquée) |
| 7 | BUSY : ARM refusé (démarrage ou test moteur en cours) |
| 8 | BADFMT : trame trop courte ou commande inconnue |
| 9 | RUNNING : IDENT refusé, moteur en marche ou consigne non nulle |

**Trame d'état et mise à jour**

Sur une carte neuve, la trame d'état est émise à 5 Hz. Sur un ESC mis à jour depuis le tout premier firmware (1.0.0), elle est **coupée** pour ne pas ajouter de trafic sur un bus déjà en service : activez-la dans ESC-PLUS Tool (Configuration › CAN › Trame d'état) si vous voulez les codes d'erreur et de défaut en temps réel. Sans elle, l'état et la consigne restent lisibles dans la télémétrie `0x180 + ID`.

## 5.5 Trames groupées : quatre ESC par trame (option)

Pour un contrôleur qui pilote plusieurs propulseurs, une seule trame porte les consignes de quatre ESC. L'option « Trame groupée (4 ESC) » doit être activée dans chaque ESC (désactivée par défaut : les identifiants `0x0F0` à `0x0FF` doivent être libres sur votre bus).

| Identifiant | Contenu (8 octets) |
|---|---|
| `0x0F0 + g` | Consignes de puissance des quatre ESC du groupe g : int16 little-endian chacune, −1000 à +1000 ‰ (hors bornes : refusée, erreur 5) |
| `0x0F8 + g` | Consignes de régime en tr/min des quatre ESC du groupe g : int16 little-endian chacune (−32 768 à +32 766) |

- Groupe g = (ID − 1) ÷ 4 (division entière), emplacement = (ID − 1) modulo 4 : les ESC 1 à 4 forment le groupe 0 (octets 0-1, 2-3, 4-5, 6-7), les ESC 5 à 8 le groupe 1, … les ESC 29 à 31 le groupe 7.
- La valeur **`0x7FFF`** (ou une trame trop courte pour l'emplacement) signifie « pas de consigne pour cet ESC » : sa consigne n'est pas entretenue.
- Même traitement qu'une consigne individuelle : ESC armé requis, entretien au moins 10 fois par seconde, erreurs dans la trame d'état. L'armement, le désarmement, QUERY et IDENT restent individuels (`0x100 + ID`).
- Exemple - ESC 1 à +300, ESC 2 à −200, ESC 3 et 4 sans consigne : `0x0F0 [8] 2C 01 38 FF FF 7F FF 7F`.

## 5.6 Comportements particuliers du bus

- **Conflit d'identifiant** : si l'ESC voit passer une trame `0x180 + son ID` émise par un autre équipement, il passe en défaut 5 (armement refusé) jusqu'à 10 s après la dernière trame étrangère. Donnez un ID différent à chaque ESC.
- **Bus-off** : l'ESC se remet en service tout seul (contrôle toutes les 500 ms), alerte `CAN_BUSOFF` pendant l'incident ; le moteur est forcé à 0 après T1 si la consigne n'arrive plus.
- **Changement de débit** : le nouveau débit est pris en compte après « Écrire dans l'ESC » dans ESC-PLUS Tool ; tous les équipements du bus doivent être au même débit.

## 5.7 Séquence type (ID 3)

```
← 0x183 [8] 01 00 00 00 00 19 64 60     télémétrie : état 1 (attente), 25 °C, 100 %, 24,0 V
→ 0x103 [1] 02                          ARM
← 0x183 [8] 02 ...                      état 2 (armement) pendant 2 s
← 0x183 [8] 03 ...                      état 3 (armé)
→ 0x103 [3] 01 2C 01                    puissance +300, puis toutes les 100 ms
← 0x183 [8] 03 2C 01 54 0B 1A 62 5F     armé, consigne 300, 2900 tr/min, 26 °C, 98 %, 23,75 V
→ 0x103 [3] 01 00 00                    consigne 0 (arrêt, roue libre)
→ 0x103 [1] 03                          DISARM
← 0x183 [8] 01 ...                      état 1 (attente)
```

Si l'état ne passe pas à 2 dans les 100 ms qui suivent ARM : renvoyez ARM (règle 3.6).

# 6. Modes PWM et analogique (rappel)

| | PWM RC (broche 7) | Analogique 0-5 V (broche 8) |
|---|---|---|
| Bidirectionnel | 1100 µs arrière max, 1500 µs neutre, 1900 µs avant max (zone morte ±20 µs) | 0,5 V arrière, 2,5 V neutre, 4,5 V avant (zone morte ±0,1 V) |
| Unidirectionnel | 1100 µs arrêt → 1900 µs plein régime | 0,5 V arrêt → 4,5 V plein régime |
| Armement | Neutre (±50 µs) tenu pendant la durée d'armement ; le moteur ne part qu'après un passage par la zone morte | Neutre (zone morte) tenu pendant la durée d'armement |
| Impulsions acceptées | 500 à 2500 µs, écrêtées aux bornes réglées ; hors de cette plage : signal invalide | Coupure immédiate (failsafe) si l'entrée sort de la plage de plus de 0,2 V pendant 100 ms |
| Perte de signal | Moteur à 0 après 250 ms (T1), failsafe après 5 s (T2) | Coupure immédiate ; réarmement au neutre |

Consigne d'installation PWM : réglez le failsafe du récepteur radio sur « neutre » ou sur « coupure du signal » ; un récepteur qui maintient la dernière position est invisible pour l'ESC.

# 7. Signalisation : LED et bips

La LED bleue « ESP RUN » et les bips du moteur donnent l'état de l'ESC sans aucun écran (les bips exigent la puissance branchée et le moteur arrêté).

| Situation | LED | Bips |
|---|---|---|
| Démarrage, configuration valide | Allumée | Bips 1-2-3, puis autant de bips que le numéro du mode (1 PWM, 2 analogique, 3 UART, 4 CAN) |
| Démarrage, configuration illisible (valeurs par défaut) | Allumée | Bips 3-2-1, deux fois |
| Attente sans signal / avec signal mais pas d'ARM | 1 Hz | 1 bip grave / 2 bips graves toutes les 4 s (réglable : 10 s, 30 s, 60 s, jamais) |
| Armement | 2,5 Hz | Bips courts réguliers, puis 1 bip aigu long à l'armement |
| Armé | Fixe | Silence |
| Suspendu (consigne absente depuis T1) | 5 Hz | Silence |
| Failsafe | 5 Hz | 3 bips aigus rapides toutes les 3 s |
| Entrée analogique hors plage | 10 Hz | Aigu-grave-aigu toutes les 3 s |
| Défaut | 10 Hz | Autant de bips graves longs que le code du défaut (2 VBAT basse ou haute, 3 température, 4 hélice bloquée, 5 conflit d'ID), toutes les 3 s (10 s pour la température) |
| Identification | 10 Hz pendant 5 s | Mélodie 1-3-5-3-1, deux fois |

# 8. Ce qu'il faut savoir sur le moteur et la télémétrie

- **Démarrage du moteur** : la première consigne non nulle part en 150 à 500 ms (l'étage de puissance a une temporisation interne). À bas régime, la qualité de la télémétrie (champ RX %) descend normalement à 55-85 % ; au-dessus d'environ 1300 tr/min elle dépasse 90 %.
- **Arrêt et inversion** : consigne 0 = roue libre (pas de freinage, pas de surtension sur l'alimentation). Une inversion de sens attend que le moteur ralentisse en roue libre (jusqu'à 2 s par défaut, réglable de 0 à 5 s) avant de repartir.
- **Régime** : mesuré par l'étage de puissance (14 pôles pour le moteur de référence, réglable). En consigne de régime, le régime visé renvoyé est la cible en cours de rampe.
- **Température** : celle de l'étage de puissance, envoyée environ toutes les 1 à 2 s ; valeur confirmée par plusieurs mesures ; alerte `NO_TEMP` si elle manque depuis 7 s (fréquent juste après le démarrage : pas avant 15 s).
- **Réduction de puissance** (1.2.0) : entre 70 et 80 °C (par défaut), le plafond de puissance descend de 100 % à 25 % (alerte `DERATE`, `DRT=` dans `DIAG`) ; de même à moins de 1 V au-dessus de la coupure VBAT basse si celle-ci est réglée. La coupure elle-même reste active.
- **Courbe, limites et rampes** (1.2.0, neutres par défaut) : l'ESC peut appliquer une courbe exponentielle, limiter la puissance maximale avant / arrière et lisser la consigne avec des rampes de montée / descente. Tout se règle dans ESC-PLUS Tool ; les coupures de sécurité restent immédiates.
- **Boîte noire** (1.2.0) : les derniers événements (états, défauts, alertes) sont gardés en flash et lisibles avec ESC-PLUS Tool, utile au support.

# 9. Configuration de l'ESC (ESC-PLUS Tool, par USB)

Branchez l'ESC au PC par son connecteur USB, lancez ESC-PLUS Tool : l'ESC est détecté seul. Réglez, puis « Écrire dans l'ESC ». Réglages qui concernent l'intégration :

| Réglage | Où | Valeur par défaut | À faire |
|---|---|---|---|
| Mode de commande | Configuration | PWM | Choisir UART ou CAN |
| ID | Configuration | 1 | Un numéro différent par ESC sur un même bus (1 à 31) |
| Direction, Sens | Configuration | Bidirectionnel, Normal | Unidirectionnel si le propulseur ne tourne que dans un sens ; Inversé pour échanger avant et arrière sans recâbler |
| KV du moteur, pôles | Configuration › Moteur | 0 (inconnu), 14 | Renseigner le KV pour la consigne de régime et VMAX |
| Débit UART, télémétrie TEL | Configuration › UART | 115 200 bauds, 10 Hz | Selon votre contrôleur |
| Mode bus (@ID) | Réglages avancés › UART | Désactivé | Activer pour un bus RS-485 |
| Débit CAN, télémétrie, trame d'état | Configuration › CAN | 500 kbit/s, 10 Hz, 5 Hz | Débit du bus ; trame d'état à activer après une mise à jour depuis la 1.0.0 |
| Trame groupée (4 ESC) | Réglages avancés › CAN | Désactivée | Activer si vous utilisez `0x0F0` / `0x0F8` |
| Consigne de régime (UART, CAN) | Réglages avancés › Boucle fermée | Désactivée | Activer pour `S` : `/0x06` ; régler VMAX (%) et les gains sur votre propulseur |
| Durée d'armement, T1, T2 | Sécurités, Réglages avancés › Sécurité | 2 s, 500 ms, 5 s | Garder les valeurs par défaut sauf besoin particulier |
| Coupure VBAT basse / haute, coupure température | Sécurités | Désactivées, 80 °C | Régler les seuils VBAT selon la batterie (6 V au moins) |
| Bip en attente | Sécurités | Toutes les 4 s | 10 s, 30 s, 60 s ou jamais |

Un ESC se met à jour par le même câble USB (onglet Mise à jour) : la configuration est conservée. Un nouveau firmware reste toujours compatible avec les contrôleurs écrits pour une version précédente.

## 10. Aide-mémoire

| Je veux… | UART | CAN (ID = n) |
|---|---|---|
| Savoir si l'ESC est prêt | Ligne `BOOT:` reçue, ou `Q` → `TEL:WAIT,…` | Télémétrie `0x180+n` avec l'octet 0 = 1 |
| Armer | `A` → `OK:ARMING` → `EVT:ARMED` | `0x100+n [1] 02` → octet 0 de la télémétrie = 2 puis 3 |
| Avancer à 30 % | `T:300` toutes les 100 ms | `0x100+n [3] 01 2C 01` toutes les 100 ms |
| Reculer à 50 % | `T:-500` toutes les 100 ms | `0x100+n [3] 01 0C FE` toutes les 100 ms |
| Tenir 3000 tr/min | `S:3000` toutes les 100 ms (option activée, KV réglé) | `0x100+n [5] 06 B8 0B 00 00` toutes les 100 ms |
| Arrêter le moteur | `T:0` | `0x100+n [3] 01 00 00` |
| Désarmer (arrêt immédiat) | `D` → `OK:DISARMED` | `0x100+n [1] 03` |
| Lire l'état et les défauts | Champs 1, 9 et 10 de `TEL:` ; `F` pour le détail | Octet 0 de la télémétrie ; octets 0 à 3 de la trame d'état `0x1C0+n` |
| Identifier un propulseur (bips) | `I` | `0x100+n [1] 05` |
| Après un failsafe (état 4) | `A` de nouveau | `0x100+n [1] 02` de nouveau |
| Commande sans réponse | Renvoyer après 200 ms | Vérifier la télémétrie, renvoyer |