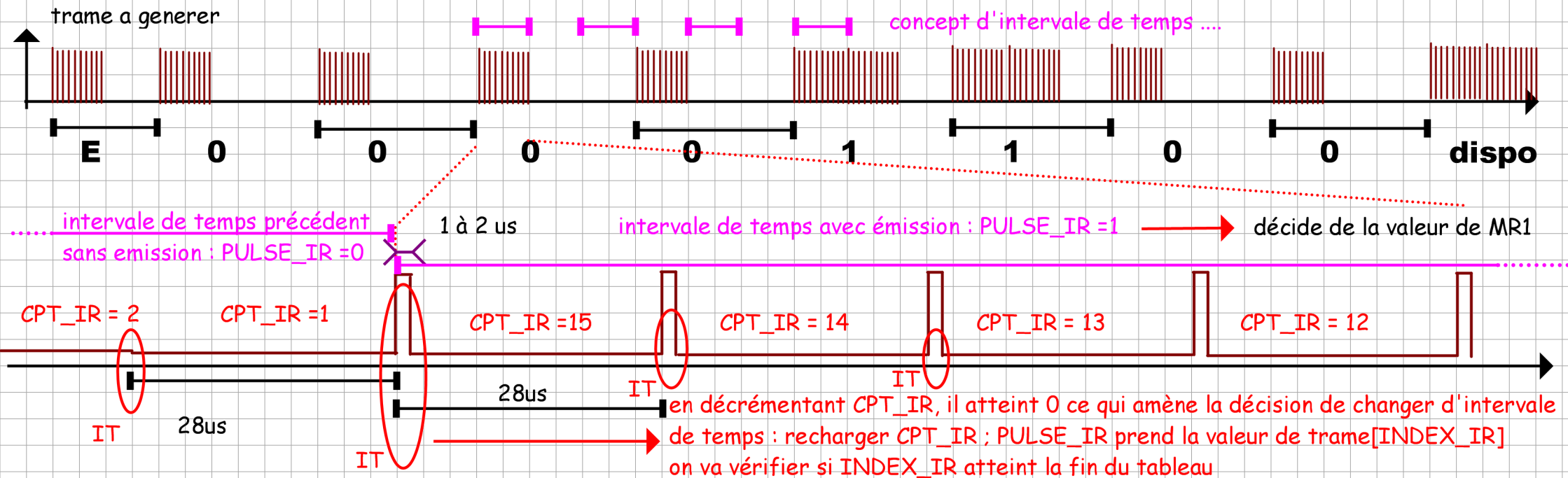


## Schéma bloc transmission IR

trains impulsions avec pour chaque pulse = 2us max toutes les 1/36000 sec



emettre une trame à l'aide d'un PWM : On génère une interruption à chaque cycle PWM et on y décide ce qu'on fait pour la suivante.

Dans cette interruption, on doit savoir si on doit emettre un rapport cyclique nul ou de la valeur qui permet de générer une impulsion IR en chargeant PWM1MR1 avec la bonne valeur.

Pendant tout un intervalle de temps, on doit emettre ou pas :

on sait qu'on est toujours dans le meme intervalle de temps parce que CPT\_IR est non nul.

En fin d'intervale de temps, détecté par un compteur atteignant 0, on prépare le comportement durant tout l'intervale suivant

A chaque changement d'intervale de temps se pose la question de savoir si on est à la fin de trame ou pas :

En fin de trame , soit on reprend en début de trame, soit on emet un très long blanc : tout dépend de la valeur de NB\_trame....

CPT\_IR décompte les IT restantes avec de changer d'intervale de temps (peut être une valeur de 15 cycles pour un intervalle de temps ou une valeur très grande correspondant à 7 trames)

PULSE\_IR décide si les pulses IR sont émis ou pas dans cet intervalle de temps (pulse ou blanc)

INDEX\_IR mémorise le numéro d'intervale de temps pour savoir quoi envoyer après

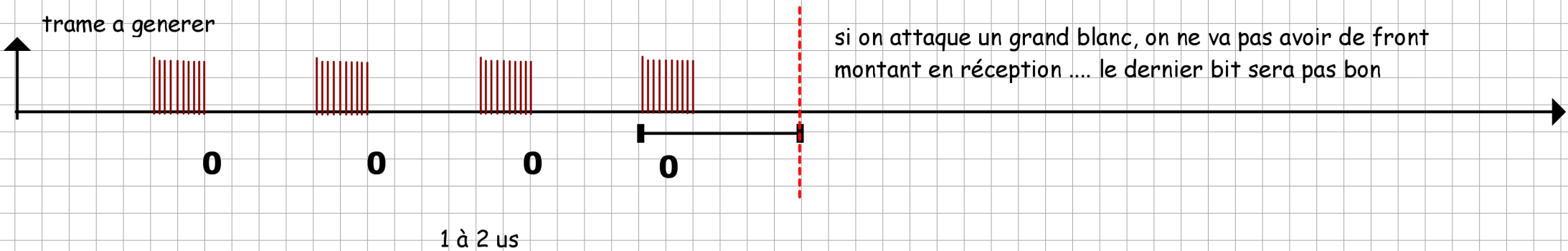
avec juste la mise à jour de PULSE\_IR et de CPT\_IR pour le prochain intervalle de temps...

INDEX\_IR est un index de tableau, (tableau mis à jour dans le main), on doit tester si on atteint la fin (comptage ou décomptage)

NB\_TRAME mémorise le nombre de trames déjà émises... pour décider si on fait un grand blanc ou si on réémet immédiatement

En fin de trame , on est dans la situation suivante :

## Schéma bloc transmission IR



pour résoudre cela , en fin de message, il faut avoir un dernier motif reconnaissable pour comprendre la trame....

ici le motif qui fait comprendre qu'une trame est finie, c'est le début de la suivante .... donc il faut donc rajouter des entetes en fin de trames :

une vraie trame qui donne la garantie d'être comprise entièrement necessite d'avoir une queue marquant la fin

E n n n n s s s E E pour garantir que la trame est reconnue

enchainer les trames fait apparaitre la séquence E n n n n s s s E E E n n n n s s s E E E n n n n s s s E E E n ...

```
void IT_PWM(void) __irq
{ CPT_IR--; if( PULSE_IR==1) {} else {}
if (CPT_IR==0) //fin d'intervale de temps
{
....
// fin de trame ?
}
// les autres choses à faire dans l'it

//acq it
}
```

→ anomalies détectées : trame vide qui signifie "rien à signaler"

## AUTRE TECHNIQUE : Emettre la trame dans le fil

Emettre la trame dans le fil est très simple si on constate que tous les motifs finissent par la même durée de blanc....

Il suffit d'emettre des périodes à 50KHz puis des blancs d'une certaine durée

on va tout décrire en nombre de périodes à 50KHz et on va décompter le nombre de périodes déjà écoulées dans l'émission d'un motif

```
#define MOTIF_E      .... /* nb période à 50KHz que dure un motif Entete , blanc compris*/  
#define MOTIF_UN     ...  
#define MOTIF_ZERO  ....  
#define SEUIL_BLANC ... /* nb période à 50KHz que dure un blanc */
```

On va emettre via un PWM (rapport cyclique 50%(signal) 0%(blanc)) en comptant les cycles à chaque it pour décider du rapport cyclique

it\_pwm(void) :

selon si cpt\_motif > SEUIL\_BLANC: rapport cyclique à 50% ou 0%

cpt\_motif = cpt\_motif -1; // il y a eu une période de plus dans le meme motif

si le motif est fini (cpt\_motif atteint 0), on passe au suivant ...

On postule qu'on dispose d'un tableau qui contient la succession des durées des motifs à envoyer, il suffit donc de recharger cpt\_motif

cpt\_motif = tab\_motif[index\_motif]; // permet de recharger facilement

au lieu d'incrémenter index\_motif, on va aussi le décrémenter, et seulement s'il n'a pas atteint 0

dans la case 0 du tableau tab\_motif, on charge donc le motif qui doit être envoyé quand on a plus rien à transmettre : MOTIF\_E

Dans le main()

Quand on veut envoyer un message on sait que le précédent est fini de transmettre parce que l'index est revenu à zéro ....

il suffit de charger le tableau avec la bonne succession de motif (à l'index le plus élevé, le premier motif à transmettre)

et on déclenche l'envoi en écrivant en dernière ligne du code de création du message : index\_motif=DERNIER\_INDEX;

dès que le motif\_E sera fini de transmettre, le premier motif du message sera automatiquement chargé et transmis

Là encore, la routine d'envoi prend quelques lignes....